Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor

Moti Bekuma Dinagde

A Thesis Submitted to

The department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

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

Office of Graduate Studies

Adama Science and Technology University

June, 2022

Adama, Ethiopia

Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor

Moti Bekuma Dinagde

Advisor: Dr. Shubhashish Bhakta

A Thesis Submitted to

The department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

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

Office of Graduate Studies
Adama Science and Technology University

June, 2022

Adama, Ethiopia

DECLARATION

I hereby declare that this Master Thesis entitled "Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor" is my original work. That is, it has not been submitted for the award of any academic degree, diploma, or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation

--	--------------------------------	--------------------------
Name of student	Signature	Date

RECOMMENDATION

I, the advisor of this thesis, hereby certify that I have read the revised version of the thesis entitled "Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor" prepared under my guidance by Moti Bekuma Dinagde submitted in partial fulfillment of the requirements for the degree of Mater's of Science in Control Engineering. Therefore, I recommend the submission of a revised version of the thesis to the department following the applicable procedures.

--	--------------------------------	--------------------------
Major Advisor	Signature	Date

APPROVAL PAGE

I, the advisors of the thesis entitled “Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor” and developed by Moti BekumaDinagde, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

--	--------------------------------	--------------------------
Major Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by Moti Bekuma Dinagde have read and evaluated the thesis entitled Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor ” and examined the candidate during open defense. This is, therefore, to certify that the thesis is accepted for partial fulfillment of the requirement of the degree of Master of Science in Control Engineering.

--	--------------------------------	--------------------------
Chairperson	Signature	Date
--	--------------------------------	--------------------------
Internal Examiner	Signature	Date
--	--------------------------------	--------------------------
External Examiner	Signature	Date

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

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

ACKNOWLEDGMENT

First of all, thank to God for his provision of peace in all the ways that I have gone. Next, I would like to thank my dear advisor **Dr. S. Bhakta** for his support and contribution. In fact, your guidance has been the success of my career. And thank you to my classroom teachers: **Dr. Tafase, Dr. Mola, Dr. Harish, Pro. S. Kim, Pro. Jin,** and **Pro. Ma.** They have been great sources of inspiration to me and I thank them from the bottom of my heart. I would like to thank all school members of Electrical and Computing Engineering and the postgraduate office. I also thank my classmates friends for the effective contribution they have rendered me in the accomplishment of my research successfully. Last but not the least, I would like to thank my beloved and respected wife **Meti Dosha**. She is strong women and a source of my strength.

Moti Bekuma Dinagde

TABLE OF CONTENT

CONTENTS **PAGES**

LIST OF FIGURE

LIST OF TABLE

LIST OF ABBREVIATION

BSMT	Balanced Stochastic Model Truncation
CF	Coprime Factorization
DOF	Degree Of Freedom
EIV	Error-In-Variable
EKF	Extended Kalman Filter
ESC	Electronic Speed Controller
GAM	Generalized Attitude Model
HTOL	Horizontal Take-Off and Landing
LQR	Linear Quadratic Regression
MAE	Maximum Absolute Error
MATLAB	Matrix Laboratory
MAV	Manned Aerial Vehicle
MRE	Maximum Relative Error
PID	Proportional-Integral-Derivative
PRBS	Pseudo Random Binary Sequence
RPAS	Remotely Piloted Aerial Systems
RPV	Remotely Piloted Vehicle
VTOL	Vertical Take-Off and Landing
SSE	Sum of Squared Error
4SID	SubSpace State Space IDentification
UAV	Unmanned Aerial Vehicle
ZOH	Zero Order Hold

LIST OF SYMBOL

U_4	Yaw control quantity
k_1	ESC constant
k_2	Motor constant
U_d	Armature voltage
ψ (t)	Output yaw angle
T	Torque
Ω_i	i'th Propeller's speed (i =1,2,3,4)
$\psi_r(k)$	Reference input angle
$u_m(k)$	Measured input
$\psi_m(k)$	Measured output angle
$x_{k\|k}$	Filtered state
$P_{k\|k}$	Filtered covariance for $x_{k\|k}$,
$x_{k\|k-1}$	predicted state for filter
$x_{k+1\|k}$	Predicted state for smoother
$P_{k+1\|k}$	Predicted covariance for $x_{k+1\|k}$
K_k	Kalman gain
u_k	Measurement input in kalman filter
y_k	Measurement output in kalman filter
$x_{k\|N}$	Smoothed state in kalman smoother
$P_{k\|N}$	Smoothed covariance for $x_{k\|N}$
S_k	Immediate matrix
x_k	Estimated state

k_d	Drag factor
k_t	Coefficient of thrust factor
$\ddot{\Psi}$	Yaw angular acceleration
$\ddot{\theta}$	Pitch angular acceleration
$\ddot{\Phi}$	Roll angular acceleration
$\dot{\Psi}$	Change in angle of yaw
$\dot{\Theta}$	Change in angle of pitch
$\dot{\Phi}$	Change in angle of roll
ψ	Yaw angle
θ	Pitch angle
Φ	Roll angle
$xh1$	Angular Position state
$xh2$	Angular Velocity state
J_{xx}	Inertia of the quadrotor along the x-axis,
J_{yy}	Inertia of the quadrotor along the y-axis,
J_{zz}	Inertia of the quadrotor along the z-axis,
M_1	Throttle torque
M_2	Roll torque
M_3	Pitch torque
M_4	Yaw torque

ABSTRACT

Quadcopter has many uses in everyday life. So it requires advanced controller to control its attitude. Due to their good applicability to nonlinear multivariable complex control objectives, numerous advanced state-feedback controller design methods have been applied in the quadrotor attitude control. State estimator plays great role in controller design in which state of a plant is feedback to input side. However, traditional observer have reduced performance due to input noise and ouput noise. This thesis proposes a subspace identification-based state estimation method suited to the state-variable estimation for the quadrotor attitude control system in the absence of models with disturbing noises. Above all, an observer-based discrete state feedback control system is designed starting from attitude control performances given by users. In order to obtain a high accuracy model and state estimation, a sufficiently excited attitude-angle reference input is designed as a pseudo-random binary series. Then, the control quantity and the real attitude angle being input and an output for the quadcopter, respectively, can be collected as outputs. The existing general attitude model for the yaw channel is used to test the validation of data. Next, this thesis proposes a 4SID-based state estimation method to deal with the yaw-axis. Quadcopter yaw angle attitude state estimation problem under the conditions that the measurements for the control quantity and the output attitude are corrupted with disturbing noises and that there are no available models. Based on identified model for the generalized attitude model from the proposed state estimation method, the Kalman filter and smoother are applicable and estimation-accuracy comparison done. Finally, the simulation results showed that among the stated estimtors, a 4SID based state estimation method is the best estimator.

Keywords: Quadrotor, State estimation, Subspace identification based state estimation, Kalman filtering, Kalman smoothing.

CHAPTER ONE

1. INTRODUCTION

1.1 Research Background

Drone systems are complex,underactuated systems. It has six degree of freedom (6 DOF) and four inputs. It is type of unmanned aerial vehicle (UAV) driven by four rortors. These four rotors are connected in pairs. The first two rotate clockwise (CW) and the other two rotate anti clockwise (ACW) (Tofigh et al., 2018). It has many rotors and goes in a straight line when it takes off and lands (Kabra et al., 2017). A quadcopter is a sophisticated, highly nimble aircraft with a straightforward design and structure (Singh et al., 2019). It has many applications in this century due to its simplicity of usage and small in size. At the current time, it becomes popular because it is good in maneuverability and survivability (Liang et al., 2022). UAV have increased in popularity among the military because of their capacity to operate in hazardous environments while keeping human operators at a safe distance. It provide a dependable, long-duration, cost-effective reconnaissance platform. They have evolved into a necessary tool for the military (Patel et al., 2016). In today's world, military use of drones or Remotely Piloted Aerial Systems (RPAS) has become the norm. Drones have long been a feature of military forces around the world, serving as target decoys, combat operations, research and development, and surveillance. To deliver equipment from one place to another, to visit someone in need, to bring information to a place where humans cannot reach, to go to a fire hazard and bring the situation. Delivering medicine to a sick person, checking on a wounded soldier on the battlefield. It is highly useful in crowded places ,where there is a lot of traffic. in video recording, the only way to shoot up and down is with a drone. They serve to guard a country's borders, track refugees, and control the crossing of various drugs (Ahirwar et al., 2019). Drones with thermal sensors have night vision, making them an effective surveillance tool. Drones are capable of locating missing people and unlucky victims, especially in difficult terrains or severe situations. A drone can drop supplies to hard-to-reach areas in war-torn or disaster-stricken countries, in addition to locating casualties. By using drones we can locate a soldier to deliver food and clothing to an injured and surrounded soldier (Ahirwar et al., 2019),(Roush, n.d.-a). Drones are increasingly being used to record footage that would normally necessitate the utilization of costly helicopters and cranes. Aerial drones are used to film fast-paced action and sci-fi scenarios, making cinematography easier. These self-piloted aircraft are also employed in real estate and sports photography (Ahirwar et al., 2019; Urbanov, 2017). A farmer can use drones to promote his farm. They can research the weather and predict what to do and earn a lot of money (Ahirwar et al., 2019). Hunters use drones to spot the location of the animal they are looking for and figure out how to catch it. They offer unrivaled security to wildlife such as elephants, rhinos, and big cats, which are popular targets for poachers. Because of their thermal cameras and sensors, drones have the ability to operate at night. This allows people to study and watch wildlife without disturbing them, as well as learn about their routines, behavior, and habitat (Ivošević et al., 2015). After a natural or man-made disaster, drones provide a speedy way to gather information and navigate debris and wreckage in search of injured patients. Its high-definition cameras, sensors, and radars provide a larger field of view to rescue teams, reducing the need for manned helicopters. Drones, due to their small size, can provide a close-up view of regions where larger aerial vehicles would be dangerous or inefficient (Mohd Daud et al., 2022; Restas, 2015).

State feedback control system is to use state of system for control purpose. It is applicable in a system require high pricition. Attitude estimation is important for the drone since it allows finer control of the drone attitude especially in autonomous modes (Roush, n.d.-a). In control system it is something that is often difficult, which is to estimate the true state of a system from observation data. Subspace State space (4SID)-based state estimation method has attracted much attention during the past few years because it is simple in parametrization, the compact model is minimal realization, no linear optimization technique is required and it is Kalman filter framework. This interest is raised to provide accurate state-space models for multivariable systems directly from input-output data. In this research, using the 4SID method is very important to estimate the state parameters using subspace algorithms. State estimators can be used for process monitoring and feedback control since they provide information about unmeasured states, which is necessary for a plant's proper operation. Estimator design is based on the concept of system observability, which stipulates that a set of measurements must offer enough data to estimate all of the system's states [L1].

First, the General attitude model (GAM) is estimated. Then the quadrotor Error-in-variable (EIV) is developed as if there is an error in the input and output. Take Pseudo random biniary series (PRBS) as input to estimate the linear model. Hence input and output are collected.

1.2 Statement of Problem

Quadcopters have a wide range of uses, in agriculture, education, cinematography, military etc and the presence of inaccuracy in state estimation can lead to a lot of problems. Quadcopter state must be precisely estimated in military related applications, in particular. The current approach of estimating quadcopter status is not enough to determine the quadcopter's states. Hence,this study compares the efficiency of subspace identification based state estimation, Kalman Filter, and Kalman Smoother to estimate the quadcopter state and determine the best one.

1.3 Objective of the research

1.3.1 General Objective

The main objective of this research is Pseudo-Random Binary Series Based Comparative Study on State Estimation for a Quadrotor

1.3.2 Specific Objective

The Specific Objective is:

- ✓ To Design Pseudo-Random Binary Series input for identification.
- ✓ To Identify GAM of quadrotor yaw axis with above 95% of estimation accuracy.
- ✓ To Design Kalman Filter estimator
- ✓ To Design Kalman Smoother estimator
- ✓ To Compare estimators using different constraints

1.4 Significance of the research

The main objective is to provide states for a quadrotor GAM. It is not simple to estimate the state when there are errors in the input and outputs. But, 4SID based state estimation method is employed to estimate the state of the closed-loop EIV system by using the available reference in the input and outputs measurements. First, the state-space model for GAM is be estimated, from which its estimation model state can then be easily extracted in the presence of disturbances in the input and outputs. Hence, the subspace identification method can propose a model for Kalman filtering and Kalman smoothing to estimate quadrotor GAM states. Based on the estimated quadrotor GAM, it estimate states. Hence, Kalman filtering and Kalman smoothing can be applied to the proposed model to estimate the quadrotor GAM state in the available reference input and outputs measurements. Controlling the plant becomes easier when the status is precisely estimated, which improves image clarity and decreases target loss in military applications.

1.5 Scope of the study

The scope of this research work is identifying a GAM of quadcopter from input-output data and estimating its state for yaw axis.

1.6 Limitation of the study

Due to the lack of access of actual yaw axis model, this research uses secondary data for validation purpose. And also lack of measuring device for input and output angle the researcher work is limited on simulation only.

1.7 Motivation of the study

A quadcopter has wide range of application in military system. Especially the event happened between Israel and Palestine on 10 May 2021 create huge vision of estimation in my mind (Adnan & Khamis, 2022).

1.8 Thesis Organization

The thesis is breakdown into the following chapters:

Chapter 1: Contains the background of UAVs, statement of the problem, objective of the thesis, scope, limitation, motivation, and significance of the thesis.

Chapter 2: Deals with the literature works previously done.

Chapter 3:Identification of the linear model of quadcopter yaw axis based on subspace identification. Then based on the identified model, it computes state estimation methods for quadcopter yaw axis.

Chapter 4: Apply designed estimators and Simulate.

Chapter 5: Contains simulation results a and detailed numerical analysis of the results

CHAPTER TWO

2. LITERATURE REVIEW

2.1 Chapter Overview

A basic description, kinds, and applications of the UAV quadcopter are described in this chapter and the manner in which it is constructed will be described. Finally, works that are closely connected are examined.

2.2 Unmanned Aerial Vehicles

Unmanned aerial vehicles is takes off and lands without the need for a pilot. Unmanned aircraft systems are made up of an aircraft component, sensor payloads, and a ground control station. They can be controlled using onboard electronic equipment or ground-based control systems. An RPV is a vehicle that can be piloted remotely from the ground and requires reliable wireless communication to function. Large UAVs may require specialized control systems that can be put on automobiles or trailers to allow close proximity to UAVs with restricted range or communication capabilities [L3].

2.3 Classification of UAVs

UAVs are classified based on a variety of factors (Singhal et al., 2018). Classification of UAVs are shown Fig. 2.1 below.

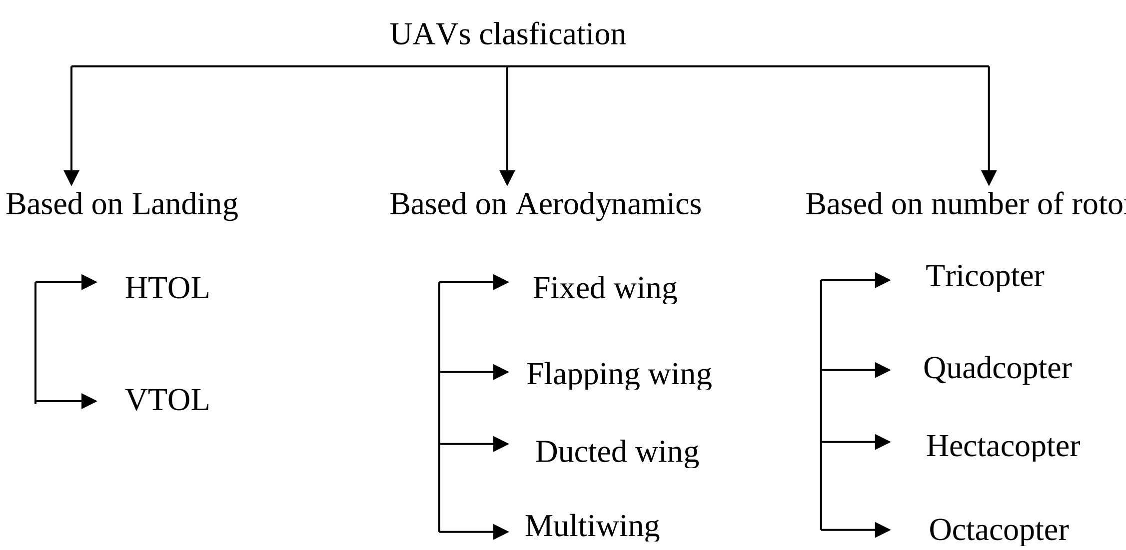

Figure 2.1 Classification of Unmanned Aerial Vehicle.

2.4 Q

A quadcopter is a type of multiotor drone. Multirotor drones are unmanned aerial vehicles having many rotors that generate lift, allowing them to fly. The operating idea is that one pair of rotors rotates clockwise while the other rotates anti-clockwise, and by adjusting the speeds, thrust and turning motions can be generated [L4].

Figure 2.2: Quadcopter image [L5]

2.5 Related Works

In this chapter, previous projects related to design, state estimation, and control of quadcopters are discussed.

Authors in reference (Jiang et al., 2019) have done on controlling attitude of quadcopter by using cascaded PID controller algorithm. They attempted to model the quadrotor's dynamic model using the Euler angle. Euler angle is not enough to express a dynamic model of quadrotor due to gimbal lock, this creates model error between actual properties and mathematical model of our system (Abeywardena et al., 2013). Also, they applied PID controller to control the attitude of the quadcopter and but PID controller's performance decreases when used in a nonlinear system, even though they linearize the model by considering hovering point as equilibrium, it is not only a single equilibrium. Another point is not regarded to be equilibrum. In this research, linearized model is estimated by Subspace system identification.

Authors in reference (Abeywardena et al., 2013) have worked on Quadrotor MAVs with Improved State Estimation. It is a modified version of the first reference because the implement extended Kalman filter estimation which is developed for nonlinear system firstly. Even though they implement EKF they own self-stated as their model has an error due to this estimated state also have error. This research reduces model error by using 4SID method.

Authors in (Sarim et al., 2019) estimates the state of the quadcopter by EKF. It uses a linearized model for state space model to apply EKF. This create model error in aerodynamic system and also error in estimated state. This research reduce model error by application of subspace identification.

Authors in reference (Martin & Salaün, 2010) have done base of adaptive controller on quadcopter. They tried to upgrade the quadcopter's model as possible and developed L1 control method of the quadcopter. In addition, the quadcopter's stability is improved by an X-configuration design (Thu & Gavrilov, 2017a). They also use the Euler angle to describe the dynamic model. Authors in reference (Liang, n.d.-a) have estimated the attitude of quadcopter orientation based on an EKF. The paper applies step input signal to test the performance of the system separately Gaussian noise disturbance to test the effect of noise, but in a real system disturbance and measurement can't be separated or treated separately. He also uses Euler angle to describe the dynamic system.

