## U.R. RAO SATELLITE CENTRE

**A STUDENT**
**INTERNSHIP REPORT**
**ON**

# Attitude Control System Design for Satellite Attached by Non-Cooperative Object using Classical and Adaptive Control Techniques

A student internship report submitted to the Advanced Technology Development Group (ATDG) at U.R. Rao Satellite Centre as part of the summer internship program.

**M Tech in Systems, Control and Automation.**

**INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI (IITG)**


**GUIDED BY: -**

**Mrs. V Chinna Ponnu**

**Group Head**

**Advanced Technology Development Group (ATDG)**

**U.R. Rao Satellite Centre (URSC).**

**SUBMITTED BY: -**

**Meesala Ganesh**

**Branch: -EEE (MTech,IITG)**

**Roll No.: - 234102505.**


**TO THE**
**Advanced Technology**
**Development Group (ATDG)**
**URSC-ISRO, VIMANAPURA, BENGALURU-560017, INDIA.**

# CERTIFICATION

This is to certify that this brief report on ***Attitude Control System Design for Satellite Attached by non-cooperative Object using Classical and Adaptive Control Techniques*** written by ***Meesala Ganesh*** student (MTech) of Indian Institute of Technology Guwahati (IITG), is an outcome intern undergone at Advanced Technology Development Group (ATDG) of URSC, ISRO Bengaluru, from 3rd June 2024 to 17th July 2024 under the guidance of ***Mrs. V Chinna Ponnu (Group Head, ATDG)*** and ***Dr. J Krishna Kishore (Group Director)*** .

He has learnt the fundamentals of satellite control system quickly. During his period, he was sincere and showed interest in doing his work, thus completing it in the stipulated time.

We wish him all success in his future endeavors.

…..............................

**Mrs. V Chinna Ponnu**

**Group Head**

**Advanced Technology Development Group**

**USRC-ISRO, BENGALURU.**

…..............................

**Dr. J Krishna Kishore**

**Group Director**

**Advanced Technology development group**

**USRC-ISRO, BENGALURU.**

# DECLARATION

I, Meesala Ganesh, hereby declare that an internship titled "Attitude Control System Design for Satellite Attached by Non-Cooperative Object using Classical and Adaptive Control Techniques." All the control techniques, methodologies, design and simulations which are presented in this report are outcomes of my internship work. Any reference to external sources, including literature survey, research papers have been duly acknowledged and cited in accordance with academic standards.

I further declare that my internship has been completed under the supervision and guidance of Mrs. V Chinna Ponnu (Group Head (ATDG) and approved by Dr. J Krishna Kishore (Group Director, ATDG) at U. R. Rao Satellite Centre, ISRO, Bengaluru. I take full responsibility for the content and accuracy of the information presented in this internship report.

……………………………

Meesala Ganesh(234102505)

Intern

M Tech

Systems, Control and Automation

IIT Guwahati.

## ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincere appreciation to all those who have supported and assisted me throughout the completion of this internship program entitled "Attitude Control System Design for Satellite Attached by Non-Cooperative Object using Classical and Adaptive Control Techniques."

I am deeply grateful to my internship guide Mrs. V Chinna Ponnu (Group Head of Advanced Technology development Group- ATDG) and Dr. J Krishna Kishore (Group Director of Advanced Technology Development Group-ATDG), for their invaluable guidance, encouragement, and expert advice throughout the duration of this internship. Their mentorship has been instrumental in shaping the direction and focus of my training.

I extend my sincere thanks to Dr. Ravi Kumar L (Scientist/Engineer-G) for providing system facilities to carry out the simulations throughout the period.

Lastly, I am grateful to my family, friends and faculty members of IITG for their unwavering encouragement towards my internship program at URSC (ISRO, Bengaluru). Their blessings and support have been a constant source of motivation and strength.

……………………

Meesala Ganesh

Intern

Advanced Technology Development Group

U. R. Rao Satellite Center

Bengaluru.

**An Internship report on**

# Attitude Control System Design for Satellite Attached by Non-Cooperative Object using Classical and Adaptive Control Techniques

*submitted*

*by*


**Meesala Ganesh**

Roll No234102505


*Under the instruction and guidance of*

**Mrs. Chinna Ponnu (Group Head,**
**Advanced Technology Development Group, ATDG)**

*Approved by*

**Dr. J Krishna Kishore (Group Director,**
**Advanced Technology Development Group, ATDG)**


**Department of Electronics and Electrical Engineering**
**Indian Institute of Technology Guwahati**

**July 2024**

# Contents

# 1 Abstract

This document details the design and implementation of control techniques (both classical control techniques (PD Controller) and adaptive control technique) for a satellite attitude control system. The PD controller's design equations are derived and demonstrated the controller's effectiveness by computing the gains ($K_p$ and $K_d$) for specified system dynamics.

The document further explains the Euler's moment equations and the state-space representation of spacecraft dynamics are discussed, providing a comprehensive framework for control system design. This document presents Simulink models for three-axis attitude control systems and their simulation results.

**Problem Statement:** When a non-cooperative object (such as space debris or an unintended attachment) attaches to a satellite, it introduces unknown inertia and changes in the system's momentum. This can lead to instability in the satellite's attitude control system.

**Proposed Solution:** To address this problem, the document proposes an adaptive control scheme that estimates the inertia of the new combined system (satellite + attached object). This adaptive control method helps the system reach a stable state quickly and accurately. The control scheme includes a method for estimating the inertia of the combined system, which is crucial for maintaining stability. The stability of the adaptive control scheme is proven using a Lyapunov function, a mathematical concept used to prove the stability of dynamic systems. The proposed adaptive control scheme is robust, effective and capable of handling the uncertainties introduced by the attachment of non-cooperative objects to satellites.

# 2 Introduction

This document presents a comprehensive guide to designing and implementing control techniques for spacecraft attitude control, specifically focusing on proportional-derivative (PD) controllers and adaptive control schemes. The content covers theoretical aspects, design calculations, and detailed explanations of various control mechanisms.

This section explains the fundamentals of designing a PD controller, including the closed-loop transfer function, characteristic equation, and calculations for control gains ($K_p$ and $K_d$). The PD controller design is particularly emphasized for systems requiring fast response and stability improvements. Detailed explanations of the classical control system block diagram are provided, outlining how each component interacts to achieve spacecraft attitude control. The document delves into the mathematical formulations of spacecraft attitude kinematics using quaternions and the dynamics governed by Euler's Moment Equations. This theoretical foundation is essential for understanding the control mechanisms. A detailed Simulink model for controlling the spacecraft's three axes (pitch, roll, and yaw) is presented in which includes diagrams, explanations of subsystems, and the process of summing control torques and converting them to angular velocities for reaction wheels. The document explains linearization of nonlinear systems, stability analysis using eigenvalues, and the design of state feedback control to ensure system stability. In real space applications and real scenarios, spacecraft often encounter attachment issues either with cooperative or non-cooperative objects. When a non-cooperative object attaches to spacecraft, it significantly impacts the attitude (spacecraft attitude control system). This attachment can lead to changes in the spacecraft's attitude angles and inertia parameters that need to be addressed carefully. The study primarily focuses on non-cooperative attachments where the object's dimensions, centroid position and movement characteristics are unknown. Non-cooperative attachment can occur through various methods such as mechanical capture or adhesion. The primary challenge is that these attachments alter the combined system's inertia and momentum, leading to instability. Previous studies have explored various strategies for handling non-cooperative attachments, including fuzzy-PD and gain scheduled PD controllers. These methods focus on managing system response speed without explicitly addressing the unknown inertia parameters. The main objective is to develop an adaptive control scheme that can effectively stabilize the target spacecraft after a non-cooperative attachment. This control scheme should be able to estimate the new combined system's inertia and maintain stability despite the unknown parameters and disturbances introduced by the attachment. The proposed adaptive control scheme is based on inertia estimation. It aims to allow the new combined system (satellite and attached non-cooperative object) to reach a stable state quickly. The stability of the adaptive control scheme is proven using a Lyapunov function. The simulation demonstrates that proposed control scheme effectively reduces attitude errors and maintain stability even under unfavorable conditions, such as unknown inertia and additional momentum from the non-cooperative object.

# 3 Classical Control scheme for spacecraft:

The classical control scheme depicted in the block diagram is designed to control the attitude angles of a spacecraft using a proportional-derivative (PD) controller. Below is a detailed explanation of the control scheme and its components:

- **Reference Input (Setpoint):**
  - This is the desired orientation that we want the spacecraft to achieve. It is usually provided as a step input, representing a sudden change to a new desired orientation.
- **Error Signal Calculation:**
  - The error signal is calculated by subtracting the actual orientation (feedback) from the reference input. This error signal represents the difference between the desired and actual orientations.
- **PD Controller:**
  - The proportional controller multiplies the error signal by a proportional gain $K_p$. This term produces an output that is proportional to the current error value. The proportional control term helps reduce the overall error by providing a control action that is directly proportional to the error.
  - The derivative controller takes the derivative of the error signal with respect to time, which is represented by $\frac{\Delta u}{\Delta t}$ in the block diagram. The derivative of the error is then multiplied by a derivative gain $K_d$. The derivative control term helps provide a damping effect and reduce overshoot and oscillations.
- **Summation of Control Actions:**
  - The outputs of the proportional and derivative controllers are summed together to form the total control signal. This total control signal represents the combined effect of the proportional and derivative actions on the error.
- **Torque Application:**
  - The control signal is then applied as a torque to the spacecraft. This torque is used to adjust the orientation of the spacecraft.
- **Spacecraft Dynamics:**
  - The spacecraft dynamics are modeled using two integrators in series, each with a transfer function $\frac{1}{s}$. This represents the double integration of torque to compute the angular rate and orientation.
  - The first integrator computes the angular rate from the applied torque.
  - The second integrator computes the orientation from the angular rate.
- **Feedback Loop:**
  - The actual orientation and angular rate are fed back to the summing junction to compute the error signal again, completing the feedback loop. This feedback mechanism ensures that the control system continuously adjusts the torque applied to the spacecraft to minimize the error and achieve the desired orientation.

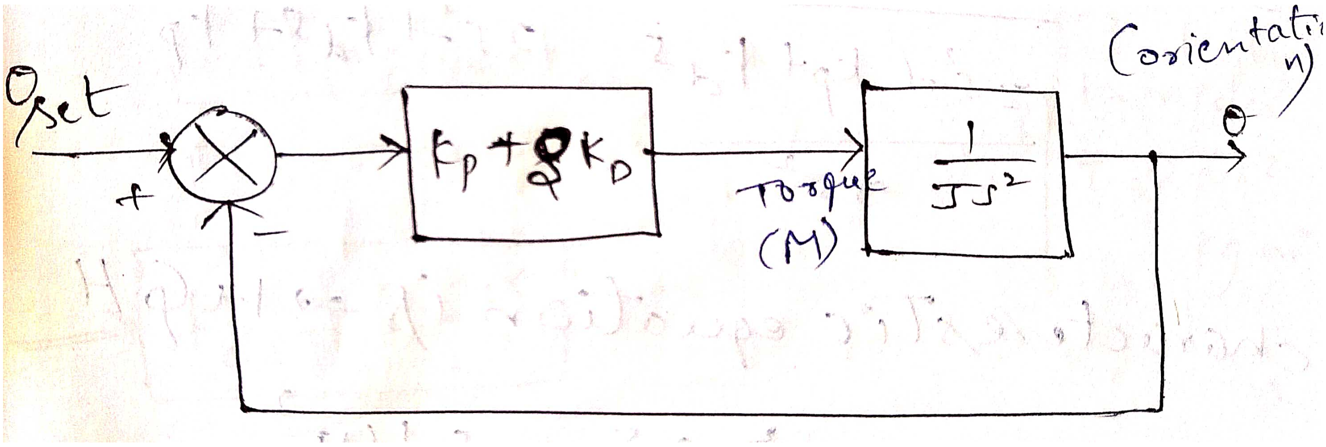


Figure 1: Classical control system

$$\frac{d}{dt}(I\dot{\omega}) = M$$
$$I\frac{d^2\theta(t)}{dt^2} = M$$

Apply Laplace transform on both sides, we get:

$$IS^2\Theta(s) = M(s)$$
$$\frac{\Theta(s)}{M(s)} = \frac{1}{IS^2}$$

Where $I$ is the moment of inertia.
Closed loop transfer function (CLTF) is:

$$\begin{aligned}\text{CLTF} &= \frac{\Theta(s)}{\Theta_{\text{set}}(s)} \\ &= \frac{H(s)}{1+G(s)H(s)} \\ &= \frac{(K_p+K_d s)\,\frac{1}{Js^2}}{1+(K_p+K_d s)\,\frac{1}{Js^2}} \\ &= \frac{\frac{K_p+K_d s}{Js^2}}{1+\frac{K_p+K_d s}{Js^2}}\end{aligned}$$

$$\begin{aligned}\text{CLTF} &= \frac{K_p+K_d s}{Js^2} \\ &= \frac{K_p+K_d s}{Js^2+K_d s+K_p}\end{aligned}$$

Characteristic equation is $1+G(s)H(s)$:

$$Js^2+K_d s+K_p = s^2+2\zeta\omega_n s+\omega_n^2$$
$$s^2+\left(\frac{K_d}{J}\right)s+\left(\frac{K_p}{J}\right) = s^2+2\zeta\omega_n s+\omega_n^2$$

Equating coefficients, we get:

$$\begin{aligned}2\zeta\omega_n &= \frac{K_d}{J}\\ \omega_n^2 &= \frac{K_p}{J}\end{aligned}$$

Solving for $K_d$ and $K_p$:

$$\begin{aligned}K_d &= (2\zeta\omega_n)J\\ K_p &= \omega_n^2 J\end{aligned}$$

If $\zeta = 0.7$ and $t_s$ (settling time) is specified:

$$\begin{aligned}t_s &= \frac{3}{\zeta\omega_n} \quad \text{for 5\% tolerance}\\ \zeta\omega_n &= \frac{3}{t_s}\\ \omega_n &= \frac{3}{t_s\zeta}\end{aligned}$$

## 3.1 Gain Calculations:

$$\omega_n = \frac{3}{t_s\zeta} = \frac{3}{(5)(0.7)} = \frac{3}{3.5} = 0.857 \text{ rad/s}$$

$$\begin{aligned}K_d &= (2\zeta\omega_n)J\\ K_d &= 2 \times 0.7 \times 0.857 \times 2.683 \quad (\text{consider } J = 2.683 \text{ kg-m}^2)\\ K_d &= 3.219\end{aligned}$$

$$\begin{aligned}K_p &= \omega_n^2 J\\ K_p &= (0.857)^2 \times 2.683\\ K_p &= 1.97\end{aligned}$$

