IE415
Control Of Autonomous System
Project Report

Design and Simulation of a Satellite Attitude Control System Using State-Space Modeling, Feedback Control, and State Estimation

Name - Tasmay Patel

Student ID - 202201129

Contents

1 Introduction

1.1 Satellites and Their Functioning

Satellites enable global interconnectivity by providing telecommunication services regardless of distance or bandwidth. They require precise guidance systems to maintain their orientation relative to the Earth. Different types of satellites, such as communication, weather, and remote-sensing satellites, have specific subsystems, including propulsion, thermal control, power supply, telemetry, and attitude/orbit control systems.

1.2 Launching and Orbit Correction

Satellites are launched into an approximate orbit by large rockets, which may carry multiple satellites. Due to external forces in space causing orbit deviations, satellites use propulsion systems for fine adjustments to reach their final orbit. Once there, small thrusters handle orientation and station-keeping corrections.

1.3 Common Satellite Orbits

A widely used orbit for fixed communication is the 24-hour geostationary orbit at 35,786 km altitude. These orbits are ideal for consistent communication and tracking. Modern systems often utilize geostationary satellites for reliable round-the-clock service.

1.4 Control Systems for Satellites

Satellite control involves managing angular speed and position using a feedback controller based on state variables. This system ensures precise control by applying radial and tangential inputs, employing techniques like pole placement for stability and accuracy.

1.5 Principles of Orbiting Satellites

Satellite motion is governed by two forces:

- **Centripetal Force:** Due to Earth's gravity, pulling the satellite towards the center.
- **Centrifugal Force:** Acts outward due to the satellite's velocity.

These forces balance each other, enabling circular or elliptical orbits. Without Earth's gravity, the satellite would move in a straight line, but gravity transforms this motion into an orbit, following Newton's third law of motion.

1.6 Satellite Attitude Control System

An attitude control system for a satellite vehicle within the Earth's atmosphere is essential for maintaining its orientation. The control system adjusts the satellite's angular speed $\omega(t)$ and angular position $\theta(t)$ for complete control. This system requires two inputs:

- **Radial Direction Input:** Controls movement towards or away from the Earth's center.
- **Tangential Direction Input:** Controls movement along the satellite's orbit.

A state variable feedback controller is utilized for this purpose. The design of the controller employs the **pole placement technique**, which ensures stability and provides a unique solution for precise satellite control.

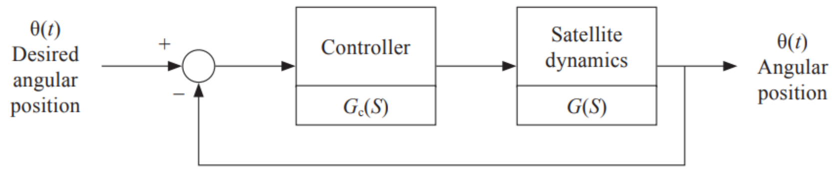

Figure 1: Block Diagram of Satellite Attitude Control System

2 State-Space Model

The satellite attitude control system can be modeled mathematically using state-space representation. The dynamics of the system are described by angular position $\theta(t)$ and angular speed $\omega(t)$, which are controlled by inputs in the radial and tangential directions.

2.1 General State-Space Representation

The general form of a linear time-invariant (LTI) system in state-space representation is given by:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)$$

Where:

- $\mathbf{x}(t)$ is the state vector.
- $\mathbf{u}(t)$ is the input vector.
- $\mathbf{y}(t)$ is the output vector.
- $\mathbf{A}$ is the state matrix.
- $\mathbf{B}$ is the input matrix.
- $\mathbf{C}$ is the output matrix.
- $\mathbf{D}$ is the feedthrough (or direct transmission) matrix.

2.2 Satellite State Variables

For the satellite attitude control system:

$$\mathbf{x}(t) = \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix}, \quad \mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}, \quad \mathbf{y}(t) = \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix}$$

Where:

- $\theta(t)$: Angular position.
- $\omega(t)$: Angular speed.
- $u_1(t)$, $u_2(t)$: Inputs in the radial and tangential directions.