Authors in reference (Li et al., 2019) have done Experimental Results on Robust Attitude Control of a 3-DOF Helicopter under Wind Disturbances. Wind gusts can readily impair helicopter flight performance, especially when performing aggressive activities. A helicopter's attitude control performance can also be influenced by actuator input saturation. In order to address these issues, they first develop a helicopter dynamics model that takes input saturation constraints into account, and then use H∞ performance to construct a robust controller to improve the helicopter's wind resistance. Finally, the usefulness of the developed attitude robust controller is confirmed. The proposed control technique suppresses the negative impacts of wind disturbances better than the PID and LQR systems, according to experimental results. Overshoot is also included in this paper.

Authors in reference (Ding & Wang, 2018) have done on the robust controller for quadcopter with consideration of air disturbance. For stability management of an aerial robot quadrotor under wind gusts, a strong combat controller based on LDRAC is proposed. The Newton-Euler approach is used to create a nonlinear dynamical model of the quadrotor that takes into account wind disturbance. When the yaw angle is 90 degrees, the Euler formula is ineffective in describing the aerodynamic system (Roush, n.d.-b).

A quadrotor used in this research has four rotors. Two rotors rotate CW, while the other two rotate ACW. flight control is accomplished by varying the speed of each rotor, and hence the lift and torque. Pitch and roll are controlled by the net center of thrust, while yaw is controlled by the net torque. If all four rotors rotate at the same angular velocity, with two rotating CW and two turning ACW, the net torque at the yaw axis is zero. This eliminates the need for a tail rotor, which is required in quadrotors. If all four rotors rotate at the same angular velocity, with two rotating clockwise and two turning counterclockwise, the net torque at the yaw axis is zero (Thu & Gavrilov, 2017b). The configuration of quadcopter used in this study is show in Fig. 2.3 below.

Figure 2.3: A quadrotor's 'x' configuration *(Aboytes Reséndiz & Rivas Araiza, 2016)*

In its motion, The three major factors of a quadrotor are roll, pitch, and yaw. The quadrotor processing its speed based on roll, pitch, and yaw. The yaw axis starts at the center of gravity and runs perpendicular to the propeller and the fuselage reference line to the bottom of the quadrotor. The motion about the y axis is known as yaw motion (Dong et al., 2014; Jackson et al., 2008; Jennings et al., 2008). The quadrotor's nose moves to the right with a positive yawing action. The pitch axis (also known as the transverse or lateral axis) originates at the center of gravity and runs parallel to a line drawn from wingtip to wingtip. Pitch is the motion around y axis. With a positive pitching action, the quadrotor's nose rises and the tail descends. The roll axis (or longitudinal axis) is oriented forward and parallel to the fuselage reference line, with its origin at the center of gravity. Roll is the motion around x axis. A positive rolling motion lifts the left propeller and lowers the right propeller (Schon, 2006).

2.6 State-space quadrotor generalized attitude model

A state-space representation is a mathematical model that connects input, output and state variabele by first-order differential equations (Tahir et al., 2016). The term "state-space" refers to a space in which the state variables are the axes (Moin et al., 2017). Within that space, the system's state can be represented as a vector. The quadrotor attitude model dynamics can be given as (Liang, 2017). Let's consider the inputs that can be applied to the GAM to control the behavior of the quadrotor. There are three degrees of freedom and four propellers. The control inputs that will be discussed here are commonly used for vertical thrust across yaw and quadrotor control. The direction of proppeler's rotation are assumed as Fig. 2.4 below.

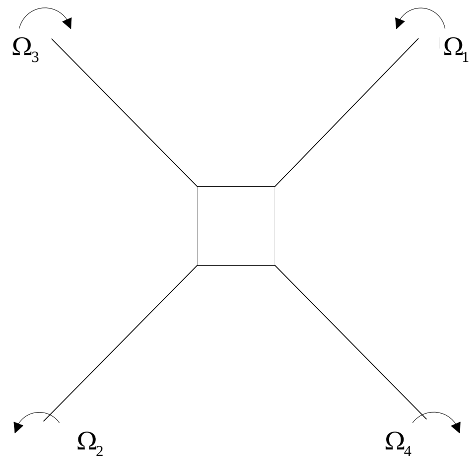

Figure 2.4: Quadcopter configuration

Let us consider the for the quadrotor yaw angle, quantities of input forces and torques proportionate to the squared speeds of the rotors (Liang, n.d.-a), The relation between input

$$\begin{aligned} M_1 &= k_t \sum_{i=1}^{4} \Omega_i^2 \\ M_2 &= k_t l(\Omega_1^2 - \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \\ M_3 &= k_t l(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \\ M_4 &= k_d(\Omega_1^2 + \Omega_2^2 - \Omega_3^2 - \Omega_4^2) \end{aligned} \tag{2. 1}$$

torque and speed of propellers is given as Eq. (2.1).

Where k_d the coefficient of drag factor is, k_t is the coefficient of thrust factor, l is the distance between any rotor and the quadrotor's center , $\Omega_1, \Omega_2, \Omega_3, \Omega_4$ are the four propellers' speed of the quadrotor. Equation of quadcopter angular velocity is given by Eq. (2.2)

$$\begin{cases} \ddot{\phi} = \left[M_2 + \dot{\theta}\dot{\psi}(J_{yy} - J_{zz}) / J_{xx} \right] \\ \ddot{\theta} = \left[M_3 + \dot{\phi}\dot{\psi}(J_{zz} - J_{xx}) / J_{yy} \right] \\ \ddot{\psi} = \left[M_4 + \dot{\phi}\dot{\theta}(J_{xx} - J_{yy}) / J_{zz} \right] \end{cases} \quad (2.\ 2)$$

where J_{xx} is the inertia of the quadrotor along the x-axis, J_{yy} is inertia along the y-axis, J_{zz} is inertia along the z-axis, M_2 is roll torque, M_3 is pitch torque, M_4 is yaw torque of the quadrotor attitude model inputs, and ϕ , θ , ψ are roll, pitch, and yaw angles respectively. The amount of change in angle during hovering $\dot{\phi}$ $\dot{\theta}$, and $\dot{\psi}$ is always very small and it is nearly zero. Finally, the three quadrotor attitude angle models can be obtained as in Eq. (2.3) (Liang, n.d.-b).

$$\begin{cases} \ddot{\phi} = \left[M_2 / J_{xx} \right] \\ \ddot{\theta} = \left[M_3 / J_{yy} \right] \\ \ddot{\psi} = \left[M_4 / J_{zz} \right] \end{cases} \quad (2.\ 3)$$

The generalized quadrotor attitude model in this research contains, the actuator, ZOH, quadrotor attitude model, and an observer. The actuator contains an ESC and motor (Chin & Lau, 2020). The ESC is used to regulate and control the quadcopter motor (Chamola et al., 2021). It regulates the speed by transferring current under constant voltage and pulse width modulation is chopping it to discrete parts and delivered to the motor. Hence propellers rotary motion will provide torque. ZOH is used for converting signal (Moir, 2022). The four propellers of the quadrotor are connected with motors. The simple mechanism of direct connection allows the quadrotor to obtain the force of rotating the fuselage by changing the motor speed through the electronic speed controller. For the quadrotor attitude yaw model of Eq. (2.3) above, the generalized quadrotor attitude model can be shown in the following figure.

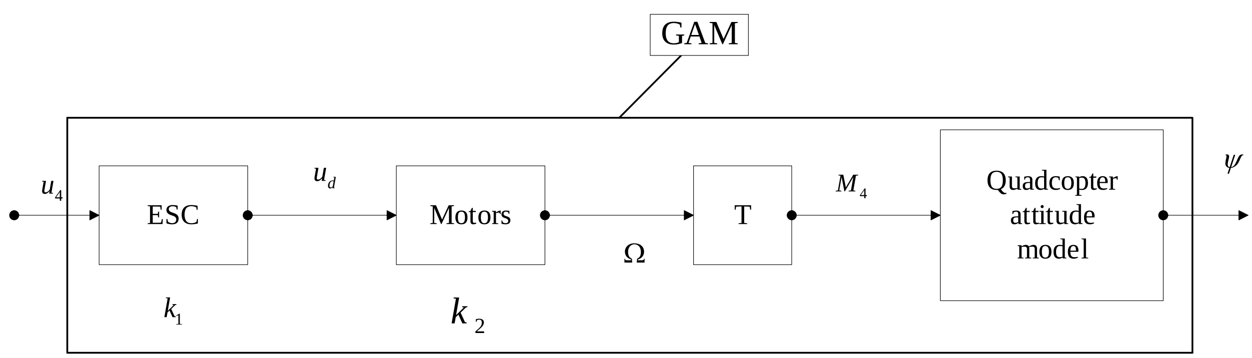

Figure 2.5: Generalized quadrotor attitude mode (Lin et al., 2021)

In Fig 2.5, U_4 is control quantity. The electronic speed controller $k_1$controls the motor speed by varying the voltage U_d on its armature and drives a motor k_2. Hence, by adjusting the duty cycle or switching frequency of pulse width modulation, the speed of the motor can be controlled and propeller speed can be produced. It needs to transform the four propeller speeds into torques for the axis of rotation of the quadrotor. Since we have four propellers it needs to transform propellers rotational speeds into torque. The quadrotor yaw attitude model is given as Eq. (2.4):

$$\frac{1}{J_{zz}s^2} \tag{2. 4}$$

Therefore, it is possible to give the transfer function of Fig 2.5. But, for simplification and to minimize redundancy, let's concentrate only on the yaw axis. Hence, for the given reference angle u_4 for the quadrotor to produce a yaw angle ψ. Hence, the quadrotor GAM can be given as Eq. (2.5) (Qin et al., 2021).

$$G(s) = \frac{\psi(s)}{u_4(s)} = \frac{k}{s^2} \tag{2. 5}$$

where $k \simeq k_d k_1^2 k_2^2 / J_{zz}$ whose value is estimated as 0.1.

According to the above transfer function, the state-space model of a quadrotors GAM for the yaw axis can be given by Eq. (2.6):

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \\ C &= \begin{bmatrix} 1 & 0 \end{bmatrix} \quad D = 0 \end{aligned} \tag{2. 6}$$

Using the state-space model in Eq. (2.6), the second-order state-space model can be expressed as follows:

$$\begin{aligned} x_1(t) &\triangleq \psi \\ x_2(t) &\triangleq \dot{\psi} \end{aligned} \tag{2. 7}$$

Where ψ output is yaw angle, $\dot{\psi}$ is its angular velocity of the quadrotor yaw-axis According to this, the state equation for the quadrotor yaw model can be given as Eq. (2.8)

$$\begin{bmatrix} \dot{x_1}(t) \\ \dot{x_2}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_1(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} u_4(t)$$
$$\psi(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_1(t) \end{bmatrix} + 0u_4(t) \qquad (2.\ 8)$$

Where $u_4(t)$ is inputs torque and $\psi(t)$ are yaw angle. From the above Eq. (2.8), it can be seen from the obtained attitude model that the structure of the three attitude angles are similar. Hence, my concern in this research is only on the yaw axis

2.7 Discretization of the quadrotor GAM

2.7.1 Discretization

Discretization is the process of transferring analog signal into discrete signal and reconstaracted signal is obtained by application of ZOH (Shim et al., 2022) (Song & Scaramuzza, 2022) and it is very important in this research because the quadrotor state-space model identified is in continuous state-space form. Hence, transferring the continuous state-space form of the quadrotor attitude model into discrete counterparts is crucial. This process is usually carried out as a first step toward making them suitable for observer design stated in the chapters. For the generalized quadrotor state-space model, it is possible to change it to discrete forms using continuous to discrete transformation techniques. Hence, the associated discrete state-space model can be given by Eq. (2.9):

$$\begin{cases} x(k+1) = Fx(k) + Gu_4(k) \\ \psi(k) = Cx(k) + Du_4(k) \end{cases} \qquad (2.\ 9)$$

Hence, the discrete state-space model of a quadrotor at the sampling period of $T = 0.1$ seconds is given by Eq. (2.10)

$$F = e^{AT} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \quad G = \int_0^T (e^{AT} B)\, dT = \begin{bmatrix} T^3/3 \\ T^2 \end{bmatrix}$$
$$C = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad D = 0 \qquad (2.\ 10)$$

Where T is the sampling period and F, G, C and D is a discretized state-space model of a quadrotor

2.7.2 Stability analysis

The main thing to consider here is whether the quadcopter is stable or not. Because it is useful in the next task. Based on Eq. (2.10), discrete characteristic equation is give as Eq. (2.11)

$$(z - 1)(z - 1) = 0 \qquad (2.\ 11)$$

Hence, since its pole is not located in the unit circle, the identified generalized attitude state-space model in Eq. (2.11) is said to be unstable.

2.7.3 Controllability and observability

The key to estimating the quadrotor attitude model state is to determine whether the system is controllable and observable, because open loop transfer function is unstable and stabilized by state feedback. It is the requirement for continuous system's total controllability. Based upon the identified quadrotor attitude model, the controllability can be checked using the following formula.

$$Q_C = \begin{bmatrix} G & FG \end{bmatrix} = \begin{bmatrix} 0.0005 & 0.0015 \\ 0.01 & 0.0100 \end{bmatrix} \qquad (2.\ 12)$$

Based on the result obtained in Eq. (2.12) determinant of Q_c is different from zero and it is equals to 0.162 and rank of 2 and the system is second order. Therefore, it is controllable. Hence state feedback control law will be designed in later chapters. A quadrotor attitude model needs to determine its observability based on Eq. (2.13).

$$Q_O = \begin{bmatrix} C^T & F^T C^T \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 0.1 \end{bmatrix} \tag{2. 13}$$

Based on Eq. (2.13), the determinant of Q_0 is equals 0.1. Here, we can see that the determinant is different from zero. The rank is 2 and the system is second order. Hence it is observable. Hence, it is possible to design observer in later chapters.

CHAPTER THREE

3. MATERIALS AND METHODS

3.1. Chapter Overview

This chapter deals with the methods used to complete this research work and estimation of the quadcopter's yaw axis states based on subspace-based state estimation, Kalman filter and Kalman smoother.

3.2. Materials

In this research work, software such as MATLAB, Microsoft office, Microsoft Visio, Excel and Math Type is used. MATLAB is the computing software that consists of technical toolboxes and Simulink. Microsoft office is used for editing the thesis documentation. Microsoft Visio is drawing software used to draw a variety of diagrams. Math Type is used for writing mathematical formulas and equations in Microsoft offices word. Excel is used for numerical analysis for end result and PowerPoint presentation tools and Mendeley is used for citation tool.

3.3 Methods

Fig 3.1 illustrates all of these study methods. Data is gathered after an examination of closely relevant publications and literature. For the purpose of identifying system modeling, the acquired data is evaluated.

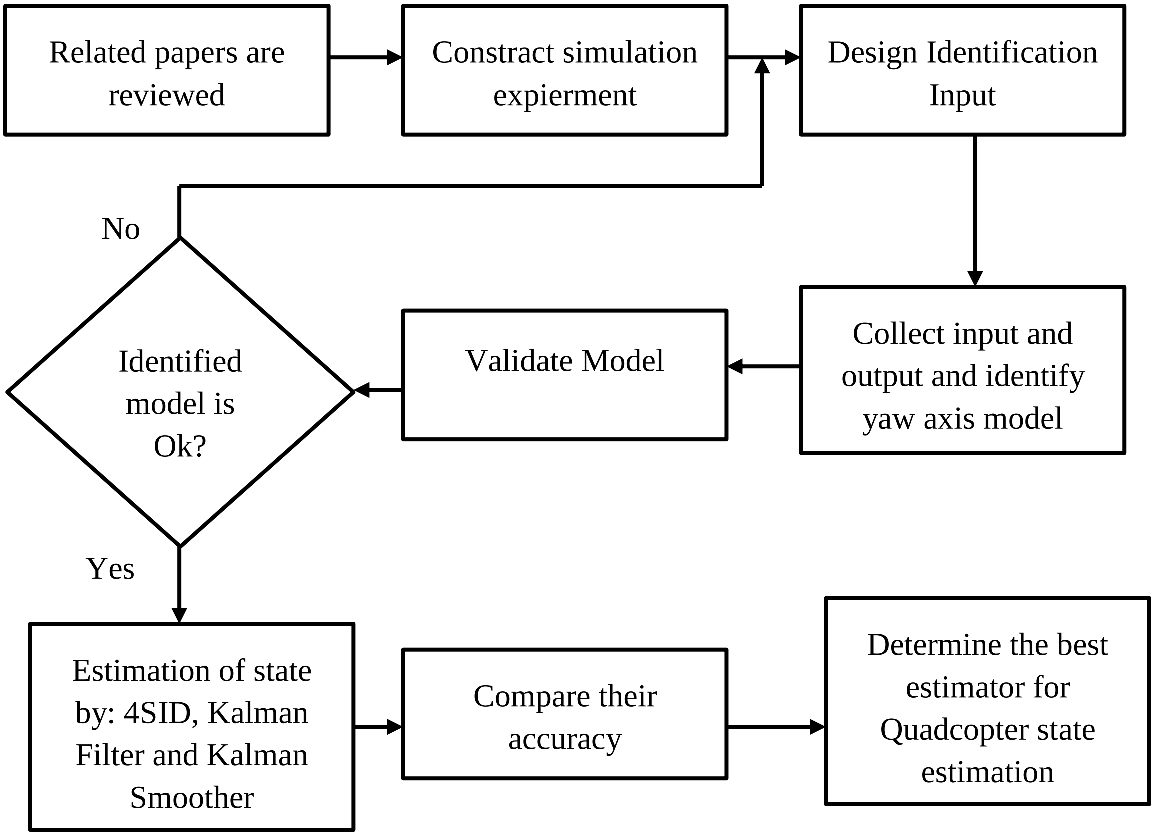

Figure 3.1: Research methodology

3.4 Related state estimation methods

3.4.1 Closed-loop Error in variables (EIV) model

A quadrotor GAM state under the closed-loop error in variable condition is problem that has arisen in many industries. Initiated by an emerging interest of estimation of state in the available errors in the inputs and output have been extended to handle the problem of estimating approximate quadrotor GAM states from the EIV system. In this research different estimation have been explored to review and compare their characteristic properties. 4SID method is applied to estimate state of GAM based on Error-in-variable block diagram in Fig 3.2, in which the quadrotor state-space model is stabilized by state feedback of discrete-time observer state-space model.