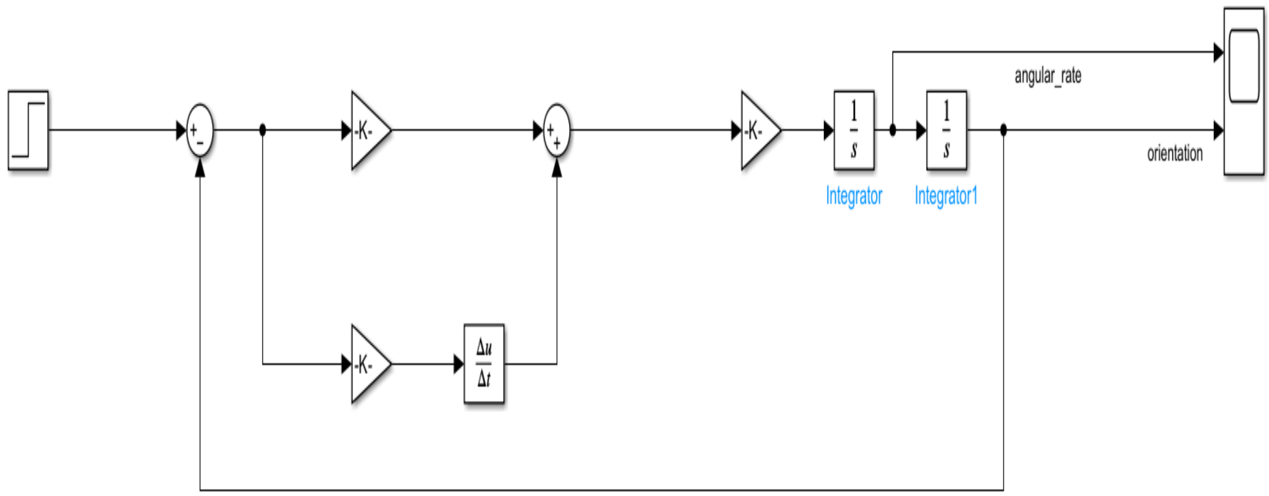


Figure 2: Block diagram of the classical control system

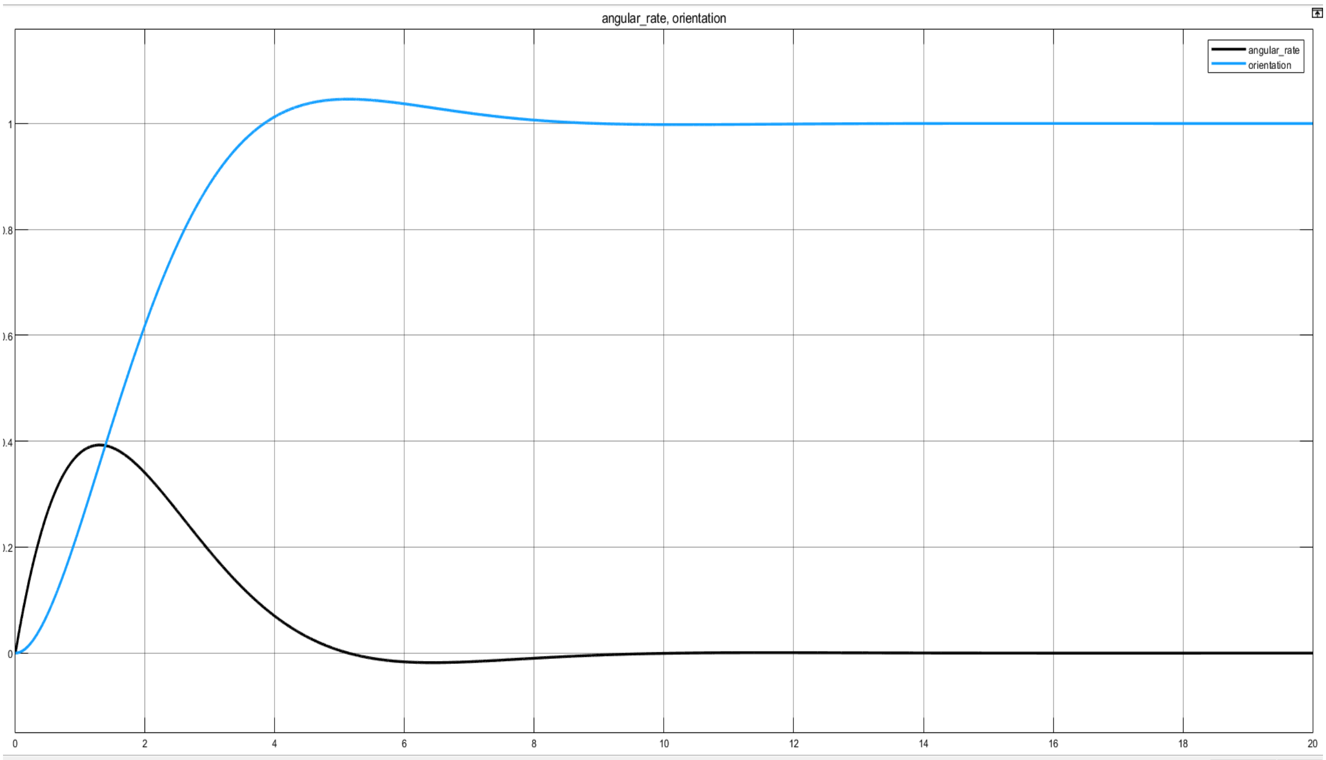


Figure 3: Simulation results of the classical control system

## 3.2 Euler's Moment Equations

These equations are fundamental in the study of dynamics, particularly in the context of rigid body rotation.

### Definition and Basic Concept

- **Angular Momentum (h)**:
  - Angular momentum is a vector quantity that represents the rotational inertia and rotational velocity of a body.
  - For a rigid body, the angular momentum in the inertial frame ($\mathbf{h}_I$) and in the body frame ($\mathbf{h}_B$) are related through the moment of inertia tensor and the angular velocity vector.
- **Moment (M)**:
  - The moment (or torque) is a vector that represents the rotational force applied to the body.

The moment $\mathbf{M}$ is given by:

$$\mathbf{M} = \dot{\mathbf{h}}_I = \dot{\mathbf{h}}_B + \boldsymbol{\omega} \times \mathbf{h}_B$$

where:

- $\mathbf{h}_I$ is the angular momentum in the inertial frame.
- $\mathbf{h}_B$ is the angular momentum in the rotating body frame.
- $\boldsymbol{\omega}$ is the angular velocity of the rotating body.
- $\boldsymbol{\omega} \times \mathbf{h}_B$ is the cross product of angular velocity and angular momentum in the body frame, accounting for the change in angular momentum due to rotation.

- **Moments of Inertia**:
  - $I_x, I_y, I_z$ are the principal moments of inertia about the principal axes $X_B, Y_B, Z_B$.
  - These moments of inertia quantify the body's resistance to angular acceleration about each principal axis.

Assuming $X_B, Y_B, Z_B$ are the principal axes, the vector equation can be expanded into three scalar equations:

$$\begin{aligned}
M_x &= I_x \dot{\omega}_x + \omega_y \omega_z (I_z - I_y) \\
M_y &= I_y \dot{\omega}_y + \omega_x \omega_z (I_x - I_z) \\
M_z &= I_z \dot{\omega}_z + \omega_x \omega_y (I_y - I_x)
\end{aligned}$$

where:

- $M_x, M_y, M_z$ are the components of the applied moment about the principal axes.
- $\omega_x, \omega_y, \omega_z$ are the components of the angular velocity about the principal axes.
- $\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z$ are the time derivatives of the angular velocity components (angular accelerations).
- The scalar form of Euler's equations is nonlinear because the equations involve products of angular velocity components.
- These nonlinear equations typically do not have an analytical closed-form solution and are often solved numerically.

- The nonlinearity arises from the $\boldsymbol{\omega} \times \mathbf{h}_B$ term, which involves the cross product of angular velocity and angular momentum.

Euler's Moment Equations are widely used in various fields, including:

- **Aerospace Engineering:** Attitude control of spacecraft.
- **Mechanical Engineering:** Dynamics of rotating machinery.
- **Robotics:** Control of robotic arms and manipulators.
- **Physics:** Study of rotational motion in rigid body dynamics.

### 3.3 Spacecraft Attitude Kinematics

The quaternion kinematics equation is

$$\dot{q} = \frac{1}{2}[\omega]q$$

where

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}$$

$[\omega]$ - skew symmetric matrix

$$[\omega] = \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix}$$

### 3.4 Spacecraft Attitude Dynamics

$$\mathbf{M} = \dot{\mathbf{h}}_I = \dot{\mathbf{h}}_B + (\omega \times \mathbf{h})$$

$$\mathbf{M} = \mathbf{h}_I = \mathbf{I}\dot{\omega} + (\omega \times \mathbf{I}\omega)$$

$$\mathbf{M} = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}, \quad \dot{\omega} = \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$$

$\mathbf{M} \rightarrow$ External moment applied on the spacecraft (3x1 vector)

$\mathbf{I} \rightarrow$ Inertia matrix (3x3 matrix)

$\omega \rightarrow$ Angular velocities (3x1 vector)

$\mathbf{h}_B \rightarrow$ Angular momentum due to body's motion

$$\dot{\omega} = \mathbf{I}^{-1}(\mathbf{M} - (\omega \times \mathbf{I}\omega))$$

It is in the form of $\dot{x} = Ax + Bu$.

$$\text{Let, } A = \begin{bmatrix} A_q & O_{4\times3} \\ O_{3\times4} & A_\omega \end{bmatrix}$$

where

$$A_q \to 4\times4, \quad A_\omega \to 3\times3$$

$$B = \begin{bmatrix} O_{4\times3} \\ I_{3\times3} \end{bmatrix}$$

$$\mathbf{M} = \dot{\mathbf{h}} = \dot{\mathbf{h}}_B + (\omega \times \mathbf{h})$$

$$\mathbf{M} = \mathbf{I}\dot{\omega} + (\omega \times \mathbf{I}\omega)$$

Let us represent the dynamic equation in state space form.

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad \text{consider} \quad \mathbf{I} \quad \text{as a diagonal matrix} \quad \mathbf{I} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

$$\mathbf{I}\omega = \begin{bmatrix} I_{xx}\omega_x \\ I_{yy}\omega_y \\ I_{zz}\omega_z \end{bmatrix} \quad (3\times1)$$

$$\omega \times \mathbf{I}\omega = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \omega_x & \omega_y & \omega_z \\ I_{xx}\omega_x & I_{yy}\omega_y & I_{zz}\omega_z \end{vmatrix}$$

$$= \begin{bmatrix} \omega_y\omega_z I_{zz} - \omega_z\omega_y I_{yy} \\ \omega_z\omega_x I_{xx} - \omega_x\omega_z I_{zz} \\ \omega_x\omega_y I_{yy} - \omega_y\omega_x I_{xx} \end{bmatrix}$$

$$\omega \times (\mathbf{I}\omega) = \begin{bmatrix} (I_{zz} - I_{yy})\omega_y\omega_z \\ (I_{xx} - I_{zz})\omega_x\omega_z \\ (I_{yy} - I_{xx})\omega_x\omega_y \end{bmatrix}$$

$$\mathbf{I}^{-1} = \begin{bmatrix} \frac{1}{I_{xx}} & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix}$$

$$\mathbf{I}^{-1}(\omega \times (\mathbf{I}\omega)) = \begin{bmatrix} \frac{1}{I_{xx}} & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \begin{bmatrix} (I_{zz} - I_{yy})\omega_y\omega_z \\ (I_{xx} - I_{zz})\omega_x\omega_z \\ (I_{yy} - I_{xx})\omega_x\omega_y \end{bmatrix}$$

$$= \begin{bmatrix} \frac{(I_{zz}-I_{yy})}{I_{xx}}\omega_y\omega_z \\ \frac{(I_{xx}-I_{zz})}{I_{yy}}\omega_x\omega_z \\ \frac{(I_{yy}-I_{xx})}{I_{zz}}\omega_x\omega_y \end{bmatrix}$$

Euler's Moment Equations describe the dynamics of rotational motion for a rigid body. The equations can be represented in matrix form, leading to a state-space representation.

### Solving for Angular Acceleration

Rearranging the above equation to solve for $\dot{\boldsymbol{\omega}}$:

$$\dot{\boldsymbol{\omega}} = I^{-1}(\mathbf{M} - (\boldsymbol{\omega} \times I\boldsymbol{\omega}))$$

### Matrix Form

Expanding the equations, we have:

$$\begin{aligned}
\dot{\omega}_x &= \left(\frac{I_{zz} - I_{yy}}{I_{xx}}\right)\omega_y\omega_z + \frac{1}{I_{xx}}M_x \\
\dot{\omega}_y &= \left(\frac{I_{xx} - I_{zz}}{I_{yy}}\right)\omega_x\omega_z + \frac{1}{I_{yy}}M_y \\
\dot{\omega}_z &= \left(\frac{I_{yy} - I_{xx}}{I_{zz}}\right)\omega_x\omega_y + \frac{1}{I_{zz}}M_z
\end{aligned}$$

### Matrix Representation in $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ Form

These equations can be written in matrix form as:

$$\dot{\boldsymbol{\omega}} = \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = \begin{bmatrix} -\frac{I_{zz}-I_{yy}}{I_{xx}}\omega_y\omega_z \\ -\frac{I_{xx}-I_{zz}}{I_{yy}}\omega_x\omega_z \\ -\frac{I_{yy}-I_{xx}}{I_{zz}}\omega_x\omega_y \end{bmatrix} + \begin{bmatrix} \frac{1}{I_{xx}} & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}$$

In $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ form, the matrices $A$ and $B$ are:

$$A = \begin{bmatrix} 0 & -\frac{I_{zz}-I_{yy}}{I_{xx}}\omega_z & -\frac{I_{zz}-I_{yy}}{I_{xx}}\omega_y \\ -\frac{I_{xx}-I_{zz}}{I_{yy}}\omega_z & 0 & -\frac{I_{xx}-I_{zz}}{I_{yy}}\omega_x \\ -\frac{I_{yy}-I_{xx}}{I_{zz}}\omega_y & -\frac{I_{yy}-I_{xx}}{I_{zz}}\omega_x & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{1}{I_{xx}} & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix}$$

## State-Space Representation

The state-space representation of the rotational dynamics of a rigid body incorporates quaternions to describe orientation.

The state-space representation combines the quaternion and angular velocity dynamics. The state vector $\mathbf{x}$ is defined as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix}$$

where:

$$\mathbf{X}_1 = \mathbf{q}, \quad \mathbf{X}_2 = \boldsymbol{\omega}$$

The combined state-space form is:

$$\begin{bmatrix} \dot{\mathbf{X}}_1 \\ \dot{\mathbf{X}}_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}[\mathbf{Q}] & \mathbf{0} \\ \mathbf{0} & -I^{-1}(\boldsymbol{\omega} \times I) \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ I^{-1} \end{bmatrix} \mathbf{u}$$

### State Vector (X)

The state vector $\mathbf{X}$ includes the quaternion components and angular velocity components:

$$\mathbf{X} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

### Input Vector (u)

The input vector $\mathbf{u}$ consists of the control moments:

$$\mathbf{u} = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}$$

The state-space representation combines the quaternion and angular velocity dynamics into a single framework. It is given by:

$$\dot{\mathbf{X}} = A\mathbf{X} + B\mathbf{u}$$

where:

$$A = \begin{bmatrix} 0 & -\frac{1}{2}\omega_x & -\frac{1}{2}\omega_y & -\frac{1}{2}\omega_z & 0 & 0 & 0 \\ \frac{1}{2}\omega_x & 0 & \frac{1}{2}\omega_z & -\frac{1}{2}\omega_y & 0 & 0 & 0 \\ \frac{1}{2}\omega_y & -\frac{1}{2}\omega_z & 0 & \frac{1}{2}\omega_x & 0 & 0 & 0 \\ \frac{1}{2}\omega_z & \frac{1}{2}\omega_y & -\frac{1}{2}\omega_x & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.5\left(\frac{I_{zz}-I_{yy}}{I_{xx}}\right)\omega_y & -0.5\left(\frac{I_{zz}-I_{yy}}{I_{xx}}\right)\omega_z \\ 0 & 0 & 0 & 0 & -0.5\left(\frac{I_{xx}-I_{zz}}{I_{yy}}\right)\omega_z & 0 & -0.5\left(\frac{I_{xx}-I_{zz}}{I_{yy}}\right)\omega_x \\ 0 & 0 & 0 & 0 & -0.5\left(\frac{I_{yy}-I_{xx}}{I_{zz}}\right)\omega_x & -0.5\left(\frac{I_{yy}-I_{xx}}{I_{zz}}\right)\omega_y & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{I_{xx}} & 0 & 0 \\ 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix}$$

The image presents a linear state-space representation for a control system. The key elements and their explanations are as follows:

- **Output Equation**:

  $$Y = CX + DU$$

  This equation relates the output $Y$ to the state vector $X$ and the control input $U$ through the matrices $C$ and $D$.