2.3 Dynamic Equations

The system dynamics are expressed as:

$$\dot{\theta}(t) = \omega(t) \quad \text{(Angular velocity is the rate of change of angular position.)}$$
$$\dot{\omega}(t) = -k\theta(t) + bu(t) \quad \text{(Torque balance considering restoring and control torques.)}$$

Here:

- $k > 0$: Restoring torque constant.
- $b > 0$: Control input constant.

2.4 State-Space Representation

Combining the above, the state-space representation is:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -k & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 & 0 \\ b & b \end{bmatrix} \mathbf{u}(t)$$

$$\mathbf{y}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{u}(t)$$

Where:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -k & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ b & b \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

This state-space model provides a complete mathematical representation of the satellite attitude control system, describing its angular position and speed dynamics.

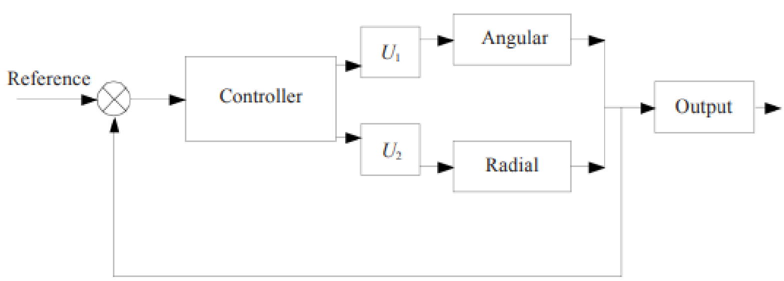

Figure 2: Block Diagram of State Space Representation

3 State Feedback Control Using PID

State feedback control combined with a PID controller is an effective approach for regulating the angular position $\theta(t)$ and angular velocity $\omega(t)$ of a satellite. The following subsections describe the dynamics, PID control, combined control law, and closed-loop system equations.

3.1 System Dynamics

The satellite system's state-space representation is:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t),$$

where:

$$\mathbf{x}(t) = \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

The control input $\mathbf{u}(t)$ consists of a state feedback term and a PID control term:

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) + \mathbf{u}_{\text{PID}}(t),$$

where $\mathbf{K} = \begin{bmatrix} K_1 & K_2 \end{bmatrix}$ is the state feedback gain vector.

3.2 PID Control

The PID controller regulates the angular position $\theta(t)$ based on the error $e_\theta(t)$:

$$e_\theta(t) = \theta_d(t) - \theta(t),$$

where $\theta_d(t)$ is the desired angular position.
The control law is:

$$u_{\text{PID}}(t) = K_p e_\theta(t) + K_i \int e_\theta(t)\, dt + K_d \frac{de_\theta(t)}{dt},$$

where K_p, K_i, and K_d are proportional, integral, and derivative gains, respectively.

3.3 Combined Control Law

The total control input is:

$$u(t) = -K_1\theta(t) - K_2\omega(t) + K_p e_\theta(t) + K_i \int e_\theta(t)\, dt + K_d \frac{de_\theta(t)}{dt}.$$

Substituting $e_\theta(t) = \theta_d(t) - \theta(t)$, the control input becomes:

$$u(t) = (K_p\theta_d(t) + K_i \int \theta_d(t)\, dt + K_d \frac{d\theta_d(t)}{dt}) - [(K_p + K_1)\theta(t) + K_i \int \theta(t)\, dt + (K_d + K_2)\dot{\theta}(t)].$$

3.4 Closed-Loop Dynamics

The state-space equations are updated with the input $u(t)$:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t).$$

Expanding for each state variable:

- Angular Position:

$$\dot{\theta}(t) = \omega(t).$$

- Angular Velocity:

$$\dot{\omega}(t) = K_p(\theta_d(t) - \theta(t)) + K_i \int (\theta_d(t) - \theta(t))\, dt + K_d \frac{d(\theta_d(t) - \theta(t))}{dt} - K_1\theta(t) - K_2\omega(t).$$

3.5 Gain Selection

The gains K_1 and K_2 in state feedback are selected to place the system poles in desired locations for stability. The PID gains K_p, K_i, and K_d are tuned for performance, ensuring fast transient response and minimal steady-state error.