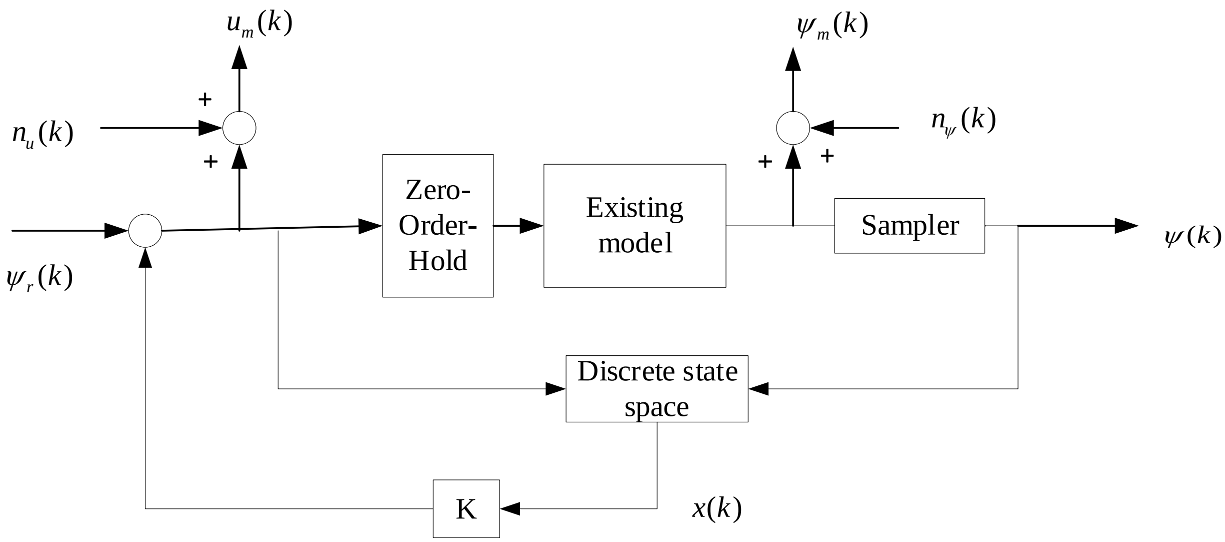

Figure 3.2: A closed-loop EIV system with an observer-based state feedback [106]

where ψ_r is the reference input angle for the quadrotor, and $\psi(k)$ is real attitude angle of the quadrotor to be estimated, $u_4(k)$ is the control quantity for yaw axis, $\hat{x}(k)$ is denoting the estimated state, and the disturbance noises in the input and output can then be given as $\psi_m(k)=\psi(k)+n_\psi(k)$ and $u_\psi(k)=u_4(k)+n_u(k)$ in which $n_u(k)$ and $n_\psi(k)$ are the measurement noises disturbing on the output $u_\psi(k)$ and $\psi_n(k)$ respectively. The main objective of this project is to estimate state $\hat{x}(k)$, by using input and output data in application of 4SID based state estimation. 4SID-based state estimation, Kalman smoothing and Kalman filtering will be applied. Since they are inapplicable without model, GAM for yaw axis should be identified. The identification problem is solved in the below flow diagram.

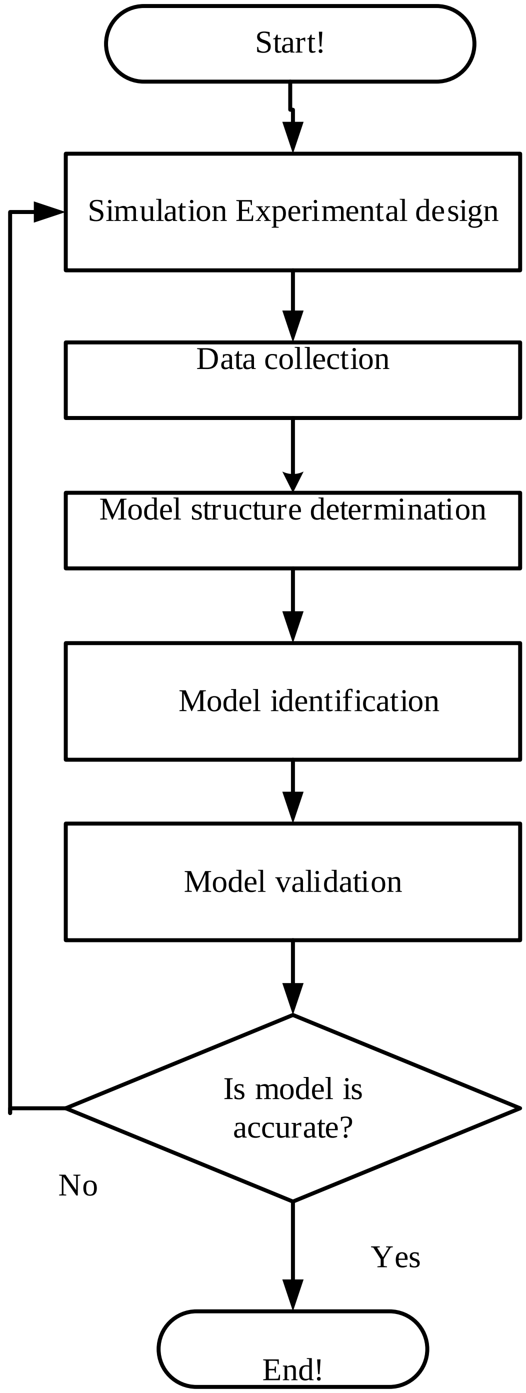

Figure 3.3: Identification procedure [L6].

The identified model accuracy is compared with existing model and use for further application, that mean estimators used the identified model to estimate state.

Models can also be constructed by one of the following method:

- Physical law
- Time series data
- From experimental data.

When a model is complex and inaccurate difficult to find, experimental data are used to obtain model in such cases (Salameh et al., 2015).

With the above procedure, there are two type of identification experiments. Open loop experiment and closed loop experiment. The closed loop experiment designed when (Wahlberg et al., 2010).

- If the plant is being monitored for safety reasons.
- Maintaining a high standard of production
- If an open loop causes the system to become unstable.

For this study closed loop expierment is applied and input signal is the reference angle the and output signal is measured yaw angle.

3.4.1.1 4SID-based estimation of state

The ability to produce correct state-space models for multivariable systems directly from input-output data makes the subspace identification-based state estimation method particularly appealing. Its aim is to estimate the state quadcopter attitude model in the presence of noisy. In this study, a 4SID-based state estimation approach for the closed-loop EIV system was suggested to estimate state when the PRBS is used as input signals rather than actual data for the quadrotor GAM state. Due to its effectiveness to estimate state space model by using input-output data which contaminated by noise 4SID-based state estimation is applicable (Coenen et al., 2019). Depending on the estimated quadrotor state-space model, it can extract and estimates states. Hence, the subspaidentification-basedsed state estimation method is very crucial to provide the state for a quadrotor attitude model. Then Kalman filtering and Kalman smoothing can be directly applied to the estimated quadrotor state-space model to estimates states. Estimation using Kalman filtering and Kalman smoothing cannot be applied if there is no model. Hence in this research, the subspace identification method can provide the quadrotor attitude state space model. Based on the estimated quadrotor GAM state-space model, subspace identification based state estimation, Kalman filtering, and Kalman smoothing methods can be applied to estimate the quadrotor GAM states. Hence, the model is important for Kalman filtering and Kalman smoothing to estimate state.

Model can be identified from Fig 3.2, by experimental measurements contaminating with output noises as well as state noise. Identified Coprime factor model is given by Eq. (3.1).

$$\begin{cases} x(k+1) = A_1 x(k) + B_1 \psi_r(k) + e(k) \\ \begin{bmatrix} u_\psi(k) \\ \psi_m(k) \end{bmatrix} = C_1 x(k) + D_1 \psi_r(k) + \begin{bmatrix} n_u(k) \\ n_\psi(k) \end{bmatrix} \end{cases} \tag{3. 1}$$

The covariance for e (k) and $\begin{bmatrix} n_{u_\psi} & n_\psi \end{bmatrix}^T$ are W and R. The matrices C_1 and D_1 in Eq. (3.1) are partitioned as Eq. (3.2):

$$\begin{aligned} C_1 &\triangleq \begin{bmatrix} C_{11} \\ C_{12} \end{bmatrix} \\ D_1 &\triangleq \begin{bmatrix} D_{11} \\ D_{12} \end{bmatrix} \end{aligned} \tag{3. 2}$$

3.4.1.1.1 Derivation of estimated state

A state variable representation of a system is not exclusive. There are infinitely many representations. Methods for transforming one set of state variables to another is discussed below. The way to transfer from one internal representation to another is called a similarity transformation (Coenen et al., 2019). To perform the transformation P, two different sets of state-space models Eqs. (3.3) and (3.4) can be given as:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases} \quad (3.\ 3)$$

$$\begin{cases} z(k+1) = Az(k) + Bu(k) \\ y(k) = Cz(k) + Du(k) \end{cases} \quad (3.\ 4)$$

$$\begin{aligned} z(k) &= Px(k) \\ z(k+1) &= Px(k+1) \\ z(k+1) &= P(Ax(k) + Bu(k)) \\ z(k+1) &= P(AP^{-1}z(k) + Bu(k)) \end{aligned} \quad (3.\ 5)$$

$$\begin{aligned} z(k+1) &= PAP^{-1}z(k) + PBu(k) \\ y(k) &= Cx(k) + Du(k) \\ y(k) &= CP^{-1}z(k) + Du(k) \end{aligned} \quad (3.\ 6)$$

The transformed matrices ($\overline{A}$, $\overline{B}$, $\overline{C}$, $\overline{D}$ of the new transformed state-space model can be given by Eq. (3.7):

$$\begin{aligned} \overline{A} &= PAP^{-1} \\ \overline{B} &= PB \\ \overline{C} &= CP^{-1} \\ \overline{D} &= D \end{aligned} \quad (3.\ 7)$$

For any nonsingular matrix P, these equations represent the same dynamical system, and hence, can regard these representations as equivalent. The input/output transfer matrix does not change as a result of the coordinate transformation. It can be simply possible to find the linear transformation matrix P as Eq. (3.8)

$$C_1 P^{-1} = C_0$$
$$P = C_0^{-1} C_1 \tag{3.8}$$

3.4.1.1.2 Balanced stochastic model truncation (BSMT)

The main aim of balanced stochastic model truncation in this research is to find a reduced order of the quadrotor state-space GAM which approximates the input-output behavior of the system. There may be a fairly complex large-scale state-space model. It may be difficult to obtain the reduced quadrotor state-space GAM. Hence, in model reduction, BSMT is very important. It is the most popular model reduction method. There are infinitely many state-space realizations for a given transfer matrix. However, its realization is a minimal realization of an asymptotically stable system (Abeywardena et al., 2013)

The implementation procedure for BSMT can be summarized as:

1) For a state-space model realization $\{A_1, B_1, C_1, D_1\}$ from Eq. (3.1), find the controllability grammarian P_C and observability grammariann Q satisfying Lyapunov and Riccati equations in Eq. (3.9) .

$$A_1 P_c + P_c A_1^T + B_1 B_1^T = 0$$
$$B_W = P_c C_1^T + B_1 D_1^T \tag{3.9}$$
$$QA_1 = A_1^T Q + (QB_W - C_1^T)(-D_1 D_1^T)(QB_W - C_1^T)^T = 0$$

2) The state-space model in step 1 and transform and partition of the state space model transformation to a balanced realization $\{A_1, B_1, C_1, D_1\}$ if and only if it is asymptotically stable and minimal.
3) Obtain subsystem realization (A_1, B_1, C_1, D_1) via BSMT. Then the realization (A_1, B_1, C_1, D_1) is balanced and stable.

3.4.1.2 Implementation procedure

The implementation procedure to obtain the continuous-time GAM from experimental data collection using the application of 4SID-based estimation of state method is shown in the Fig. 3.4 below. After state is estimated, by using different constrients the error has evaluated.

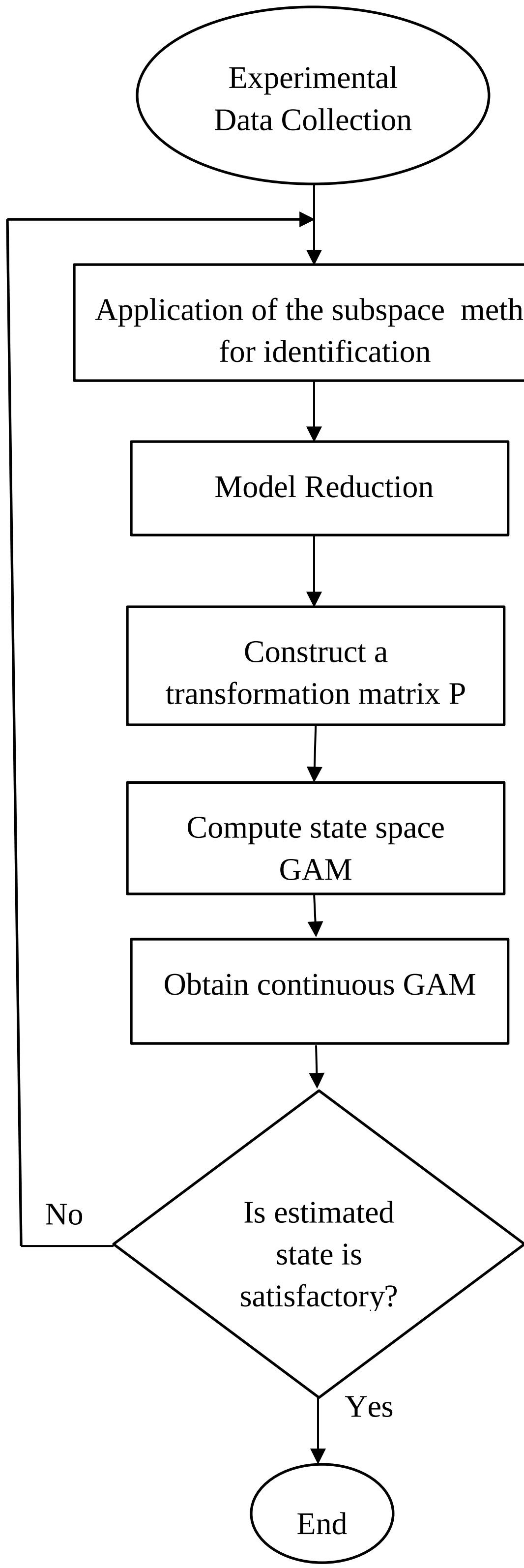

Figure 3.4: Subspace identification based state estimation implementation procedure.

3.4.2 Kalman filter

The Kalman filter aligns the model to the measurement and improves both the state and its parameters by lowering measurement noise and boosting their accuracy. It was developed by Rudolf Kalman, who founded the method to estimate dynamic states. It estimate dynamical systems states based on state-space format. The Kalman filter matches the model with the measurement and, in the process, improves both by reducing measurement noise and increasing the accuracy of the state and its parameters (Kurak & Hodzic, 2018; Welch, 2014; Zhu et al., 2019). Kalman filter is a recursive procedure that can provide the best state estimation for the quadrotor attitude model states. It is used to predict the current situation based upon the past event. The Kalman filter is a recursive algorithm for computing optimal state estimates given noisy observations and knowledge of the system dynamics. Due to its capacity to optimally estimate the system's error covariance and iteratively apply the prediction to improve the system measurements from time to time, the Kalman filter is stated to be an ideal estimator. Kalman filtering is an optimum estimator that infers parameters of interest from indirect, incorrect, and uncertain observations. It is a common technique used to tackle observer problems in control engineering. It's recursive, which means it can handle new metrics as they come in. Kalman Filtering is so popular in practice due to its optimality and structure. The Kalman filter requires system models to estimate the state. In this research, the model estimated by the subspace identification method. Therefore, estimation using Kalman filtering depends on the subspace identification method

3.4.2.1 Kalman filtering main iterative formulas

Applying the subspace identification method to an experiment for the closed-loop error in the variable system as in Fig.3.2, the discrete state-space model of the quadrotor attitude model is obtained and the transformation matrix and realized discrete state-space model of the quadrotor GAM can be estimated. Then, using the discretize to continuous transformation technique, we can obtain a continuous-time state-space model of the quadrotor. Then, predict the next state based on the obtained transformation matrix. Kalman filter keeps track of the estimated state of the system, and the variance or uncertainty of the estimate. To estimate the internal state of a set of noisy data, one must use the Kalman filter framework to model the process. The main Kalman filter recursive formulas(Kim & Bang, 2019; Kurak & Hodzic, 2018; Welch, 2014; Zhu et al., 2019), for an estimated coprime factor model of the quadrotor GAM in Eq. (3.1) is given by Eqs. (3.10)-(3.14).

$$x_{k|k} = x_{k|k-1} + K_k(y_k - C_1 x_{k|k-1} - D_1 u_k) \tag{3.10}$$

$$P_{k|k} = P_{k|k-1} - K_k C_1 P_{k|k-1} \tag{3.11}$$

$$x_{k+1|k} = A_1 x_{k|k} + B_1 u_k \tag{3.12}$$

$$P_{k+1|k} = A_1 P_{k|k} A_1^T + W \tag{3.13}$$

$$K_k = P_{k|k-1} C_1^T (C_1 P_{k|k-1} C_1^T + R)^{-1} \tag{3.14}$$

$x_{k|k}$ Is the filtered state, $P_{k|k}$ is the filtered covariance for $x_{k|k}$, $x_{k+1|k}$ is predicted state, $P_{k+1|k}$ predicted covariance for $x_{k+1|k}$ and K_k is Kalman gain, the input u_k, and the measurement output y_k is given by Eq. (3.15):

$$u_k \triangleq \psi_r \qquad y_k \triangleq \begin{bmatrix} u_m(k) \\ \psi_m(k) \end{bmatrix} \tag{3.15}$$

The quadrotor state-space model is very important to estimate the state using Kalman filtering and is nothing without a model. Hence, the quadrotor state-space model can be estimated using the subspace identification method is used. Depend upon the estimated state-space model of the quadrotor, the Kalman filtering estimate the state.

3.4.2.2 Implementation procedure

Kalman filtering algorithm can be implemented from the initial predicted state and covariance by setting k =1 for the given reference input angle $\psi_r(k)$ the outputs $u_\psi(k)$ and $\psi_m(k)$ as well as the predicted state $x_{1|0}$ and covariance $P_{1|0}$. Kalman gain must be updated from its starting values. As a result, the gain of Kalman filtering is updated according to Eq. (3.14). Then, Eq. (3.10) and (3.11) are used to update the Kalman filtering state and covariance estimate, while Eqs. (3.12) and (3.13) are used to update the projected state and covariance estimate. After that, update the data and get ready for the next cycle of implementation. Return to the step for Kalman filtering gain update and follow the procedure again if it does not come to the end of the data. Otherwise, the Kalman filter algorithm terminates. The implementation procedures for Kalman filtering generalized in the flowchart shown in Fig. 3.5. In this research initial covariance is taken based on standard deviation of noises.