- **Matrix $C$**:

$$C = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

The matrix $C$ is a 7x7 identity matrix, indicating that all states are considered as outputs directly.

- **Matrix $D$**:

$$D = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

The matrix $D$ is a 6x3 zero matrix, indicating that the control input $U$ does not directly affect the output $Y$.

## 3.5 Linearization of Non-Linear System

To linearize a non-linear system using the Jacobian linearization method, we start with the non-linear state-space representation of the system.

### Non-linear State-Space Representation

For a spacecraft, the non-linear state-space equations can be represented as:

$$\dot{x} = f(x, u)$$
$$y = g(x, u)$$

where:

- $x$ is the state vector,
- $u$ is the input vector (control torques),
- $f(x, u)$ represents the dynamics,
- $g(x, u)$ represents the output equations.

For the spacecraft's rotational dynamics and kinematics, let the state vector include the orientation (Euler angles) and angular velocities.

Let's use Euler angles $(\phi, \theta, \psi)$:

$$\phi : \text{roll}$$
$$\theta : \text{pitch}$$
$$\psi : \text{yaw}$$

The state vector $x$ and input vector $u$ are defined as:

$$x = \begin{bmatrix} \phi \\ \theta \\ \psi \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad u = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}$$

The non-linear dynamics are given by the following equations:

$$\dot{\omega}_x = \frac{1}{I_{xx}} \left( M_x - (I_{yy} - I_{zz})\omega_y \omega_z \right)$$

$$\dot{\omega}_y = \frac{1}{I_{yy}} \left( M_y - (I_{zz} - I_{xx})\omega_z \omega_x \right)$$

$$\dot{\omega}_z = \frac{1}{I_{zz}} \left( M_z - (I_{xx} - I_{yy})\omega_x \omega_y \right)$$

where:

- $I_{xx}, I_{yy}, I_{zz}$ are the moments of inertia about the x, y, and z axes, respectively.

The angular velocity vector of the body frame relative to the inertial frame $\omega_{BI}$ can be expressed as:

$$\omega_{BI} = \omega_{BR} + \omega_{RIB}$$

- $\omega_{BR}$ is the angular velocity vector of the body frame relative to the reference frame.
- $\omega_{RIB}$ is the angular velocity vector of the reference frame relative to the inertial frame.

## Transformation Using ZYX Euler Angles

For the transformation $\psi - \theta - \phi$:

The non-linear kinematic equations can be derived from the rotation matrices and the transformation between Euler angles and angular velocity components.

## Angular Velocity to Euler Angle Rates Transformation

For the transformation order $\phi - \theta - \psi$:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi \tan\theta & \cos\phi \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi / \cos\theta & \cos\phi / \cos\theta \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

The equations for the angular velocities are:

$$\dot{\phi} = \omega_x + (\omega_y \sin\phi + \omega_z \cos\phi) \tan\theta$$

$$\dot{\theta} = \omega_y \cos\phi - \omega_z \sin\phi$$

$$\dot{\psi} = \frac{\omega_y \sin\phi + \omega_z \cos\phi}{\cos\theta}$$

### Linearization Around Equilibrium Point

To linearize these equations around the equilibrium point $(\phi, \theta, \psi) = (0, 0, 0)$, the partial derivatives of each equation with respect to the state variables are calculated:

$$\frac{\partial \dot{\phi}}{\partial \phi} = 0, \quad \frac{\partial \dot{\phi}}{\partial \theta} = 0, \quad \frac{\partial \dot{\phi}}{\partial \psi} = 0$$

$$\frac{\partial \dot{\phi}}{\partial \omega_x} = 1, \quad \frac{\partial \dot{\phi}}{\partial \omega_y} = 0, \quad \frac{\partial \dot{\phi}}{\partial \omega_z} = 0$$

$$\frac{\partial \dot{\theta}}{\partial \phi} = 0, \quad \frac{\partial \dot{\theta}}{\partial \theta} = 0, \quad \frac{\partial \dot{\theta}}{\partial \psi} = 0$$

$$\frac{\partial \dot{\theta}}{\partial \omega_x} = 0, \quad \frac{\partial \dot{\theta}}{\partial \omega_y} = \cos\phi \approx 1, \quad \frac{\partial \dot{\theta}}{\partial \omega_z} = -\sin\phi \approx 0$$

$$\frac{\partial \dot{\psi}}{\partial \phi} = 0, \quad \frac{\partial \dot{\psi}}{\partial \theta} = 0, \quad \frac{\partial \dot{\psi}}{\partial \psi} = 0$$

$$\frac{\partial \dot{\psi}}{\partial \omega_x} = 0, \quad \frac{\partial \dot{\psi}}{\partial \omega_y} = \frac{\sin\phi}{\cos\theta} \approx 0, \quad \frac{\partial \dot{\psi}}{\partial \omega_z} = \frac{\cos\phi}{\cos\theta} \approx 1$$

Remaining elements are zeros.

## 3.6 Linearization Using Jacobian

To linearize the system, we compute the Jacobian matrices of $f(x, u)$ with respect to $x$ and $u$.

The Jacobian matrices are defined as:

$$A = \left.\frac{\partial f}{\partial x}\right|_{(x_0, u_0)}, \quad B = \left.\frac{\partial f}{\partial u}\right|_{(x_0, u_0)}$$

where $x_0$ and $u_0$ are the equilibrium points.

For the rotational dynamics, the partial derivatives (Jacobian elements) are computed as follows:

$$\frac{\partial \dot{\omega}_x}{\partial \omega_x} = 0, \quad \frac{\partial \dot{\omega}_x}{\partial \omega_y} = \frac{-(I_{yy} - I_{zz})}{I_{xx}}\omega_z, \quad \frac{\partial \dot{\omega}_x}{\partial \omega_z} = -\frac{(I_{yy} - I_{zz})}{I_{xx}}\omega_y$$

$$\frac{\partial \dot{\omega}_y}{\partial \omega_x} = -\frac{(I_{zz} - I_{xx})}{I_{yy}}\omega_z, \quad \frac{\partial \dot{\omega}_y}{\partial \omega_y} = 0, \quad \frac{\partial \dot{\omega}_y}{\partial \omega_z} = \frac{-(I_{zz} - I_{xx})}{I_{yy}}\omega_x$$

$$\frac{\partial \dot{\omega}_z}{\partial \omega_x} = -\frac{(I_{xx} - I_{yy})}{I_{zz}}\omega_y, \quad \frac{\partial \dot{\omega}_z}{\partial \omega_y} = -\frac{(I_{xx} - I_{yy})}{I_{zz}}\omega_x, \quad \frac{\partial \dot{\omega}_z}{\partial \omega_z} = 0$$

At the equilibrium point $(\omega_x, \omega_y, \omega_z) \approx (0, 0, 0)$, the elements are:

$$\frac{\partial \dot{\omega}_x}{\partial \omega_x} = 0, \quad \frac{\partial \dot{\omega}_x}{\partial \omega_y} = 0, \quad \frac{\partial \dot{\omega}_x}{\partial \omega_z} = 0$$

$$\frac{\partial \dot{\omega}_y}{\partial \omega_x} = 0, \quad \frac{\partial \dot{\omega}_y}{\partial \omega_y} = 0, \quad \frac{\partial \dot{\omega}_y}{\partial \omega_z} = 0$$

$$\frac{\partial \dot{\omega}_z}{\partial \omega_x} = 0, \quad \frac{\partial \dot{\omega}_z}{\partial \omega_y} = 0, \quad \frac{\partial \dot{\omega}_z}{\partial \omega_z} = 0$$

### Linearized Dynamics Around Equilibrium Point

The linearized dynamics around the equilibrium point are:

$$\dot{\omega}_x = \frac{1}{I_{xx}} M_x$$

$$\dot{\omega}_y = \frac{1}{I_{yy}} M_y$$

$$\dot{\omega}_z = \frac{1}{I_{zz}} M_z$$

The equations for the angular velocities are:

$$\dot{\phi} = \omega_x$$
$$\dot{\theta} = \omega_y$$
$$\dot{\psi} = \omega_z$$

The linearized system can be represented as:

$$\dot{X} = AX + BU$$

$$Y = CX + DU$$

The state-space matrices $A$ and $B$ for the linearized system are given below.

### Matrix $A$

Matrix $A$ is a 6x6 matrix representing the linearized system dynamics:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

### Matrix $B$

Matrix $B$ is a 6x3 matrix representing the control input influence on the system dynamics. Given the inertia matrix $I$:

$$I = \begin{bmatrix} 2.683 & 0 & 0 \\ 0 & 2.326 & 0 \\ 0 & 0 & 1.897 \end{bmatrix}$$

Using the inverse of $I$, the elements of matrix $B$ are computed as:

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.3727 & 0 & 0 \\ 0 & 0.4299 & 0 \\ 0 & 0 & 0.5271 \end{bmatrix}$$

### 3.7 Stability Analysis of the Linearized System

To analyze the stability of the system (linearized system), we find the eigenvalues by solving the characteristic equation.

#### Characteristic Equation

The characteristic equation is given by:

$$|\lambda I - A| = 0$$

where $\lambda$ represents the eigenvalues, $I$ is the identity matrix, and $A$ is the state matrix.

The matrix $\lambda I - A$ for the given system is:

$$\lambda I - A = \begin{bmatrix} \lambda & 0 & 0 & -1 & 0 & 0 \\ 0 & \lambda & 0 & 0 & -1 & 0 \\ 0 & 0 & \lambda & 0 & 0 & -1 \\ 0 & 0 & 0 & \lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda \end{bmatrix}$$

Solving the determinant of the above matrix, we get:

$$\det(\lambda I - A) = 0$$

This leads to the eigenvalues:

$$\lambda_1, \lambda_2, \ldots, \lambda_6 = 0$$

Since all the eigenvalues are zero, the system is marginally stable.

### 3.8 Stabilization Techniques: State Feedback Control

One common method to use is state feedback control.

State feedback control involves designing a control input $u$ as a function of the state vector $x$ to modify the system dynamics. The control law can be designed as:

$$u = -Kx$$

where $K$ is the state feedback gain matrix.

#### Augmented System for Control Design

Given the state-space representation:

$$\dot{x} = Ax + Bu$$

Substituting the control law $u = -Kx$:

$$\dot{x} = Ax + B(-Kx)$$

Simplifies to:

$$\dot{x} = (A - BK)x$$

Thus, the new state matrix is $A - BK$.

## Designing State Feedback Control for Stability

We need to choose $K$ such that the eigenvalues of $(A - BK)$ are in the left half of the complex plane, ensuring stability.

### Steps for Designing State Feedback Control

1. **Define Desired Eigenvalues:** Define the desired eigenvalues for the closed-loop system to ensure stability.

2. **Compute State Feedback Gain Matrix:** Compute the state feedback gain matrix $K$ using methods like pole placement or Linear Quadratic Regulator (LQR).

   *Use the `place` command for pole placement technique.*

3. **Closed-Loop System:** With the gain matrix $K$, the closed-loop system becomes:

$$\dot{x} = (A - BK)x$$

   This approach is also called a regulatory problem since there is no reference input ($r = 0$).

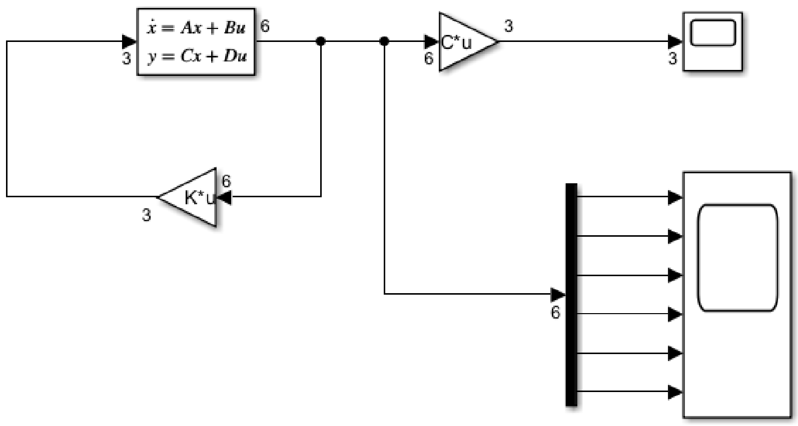


Figure 4: Block diagram of Regulatory System

## Augmented State-Space Representation

The process of augmenting the state-space matrices includes adding additional states, such as the integral of the output errors, to improve control performance by minimizing steady-state error and ensuring the system accurately tracks reference commands.

1. **Augmented Matrices:**

   - The augmented matrix $\bar{A}$ is constructed as follows:

$$\bar{A} = \begin{bmatrix} A & \text{zeros}(6,3) \\ -C & \text{zeros}(3,3) \end{bmatrix}$$

   - The augmented matrix $\bar{B}$ is constructed as follows:

$$\bar{B} = \begin{bmatrix} B \\ \text{zeros}(3,3) \end{bmatrix}$$

2. **Finding $K$ Matrix:**

   - The $K$ matrix is found using the ‘place‘ command:

$$K = \text{place}(\bar{A}, \bar{B}, \text{desired\_poles})$$

3. **Augmented Plant Dynamics:**

   - These dynamics include additional states (integral of the output errors) to improve control performance, minimize steady-state error, and ensure accurate tracking of reference commands.