3.6 Equations Summary

1. State Variables:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t).$$

2. Control Input:

$$u(t) = K_p(\theta_d(t) - \theta(t)) + K_i \int (\theta_d(t) - \theta(t))\, dt + K_d \frac{d(\theta_d(t) - \theta(t))}{dt} - K_1\theta(t) - K_2\omega(t).$$

3. Closed-Loop Dynamics:

$$\dot{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}(t) + \mathbf{B}\mathbf{u}_{\text{PID}}(t).$$

4 Simplified System in Roll, Pitch, and Yaw

In this section, we simplify the satellite's attitude control system by expressing the dynamics in terms of roll, pitch, and yaw components. These three rotational components define the orientation of the satellite relative to the Earth and are described using the Euler angles.

4.1 State Variables and Control Inputs

The state vector is defined as:

$$\mathbf{x}(t) = \begin{bmatrix} \phi(t) \\ \theta(t) \\ \psi(t) \\ \dot{\phi}(t) \\ \dot{\theta}(t) \\ \dot{\psi}(t) \end{bmatrix}$$

Where: - $\phi(t)$, $\theta(t)$, and $\psi(t)$ are the roll, pitch, and yaw angles, respectively. - $\dot{\phi}(t)$, $\dot{\theta}(t)$, and $\dot{\psi}(t)$ are the angular velocities.

The control inputs are:

$$\mathbf{u}(t) = \begin{bmatrix} u_\phi(t) \\ u_\theta(t) \\ u_\psi(t) \end{bmatrix}$$

Where $u_\phi(t)$, $u_\theta(t)$, and $u_\psi(t)$ represent the torques applied to the satellite in the roll, pitch, and yaw directions, respectively.

4.2 Yaw, Pitch, and Roll Dynamics

The dynamics of yaw, pitch, and roll can be expressed as:

1. Yaw (ψ):

$$\dot{\psi}(t) = \omega_\psi(t)$$

$$\omega_\psi(t) = \int \dot{\psi}(t)\, dt$$

Where $\omega_\psi(t)$ is the angular velocity in the yaw direction.

2. Pitch (θ):

$$\dot{\theta}(t) = \omega_\theta(t)$$

$$\omega_\theta(t) = \int \dot{\theta}(t)\, dt$$

Where $\omega_\theta(t)$ is the angular velocity in the pitch direction.

3. Roll (ϕ):

$$\dot{\phi}(t) = \omega_\phi(t)$$

$$\omega_\phi(t) = \int \dot{\phi}(t)\, dt$$

Where $\omega_\phi(t)$ is the angular velocity in the roll direction.

4.3 State-Space Representation

The state-space representation of the system can be written as:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} \dot{\phi}(t) \\ \dot{\theta}(t) \\ \dot{\psi}(t) \\ \ddot{\phi}(t) \\ \ddot{\theta}(t) \\ \ddot{\psi}(t) \end{bmatrix} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

Where:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ g_\phi & 0 & 0 \\ 0 & g_\theta & 0 \\ 0 & 0 & g_\psi \end{bmatrix}$$

4.4 PID Control for Each Component

The PID control law for each of the three components (yaw, pitch, and roll) is given by:

$$u_\phi(t) = K_{p\phi} e_\phi(t) + K_{i\phi} \int e_\phi(t)\, dt + K_{d\phi} \frac{de_\phi(t)}{dt}$$

$$u_\theta(t) = K_{p\theta} e_\theta(t) + K_{i\theta} \int e_\theta(t)\, dt + K_{d\theta} \frac{de_\theta(t)}{dt}$$

$$u_\psi(t) = K_{p\psi} e_\psi(t) + K_{i\psi} \int e_\psi(t)\, dt + K_{d\psi} \frac{de_\psi(t)}{dt}$$