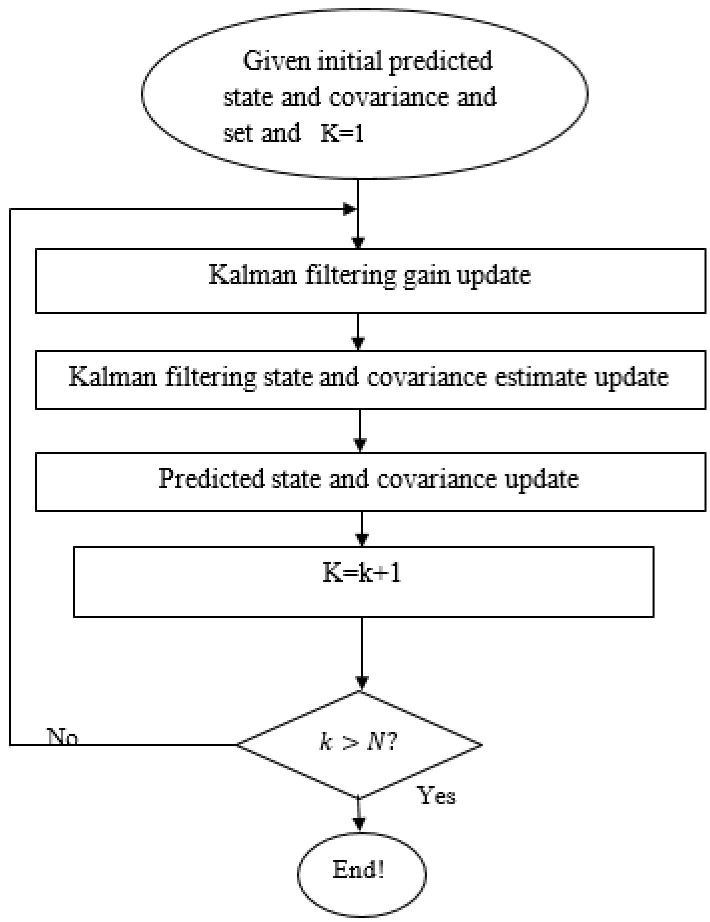

Figure 3.5: Kalman filtering algorithm implementation procedure

3.4.3 Kalman smoothing

Kalman smoothing is to predict the current state-based upon the past and future states (Aravkin et al., 2017; Fillion et al., 2020). Thus, smoothing does not provide current estimates at a time, but only estimates from past and future states. If there is a plant model, Kalman smoothing can produce estimates based on previous and future data. Because the model is required, this study estimates the model using the subspace identification approach. Based on the model derived using the subspace identification method, Kalman smoother estimate a model state. Hence, Kalman smoothing depends on the model estimated by the subspace identification method.

3.4.3.1 Kalman smoothing main iterative formulas

Based on the designed experiment in Fig. 3.2 and collect the reference inputs and errors in input and output with an appropriate sampling period. Then, identify the estimated attitude model of GAM. Obtain a transformation matrix and realize a discrete state-space model parameter. Estimate the realized quadrotor state-space model for the quadrotor yaw attitude model and obtain a quadrotor state-space model in discrete form. Then apply Kalman smoother to a quadrotor state-space model estimated in the available error in the inputs and output. In the beginning, the transformation matrix should be obtained. Hence, the quadrotor attitude model state can be estimated using the following main iterative formula. Kalman smoothing continue from equation (3.14) and Kalman smoothing main iterative formula (Barratt & Boyd, 2020; Kowalski & Smyk, 2018), are given by Eqs. (3.16)-(3.18) based on the estimated coprime factor model for the quadrotor GAM in Eq. (3.4)

$$x_{k|N} = x_{k|k} + S_k (x_{k+1|N} - x_{k+1|k}) \tag{3.16}$$

$$P_{k|N} = P_{k|k} + S_k (P_{k+1|N} - P_{k+1|k}) S_k^T \tag{3.17}$$

$$S_k = P_{k|k} A_1^T P^{-1}_{k+1|k} \tag{3.18}$$

$x_{k|N}$ is smoothed state, $P_{k|N}$ is the smoothed covariance for $x_{k|N}$, $x_{k|k}$ is the filtered state, $P_{k|k}$ is filtered covariance for , $x_{k+1|k}$ is the predicted state, $P_{k+1|k}$ is the predicted covariance for $x_{k+1|k}$ and S_k is the immediate matrix.

Without a model, Kalman smoothing is not possible. It is dependent on the state space model estimated using the subspace identification method in this study. The Kalman smoother can be used to estimate the state based on the estimated model.

3.4.3.2 Implementation procedure

Kalman smoothing algorithm implementation procedure can be applied to the predicted initial state and covariance by setting $k = 1$ for the given reference input angle $\psi_r(k)$ the outputs $u_m(k)$ and $\psi_m(k)$ the predicted state $x_{1|0}$ and covariance $P_{1|0}$. Since Kalman smoothing is dependent on Kalman filtering, use Eqs. (3.10) to (3.14) and initialize the smoothed state and covariance, and then set $k = N$. Next, the immediate matrix computation can be done using equations (3.18). Hence, Kalman smoothing state and covariance can be updated using Eqs. (3.16) and (3.17) and update data backward and verify that $k < 1$or not. Then, return to the immediate matrix computation and follow the procedure if it is not satisfied. Otherwise, Kalman smoothing comes to an end.

Estimating the state using Kalman smoother, needs a model. In this thesis, Kalman smoother depends on the model estimated for the generalized quadrotor GAM state. Then, depending on the estimated CF for GAM, Kalman smoothing can estimate the state. It is dependent on the model that has been identified in the presence of noise (Kurak & Hodzic, 2018). It can be concluded that Kalman smoothing is also nothing without a model. Kalman smoothing implementation procedure is described by Fig. 3.6

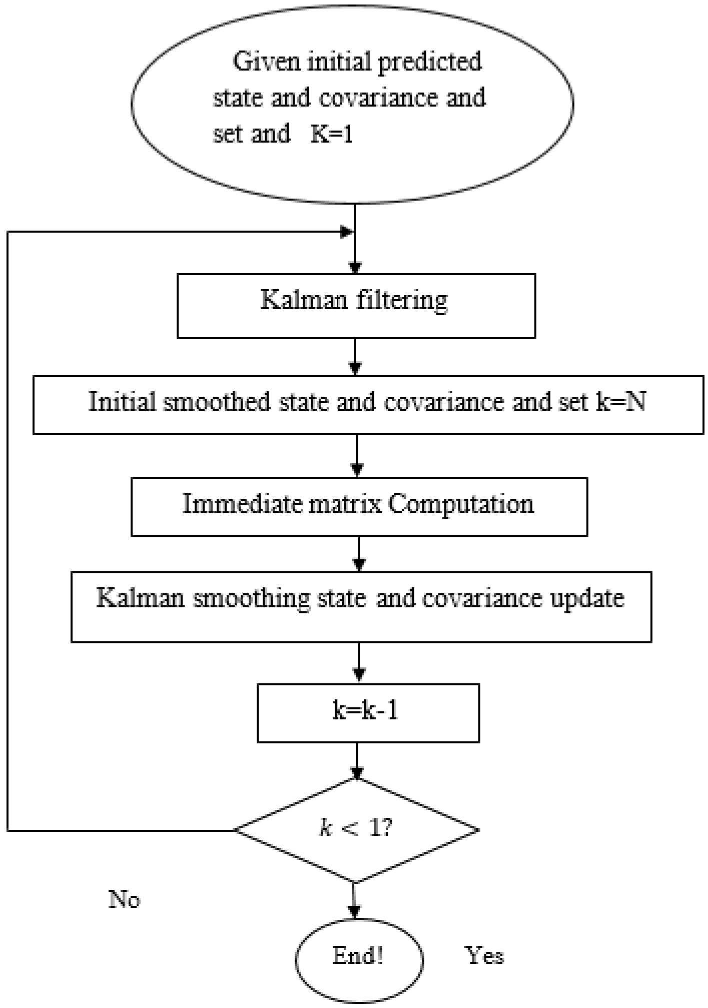

Figure 3.6: Kalman smoothing algorithm implementation procedure

3.5 Simulation Experiment Design for Quadrotor Attitude Model State

The goal of this chapter is to estimate the quadrotor GAM states. First, a linear GAM of the quadrotor state-space model is established using the subspace identification method (Qin et al., 2021), and the premise of an ideal mechanism arrangement is used to derive a linearized model. To estimate the GAM states, the reference attitude input is designed as a PRBS, and the subspace identification-based state estimation method is applied to estimate the quadrotor GAM state of the reference and measurements output.

3.5.1 Quadrotor attitude control system building with Matlab

Through the study of the quadrotor GAM in the above section, the axial model of the quadrotor system is obtained. To build a closed-loop attitude model control system it is a must to design feedback gain in the next section.

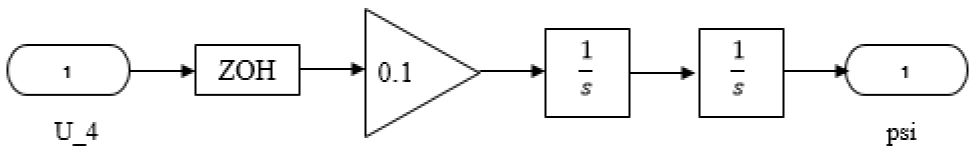

Figure 3.7: Quadcopter GAM for yaw axis

From Fig 3.7 u_4 is the control quantity for quadrotor GAM, ψ is the output yaw angle. Hence, it is possible to design a closed-loop quadrotor attitude by designing feedback gains as stated in the next section. A closed-loop attitude control design is very important. As shown in Fig 3.7, the system is an open-loop and it needs to design a feedback gain and change the system to a closed loop. Since, the other angles like a roll angle, a pitch angle is nearly similar to the yaw angle. This study focuses solely on the yaw angle.

3.5.2 Attitude control performance and poles

Control poles are selected by consideration of control performance to be achieved, which can be described by maximum overshoot and settling time. When the maximum overshoot % 0 and settling time $T_S = 1\ sec$. Hence damping ratio ζ can be given as;

$$M_p\% = e^{\pi\xi / \sqrt{1-\xi^2}} \qquad (3.19)$$

$$\xi = 1$$

The system is criticaly damped as it returns to equilibrium without overshoot. The value of ω_n can be calculated using the following formula

$$T_s = \frac{3}{\xi \omega_n} \qquad (\Delta = 5\%) \tag{3.20}$$

$$1 = \frac{3}{1 * \omega_n} \Rightarrow \omega_n = 3$$

The desired continuous control poles are selected as $(-3, -15)$

$$s_{1,2} = \xi \omega_n \pm j\sqrt{1 - \xi^2}\, \omega_n \tag{3.21}$$

$s_1 = \xi \omega_n = -3$ and $s_2 = \xi \omega_n = -15$

Pole 2 s is five times dominant pole 1. Then, it is possible to change these continuous-time desired control poles into discrete desired control poles according to the Eq. (3.22)

$$z_{1,2} = e^{T s_{1,2}} \tag{3.22}$$

$z_1 = e^{0.1*(-3)} = 0.7408$ and $z_2 = e^{0.1*(-15)} = 0.2231$

Hence, The observer must make an educated guess about states that cannot be directly viewed. Accordingly, the chosen observer poles in continuous time are $\{-60, -61\}$ which should be decay faster than control poles given in the next subsection. The desired continuous control poles can be changed to discrete form as by Eq. (3.23).

$$z_{3,4} = e^{T s_{3,4}}$$

$z_3 = e^{0.1*(-60)} = 0.0025$ and *(3.23)*

$z_4 = e^{0.1*(-61)} = 0.0022$

3.5.3 Observer design

An observer is also used to estimate the state of GAM of a quadrotor of a system. In this study, the observer's job is to provide a state estimate based on observations of the system's inputs and outputs. The observer employs a mathematical model of the quadrotor's state space realization. It can be designed as a discrete-time or continuous-time system. An observer is a dynamic system that estimates all of a system's states. A dynamic system's state variable is the variable that makes up the smallest set of variables that determine the system's state. Not all state variables are available in practice. As a result, it must estimate the values of state variables that are available. This kind of estimation is commonly called observation. Hence, this section will indicate how the closed-loop attitude control system will be designed according to the gain designed. But I have designed a feedback gain accordingly to build closed-loop attitude control specifically, for yaw angles. Designing observer-based state feedback is very important to come up with the quadrotor attitude model state. The observer is called state observer complete order if it observes all state variables of the system, regardless of whether some are available for direct measurement. Hence, for the quadrotor yaw model, designing an observer is very important. Previously in Section 2.8.3, it is proved that the quadrotor generalized attitude model is observable. As a result, an observer must be designed to estimate the quadrotor model states. Here in this section, observer-based state feedback is designed which is the main importance in this thesis because, the observer is used to provides a state estimate of the internal state of a given real system, from measurements of the input and output of the system. Based on the chosen observer poles in Eq. (3.23), the corresponding desired characteristics equation can be given as:

$$\begin{aligned} \beta^{*}(z) &= (z - 0.0025)(z - 0.0022) \\ &= z^{2} - 0.0044z + 0.0000055 \end{aligned} \quad (3.24)$$

We can have the following actual characteristic equation in Eq. (3.25)

$$\beta_L(z) = |zI_2 - (F - LC)| = \left| \begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix} - \left[\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \right] \right| \tag{3.25}$$

Using Eqs. (3.24) and (3.25) the observer feedback gain can be obtained as :

$$L = \begin{bmatrix} 1.9953 & 9.9528 \end{bmatrix}^T \tag{3.26}$$

An observer gain should be computed so that all the eigenvalues of $(F - LC)$ have negative real parts. Based on a discrete observer for the discrete GAM designed observer is gives by Eq. (3.27).

$$x(k+1) = Fx(k) + Gu_4(k) + L\left[\psi(k) - Cx(k)\right] \tag{3.27}$$

From Eq. (3.26) an observer gain matrix can be computed. Based on the results obtained above, the observer can be given as:

$$\begin{aligned} x1(k+1) &= x1(k) + T\,x_2(k) + T^3/2 + L_1\left[\psi(k) - x1(k)\right] \\ x2(k+1) &= x2(k) + T^2 u_4(k) + L_2\left[\psi(k) - x1(k)\right] \end{aligned} \tag{3.30}$$

As shown in Fig. 3.8, The control quantity u 4 is the input, while the ψ is yaw angle is the output. The observer's job is to create a state estimate based on measurements of the system's inputs and outputs. The observer employs a mathematical model of the system's state space realization.

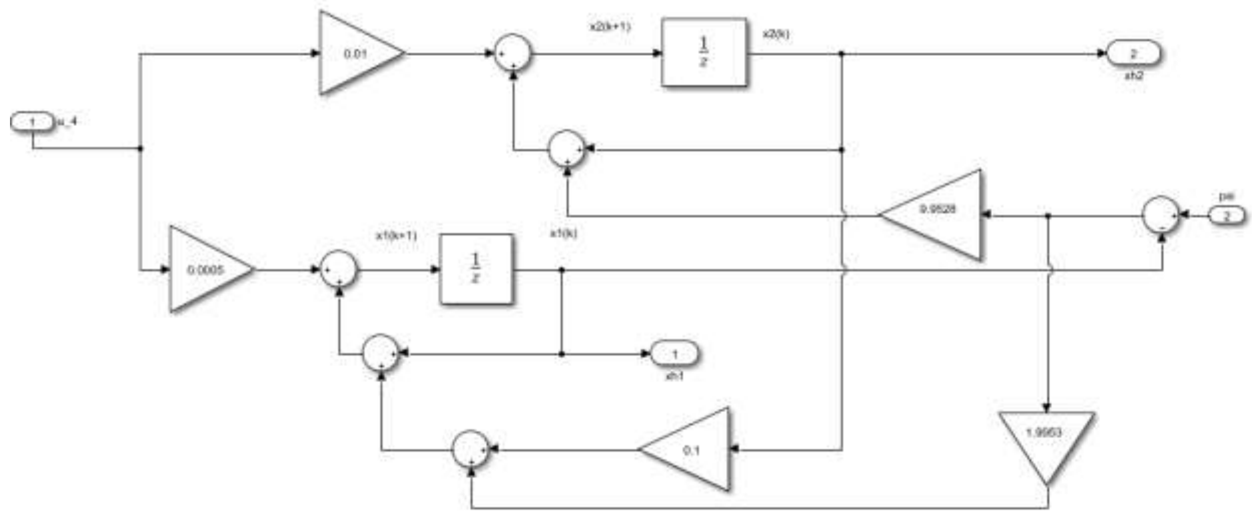

Figure 3.8: Observer design

3.5.4 State feedback law design and realization

The state is unknown to the designer. As a result, a state estimator must be built in such a way that it can give a state for feedback. We must implement state feedback in order to estimate the status. The controllability of the generalized quadrotor GAM was checked in Section 2.8.3. As a result, a feedback gain must be designed in this part. The pole assignment method is used to design a state feedback gain k based on the performance parameters.

The state feedback law can be achieved by the following equation

$$u_4(k) = \psi_r(k) - K\hat{x}(k) \qquad (3.29)$$

Where k is a state feedback gain matrix and $\hat{x}(k)$ is donating the state to be estimated.

Based on the discrete desired poles obtained in Eq. (3.22) the characteristic equation can be given as:

$$\alpha^{*}(z) = (z - 0.7408)(z - 0.2231)$$
$$= z^2 - 0.9639z + 0.1653 \qquad (3.30)$$

The state feedback gain can be calculated as:

$$\alpha_k(z) = |zI_2 - (F - GK)| = \left| \begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix} - \left[\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} T^3/3 \\ T^2 \end{bmatrix} \begin{bmatrix} K_1 & K_2 \end{bmatrix} \right] \right| \qquad (3.31)$$

$$K = \begin{bmatrix} 201.3505 & 93.5376 \end{bmatrix}$$

On the basis of the discrete state estimation from it thus is possible to realize the GAM state feedback control system using Matlab Simulink shown in Fig. 3.9. In the controller synthesis, the designed control gains are connected in closed-loop feedback by giving step input as shown in the following. After, the gain K is designed for the GAM, the state feedback control system model can be designed as shown in the following figure.

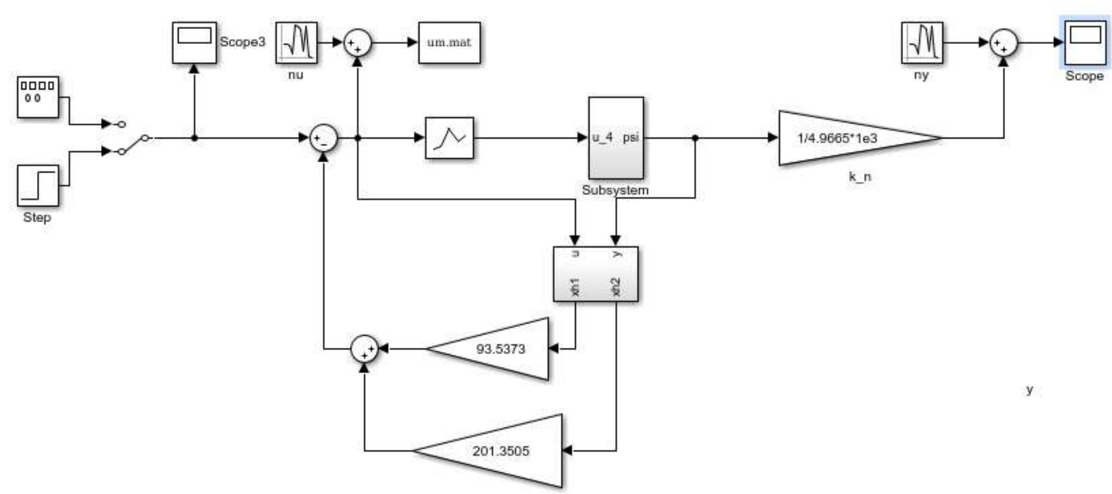

Figure 3.9: GAM state feedback control system model

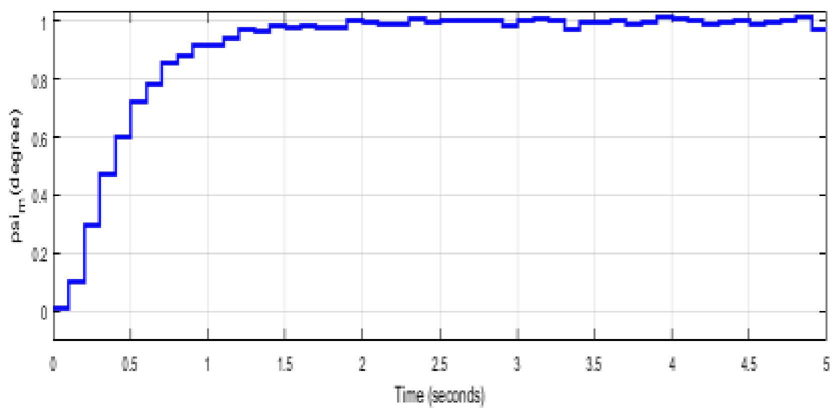

Figure 3.10: Step response

3.6 Design of a simulation experiment for quadrotor GAM identification

3.6.1 PRBS input design

The main task of the theory of identification is the collecting of maximum information which describes behavier of system accuretly. This task is completed by optimal experiment design. PRBS has many advantages when compared to other inputs. Advantages of PRBS are:

- ✓ Has the lowest crest factor.
- ✓ Frequency content can be changed by altering the clock sampling rate
- ✓ They possess white-noise-like properties.
- ✓ Can be generated by using a linear feedback shift register and the only disadvantage of PRBS is that only maximum length PRBS possesses the desired properties.