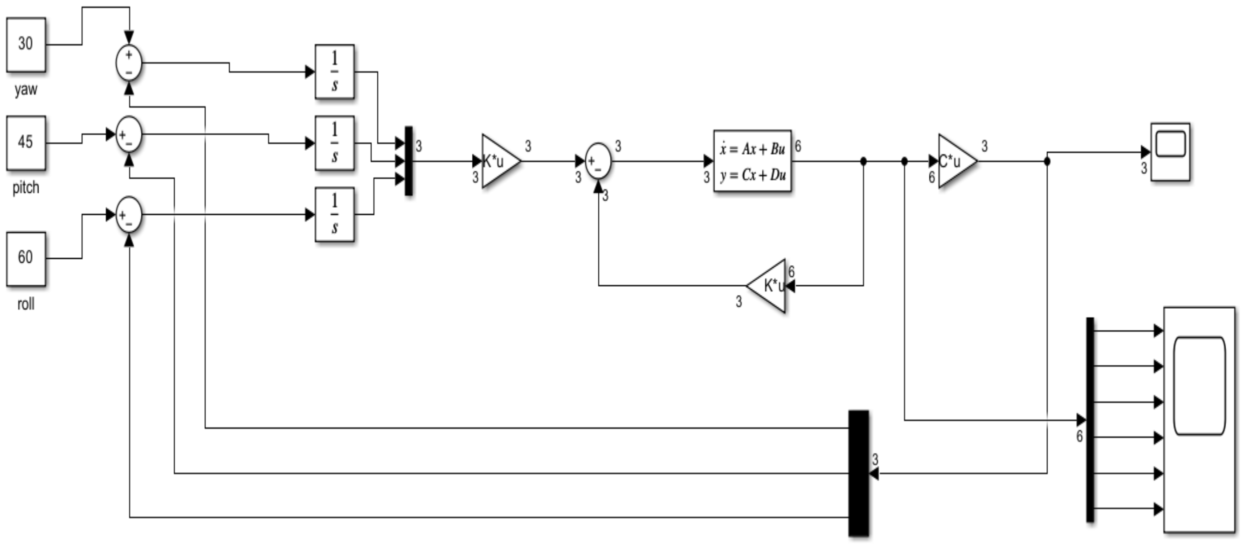


Figure 5: Block diagram of the State feedback control scheme

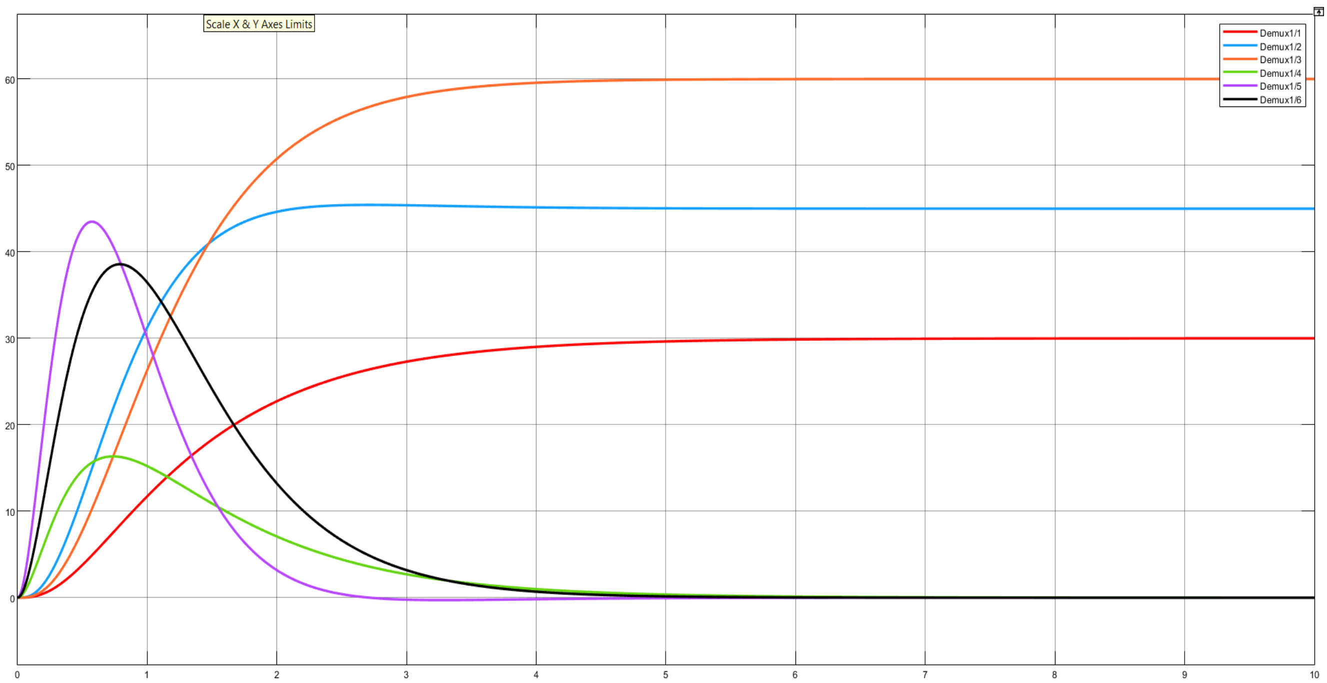


Figure 6: Simulation results of the State feedback control scheme

### 3.9 Three-Axis Attitude Control System

A three-axis attitude control system is an essential component in spacecraft to ensure that it maintains the desired attitude angles in space. This system controls the spacecraft's orientation by adjusting its pitch, roll, and yaw angles.

Sensors:

- **Gyroscopes**: Measure the angular velocity of the spacecraft in all three axes.
- **Star Trackers**: Determine the spacecraft's orientation by tracking stars.
- **Sun Sensors**: Provide orientation information relative to the Sun.
- **Magnetometers**: Measure the local magnetic field to help determine orientation.

Control Algorithms

- **Proportional-Derivative (PD) Controllers**: These controllers provide a control signal that is proportional to the error and the rate of change of the error.
- **State Feedback Controllers**: Use the state vector of the system to compute the control input.
- **Adaptive Controllers**: Adjust the control parameters in real-time to handle uncertainties and variations in the system dynamics.

Actuators

- **Reaction Wheels**: Spin up or down to create a torque that changes the spacecraft's orientation.
- **Control Moment Gyroscopes (CMGs)**: Use gimbaled flywheels to generate large torques.
- **Thrusters**: Provide small bursts of force to adjust the spacecraft's orientation.

Operation of the Three-Axis Attitude Control System

The sensors continuously measure the current orientation of the spacecraft. The measurements are processed to determine the current attitude (pitch, roll, yaw). The control system calculates the error between the desired orientation (set points) and the current orientation.This error is used to determine the necessary corrective actions. The control algorithm computes the control signals (torques or forces) required to minimize the orientation error. These control signals are sent to the actuators. The actuators apply the necessary torques or forces to adjust the spacecraft's orientation. Reaction wheels and CMGs are typically used for fine adjustments, while thrusters are used for larger corrections. The system continuously monitors the spacecraft's orientation and updates the control signals in real-time. This feedback loop ensures that the spacecraft maintains its desired orientation.

**Actuators in Spacecraft Attitude Control** Actuators are devices that produce the necessary forces and torques to control the orientation of a spacecraft. The primary types of actuators used in spacecraft attitude control systems are reaction wheels, control moment gyroscopes (CMGs), and thrusters. Each type of actuator has its own unique mechanism and applications.

**Reaction Wheels:** Reaction wheels are flywheels that spin around an axis. By changing the spin rate of the wheel, a spacecraft can change its orientation due to the conservation of angular momentum. Reaction wheels are effective for fine attitude adjustments and are commonly used in many spacecraft due to their precision and reliability.

- **Principle**: Conservation of angular momentum.
- **Advantages**: High precision, no fuel consumption, long operational life.

- **Disadvantages**: Limited by the maximum stored angular momentum, can become saturated.
- **Applications**: Used in small to medium-sized satellites for precise pointing tasks like imaging, communications, and scientific observations.

**Control Moment Gyroscopes (CMGs)** CMGs consist of a spinning rotor mounted on gimbals. By tilting the rotor using the gimbals, a torque is generated that changes the spacecraft's orientation. CMGs are capable of producing large torques and are used for rapid and substantial attitude changes.

- **Principle**: Gyroscopic effect and angular momentum transfer.
- **Advantages**: Can produce large torques, efficient for large and rapid maneuvers.
- **Disadvantages**: Complex mechanical system, potential for mechanical failure.
- **Applications**: Used in larger spacecraft and space stations where significant and quick attitude adjustments are required.

**Thrusters:**

Thrusters generate force by expelling mass (usually gas) at high velocity, according to Newton's third law of motion. They are used for both translational and rotational maneuvers and can provide significant control authority.

- **Principle**: Newton's third law of motion.
- **Advantages**: High control authority, versatile for both attitude and orbit control.
- **Disadvantages**: Limited by fuel supply, less precise compared to reaction wheels.
- **Applications**: Used in spacecraft requiring large or sustained changes in orientation or position, such as during orbital insertion, station-keeping, and deorbiting.

### 3.10 Simulink Diagrams

The provided Simulink diagram represents a control system for a spacecraft's three axes: pitch, roll, and yaw. Here is a breakdown of the diagram:

- **Inputs:**
  - Three constant blocks are used to set the desired setpoints (or commands) for the spacecraft's pitch (20), roll (30), and yaw (10) angles.
- **Processing Block:**
  - A central processing block labeled with *Pitch*, *Roll*, and *Yaw* takes these inputs and processes them to compute the corresponding attitude angles. This block represents the control algorithm that calculates the required adjustments to achieve the desired orientations.
- **Outputs:**
  - The processed signals are then outputted to three separate blocks: *pitch*, *roll*, and *yaw*. These outputs indicate the controlled attitude angles for the spacecraft's three axes.

The diagram illustrates a simple control scheme set up for three-axis control (pitch, roll, and yaw) of a spacecraft.

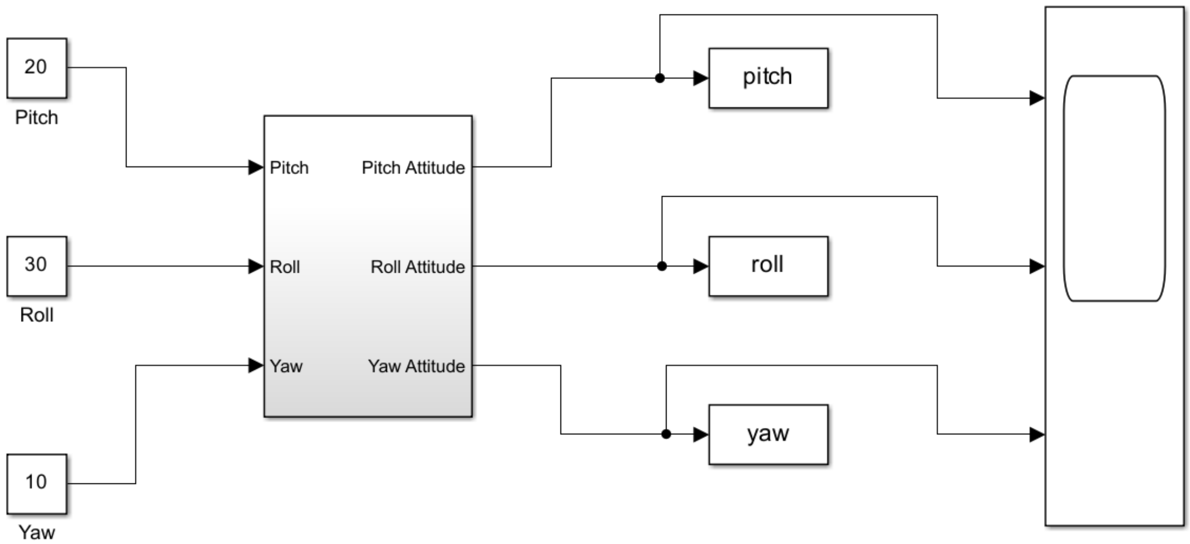


Figure 7: Main simulink Diagram for Three-Axis Control

The provided Simulink diagram represents a system for controlling three axes of a spacecraft: pitch, roll, and yaw. The three constant blocks (inputs) represent the desired set points (or commands) for the spacecraft's pitch, roll, and yaw. In summary, the diagram shows a simple control scheme set up for three-axis control (pitch, roll, and yaw).

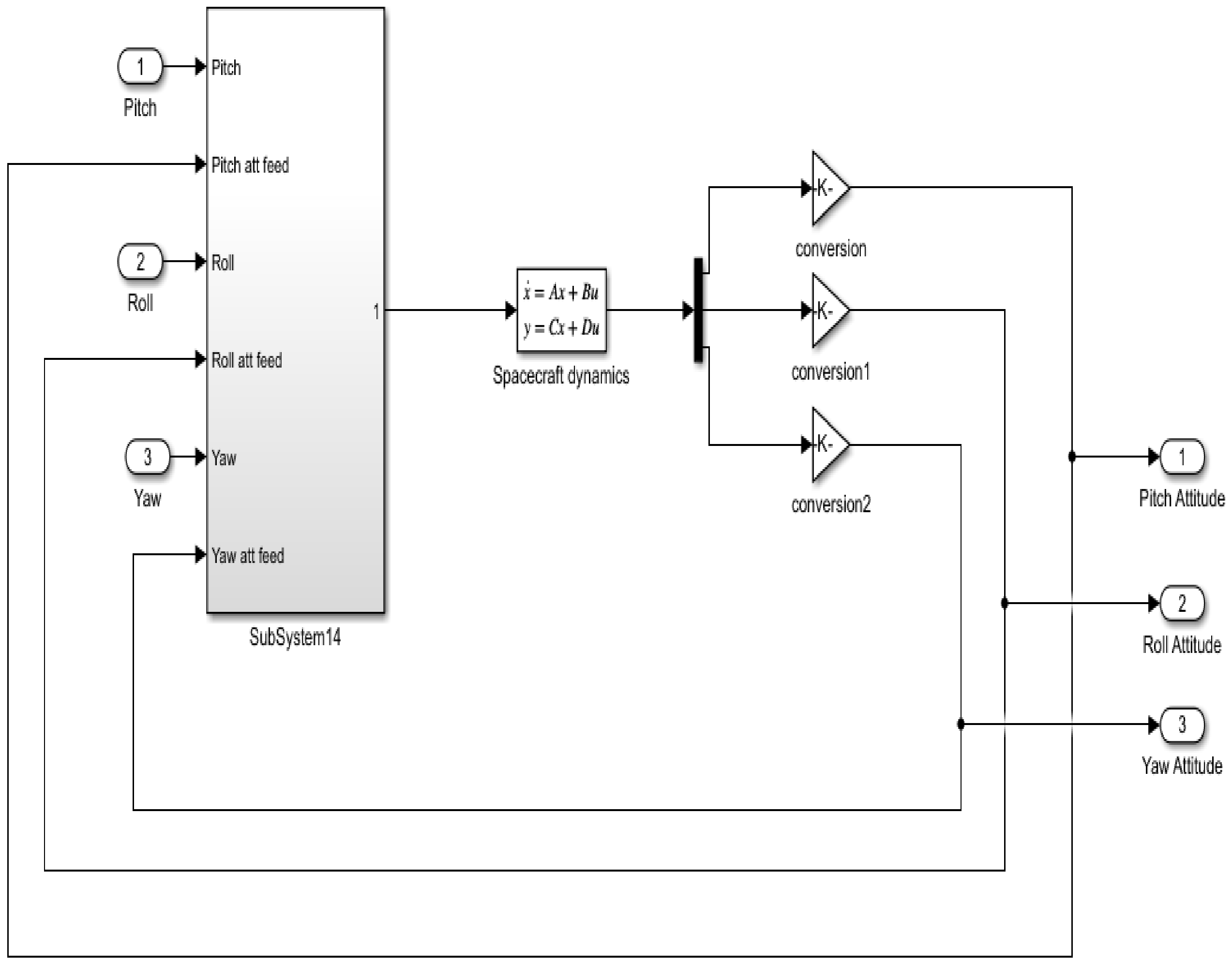


Figure 8: Simulink Diagram for subsystem present in main diagram

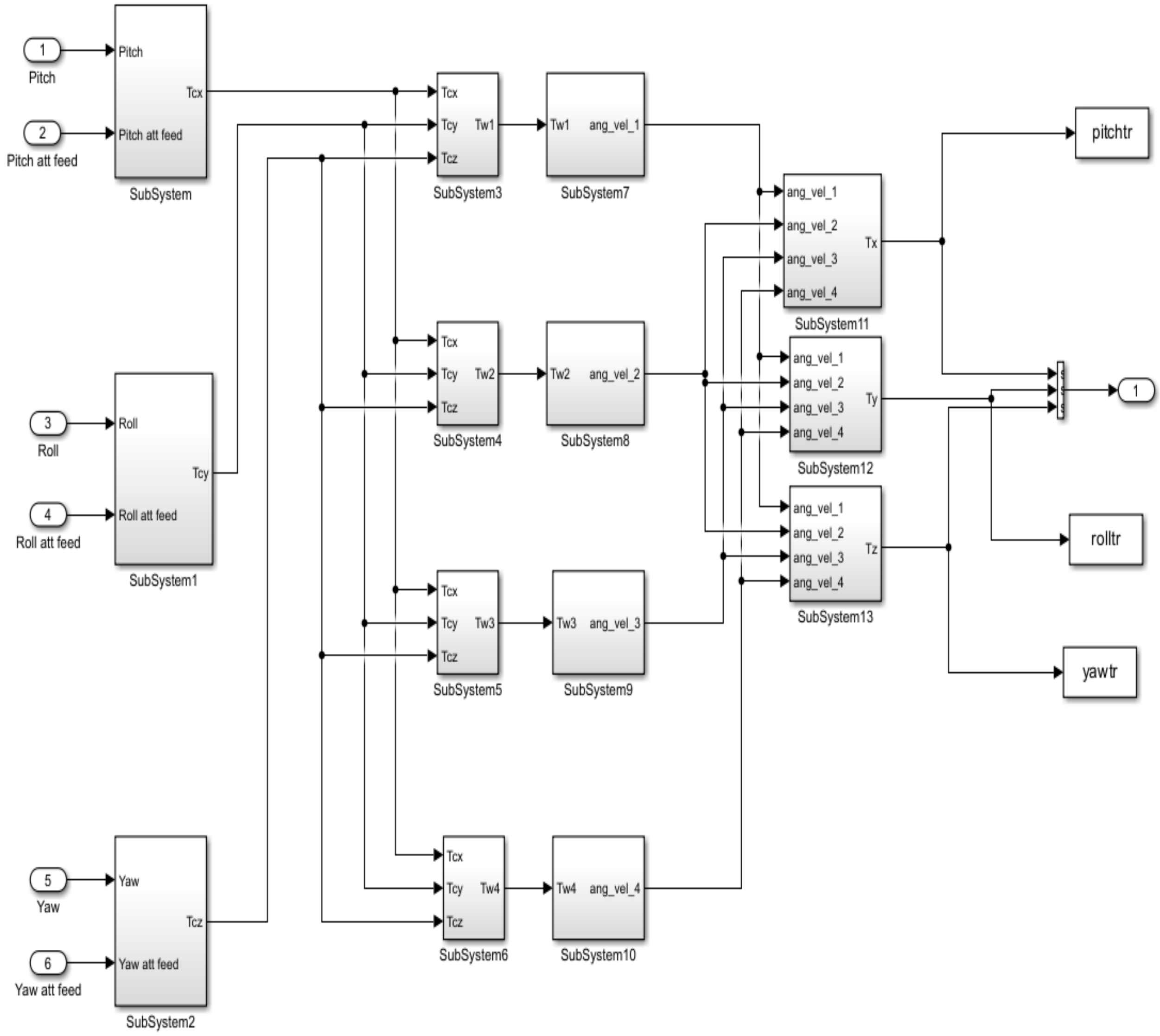


Figure 9: Simulink Diagram for susbsystem 14

### Simulink Diagram of subsystem 1

The provided Simulink diagram represents a control system for the pitch axis of a spacecraft. Here is a detailed explanation of the diagram:

- **Inputs:**
  - *Pitch*: The desired pitch angle setpoint.
  - *Pitch att feed*: The actual pitch attitude feedback from the spacecraft.
- **Processing Blocks:**
  - A summing block calculates the pitch error by subtracting the actual pitch attitude feedback from the desired pitch angle. This error signal is labeled *pitcherror*.
  - The error signal is then passed through a conversion block, which scales the error signal to appropriate units.
  - The scaled error signal is multiplied by a proportional gain ($kp$) to generate a control signal.

– A second control path involves the rate of change of the pitch error, scaled by a rate gain ($kr$). This is added to the proportional control signal.

– The combined control signal is then passed through a transfer function block, represented by *num(s)/den(s)*, which models the dynamics of the actuator or control system.

– The output of the transfer function block is labeled *Tcx*, representing the control torque applied to the pitch axis.

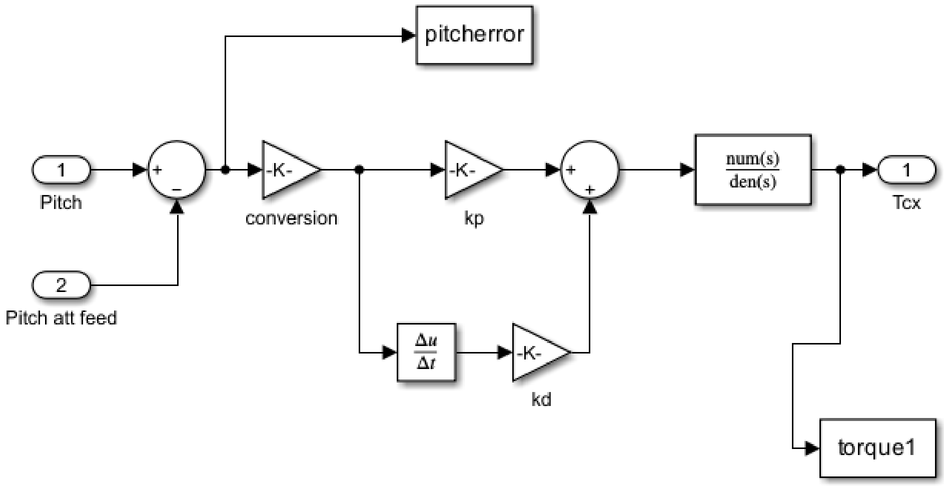


Figure 10: Simulink Diagram for subsystem1

The pitch and pitch feedback are compared to get the error signal, which is the difference between the desired pitch and the actual pitch. The error signal is then passed through a conversion block. This block is used to convert the error signal into appropriate units. The converted error signal is then multiplied by a proportional gain ($kp$) to get the proportional control action. Additionally, the rate of change of the error signal is taken and multiplied by a derivative gain ($kr$) to get the derivative control action. The proportional and derivative control actions are added together. The combined signal is then passed through a transfer function block that models the dynamics of the actuator or control system, resulting in the control torque output.

**Torque Control and Conversion to Angular Velocities(Simulink Diagram of Subsystem 3:**

The provided Simulink diagram represents a subsystem that sums the control torques along the three axes (pitch, roll, and yaw) of a spacecraft. Here is a detailed explanation of the diagram:

- **Inputs:**
  – $Tcx$: Control torque for the pitch axis.
  – $Tcy$: Control torque for the roll axis.
  – $Tcz$: Control torque for the yaw axis.

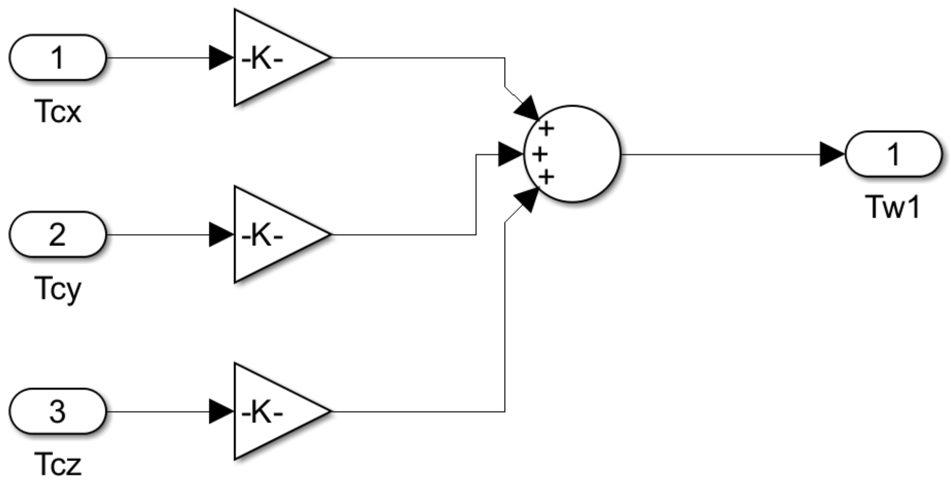


Figure 11: Simulink Diagram for Summing Control Torques

Subsystems 3, 4, 5, and 6 take the control torques as inputs and sum them up to get the wheel torques, which will be useful for the satellite's orientation. Since direct torque commands cannot be given to the reaction wheels, the wheel torques are converted into angular velocities by integrating them in subsystems 7, 8, 9, and 10. The control torques *Tx*, *Ty*, and *Tz* are summed together using a summing block and the resulting combined torque is outputted to the block labeled *Tw1*, *Tw2*, *Tw3*, and *Tw4* representing the total control torque distributed to generate individual wheel torques *Tw1*, *Tw2*, *Tw3*, and *Tw4*. These wheel torques are converted into angular velocities using integrators because the reaction wheels respond to angular velocity commands rather than torque commands.

## Simulink Diagram of Subsystem 7

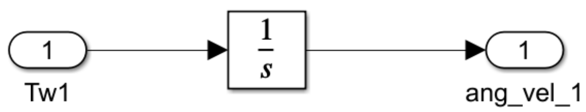


Figure 12: Subsystem 7 Diagram

**Why do we need an integrator?**
Torque to angular velocity conversion
Torque ($\tau$) is related to the angular acceleration ($\alpha$) by the moment of inertia ($I$), as follows

$$\tau = I\alpha$$

Angular acceleration is the derivative of angular velocity ($\omega$)

$$\alpha = \frac{d\omega}{dt}$$

Therefore integrating the torque gives the change in angular velocity.

$$\omega = \int \alpha \, dt = \int \frac{\tau}{I} \, dt$$

Therefore, the integrator converts the input torque (which dictates how fast the wheel should spin up or down) into an angular velocity command for the reaction wheel.
The integrator accumulates the torque over time to determine the angular velocity that the reaction wheel should achieve. This is important for controlling the wheel's speed and consequently, the satellite's orientation.

## Why this step is crucial?

**Wheel Dynamics:**
Reaction wheel control the satellite's orientation by spinning up or down (or by changing their spin rate). The rate of spin (angular velocity) determines the torque exerted on the satellite.
The reaction wheels need angular velocity commands because their motors are controlled based on how fast the wheel should spin, not directly by the torque.

**Attitude Control:**
Accurate attitude control requires precise adjustments to the wheel speed. By integrating the torque to get angular velocity, the control system can specify the exact speed needed to achieve the desired orientation.
This step ensures smooth and precise control over the satellite's attitude by continuously adjusting the wheel's speed based on the control torques.
* The motors in the reaction wheels are typically controlled by specifying a target speed, not directly by torque.

**Specific commands required for reaction wheels:**
The reaction wheels require specific commands to generate the necessary torques for attitude control:

The reaction wheels operate based on angular velocity commands because the motors in the wheels are typically controlled by specifying a target speed.

## Simulink Diagram of Subsystems 11, 12, 13

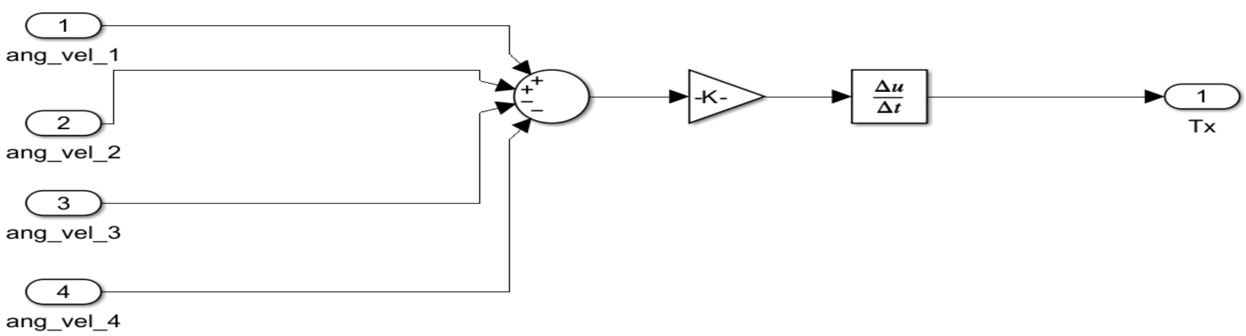


Figure 13: Simulink Diagram for Subsystems 11, 12, 13

These angular velocities are summed up and then further differentiated to get the torques. These torques are nothing but torques produced by reaction wheels.
Finally, those torques are taken out and given as input to state space modelled spacecraft dynamics.

### 3.11 Simulation Results

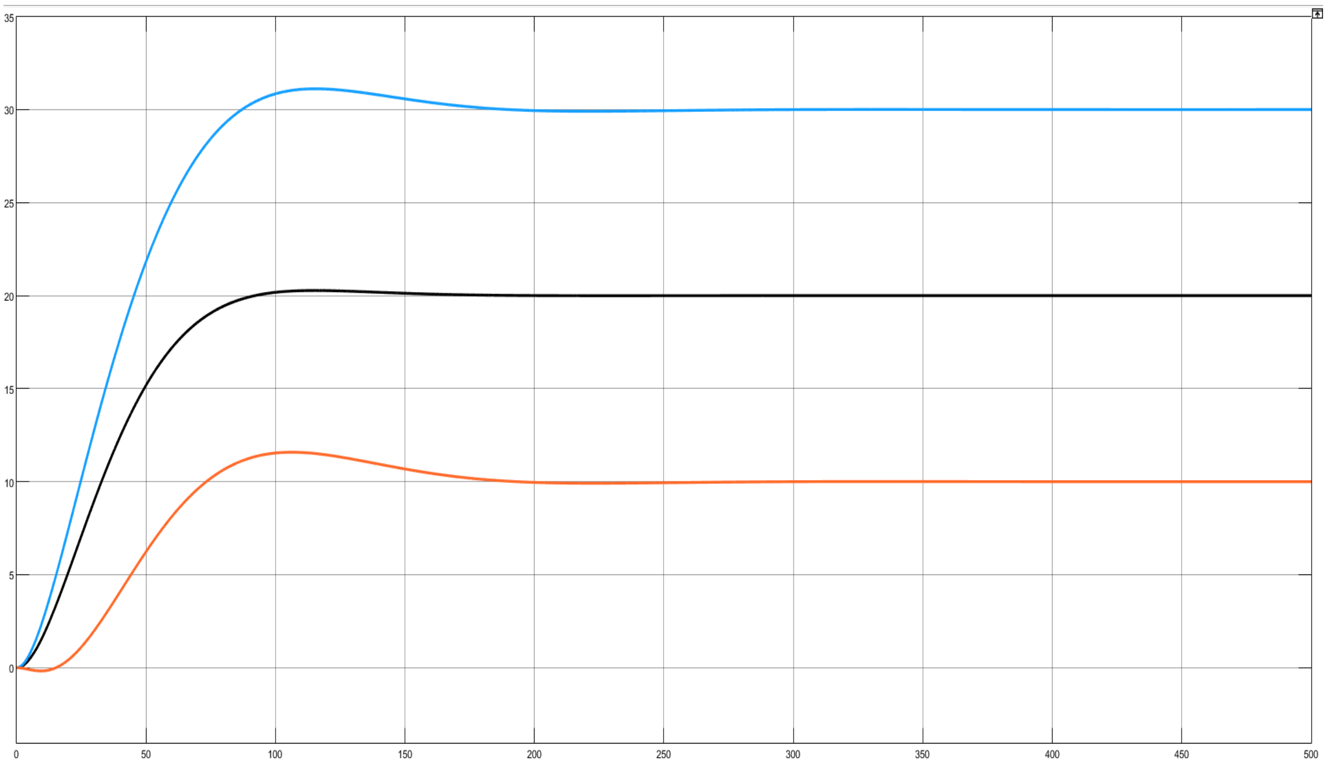


Figure 14: Simulation Results

The simulation results demonstrate that the three-axis attitude control system effectively stabilizes the satellite's orientation. The control system handles the transient response well, with acceptable overshoot and quick settling times. The final stable values for pitch, roll, and yaw are consistent with the desired set points, indicating successful attitude control.

# 4 ADAPTIVE CONTROLLER DESIGN FOR SATELLITE ATTACHED BY NON-COOPERATIVE OBJECT

## 4.1 Spacecraft model and non-cooperative attachment model

**Spacecraft attitude dynamics:**

The rigid spacecraft attitude dynamics can be expressed by the equation:

$$J\dot{\omega} + (\omega \times J\omega) = L_c + L_{dist} \quad (1)$$

where
$\omega \in \mathbb{R}^3$ angular rate vector of the spacecraft's body coordinate system relative to the inertial coordinate system.

$$\omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$L_c, L_{dist} \in \mathbb{R}^3$ are the torque vector and disturbance torque vector respectively.
$J$ is the inertia matrix of the spacecraft.

The inertia matrix is assumed to be a third-order non-symmetric matrix. Thus inertia matrix is expressed as follows:

$$J = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix} \quad (2)$$

**Spacecraft attitude kinematics:**

The spacecraft attitude kinematic equations are described by the quaternion kinematic equation expressed as follows.

$$\dot{q}_v = \frac{1}{2}[\Lambda]q_v \quad (3)$$

where

$$q_v \triangleq \begin{bmatrix} q_0 \\ q \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^3$$

$$q^T q + q_0^2 = 1$$

Quaternion vector $q_v = [q_0, q_1, q_2, q_3]^T$ is a quaternion representing the spacecraft's orientation. Quaternions provide a way to describe rotations without the singularity issues that Euler angles suffer from.