It is a Periodic binary sequence with a noise-like wave-form. PRBS is designed as a reference input signal for a quadrotor. Here a reference input and add a pulse signal to the input of the quadrotor by decreasing a pulse signal and observe the time and make the duty cycle and increase the frequency signal simultaneously if there is no change in the input value. Using the transition process T_s and high cut-off frequency f_{max}, it is possible to determine PRBS. Different paper have been published for identification by using different input. Table below show summarized articles (He et al., 2016; Evans et al., 1994; Stamati et al., 2012; Vuojolainen et al., 2017; Sarath et al., 2019; Sarim et al., 2019).

Authors	**Year**	**Input Used**	**Output command**	**Control Object**
Evans et al.	1994	Multisine	Shaft speed	Gas turbine
Ma et al.	2011	PRBS	Model identified	Servo system
Telen et al.	2012	White noise	Estimated model	Diesel air path system
Carangui et al.	2016	PRBS	Model identified	Quadcopter

Table 3.1: Review on Input for identification

He et al.	2016	Multisine	Angle	Aerodynamic system
Vuojolaininen et al.	2017	PRBS	Rotor Position	Active magnetic bearing
Eric T.Belsiki	2018	Multisine	-	Ganty axis
Yavad et al.	2019	PRBS	Level of Tank	Tank
Geng et al.	2020	PRBS	Angle	Quadcopter

The quadrotor GAM identification is a very important point in the quadrotor attitude model state estimation method. Because the state will be estimated after the model is estimated. So that a quadrotor GAM estimation is a prerequisite for the quadrotor GAM state estimation method. The data collection procedure from input to output is summarized below. The system reference input is provided and then the input and output data subjected to noise interference are collected to obtain the quadrotor GAM estimation data.

The design procedure is clearly explained when a reference signal PRBS is used as input. The design methods and steps can be summarized in the following points.

- After checking quadcopter control system is stable, connect pulse signal to the stable system and determine the time that pulse signal decreases. It is transition time Ts=1s.
- Connect square wave of 50 % of duty ratio to the system. Gradually change the frequacy of square wave upto no change on magnitude of output. It is maximum frequency f_{max} and $f_{max} = 5hz$

Giving step input, it is possible to get the following simulation curves

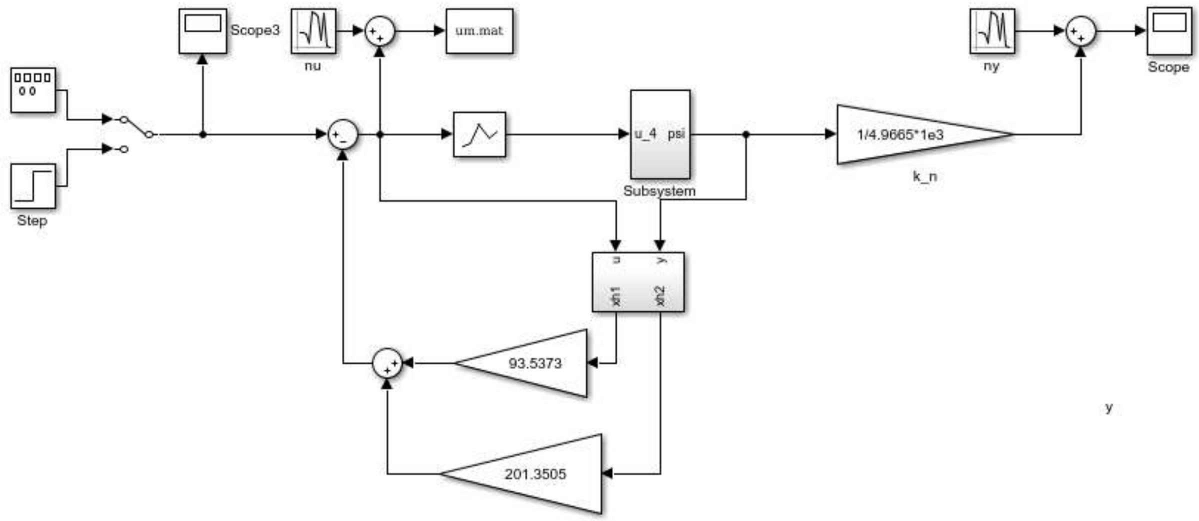

Figure 3.11: Maximum frequency determination model

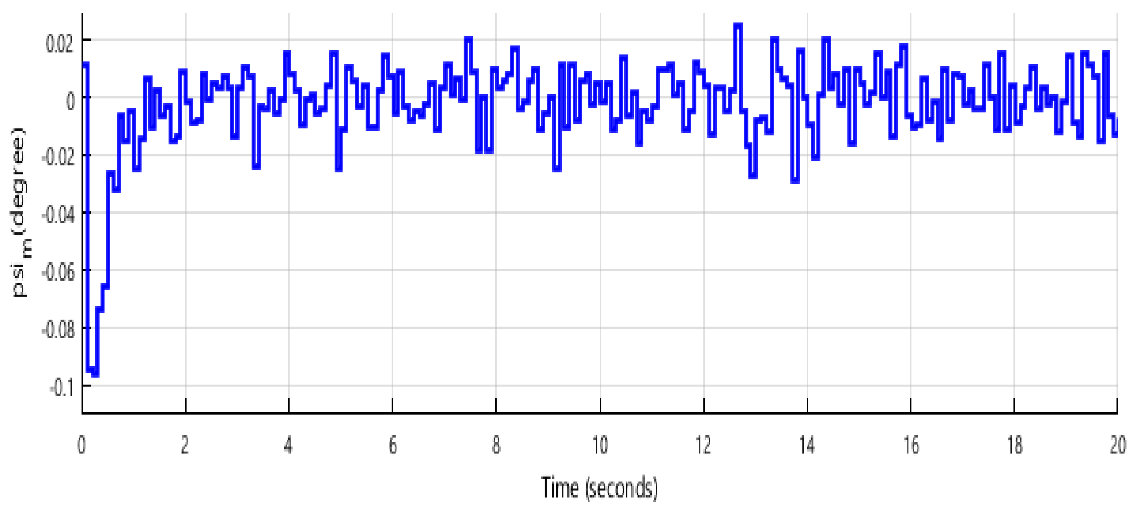

Figure 3.12: Simulation curves for a maximum frequency

After several simulation experiments, the maximum frequency max f = 5 Hz is obtained using Matlab simulation. Hence it is possible to calculate pulse interval as:

$$\frac{1}{3\Delta t} \geq f_{\max} \tag{3.32}$$

$$\vartriangle t \le 1/3f_{\max} \approx 1/3*5 \approx 0.1s \quad (3.33)$$

Where $\vartriangle t$ is the pulse interval, N_p is the number of cycles, and $f_{\max}$ is the highest operation frequency;

$$\begin{aligned}
&(N_p - 1)\vartriangle t \gg T_s \\
&N_p \gg \frac{T_s}{\vartriangle t} = \frac{1}{0.1} + 1 = 11 \\
&N_p = 2^n - 1 = 15 \gg 11 \\
&N_p = 15
\end{aligned} \quad (3.34)$$

3.6.2 Reference input and output measurements

To estimate a quadrotor GAM state, a pseudo-random binary input signal must be designed. Therefore, according to the concept, PRBS can be designed as a reference input signal for a quadrotor error in a variable system, and all of the design methodologies and phases are outlined above. Due to the influence of factors such as disturbances in the input and output in the actual experiment, these will cause excessive errors in the collected data, failing to estimate or excessive errors in the estimate of quadrotor GAM state for the quadrotor state-space parameters. Therefore, a large number of experiments are very essential in this estimation process. The first step is to design the reference input of the quadrotor GAM state. As in Fig. 3.13, PRBS given to the system as input, and then inconsideration of noise, input and output data are collected to obtain the quadrotor GAM identification data.

Because of disturbing input, input and output noise are considered as identically independent Gaussian noises distributed normally with standard deviation 0.01 and zero mean.

According to determined transition time Ts and maximum frequency f_{max}, 66-period of PRBS with a clock time of 0.1 seconds is designed.

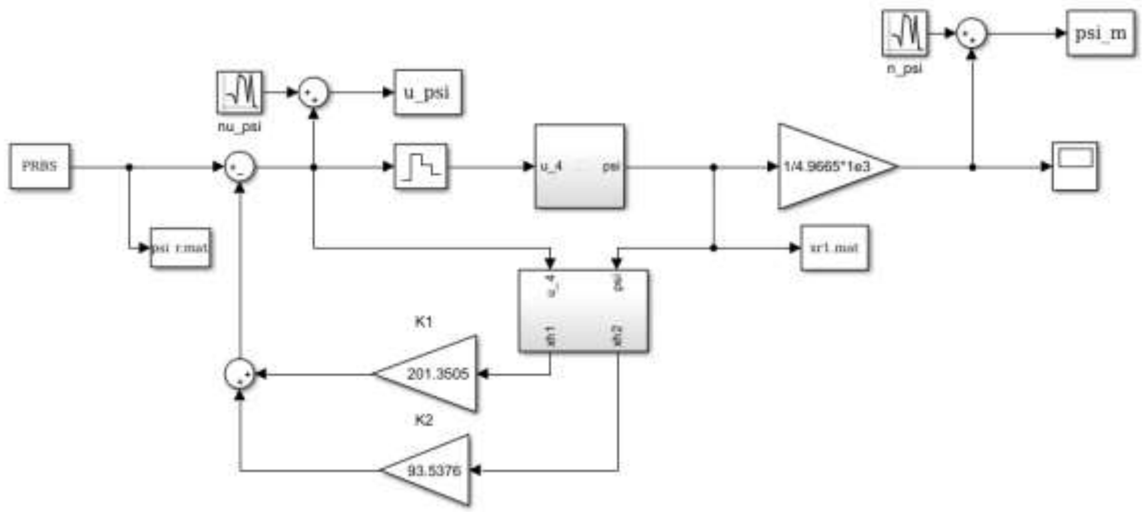

Figure 3.13: The overall quadrotor yaw attitude model

The obtained result and the given quadrotor state-space model are very nearest to each other. Calculated final prediction error is minimum and its value is 0.000103955. This is due to disturbances at the input and outputs. But the error is very small. Hence subspace identification-based state estimation is very satisfactory to estimate quadrotor attitude model state.

CHAPTER FOUR

4. RESULT AND ANALYSIS

4.1 Application of the estimation method

4.1.1 Application of 4SID-based state estimation method

Model is essential to estimate the quadrotor GAM state. Since the quadrotor is maneuvered by remote control, commands sent is the system inputs. For this research input command is desired yaw angle. After the application of the proposed model state estimation steps in which the zero-order hold for the continuous-time system, an estimation model for the yaw can be found. The quadrotor GAM obtained in this research is nearly similar to the given quadrotor state-space model. In present study, 4SID-based state estimation method is used to estimate the state of quadrotor GAM. Because it is important to estimate states when there are errors in the inputs and outputs.

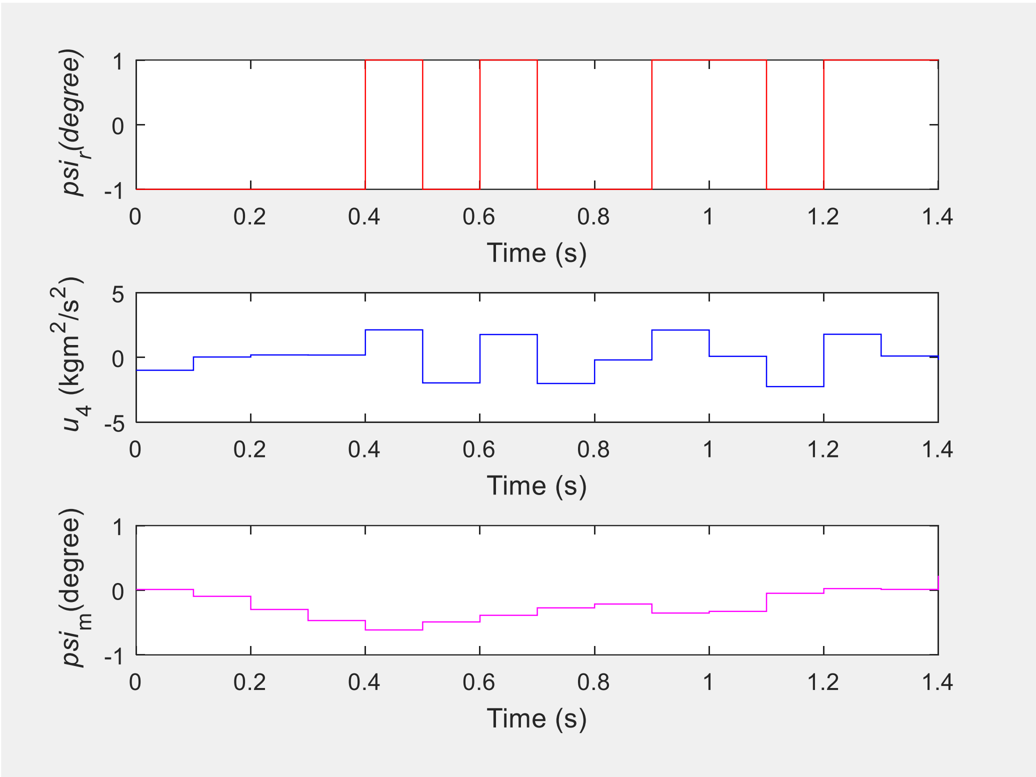

Figure 4.1: A quadrotor input and output angle for yaw axis

But it is more difficult to estimate errors in the input and output using either Kalman filtering or Kalman smoothing. Hence 4SID-based state estimation method will overcome this problem. When the designed input is connected with the quadrotor GAM in noisy envarioment and when the state feedback gain designed is connected as feedback. The continuous-time quadrotor state-space model, the observer and state feedback gain can also be designed and given as feedback to estimate state. The input is also very important for the quadrotor model state.

Present work use ‘iddata’ to create identification data to encapsulate the input-output data. To create data, the input, output, and sample are very important. Then using ‘n4sid’ to estimate a discrete-time quadrotor state-space model using identification data of 990 samples for a different number of states. The programs is already stated under appendix. To estimate a continuous-time state-space model, the sample time 0.1 second taken from the experimental data is used and ‘n4sid’ specifies the estimation options. These options can include the initial states, estimation states and subspace identification-based state estimation method. Here, it can estimate the state for a quadrotor GAM state based on its state-space model and compare its response with the measured output. The other main important point in this research is to get the transformation matrix P. Hence, the transformation matrix can be obtained from experimental data and real data. Hence C_0 can be obtained from coprime factor as:

$$C_0 = \begin{bmatrix} C - DK \\ -K \end{bmatrix} \triangleq \begin{bmatrix} 1 & 0 \\ -201.3505 & -93.5376 \end{bmatrix} \tag{4.1}$$

Applying 4SID-based state estimation method, on the experimental data, the state-space model obtained C_1 is:

$$C_1 = \begin{bmatrix} -0.0001 & -0.0001 \\ 0.0232 & -0.0106 \end{bmatrix} \tag{4.2}$$

From Eq. (3.8), the transformation matrix can be obtained as:

$$P \triangleq \begin{bmatrix} 1 & 0 \\ -201.3505 & -93.5376 \end{bmatrix}^{-1} \begin{bmatrix} 0.0001 & 0.0001 \\ -0.0232 & 0.0106 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ -2.1526 & 0.0107 \end{bmatrix} \begin{bmatrix} -0.0001 & -0.0001 \\ 0.0232 & -0.0106 \end{bmatrix} \tag{4.3}$$

$$= 10^{-3} \times \begin{bmatrix} -0.1 & -0.1 \\ 0.4635 & 0.1018 \end{bmatrix}$$

4SID-based state estimation uses the model estimated. The quadrotor state-space model is very basic in state estimation. Using the 4SID method, the quadrotor state-space model can be estimated(Qin et al., 2021), the next step will be the state estimation based on the estimated quadrotor state-space model. Then, first, estimate the state-space model in a discrete form and change it to a continuous time state-space form. Then estimate the GAM state by using the 4SID-based estimation method. If the estimation is not fitting, again check the realized state-space model and if the reference state is fitting with the estimated state, the result will come to an end.

As shown in the above Fig. 3.13, first give the excitation input signal PRBS for GAM according to the procedure is clearly explained in Fig.4.2 and collect the identification experimental data. The CF state-space model can be derived to obtain the transformation matrix for the GAM. Then after it is possible to obtain the GAM state-space model in a discrete-time state-space form. The implementation of 4SID-based state estimation method is clearly shown in Fig. 4.2. The higher-order estimation of CF for GAM can be obtained by considering the involved state observer and then a resulting quadrotor estimation state is produced by the BSMT model reduction method. The estimation accuracy for fitting is high up to 99.01 % and this estimation model is ready for its later Kalman filtering and Kalman smoothing. Mainly subspace identification-based state estimation method plays a major role in the extraction of states in this research. The results obtained using the Matlab simulation are given in Fig.4.3.

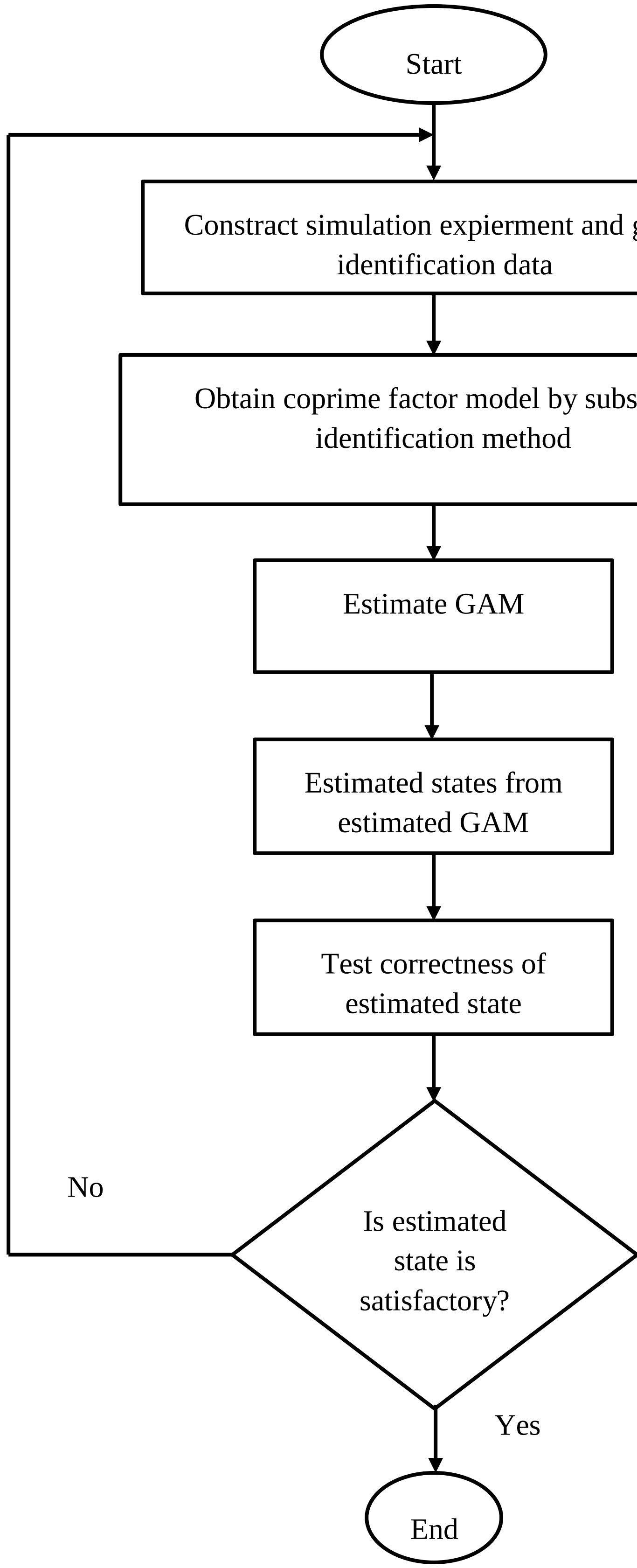
Start
Constract simulation expierment and gather identification data
Obtain coprime factor model by subspace identification method
Estimate GAM
Estimated states from estimated GAM
Test correctness of estimated state
Is estimated state is satisfactory?
No
Yes
End

Figure 4.2: 4SID-based state estimation implementation procedure

In Fig. 4.

Based on the simulation result obtained in Fig. 4.3, the estimated state x_{e1}, x_{e2} are very nearest to the true state, x_1, x_2.

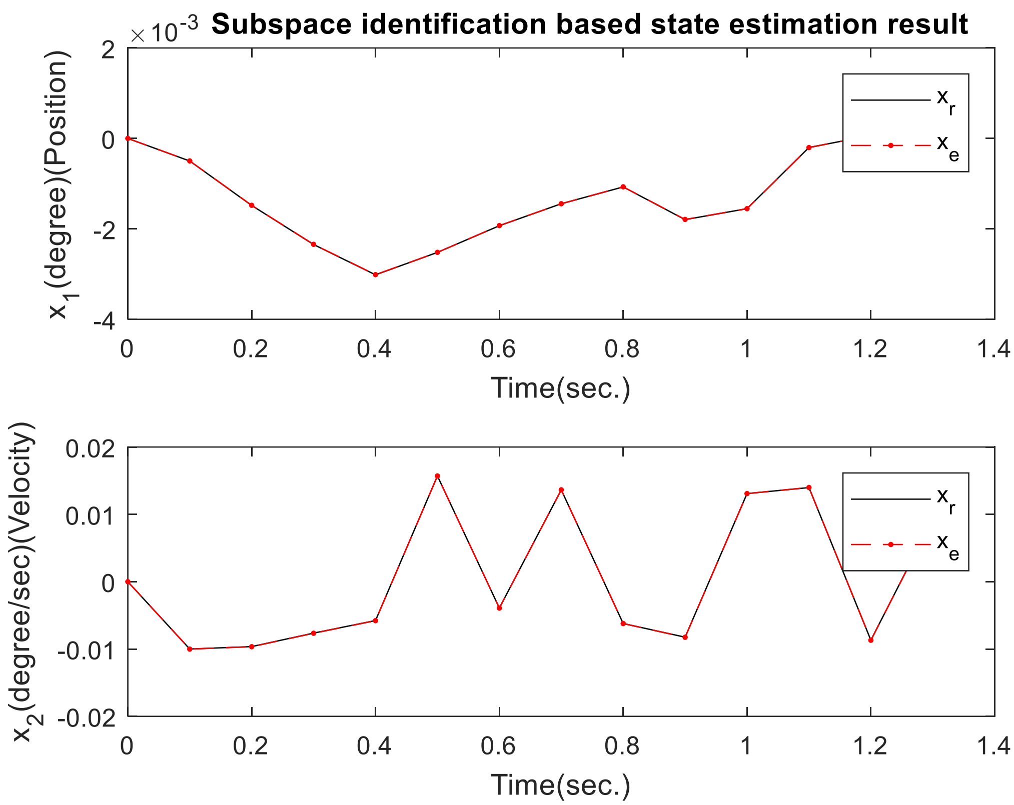

Figure 4.3:Subspace identification based state estimation result

4.1.2 Application of the Kalman filter

To estimate the state using Kaman filtering, the model is very important. Hence, the model estimated by the 4SID-based state estimation method is applied in this research.

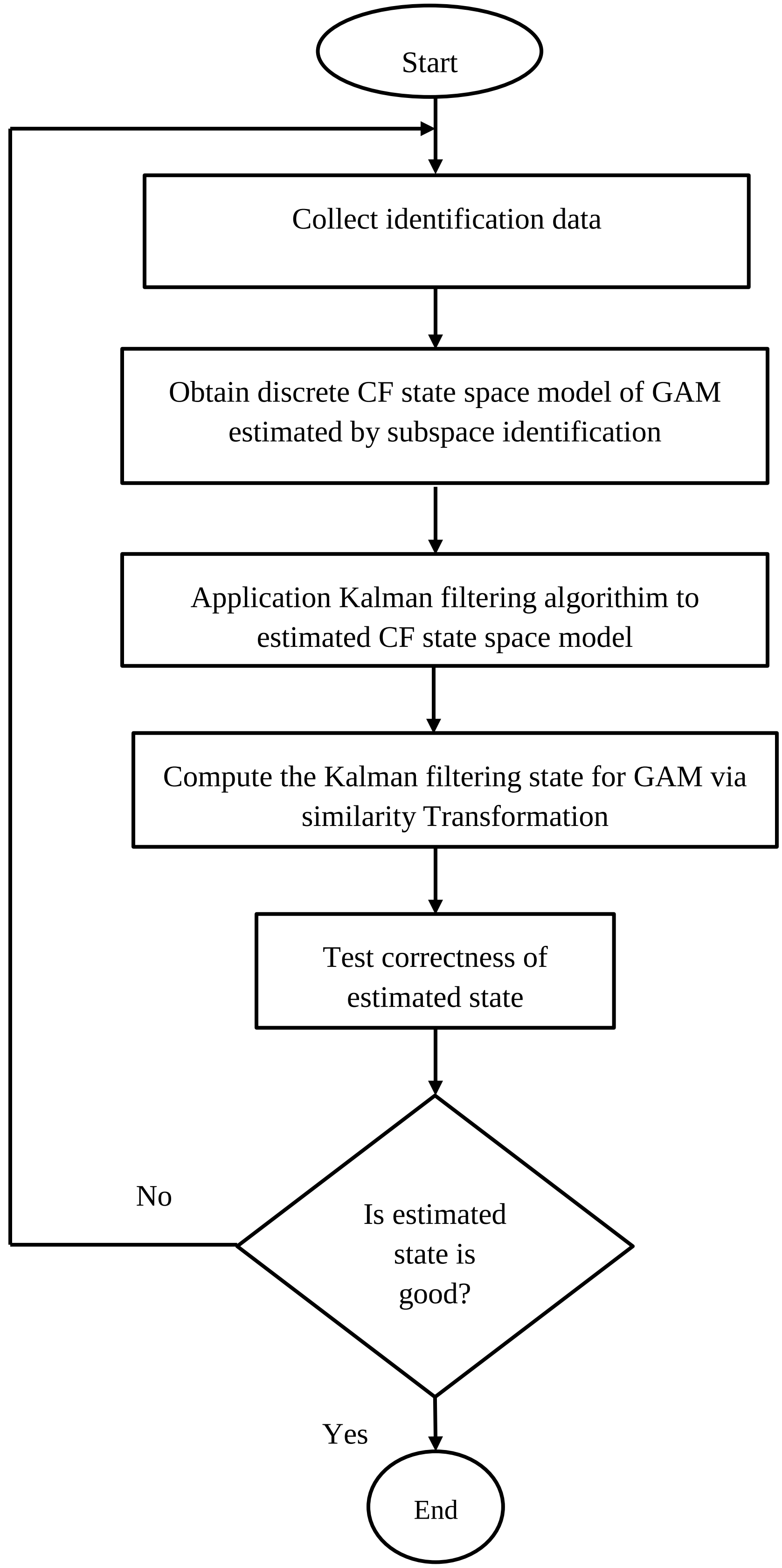

Figure 4.4: Kalman filtering implementation procedure summary

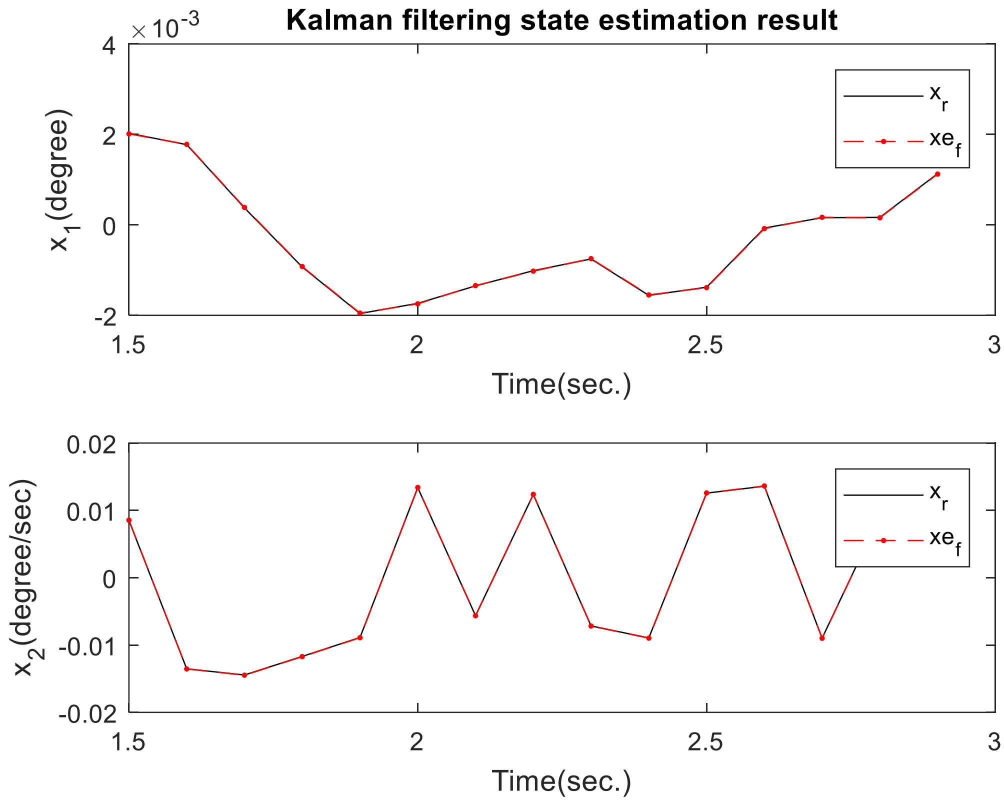

Figure 4.5: Kalman filter state estimation result

Generally, based on the model estimated using the subspace identification method, it is possible to apply Kalman filtering to estimate the quadrotor attitude model state. Because the Kalman filter is nothing without a model. Then using different norms, comparation of the estimated state will be done in Section 4.2. Depends on the results of Kalman filtering obtained the filtered state and the true state can be compared. But, looking at Fig.4.5, x_1, x_2, are the true states, and x_{ef1}, x_{ef2} are estimated states for Kalman filtering.

4.1.3 Application of the Kalman smoother

To estimate the quadrotor GAM state using Kaman smoothing, the model estimated using the subspace identification method is very important. Based on that model, Kalman smoothing can be applied to estimate the quadrotor GAM state.

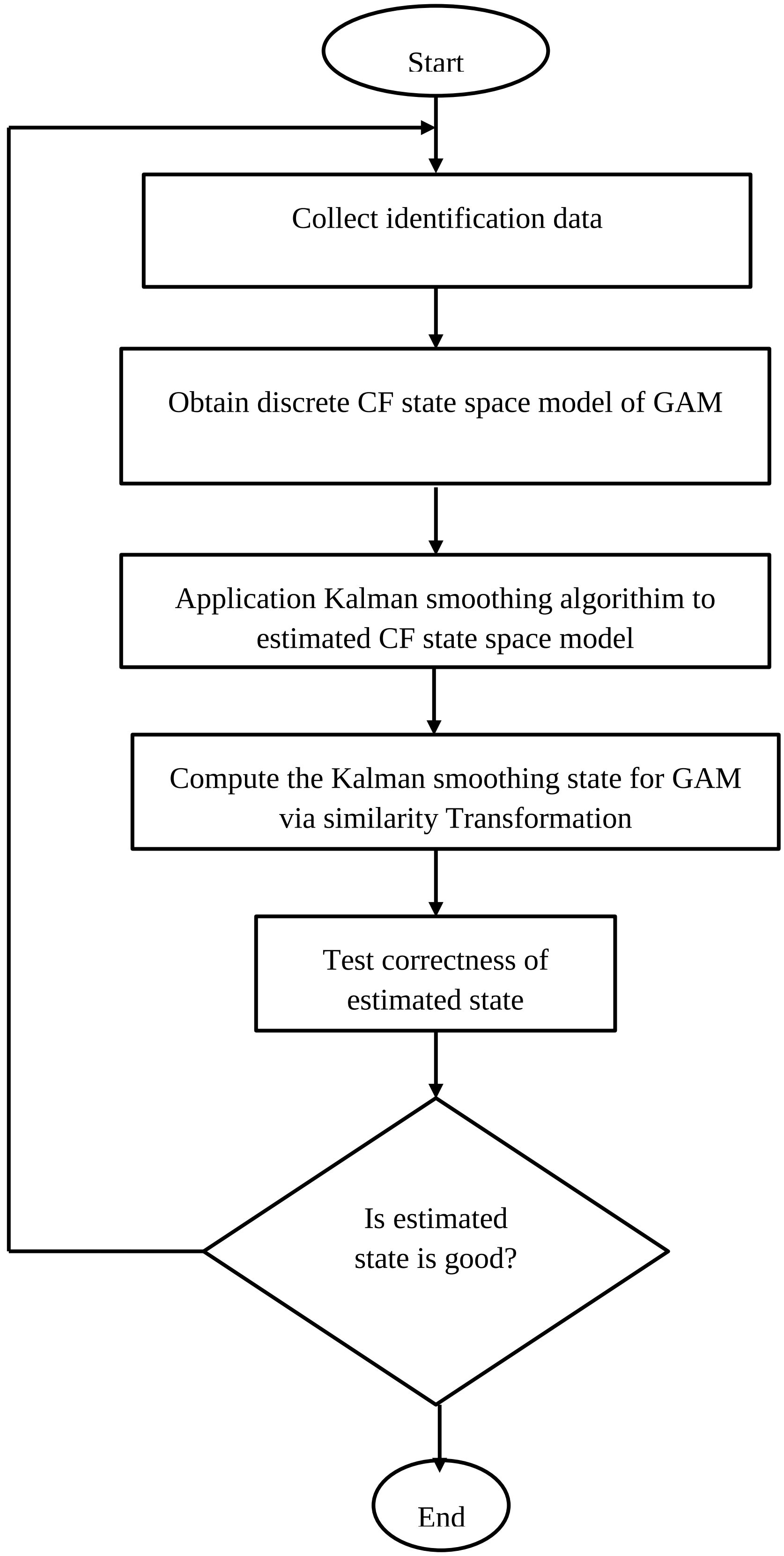

Figure 4.6: Kalman smoothing implementation procedure summary

Generally, to do estimation of state for the quadrotor GAM using Kalman smoothing, the model is important. Hence, the researcher used the model estimated using the subspace identification method and apply Kalman smoothing to estimate quadrotor GAM states. Then using different norms and he compare the estimated state with the true states. According to the procedures in this section, it is possible to generalize the implementation procedures of Kalman smoother using the flowchart below. As it is seen from Fig.4.7 below, the results of Kalman smoothing obtained are not nearest to the true state. Numerically it is shown in Table 4.1.

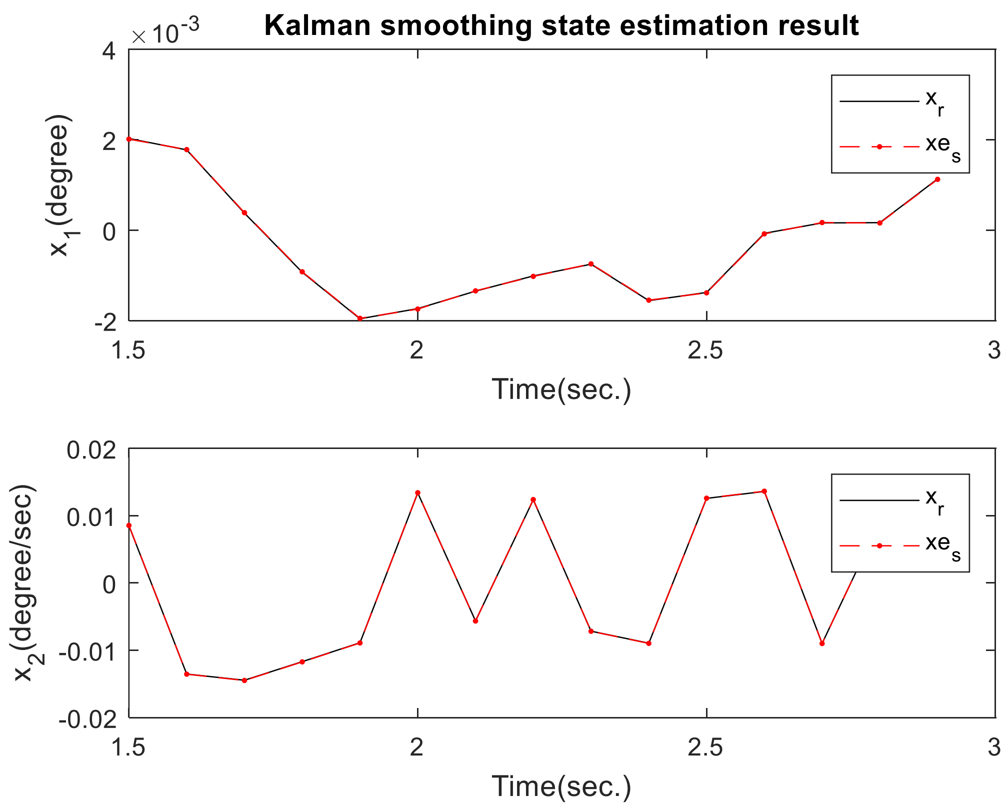

Figure 4.7: Kalman smoother state estimation result

In the above Fig.4.7 x_1, x_2, are the true states and x_{es1} , x_{es2} are Kalman smoothing estimated states.