**Error quaternion ($q_{ev}$):**

$$q_{ev} = [q_{e0}, q_{e1}, q_{e2}, q_{e3}]^T$$

is a quaternion representing the the difference between the current quaternion and the target quaternion.
It indicates the error in the spacecraft's orientation, which has to be minimized by the control scheme.

**Target quaternion $q_{cv}$**

$$q_{cv} = [q_{c0}, q_{c1}, q_{c2}, q_{c3}]^T$$

represents the desired orientation of the spacecraft.

The control scheme uses this target quaternion to determine the necessary torque to align the spacecraft to the desired orientation.

$$q_{cv} \triangleq [q_{c0}, q_c^T]^T \in \mathbb{R} \times \mathbb{R}^3$$
$$q_c^T q_c + q_{c0}^2 = 1$$

$$q_{ev} \triangleq [q_{e0}, q_e^T]^T \in \mathbb{R} \times \mathbb{R}^3$$

$$q_{ev} = \begin{bmatrix} q_{e0} \\ q_e \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_{cv}^T[\Lambda]q_v \\ [\Lambda^T(q_{cv})][\Lambda]q_v \end{bmatrix} \omega$$

$$\dot{q}_{ev} = \frac{1}{2}[\Lambda]q_{ev}\omega \quad (4)$$

## 4.2 Non-cooperative attachment model:

Non-cooperative attachment primarily refers to situations where the attached object's dimensions, centroid position, and momentum characteristics are unknown.

### Control challenges

When a non-cooperative body attaches to the target spacecraft, the resulting combined system becomes difficult to control. This is due to the significant disturbances and unknown inertia parameters introduced by the attachment.

The specific issues highlighted in this section include:

### a) Unknown outline dimensions

The attaching object's size and shape are not known, making it challenging to predict the impact on the combined system.

### b) Unknown centroid position

The center of mass of the attaching object is not known, which affects the overall balance and stability of the system.

### c) Unknown movement

The dynamics or movement patterns of the attaching object are not known, adding to the unpredictability of the system.

These unknown factors lead to significant disturbances and introduce unknown inertia parameters, making the control of the new combined system very challenging.

$\omega_0$ - orbit angular velocity.

O - $(x_0, y_0, z_0)$ is the fixed coordinate system of target spacecraft.

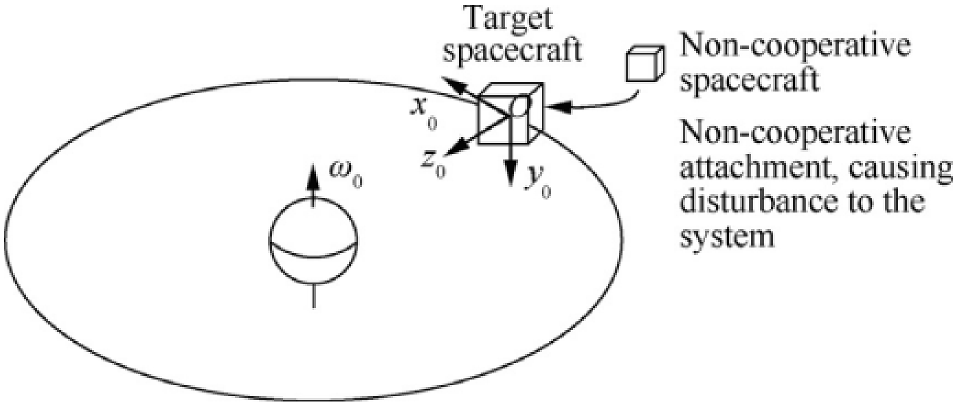


Figure 15: Schematic diagram of non-cooperative attachment

### 4.3 Controller Design Constraints

a) **Bounded Disturbance Torque:**
The disturbance torque $L_{\text{dist}}$ is bounded, meaning that its magnitude does not exceed a certain unknown constant $L_{\max}$.

$$\|L_{\text{dist}}\| \leq L_{\max}$$

upper Bounded

b) **Unknown Inertia Matrix:**
The inertia matrix $J$ of the spacecraft is unknown but is assumed to be a positive constant matrix.

This means that while the exact values of the inertia matrix elements are not known, the matrix remains constant and positive definite.

c) **Non-cooperative attachment:**
The specific process of non-cooperative body attachment is not considered in detail. Only the states/scenarios before and after attachment are taken into account.
This simplifies the control design by focusing on the overall effect of attachment rather than the attachment process itself.

d) **Attitude Maneuver control objectives:**
After the non-cooperative attachment, the goal is to make the combined system reach a stable state as soon as possible.
Additionally, the control torque applied to achieve this should be limited, ensuring the system does not apply excessively large torques that could cause damage or instability.

### 4.4 Adaptive attitude control

After a non-cooperative attachment, the original system is disturbed due to the unknown inertia and momentum carried by non-cooperative spacecraft. When the carried inertia and momentum are too large, the system control scheme may fail and the control of the system may be lost. To solve this problem, let the control torque of the target spacecraft meet the following adaptive control law:

$$L_c = -\frac{1}{2}x_1 - Y\hat{h} - kx_2 - L_{dist} \quad (5)$$

$$-\frac{1}{2}x_1 \Rightarrow \text{feedback term based on state variable } x_1.$$

$$-Y\hat{h} \Rightarrow \text{adaptive term based on the estimated inertia } (\hat{h})/(\hat{\theta})$$

$$-kx_2 \Rightarrow \text{proportional control term with positive gain } k.$$

$$-L_{dist} \Rightarrow \text{compensation for disturbance torque } L_{dist}.$$

$$Y \Rightarrow \text{coefficient matrix.}$$

$$\tilde{\theta} = D^{-1}Y^T x_2 \quad (6)$$

Equation (6) defines an update law for the estimated inertia $\hat{h}/\tilde{\theta}$, where $D$ is a positive definite matrix.

$x_1, x_2$ are state variables of three dimensions.

$$x_1 = q_e$$

$$x_2 = a \cdot x_1 + \omega = a \cdot q_e + \omega, a > 0 \quad (7)$$

$\tilde{\theta}$ is inertia estimation.

$$\tilde{\theta} = \begin{bmatrix} \tilde{J}_{11} & \tilde{J}_{12} & \tilde{J}_{13} & \tilde{J}_{21} & \tilde{J}_{22} & \tilde{J}_{23} & \tilde{J}_{31} & \tilde{J}_{32} & \tilde{J}_{33} \end{bmatrix}^T \quad (8)$$

$a \Rightarrow$ positive coefficient

$\tilde{\theta}$ is the estimation of the changing rate of the inertia.

We can prove that although a non-cooperative spacecraft attaches to the target spacecraft, the new combination can still reach a stable state using the control law outlined before.

**Proof:**

Take Lyapunov function as

$$V = \frac{1}{2}\left[x_1^T x_1 + (1 - q_{e0})^2 + x_2^T J x_2 + \Delta_0^T D \Delta_0\right]$$

$$+\frac{1}{2}\left[x_2^T J x_2 + \Delta_0^T D \Delta_0\right] > 0$$

$V > 0$ defines that the Lyapunov function is positive definite.

$\Delta_0$ is defined as the estimation error of the spacecraft inertia parameter.

$$\Delta_0 \triangleq \Theta - \tilde{\Theta}$$

Let $V_1 = \frac{1}{2}\left[x_1^T x_1 + (1 - q_{e0})^2\right]$
Thus, $V_1 = \frac{1}{2}\left[q_{e1}^2 + q_{e2}^2 + q_{e3}^2 + 1 - 2q_{e0} + q_{e0}^2\right]$

$$q_e^T q_e + q_{e0}^2 = 1$$

$$q_{e1}^2 + q_{e2}^2 + q_{e3}^2 + q_{e0}^2 = 1$$

$$V_1 = \frac{1}{2}(2 - 2q_{e0}) = 1 - q_{e0}$$

$$\therefore V_1 = 1 - q_{e0} \geq 0$$

$$\dot{V}_1 = -q_{e0} = -\frac{1}{2}\left[-q_{e1} - q_{e2} - q_{e3}\right]\omega \quad (11)$$

As $x_1 = q_e$

$$x_2 = \alpha x_1 + \omega \Rightarrow \omega = x_2 - \alpha x_1$$

$$\dot{V}_1 = \frac{1}{2} x_1^T(\omega) = -\frac{1}{2} q_e^T \omega$$

$$\dot{V}_1 = \frac{1}{2} x_1^T(x_2 - \alpha x_1)$$

$$\dot{V}_1 = \frac{1}{2} x_1^T x_2 - \frac{1}{2} \alpha x_1^T x_1 \quad (12)$$

$$\dot{V} = \dot{V}_1 + \frac{1}{2}\left(\dot{x}_2^T J x_2 + x_2^T J \dot{x}_2\right) + \frac{1}{2}\left(\dot{\Delta}_0^T D \Delta_0 + \Delta_0^T D \dot{\Delta}_0\right)$$

$$= -\frac{1}{2} \alpha x_1^T x_1 + \frac{1}{2} x_1^T x_2 + x_2^T J \dot{x}_2 + \Delta_0^T D \dot{\Delta}_0 \quad (13)$$

From equation (7), we can establish that

$$x_2 = \alpha x_1 + \omega$$

$$\dot{x}_2 = \alpha \dot{x}_1 + \dot{\omega} \Rightarrow \dot{x}_2 = \alpha \dot{q}_e + \dot{\omega} \quad (14)$$

Multiplying left and right hand sides by $J$, we easily obtain

$$J\dot{x}_2 = \alpha J \dot{q}_e + J\dot{\omega}$$

$$J\dot{x}_2 = J\dot{\omega} + J\alpha\dot{q}_e$$

$$J\dot{x}_2 = -\omega \times J\omega + L_c + L_{dist} + \alpha J \dot{q}_e \quad (15)$$

Let

$$J\dot{x}_2 = Y\Theta + L_c + L_{dist}$$

where the inertia parameters of the spacecraft are defined as

$$\Theta = \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{21} & J_{22} & J_{23} & J_{31} & J_{32} & J_{33} \end{bmatrix}^T$$

Inertia coefficient matrix, obtained as:

$$Y = \begin{bmatrix} -\alpha q_{e1} & -\alpha q_{e2} & \alpha q_{e3} & \omega_1\omega_3 & \omega_2\omega_3 & \omega_3^2 & -\omega_2\omega_3 & -\omega_2^2 & -\omega_2\omega_3 \\ -\omega_1\omega_3 & -\omega_2\omega_3 & -\omega_3^2 & -\alpha q_{e1} & -\alpha q_{e2} & \alpha q_{e3} & \omega_1^2 & \omega_1\omega_2 & \omega_1\omega_3 \\ \omega_1\omega_2 & \omega_2^2 & \omega_2\omega_3 & -\omega_1^2 & -\omega_2\omega_3 & -\omega_1\omega_3 & -\alpha q_{e1} & -\alpha q_{e2} & -\alpha q_{e3} \end{bmatrix}$$

Since the inertia of the new combination, after the non-cooperative attachment, is an unknown but constant value (regardless of the case in which non-cooperative spacecraft move on or leave the surface).

$$\dot{\Delta}_0 = \Theta - \tilde{\Theta} = -\tilde{\Theta} \quad (17)$$

Substituting equations (15) and (17) into equation (13), we get

$$\dot{V} = -\frac{1}{2}\alpha\|x_1\|^2 + x_2^T\left(\frac{1}{2}x_1 + Y\tilde{\Theta} + L_c + L_{dist}\right) + x_2^T Y \Delta\Theta - \Delta_0^T D \dot{\tilde{\Theta}}$$

$$\dot{V} = -\frac{1}{2}\alpha\|x_1\|^2 + x_2^T\left(\frac{1}{2}x_1 + \phi\hat{\theta} + L_{dist} + L_c\right) + \Delta_\theta^T D\left(D^{-1}Y^T x_2 - \dot{\hat{\theta}}\right). \quad (18)$$

substitute adaptive control law into equation (18), we get:

$$\dot{V} = -\frac{1}{2}\alpha\|x_1\|^2 - kx_2^T x_2$$

$$\dot{V} = -\frac{1}{2}\alpha\|x_1\|^2 - k\|x_2\|^2 < 0 \quad (19)$$

Therefore, while $k > 0$, $\alpha > 0$, the system is globally asymptotically stable.
i.e., when $t \to \infty$ then $x_1 \to 0$ (state)
$x_2 \to 0$ (convergence)
Further, we get $q_e \to 0$
$\omega \to 0$

In order to compensate for the uncertain disturbance torque $L_{dist}$, the compensation strategy is constructed by the non-linear damping algorithm:
substitute $L_{dist}$ with non-linear damping part $\xi x_2$, then we can get the control law as:

$$L_c = -\frac{1}{2}x_1 - \gamma\hat{\theta} - kx_2 - \xi x_2 \quad (20)$$

$\xi$ is non-linear damping coefficient
substituting equation (20) into eq (18)

$$\dot{V} = -\frac{1}{2}\alpha\|x_1\|^2 - k\|x_2\|^2 + x_2^T L_{dist} - \xi\|x_2\|^2 \quad (21)$$

$\xi$ = positive coefficient
so,

$$\xi^2\|x\|^2 + \|L_{dist}\|^2 \geq \frac{\xi x_2^T L_{dist}}{4} \quad (22)$$

Then $x_2^T L_{dist} \leq \xi\|x_2\|^2 + \frac{\|L_{dist}\|^2}{4\xi}$.
Thus we get $\dot{V}$ as

$$\dot{V} \leq -\frac{1}{2}\alpha\|x_1\|^2 - k\|x_2\|^2 + \|L_{dist}\|^2 + \frac{\|L_{dist}\|^2}{4\xi}$$

$$\dot{V} \leq -\sigma\|x\|^2 + \frac{dm}{4\xi} \quad (23)$$

As $x = \begin{bmatrix} x_1^T & x_2^T \end{bmatrix}^T$

$$\sigma = \min\left\{\frac{1}{2}\alpha, k\right\}$$

$$X_J = \lambda_{\max}(J)$$

$$X_D = \lambda_{\max}(D)$$

$\lambda_{\max}(*)$ is the maximum eigenvalue of a matrix and as we know for spacecraft $X_J \geq 1$.
According to definition of $V$, we can easily get that,

$$V \leq \frac{1}{2}X_J\|x\|^2 + \frac{1}{2}X_D\|\Delta\theta_0\|^2 \leq \frac{1}{2}X_J\|x\|^2 + \mu \quad (24)$$

$\mu$ is the upper limit of $\frac{1}{2} + \frac{1}{2}X_D\|\Delta\theta_0\|^2$.

Thus:

$$-\|x\|^2 \leq -\frac{2V}{X_J} + \frac{2\mu}{X_J} \quad (25)$$

We can get that

$$\dot{V} \leq -\lambda V + \psi \quad (26)$$

Both $\lambda = \frac{2\sigma}{X_J}$ and $\psi = \frac{2\mu}{X_J} + \frac{L_{\max}}{4\xi}$
are positive constants.
We know that the system is globally uniformly asymptotically stable due to the above equation.

## 4.5 Simulation and Evaluation

Let us assume that a non-cooperative spacecraft attaches to a small satellite. The simulation is carried out for the above system in order to verify the robustness and effectiveness of adaptive control scheme for the non-cooperative attachment.

### 4.5.1 Simulation Parameters

The original inertia of the spacecraft is

$$J = \begin{bmatrix} 2.603 & 0 & 0 \\ 0 & 2.326 & 0 \\ 0 & 0 & 1.897 \end{bmatrix} \text{kg} \cdot \text{m}^2$$

$J_1$ is close to inertia of 1:1 attachment and $J_2$ is close to inertia of 1:10 attachment.