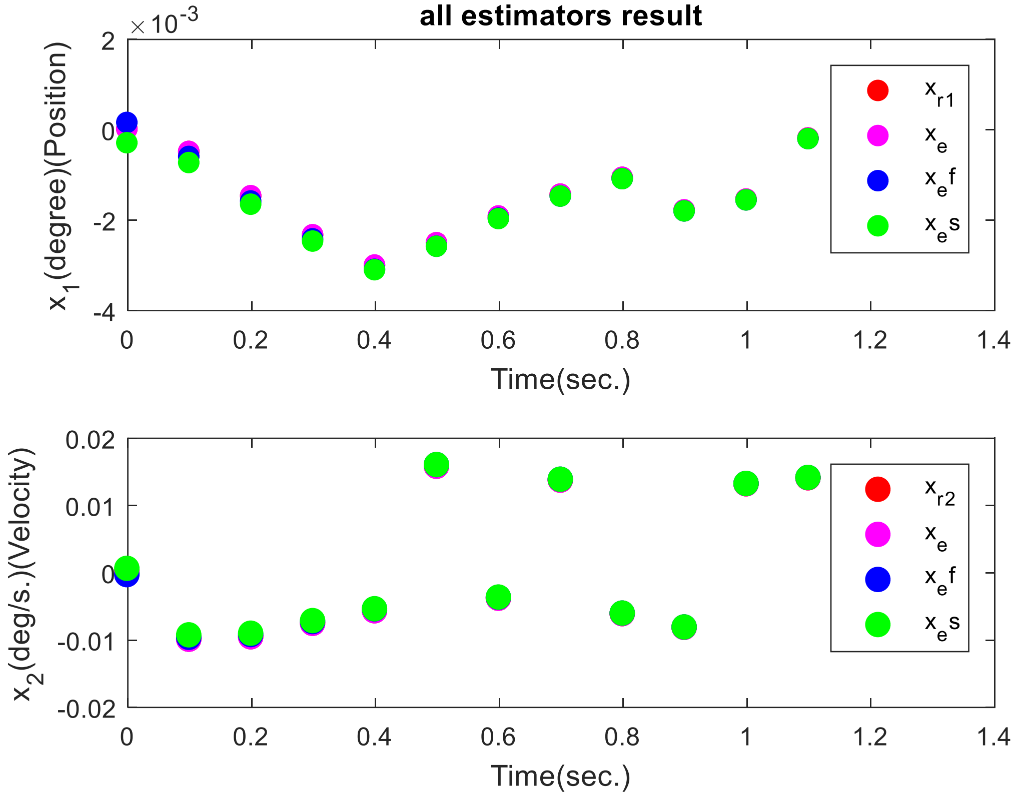

Figure 4.8: All estimators simulation

4.2 Comparison of the state estimation accuracies

A quadrotor state is estimated using the 4SID-based state estimation method, Kalman filtering, and Kalman smoothing in the previous sections. In this section, the results obtained using Matlab simulation will be compared with the true state. The comparison is done using different forms like SSE, MAE and MRE.

$$SSE \triangleq \sum_{k=1}^{N} \left[x(k) - x(k) \right]^2 \qquad (4.4)$$

$$MAE \triangleq \max_{k} \left| x(k) - x(k) \right| \qquad (4.5)$$

$$\mathrm{MRE} \triangleq \max_{k} \frac{\left| \mathrm{x(k)} - \mathrm{x(k)} \right|}{\left| \mathrm{x(k)} \right|} \quad (4.6)$$

Where $x(k)$ is the true state, $x(k)$ is estimated states and, MAE, MRE and, SSE are norms (Datong et al.,2018,; Davis et al., 2017; Itakura et al., 2019). Using the above norms, the comparison of estimated states with true states after long computation is listed in the table below.

States	Norms		
$x(k)$	SSE	MAE	MRE
x_{e1}	1.33562×10^{-9}	0.003270758	0.309855
x_{e2}	1.03867×10^{-8}	0.00003625	0.004759
x_{ef1}	7.10015×10^{-8}	0.000143378	0.183952
x_{ef2}	8.85898×10^{-7}	0.000416458	0.046123
x_{es1}	2.24542×10^{-7}	0.000307197	0.323202
x_{es2}	1.3016×10^{-6}	0.000627492	0.06695

From the state estimation comparison of subspace identification-based state estimation method, Kalman filtering, and Kalman smoothing estimation result is clearly shown in the Table 4.1, subspace identification based estimated states, Kalman filtered estimated states and Kalman smoothed estimated states have different norms. Based on the results obtained, the

Table 4.1: Assessment of state estimation using different norms.

subspace identification-based state estimation method and Kalman filtering show better estimation results.

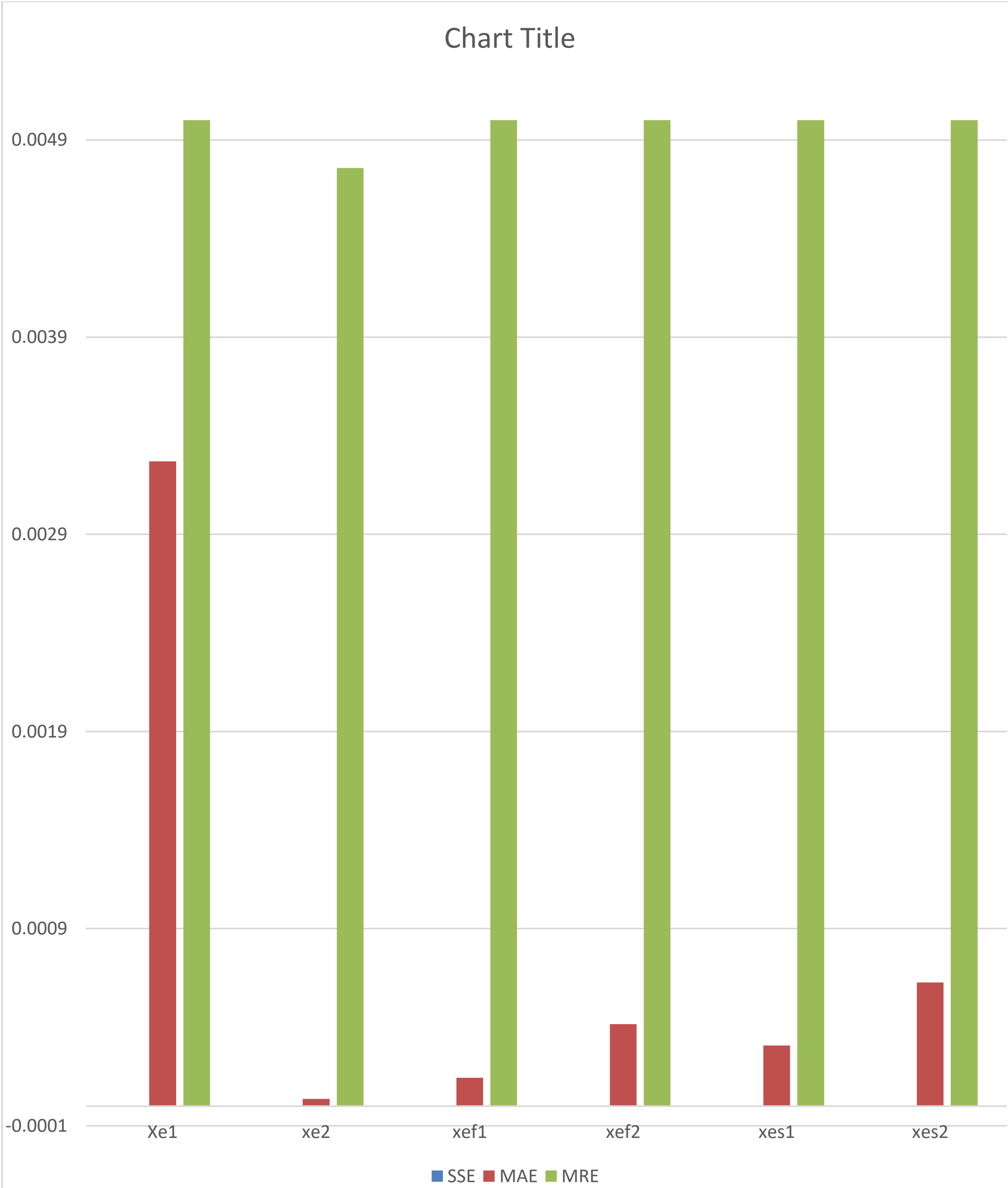

Figure 4.9: Bar graph of performance metrics

Subspace identification-based state estimation method shows a better estimation accuracy than Kalman filtering and Kaman smoothing. In terms of the performance metrics as can be seen from the simulation results obtained in Table 4.1, subspace identification-based state estimation method is better than Kalman filtering, and Kalman smoothing. Because the result SSE shows that subspace identification based state estimation is more preferable. Also as demonstrated from bar graph SSE is approximately zero because it is invisible from bar graph with in specified vertical axix.

5. CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This thesis has compared the quadrotor GAM state estimation methods like 4SID-based state estimation, Kalman filter, and Kalman smoother. Giving PRBS as input signals instead of actual data for the subspace identification method, there are some distortion when considering the influence of input and output noise. The quadrotor state-space model is estimated from identified coprime factor. The newly proposed subspace identification method is very influential in state estimation. In this study, rough analysis has done for performance metrics of subspace identification based state estimation method, Kalman smoother and Kalman filter. As demonstrated from Table 4.1 to estimate angular position state, under SSE column 1.33562×10^{-9} error is smaller than Kalman filter's and kalman smoother's error and under MRE column with minimum error of 0.000143378 Kalman filter is the best estimator to estimate angular position and Subspace identification based state estimation is the best estimator to estimate angular velocity state with minimum error of 0.00003625. Under MRE column, Kalman filter is the best estimator to estimate state of angular position with error of 0.183952 and Subspace identification based estimation is the best estimator to estimate state of angular velocity with error of 0.004759. Arise from this analysis it is verified that subspace identification-based state estimation method is best estimator.

5.2 Recommendation

This thesis uses PRBS inplace of input signal to estimate the state of quadrotor attitude model, rather than collecting data, these has little error. But the estimation method performed in the noisy environment, so the real system is simulated to a certain extent. The estimation results shows that 4SID-based state estimation method is very effective. I recommend that by using recording device collecting real expiermental data can be used for further study of estimation of the quadrotor GAM states.

Acknowledgements

The research for this paper was financially supported by Adama Science and Technology University under the grant number of ASTU/SM-R/170/20, Adama, Ethiopia.

REFERENCE

Aboytes Reséndiz, V. M., & Rivas Araiza, E. (2016). System Identification of a Quad-rotor in X Configuration from Experimental Data. *Research in Computing Science*, *118*(1), 77–86.

Abeywardena, D., Kodagoda, S., Dissanayake, G., & Munasinghe, R. (2013). Improved state estimation in quadrotor MAVs: A novel drift-free velocity estimator. *IEEE Robotics and Automation Magazine*, *20*(4), 32–39

Adnan, W. H., & Khamis, M. F. (2022). *Drone Use in Military and Civilian Application : Risk to National Security*. *15*(January), 60–70.

Ahirwar, S., Swarnkar, R., Bhukya, S., & Namwade, G. (2019). Application of Drone in Agriculture. *International Journal of Current Microbiology and Applied Sciences*, *8*(01), 2500–2505.

Aravkin, A., Burke, J. v., Ljung, L., Lozano, A., & Pillonetto, G. (2017). Generalized Kalman smoothing: Modeling and algorithms. *Automatica*, *86*, 63–86.

Barratt, S. T., & Boyd, S. P. (2020). Fitting a Kalman Smoother to Data. *Proceedings of the American Control Conference*, *2020-July*, 1526–1531.

Chamola, V., Kotesh, P., Agarwal, A., Naren, Gupta, N., & Guizani, M. (2021). A Comprehensive Review of Unmanned Aerial Vehicle Attacks and Neutralization Techniques. *Ad Hoc Networks*, *111*.

Chin, C. S., & Lau, M. W. S. (2020). Benchmark Models of Control System Design for Remotely Operated Vehicles. In *Benchmark Models of Control System Design for Remotely Operated Vehicles*.

Coenen, A., Reif, E., Kim, A. Y. B., Pearce, A., Viégas, F., & Wattenberg, M. (2019). Visualizing and measuring the geometry of BERT. *Advances in Neural Information Processing Systems*, *32*(NeurIPS).

Davis, S. E., Cremaschi, S., & Eden, M. R. (2017). Efficient Surrogate Model Development: Optimum Model Form Based on Input Function Characteristics. In *Computer Aided Chemical Engineering* (Vol. 40, Issue 1). Elsevier Masson SAS.

Datong Liu, Xuehao yin, Yuchen S. Wang L. & Yu P. (2018). An online state of health estimation of Lithium-ion battery using uncented Kalman filter. IEEE.

Ding, L., & Wang, Z. (2018). A Robust Control for an Aerial Robot Quadrotor under Wind Gusts. *Journal of Robotics*, *2018*.

Dong, W., Gu, G. Y., Zhu, X., & Ding, H. (2014). High-performance trajectory tracking control of a quadrotor with disturbance observer. *Sensors and Actuators, A: Physical*, *211*, 67–77.

Evans, C., Hill, D., Rees, D. and Jones, L., 1994. *Measurement and identification of gas turbine dynamics in the presence of noise and nonlinearities*.

Fillion, A., Bocquet, M., Gratton, S., Gürol, S., & Sakov, P. (2020). An iterative ensemble kalman smoother in presence of additive model error. *SIAM-ASA Journal on Uncertainty Quantification*, *8*(1), 198–228.

He, X., Xue, W., Fang, H., Jiang, Y. and Huang, Y., 2016. *Multisine input design for LTI systems by uniform design and its application to aerodynamic system*.

Itakura, K., Saito, Y., Suzuki, T., Kondo, N., & Hosoi, F. (2019). Estimation of citrus maturity with fluorescence spectroscopy using deep learning. *Horticulturae*, *5*(1), 1–9.

Ivošević, B., Han, Y. G., Cho, Y., & Kwon, O. (2015). The use of conservation drones in ecology and wildlife research. *Journal of Ecology and Environment*, *38*(1), 113–118.

Jackson, S., Tisdale, J., Kamgarpour, M., Basso, B., & Hedrick, J. K. (2008). Tracking controllers for small UAVs with wind disturbances: Theory and flight results. *Proceedings of the IEEE Conference on Decision and Control*, 564–569.

Jennings, A. L., Ordóñez, R., & Ceccarelli, N. (2008). An ant colony optimization using training data applied to UAV way point path planning in wind. *2008 IEEE Swarm Intelligence Symposium, SIS 2008*.

Jiang, S., Liu, S., Guo, C., Fan, X., Ma, T., & Tiwari, P. (2019). Implementation of ARINC 659 bus controller for space-borne computers. *Electronics (Switzerland)*, *8*(4), 1–8.

Kabra, T. S., Kardile, A. v, Deeksha, M. G., Mane, D. B., Bhosale, P. R., Belekar, A. M., & Mechanical, I.-. (2017). Design , Development & Optimization of a Quad-Copter for Agricultural Applications. *International Research Journal of Engineering and Technology(IRJET)*, *4*(7), 1632–1636.

Kim, Y., & Bang, H. (2019). Introduction to Kalman Filter and Its Applications. *Introduction and Implementations of the Kalman Filter*, 1–16.

Kowalski, P., & Smyk, R. (2018). Review and comparison of smoothing algorithms for one-dimensional data noise reduction. *2018 International Interdisciplinary PhD Workshop, IIPhDW 2018*, 277–281.

Kurak, S., & Hodzic, M. (2018). Control and estimation of a quadcopter dynamical model. *Periodicals of Engineering and Natural Sciences*, *6*(1), 63–75.

Liang, W. (2017). *Attitude Estimation of Quadcopter through Extended Kalman Filter*. http://preserve.lehigh.edu/etd/2685

Liang, W., Li, D., Zhu, X. X., Li, W., Chen, S., Tofigh, M. A., Mahjoob, M. J., Ayati, M., Kabra, T. S., Kardile, A. v, Deeksha, M. G., Mane, D. B., Bhosale, P. R., Belekar, A. M., Mechanical, I.-, Singh, R., Kumar, R., Mishra, A., Agarwal, A. A., … Rivas Araiza, E. (2022). A review of quadrotor UAV: Control methodologies and performance evaluation. *Proceedings of the IEEE Conference on Decision and Control*, *8*(1), 271–276.

Li, D., Zhu, X., Li, W., & Chen, S. (2019). Experimental results on robust attitude control of a 3-DOF helicopter under wind disturbances. *Chinese Control Conference, CCC*, *2019-July*, 8224–8228.

Lin, S., Zhang, Y., Wang, H., Jiang, J., & Modler, N. (2021). Geometric synthesis method of compliant mechanism based on similarity transformation of pole maps. *Mechanical Sciences*, *12*(1), 375–391.

Martin, P., & Salaün, E. (2010). The true role of accelerometer feedback in quadrotor control. *Proceedings - IEEE International Conference on Robotics and Automation*, 1623–1629.

Mohd Daud, S. M. S., Mohd Yusof, M. Y. P., Heo, C. C., Khoo, L. S., Chainchel Singh, M. K., Mahmood, M. S., & Nawawi, H. (2022). Applications of drone in disaster management: A scoping review. *Science and Justice*, *62*(1), 30–42.

Moin, L., Baig, A.-Z., & Uddin, V. (2017). State Space Model of an Aircraft Using Simulink. *International Journal of System Modeling and Simulation*, *2*(4), 1.

Moir, T. J. (2022). Rudiments of Signal Processing and Systems. *Rudiments of Signal Processing and Systems*, 2–6.

Patel, D. C., Gabhawala, G. S., Kapadia, A. K., Desai, N. H., & Sheth, S. M. (2016). *Design of Quadcopter in Reconnaissance Design of Quadcopter in Reconnaissance*. *August 2015*.