Due to the non-cooperative attachment, the inertia of the combined system of satellite and attachment may not symmetrically distributed, so, the simulation parameters of the inertia matrix are selected as follows:

$$J_1 = \begin{bmatrix} 2.683 & 0.22 & 0.43 \\ 0.18 & 2.226 & 0.24 \\ 0.29 & 0.14 & 2.897 \end{bmatrix} \text{kg} \cdot \text{m}^2$$

$$J_2 = \begin{bmatrix} 26.83 & 0.412 & 0.213 \\ 0.214 & 23.26 & 0.192 \\ 0.293 & 0.144 & 18.97 \end{bmatrix} \text{kg} \cdot \text{m}^2$$

The initial parameters of spacecraft attitude are as follows:

- initial attitude angle is: $[10, 10, 10]^\circ$
- initial quaternion matrix is:

$$q_v(0) = [0.9893 \quad 0.0789 \quad 0.0941 \quad -0.0789] \quad (30)$$

- initial angular rate is:

$$\omega(0) = [3 \quad -3 \quad 3]^T \quad (\text{deg/s}) \quad (31)$$

- The control parameters are:

$$\bar{0}.2k = 0.5\xi = 0.2 \quad \{\text{initial parameters}\}$$

- $$D = \text{diag}\{0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01\} \quad (32)$$

- control torque $L_c = [L_{cx} \quad L_{cy} \quad L_{cz}]^T$ is limited and $L_{ci} \leq 0.1\,\text{N-m}\,(i = x, y, z)$.

## 4.6 Simulation Results

The attitude simulation was conducted for a combined system resulting from the attachment of a non-cooperative spacecraft to a small satellite. The simulation was divided into two control processes:

1. **First Process:**
   - The satellite was initially in stable control of itself.
2. **Second Process:**
   - The target satellite was attached to a non-cooperative object, leading to instability.

**Impact of Non-Cooperative Spacecraft:** The non-cooperative spacecraft carries its momentum, which can significantly impact the target satellite and cause a notable change in the angular rate. The default change in the angular rate was assumed to be:

$$\omega = [10\ -10\,10]^T\ (^\circ/\mathrm{s})$$

### 4.6.1 Under Original PD Control

The original satellite attitude control was based on PD control of the quaternion. The control law was:

$$L_c = K_p q + K_d \omega$$

If a non-cooperative attachment occurred, the new system remained under the original PD control. The default angular rate change was given by:

$$\omega = [10\ -10\,10]^T\ (^\circ/\mathrm{s})$$

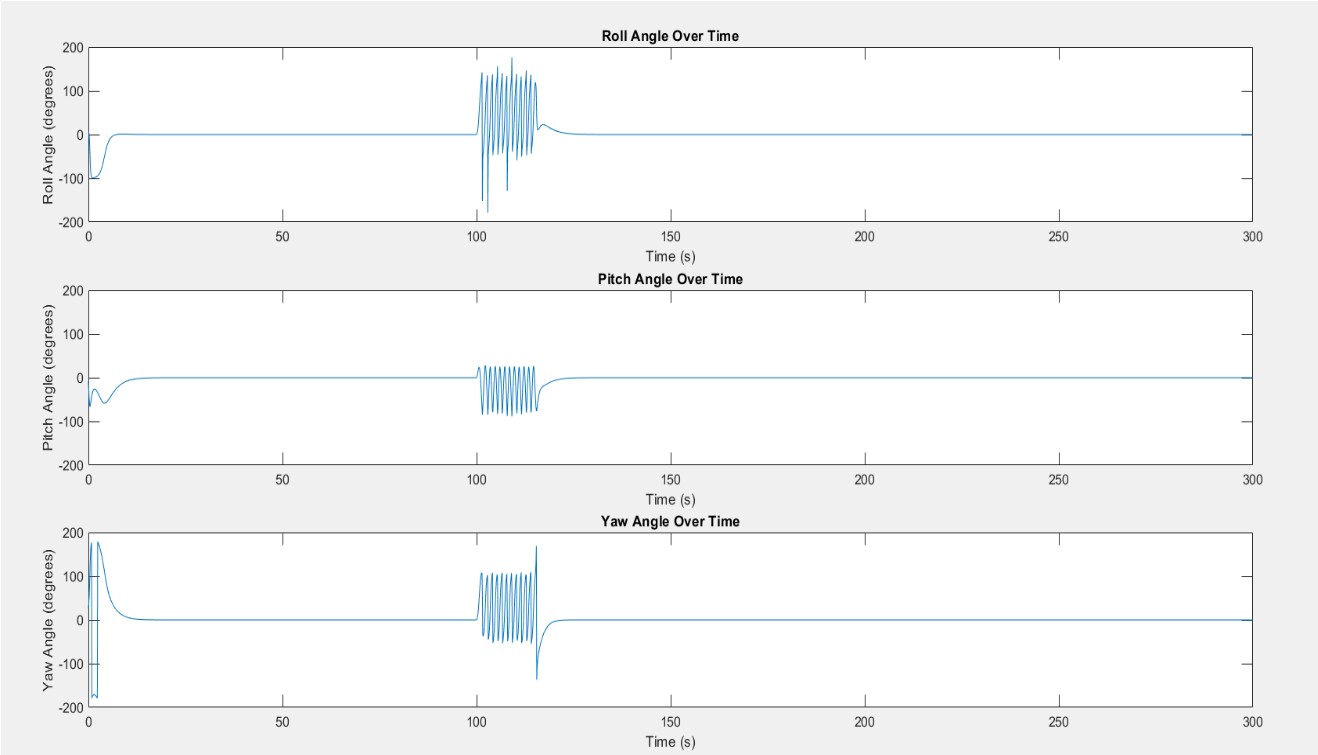


Figure 16: Diagram of attitude angles changing over time

#### 4.6.2 Under adaptive control

In this part, different cases of the system under the proposed adaptive control scheme will be simulated.

**1) Cases of 1:1 non-cooperative attachment** Non-cooperative body with an inertia matrix of $J_1$ attached to the small satellite under the proposed adaptive control is simulated.

Compared with classical (PD) control technique under the same conditions in Fig. 16 the adaptive control scheme was much more effective. Using the adaptive control scheme, the system can recover as soon as possible. These results indirectly validate the effectiveness of the proposed adaptive control scheme.

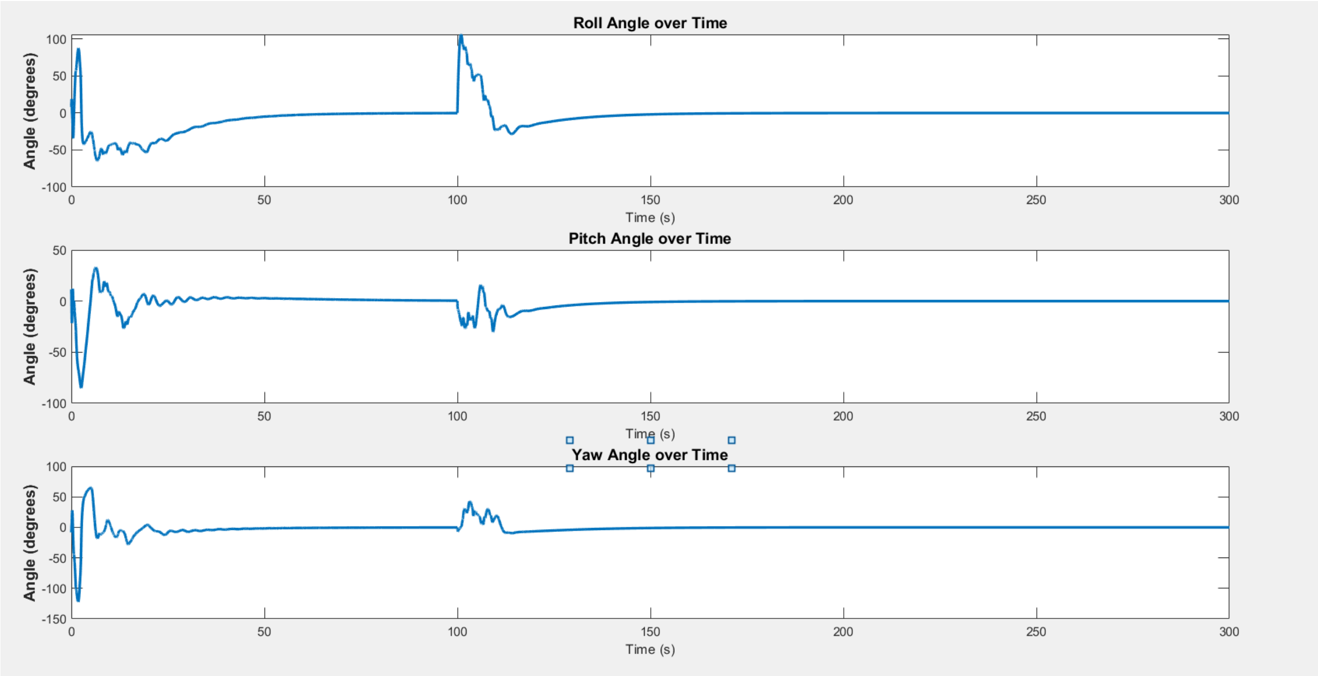


Figure 17: Diagram of attitude angles changing over time.

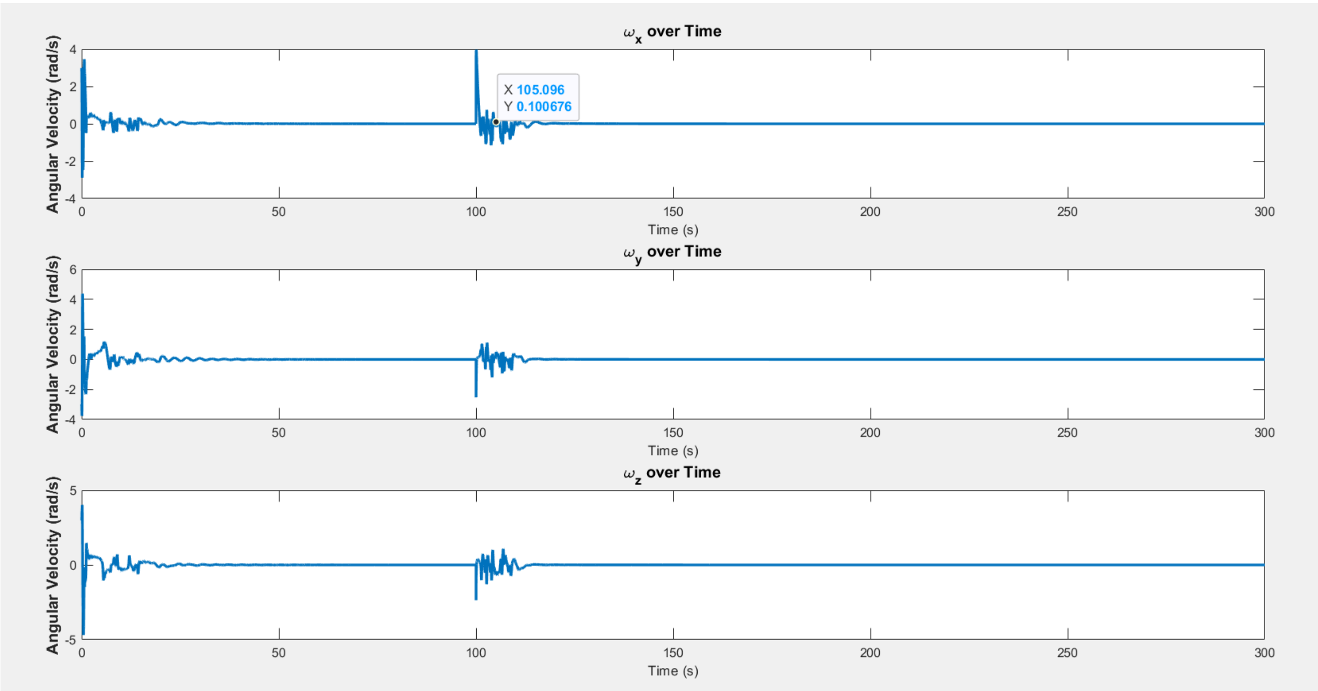


Figure 18: Diagram of attitude rates changing over time.

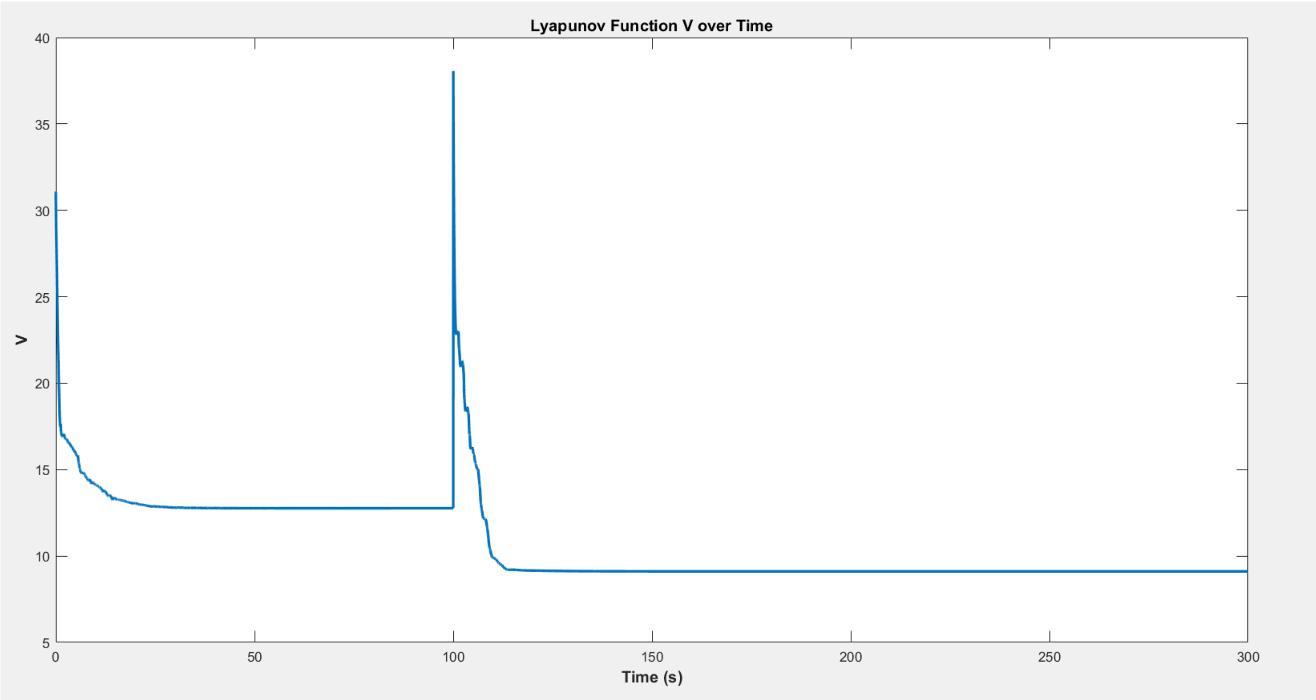


Figure 19: Lyapunov Function Over time.

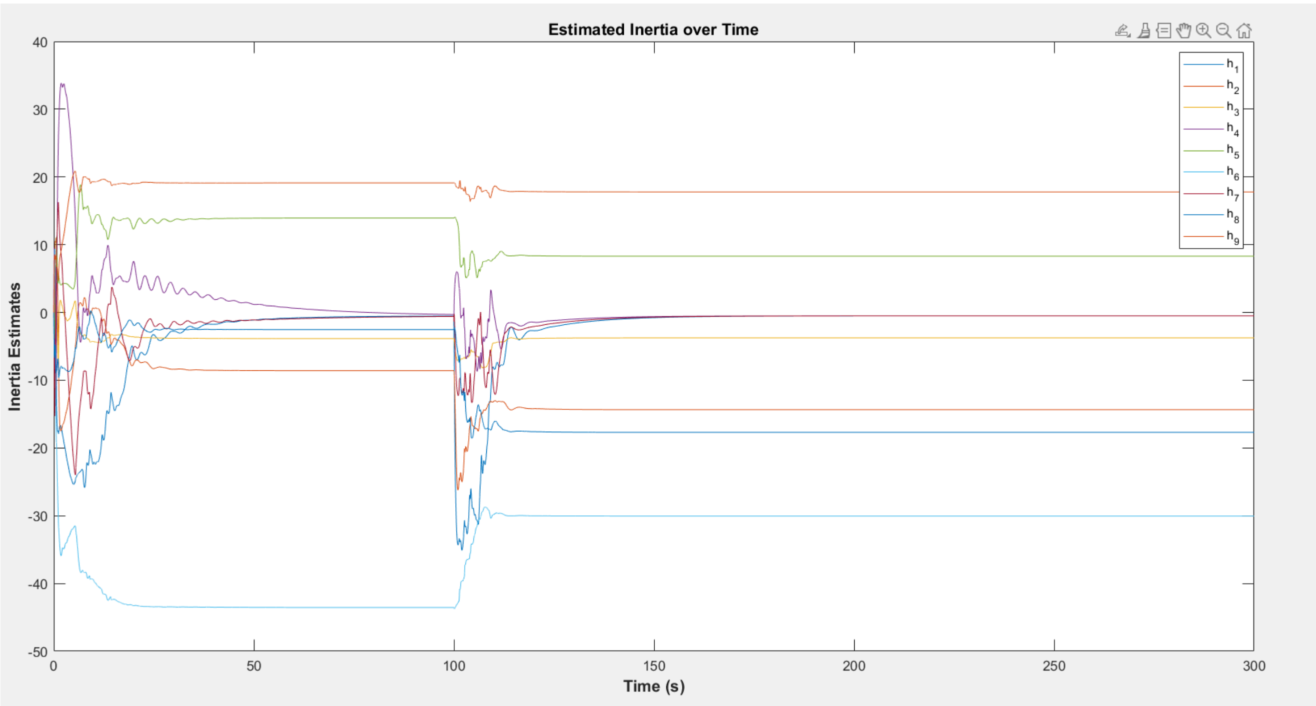


Figure 20: Estimated Inertia Over time .

**2) Case of 1:10 non-cooperative attachment** In this case, the simulation was carried out for a small satellite attached to by a non-cooperative body with an inertia matrix of $J_2$. The control effect of the 1:10 non-cooperative attachment under the proposed adaptive control is shown in Fig. 21. Due to the increased momentum, the angular rate increased.

As the inertia matrix was ten times bigger than the 1:1 attachment, the control time was only a little longer. Compared with the PD control in Fig. 16, we can easily see that, not only the control time but also the oscillation of the angular rate and attitude angle is much less. The system under the proposed adaptive control theme is more sensitive to the change of the system inertia. Therefore, the control system is robust in dealing with different sizes of inertia from the non-cooperative spacecraft.

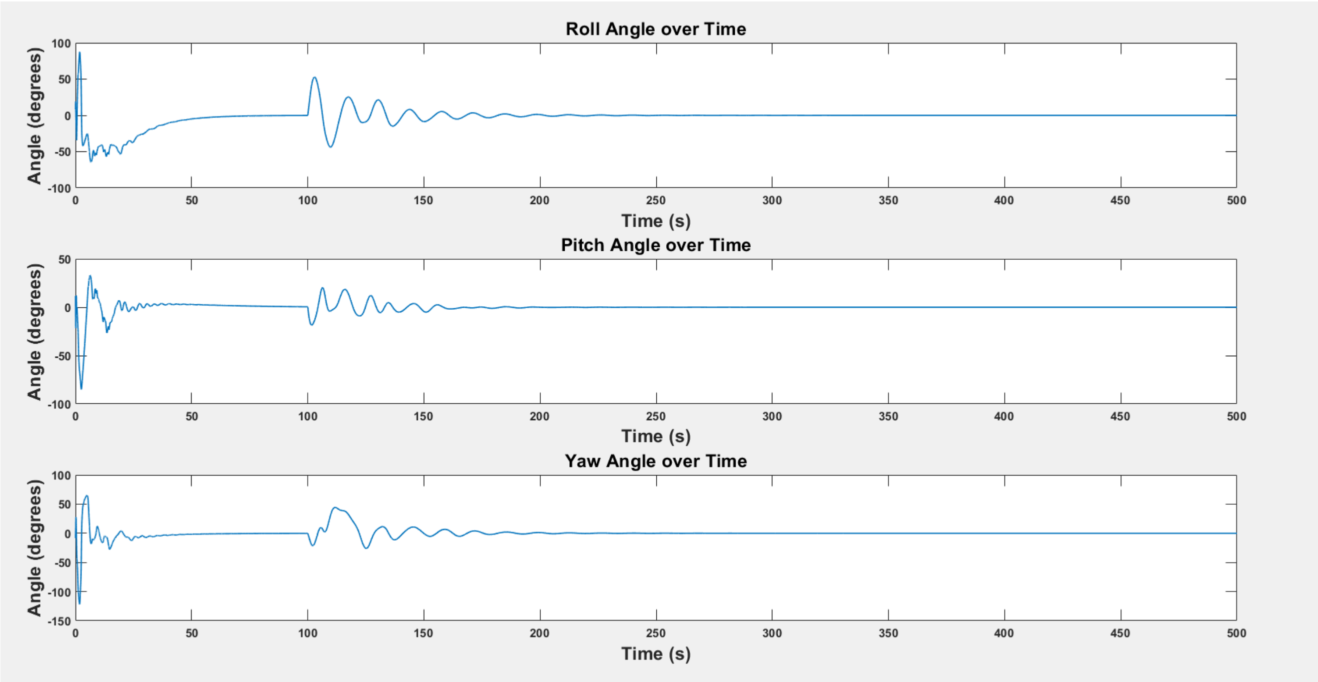


Figure 21: Diagram of attitude angles changing over time.

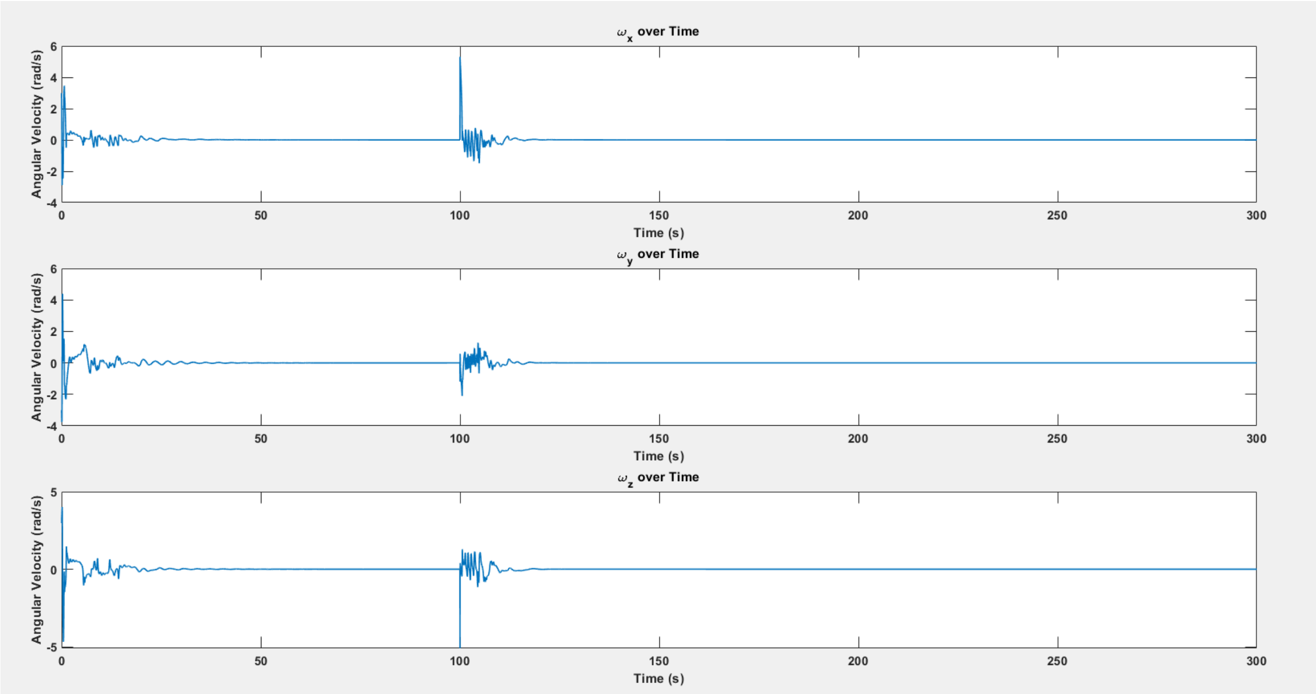


Figure 22: Diagram of attitude rates changing over time.

**3) Cases of 1:1 attachment, carrying more momentum** In this case, the simulation was carried out for a small satellite attached to by a non-cooperative body with an inertia matrix of $J_1$. Due to the increased momentum, the angular rate increased.

$$\omega = [15\ -15\,15]^T\ (°/\mathrm{s})$$

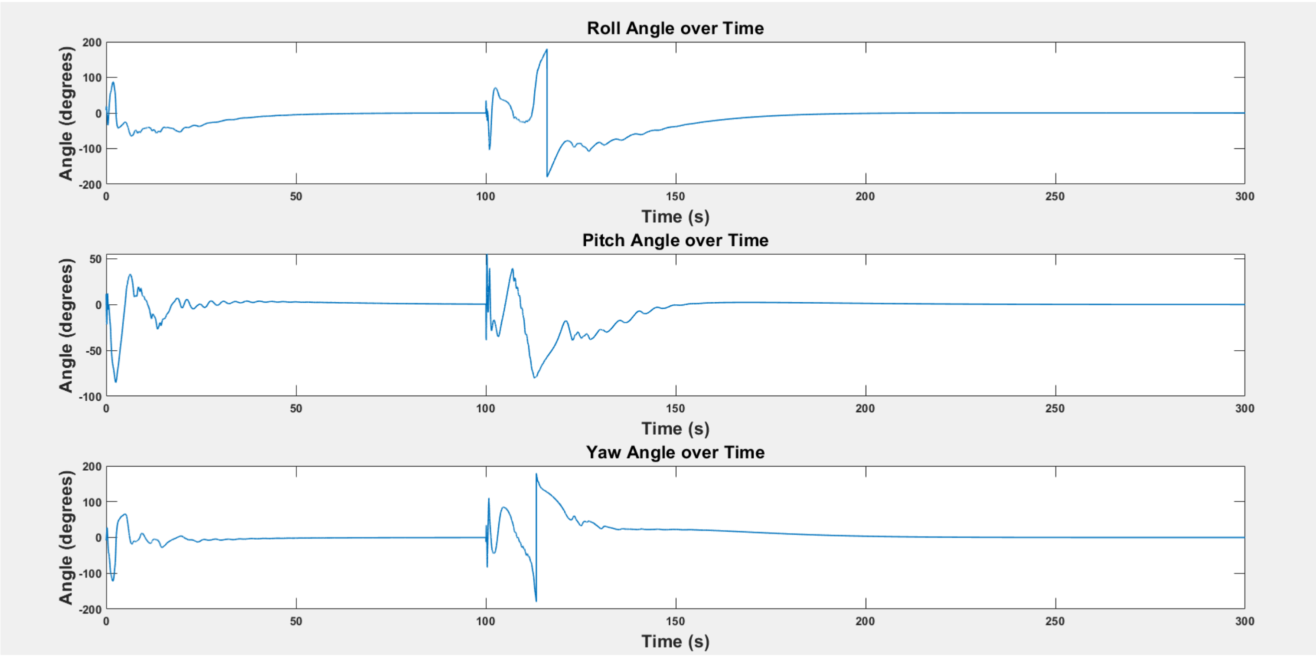


Figure 23: Diagram of attitude angles changing over time with added momentum.

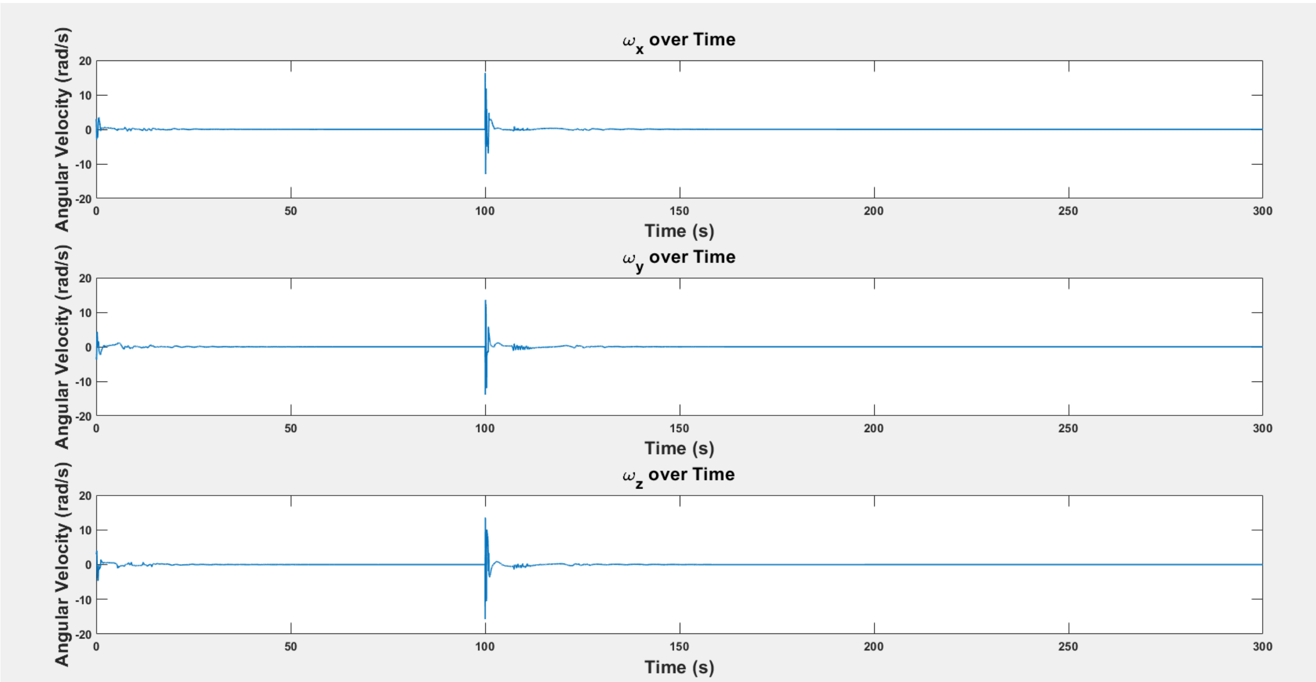


Figure 24: Diagram of attitude rates changing over time with added momentum.

## 4.7 Conclusions:

This document demonstrates all basics and fundamentals of spacecraft attitude control system. To address the problem of non cooperative body attachment, proposed both classical and adaptive control techniques. Based on kinematics and dynamics of spacecraft, an adaptive control scheme was developed which was designed based on lyapunov function and detailed proof is given for proving the global asymptotic stability of the system. The feasibility and effectiveness of the adaptive control scheme were verified by the simulation results. The adaptive control scheme can be more sensitive to the state of new combined system especially the change of system inertia. Thus adaptive control scheme not only works on the new combined system but also works better with estimating inertia.

## 4.8 Future Scope

This work can be further extended to large satellites with large inertia values and with large non cooperative object attachments.