Qin, Q. H., Liu, J. C., & Geng, L. H. (2021). Generalized Attitude Model Identification of a Quadrotor. *Journal of Physics: Conference Series*, *1887*(1).

Restas, A. (2015). Drone Applications for Supporting Disaster Management. *World Journal of Engineering and Technology*, *03*(03), 316–321.

Roush, K. (n.d.-a). *Designing for Security : A Cybersecurity Introduction for Aerospace Education*. 1–7.

Salameh, I. M., Ammar, E. M., & Tutunji, T. A. (2015). Identification of quadcopter hovering using experimental data. *2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies, AEECT 2015*.

Sarath Yadav, E. and Indiran, T., 2019. *PRBS based model identification and GPC PID control design for MIMO Process*.

Sarim, M., Nemat, A., Kumar, M. and Cohen, K., 2015. *Extended Kalman Filter Based Quadrotor State Estimation Based on Asynchronous Multisensor Data*.

Schon, T. (2006). Estimation of nonlinear dynamic systems: Theory and applications. In *Linköping Studies in Science and Technology.* (Issue 998).

Shim, H., Back, J., Eun, Y., Park, G., & Kim, J. (2022). Zero-Dynamics Attack, Variations, and Countermeasures. *Lecture Notes in Control and Information Sciences*, *489*, 31–61.

Singhal, G., Bansod, B., & Mathew, L. (2018). Unmanned Aerial Vehicle Classification , Applications and Challenges : A Review. *Preprint*, *November*, 1–19.

Singh, R., Kumar, R., Mishra, A., & Agarwal, A. (2019). Structural Analysis of Quadcopter Frame. *Materials Today: Proceedings*, *22*, 3320–3329.

Song, Y., & Scaramuzza, D. (2022). Policy Search for Model Predictive Control With Application to Agile Drone Flight. *IEEE Transactions on Robotics*, 1–17.

Stamati, I., Telen, D., Logist, F., Van Derlinden, E., Hirsch, M., Passenbrunner, T. and Van Impe, J., 2012. *Optimal experimental design for mechanistic nonlinear dynamic models using multisine inputs.*

Tahir, Z., Jamil, M., Liaqat, S. A., Mubarak, L., Tahir, W., & Gilani, S. O. (2016). State Space System Modeling of a Quad Copter UAV. *Indian Journal of Science and Technology*, *9*(27).

Thu, K. M., & Gavrilov, A. I. (2017a). Designing and Modeling of Quadcopter Control System Using L1 Adaptive Control. *Procedia Computer Science*, *103*(October 2016), 528–535.

Thu, K. M., & Gavrilov, A. I. (2017b). Designing and Modeling of Quadcopter Control System Using L1 Adaptive Control. *Procedia Computer Science*, *103*, 528–535.

Tofigh, M. A., Mahjoob, M. J., & Ayati, M. (2018). Dynamic modeling and nonlinear tracking control of a novel modified quadrotor. *International Journal of Robust and Nonlinear Control*, *28*(2), 552–567.

Urbanov, A. P. (2017). *Accepted Manuscript*.

Vuojolainen, J., Pyrh¨onen, O., Nevaranta, N. and Jastrzebski, R., 2017. *Comparison of Excitation Signals in Active Magnetic Bearing System Identification*.

Wahlberg, B., Hjalmarsson, H., & Annergren, M. (2010). On optimal input design in system identification for control. In *Proceedings of the IEEE Conference on Decision and Control*.

Welch. (2014). Kalman Filter Kalman Filter. *Fundamentals of Inertial Navigation, Satellite-Based Positioning and Their Integration*, *15*, 7–9.

Zhu, Q., Xu, M., Liu, W., & Zheng, M. (2019). A state of charge estimation method for lithium-ion batteries based on fractional order adaptive extended kalman filter. *Energy*, *187*, 271–276.

L1. Ocampo-Martínez, C., Tornil, S. and Puig, V., 2012. *State Estimator*. [online] ScienceDirect. Available at: <https://www.sciencedirect.com/topics/engineering/state-estimator> [Accessed 16 August 2021].

L2. En.wikipedia.org. 2022. *2021 Israel–Palestine crisis - Wikipedia*. [online] Available at: <https://en.wikipedia.org/wiki/2021_Israel%E2%80%93Palestine_crisis> [Accessed 30 May 2022].

L3. 2022. [online] Available at: <https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle.> [Accessed 30 May 2022].

L4. Unmanned Systems Technology. 2022. *Quadcopter Drones | Professional Quadcopter Manufacturers*. [online] Available at: <https://www.unmannedsystemstechnology.com/expo/quadcopter-drones/> [Accessed 12 February 2022].

L5. Istockphoto.com. 2022. [online] Available at: <https://www.istockphoto.com/photos/quadcopter> [Accessed 12 March 2022].

L6. Youtube.com. 2022. [online] Available at: <https://www.youtube.com/watch?v=Z1QS6FsxrJI&t=19s> [Accessed 23 June 2021].

Appendices

```
%editor: MOTI BEKUMA

%DATE:29-MAY-2022
%@ASTU
%Adivisor:Dr. S.Bakta
```

A. MATLAB Script for check Controllability

```
A = [-10];
B = [1];
C = [1];
D = 0;
Qc = ctrb(F,G);
rankQc = rank(Qc);
disp('Controllable Matrix is Qc = ');
disp(Qc);
if(rankQc == rank(F))
disp('Given System is Controllable.');
else
disp('Given System is Uncontrollable');
end
```

B. MATLAB Script for check Observablity

```
 clear;
close all;
T=0.1;
F = [1 T;0 1];
G = [T^3 /2;T^2];
C = [1 0];
D = 0;
Qb = obsv(F, C);
rankQb = rank(Qb);
disp('Observable Matrix is Qb = ');
disp(Qb);
if(rankQb == rank(F))
disp('Given System is Observable.');
else
disp('Given System is Unobservable');
end
```

C. MATLAB Script for State feedback design

```
T = 0.1;
F = [1 T;0 1];G = [T^3/2;T^2];C = [1 0];D = 0;
poles_c_s=[-3,-15];
poles_c_z = exp(T*poles_c_s);
poles_o_s=[-60,-61];
poles_o_z = exp(T*poles_o_s);
K = place(F,G,poles_c_z)
Ke = place(F',C',poles_o_z)
```

D. Function of Kalman filter and Kalman smoother

```
clc;
clear;
close all;
run PRBS_generatorTG
load PRBS.mat
run identification_experiment
sim identification_experiment
T = 0.1;intvl = 1;
T = T*intvl;
F = [1 T;0 1];G = [T^3/2;T^2];C = [1 0];D = 0;
Pd0 = tf(ss(F,G,C,D,T));
K = [ 201.3505 93.5376];
disp(num2str([F-G*K,G;[C-D*K;-K],[D;1]]))
NDd0 = tf(ss(F-G*K,G,[C-D*K;-K],[D;1],T));
disp('-----------------------------------------')
% N = length(ym);
Np =15;
L1 = 1;L2 = Np*3;
t = psi_r(1,L1:intvl:L2);N = length(t);
z1 = iddata(psi_m(2,L1:intvl:L2)',psi_r(2,L1:intvl:L2)',T);
z2 = iddata(u_psi(2,L1:intvl:L2)',psi_r(2,L1:intvl:L2)',T);
opt = n4sidOptions('Display', 'on','N4Horizon',(7:15)');
[Nd,x0n] =n4sid(z1,5,'DisturbanceModel','none','Feedthrough',false(1,1),'InputDelay',0,opt);
[Dd,x0d] = n4sid(z2,5,'DisturbanceModel','none','Feedthrough',true(1,1),'InputDelay',0,opt);
figure(1)
subplot(311)
stairs(psi_r(1,L1:intvl:Np),psi_r(2,L1:intvl:Np),'r')
xlabel('Time (s)')
ylabel('{\it psi_r(degree)}')
subplot(312)
stairs(u_psi(1,L1:intvl:Np),u_psi(2,L1:intvl:Np),'b')
xlabel('Time (s)')
ylabel('{\it u}_4 (kgm^2/s^2)')
subplot(313)
```

```
stairs(psi_m(1,L1:intvl:Np),psi_m(2,L1:intvl:Np),'m')
xlabel('Time (s)')
ylabel('{\it psi}_m(degree)')
% Any form of realization identification of Nd and Dd
ND1 = bstmr([Nd*4.9665*1e-3;Dd],2);%balancmr | bstmr | hankelmr | ncfmr | reduce |
schurmr
type = 'zoh';
d2c(tf(minreal(ND1(1)/ND1(2))),type)
Y = [psi_m(2,L1:intvl:L2);u_psi(2,L1:intvl:L2)];u = psi_r(2,L1:intvl:L2);Q = zeros(2,2);R = 1e-
5*eye(2);
[xf,xs] = Kalman_filter_smoother(ND1.a,ND1.b,ND1.c,ND1.d,Y,u,Q,R);
% normalization of realization for Nd and Dd
P = pinv([C-D*K;-K])*ND1.c
A = P*ND1.a*pinv(P)
B = P*ND1.b
C = ND1.c*pinv(P)
disp(num2str([A,B;C,ND1.d]))
NDA1 = A;NDB1 = B;NDC1 = C;NDD1 = ND1.d;
D_est1 = NDD1(1)%NDD1(1)*pinv(NDD1(2));
B_est1 = NDB1%NDB1*pinv(NDD1(2));
A_est1 = NDA1-NDB1*NDC1(2,:)%NDA1-NDB1*pinv(NDD1(2))*NDC1(2,:);
C_est1 = NDC1(1,:)-D_est1*NDC1(2,:)
canon_P_est1 = tf(canon(ss(A_est1,B_est1,C_est1,D_est1,T),'companion'));
NDc = d2c(canon_P_est1,type)
x(:,1) = zeros(2,1);
for k = 1:N
 x(:,k+1) = ND1.a*x(:,k)+ND1.b*psi_r(2,k);
end
xe = P*x;
xef = P*xf;
xes = P*xs;
t = t(Np+1:2*Np);
% % % %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Subpace identification based state estimation method
figure(5)
subplot(211)
plot(t,xr1(2,Np+1:2*Np),'k',t,xe(1,Np+1:2*Np),'r.--')
legend('boxoff')
legend('x_r','x_e')
xlabel('Time(sec.)')
ylabel('x_1(degree)')
%title('Subspace identification based state estimation result')
subplot(212)
plot(t,xr2(2,Np+1:2*Np),'k',t,xe(2,Np+1:2*Np),'r.--')
legend('boxoff')
legend('x_r','x_e')
```

```
xlabel('Time(sec.)')
ylabel('x_2(degree/sec)')
% % % %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%kalman filtering
figure(6)
subplot(211)
plot(t,xr1(2,Np+1:2*Np),'k',t,xef(1,Np+1:2*Np),'r.--')
legend('boxoff')
legend('x_r','xe_f')
xlabel('Time(sec.)')
ylabel('x_1(degree)')
%title('Kalman filtering state estimation result')
subplot(212)
plot(t,xr2(2,Np+1:2*Np),'k',t,xef(2,Np+1:2*Np),'r.--')
legend('boxoff')
legend('x_r','xe_f')
xlabel('Time(sec.)')
ylabel('x_2(degree/sec)')
```

E. Estimation of states main program: This m-file can determine the estimated states, estimated models and the transformation matrixes.

```
clc;
clear;
close all;

run new_PRBSgenerator01
load att_r01.mat
run identification_experiment
sim identification_experiment
load um;
load ym;    %psi_m jechuu fedheetu
load rh;    %psi_r
load xr1;   %true position used for testing the estimated position
load xr2;   %true velocity used for testing the estimated position

T = 0.1;intvl = 1;
T = T*intvl;
F = [1 T;0 1];G = [T^3/2;T^2];C = [1 0];D = 0; %discrete form of state space

Pd0 = tf(ss(F,G,C,D,T)); %tf discrete
K = [201.3505 93.5376];
disp(num2str([F-G*K,G;[C-D*K;-K],[D;1]])) %state space discrete with feedback gain

NDd0 = tf(ss(F-G*K,G,[C-D*K;-K],[D;1],T)); %tf discrete with feedback gain
```

```
disp('*****************************************************')
% N = length(ym);

Np = 15; %number of cycle
L1 = 1; L2 = Np*3;
t = rh(1,L1:intvl:L2);N = length(t);
z1 = iddata(ym(2,L1:intvl:L2)',rh(2,L1:intvl:L2)',T); %rh to ym transfer function identify
gochuuf
z2 = iddata(um(2,L1:intvl:L2)',rh(2,L1:intvl:L2)',T); %rh to um transfer function identify
gochuuf
opt = n4sidOptions('Display', 'off','N4Horizon',(7:15)');
[Nd,x0n] = n4sid(z1,2,'DisturbanceModel','none','Feedthrough',false(1,1),'InputDelay',0,opt);
[Dd,x0d] = n4sid(z2,2,'DisturbanceModel','none','Feedthrough',true(1,1),'InputDelay',0,opt);
 %ND1 = c2d([Nd;Dd],0.1,'tustin');
 %ND1 = hankelmr(Nd,2); %kana balleessuus dandeessa
%
%
%
figure(1)
subplot(311)
stairs(rh(1,L1:intvl:Np),rh(2,L1:intvl:Np),'r')
%ylim([-15 15])
xlabel('Time (s)')
ylabel('{\it psi}_r')
subplot(312)
stairs(um(1,L1:intvl:Np),um(2,L1:intvl:Np),'b')
xlabel('Time (s)')
ylabel('{\it u}_m')
subplot(313)
stairs(ym(1,L1:intvl:Np),ym(2,L1:intvl:Np),'m')
xlabel('Time (s)')
ylabel('{\it y}_m')
%Nd,Dd kan jedhu kun isa bifa transfer functioniitin barreeffamuudha coprime factorizatinidha
jechuudha.


% Any form of realization identification of Nd and Dd
ND1 = bstmr([Nd;Dd],2);%balancmr | bstmr | hankelmr | ncfmr | reduce | schurmr  kunimmoo
state  order fi plant order walitti identified tahu kanaaf reduce gochuuf
type = 'foh';
d2c(tf(minreal(ND1(1)/ND1(2))),type) %isa identify tahe sana continous gochuudhaaf.

Q = zeros(2,2);R = 1e-5*eye(2);Nn = zeros(2,2);
% sys = ss(ND1.a,[ND1.b],ND1.c,[ND1.d],T,'inputname',{'u'},'outputname',{'y'});
% [kest,L,X] = kalman(sys,Rn);
```

```
Y = [ym(2,L1:intvl:L2);um(2,L1:intvl:L2)];u = rh(2,L1:intvl:L2);Q = zeros(2,2);R = 1e-
5*eye(2);
[xf,xs] = Kalman_filter_smoother(ND1.a,ND1.b,ND1.c,ND1.d,Y,u,Q,R);

% normalization of realization for Nd and Dd
P = pinv([C-D*K;-K])*ND1.c
A = P*ND1.a*pinv(P)
B = P*ND1.b
C = ND1.c*pinv(P)
disp(num2str([A,B;C,ND1.d]))
tf(ss(ND1.a,ND1.b,ND1.c,ND1.d,T))

% estimation of A,B,C,D for attitude model
NDA1 = A;NDB1 = B;NDC1 = C;NDD1 = ND1.d;
D_est1 = NDD1(1)%NDD1(1)*pinv(NDD1(2));
B_est1 = NDB1%NDB1*pinv(NDD1(2));
A_est1 = NDA1-NDB1*NDC1(2,:)%NDA1-NDB1*pinv(NDD1(2))*NDC1(2,:);
C_est1 = NDC1(1,:)-D_est1*NDC1(2,:)
canon_P_est1 = tf(canon(ss(A_est1,B_est1,C_est1,D_est1,T),'companion'));
NDc = d2c(canon_P_est1,type);

x(:,1) = zeros(2,1); xx(:,1) = zeros(2,1);yy = zeros(4,N);
for k = 1:N
   x(:,k+1) = ND1.a*x(:,k)+ND1.b*rh(2,k);
end
xe = P*x;
xef = P*xf;
xes = P*xs;
t = t(Np+1:2*Np);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Estimation of subspace,k_filter and k_smoother
figure(3)
subplot(211)
plot(t,xr1(2,Np+1:2*Np),'k',t,xe(1,Np+1:2*Np),'r',t,xef(1,Np+1:2*Np),'b',t,xes(1,Np+1:2*Np),'c')
legend('xr1','xess1','xef1','xes1') %xr1 kan jettu kun referance position dha kanan secondary data
irraa argadheedha
subplot(212)
plot(t,xr2(2,Np+1:2*Np),'k',t,xe(2,Np+1:2*Np),'r',t,xef(2,Np+1:2*Np),'b',t,xes(2,Np+1:2*Np),'c')
legend('xr2','xess2','xef2','xes2') %xr2 kan jettu kun referance velocity dha kanan secondary data
irraa argadheedha
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% %%%%%%%%%%%
%Bode diagram
%P0 = tf(-0.2,[1 0 0]);
%figure(4)
%bode(P0,'k',NDc,'r.-');
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Subpace identification based state estimation method
figure(5)
subplot(211)
plot(t,xr1(2,Np+1:2*Np),'k',t,xe(1,Np+1:2*Np),'r')
legend('boxoff')
legend('x_r_1','x_e_1')
xlabel('Time(sec.)')
ylabel('','Interpreter','latex','String','$\hat x_1$')
title('Subspace identification based state estimation result')
subplot(212)
plot(t,xr2(2,Np+1:2*Np),'k',t,xe(2,Np+1:2*Np),'r')
legend('boxoff')
legend('x_r_2','x_e_2')
xlabel('Time(sec.)')
ylabel('','Interpreter','latex','String','$\hat x_2$')
% % %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%kalman filtering
figure(6)
subplot(211)
plot(t,xr1(2,Np+1:2*Np),'k',t,xef(1,Np+1:2*Np),'b')
legend('boxoff')
legend('x_r_1','x_e_f_1')
xlabel('Time(sec.)')
ylabel('','Interpreter','latex','String','$\hat x_1$')
title('Kalman filtering state estimation result')
subplot(212)
plot(t,xr2(2,Np+1:2*Np),'k',t,xef(2,Np+1:2*Np),'b')
legend('boxoff')
legend('x_r_2','x_e_f_2')
xlabel('Time(sec.)')
ylabel('','Interpreter','latex','String','$\hat x_2$')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%
% kalman smoothing
figure(7)
subplot(211)
plot(t,xr1(2,Np+1:2*Np),'k',t,xes(1,Np+1:2*Np),'b')
legend('boxoff')
legend('x_r_1','x_e_s_1')
xlabel('Time(sec.)')
```

```
ylabel('','Interpreter','latex','String','$\hat x_1$')
title('Kalman smoothing state estimation result')
subplot(212)
plot(t,xr2(2,Np+1:2*Np),'k',t,xes(2,Np+1:2*Np),'r')
legend('boxoff')
legend('x_r_2','x_e_s_2')
xlabel('Time(sec.)')
ylabel('','Interpreter','latex','String','$\hat x_2$')
```