Where: - $e_\phi(t) = \phi_d(t) - \phi(t)$ is the error in roll angle, - $e_\theta(t) = \theta_d(t) - \theta(t)$ is the error in pitch angle, - $e_\psi(t) = \psi_d(t) - \psi(t)$ is the error in yaw angle, - $K_{p\phi}, K_{i\phi}, K_{d\phi}$ are the PID gains for roll, - $K_{p\theta}, K_{i\theta}, K_{d\theta}$ are the PID gains for pitch, - $K_{p\psi}, K_{i\psi}, K_{d\psi}$ are the PID gains for yaw.

4.5 State Feedback Control for Roll, Pitch, and Yaw

State feedback control is applied to adjust the state vector:

$$\mathbf{u}(t) = -\mathbf{K}\mathbf{x}(t) + \mathbf{u}_{\text{PID}}(t)$$

Where:

$$\mathbf{K} = \begin{bmatrix} K_\phi & K_{\dot{\phi}} \\ K_\theta & K_{\dot{\theta}} \\ K_\psi & K_{\dot{\psi}} \end{bmatrix}$$

And the total control input becomes a combination of state feedback and PID control for the three rotational components.

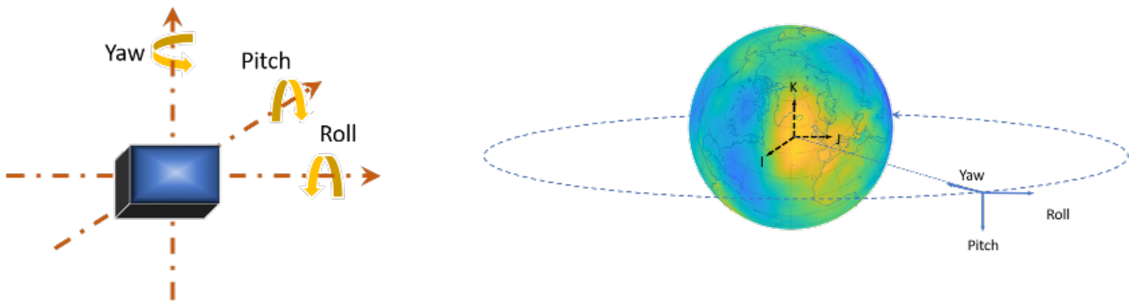

Figure 3: Roll, Pitch, Yaw of Satellite

5 State Estimation Using Kalman Filter

In this section, we discuss how to estimate the states of the satellite attitude system (yaw, pitch, and roll) using a **Kalman filter**. The Kalman filter is an optimal estimator for linear systems with Gaussian noise in both the system and measurement models.

5.1 Kalman Filter Overview

The Kalman filter works by recursively estimating the state of a system at each time step, combining predictions from the system model with actual measurements. The state update process consists of two main steps:
1. Prediction: Projecting the state forward using the system dynamics.
2. Correction: Updating the prediction with new measurements to reduce the estimation error.

5.2 State-Space Model with Noise

We define the system in state-space form as follows:

$$\mathbf{x}(t) = \begin{bmatrix} \phi(t) \\ \theta(t) \\ \psi(t) \\ \dot{\phi}(t) \\ \dot{\theta}(t) \\ \dot{\psi}(t) \end{bmatrix}$$

The system's dynamics are given by:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{w}(t)$$

Where $\mathbf{w}(t)$ is the process noise, assumed to be zero-mean Gaussian with covariance $\mathbf{Q}$.
The measurement model is given by:

$$\mathbf{z}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{v}(t)$$

Where $\mathbf{z}(t)$ is the measurement vector and $\mathbf{v}(t)$ is the measurement noise, assumed to be zero-mean Gaussian with covariance $\mathbf{R}$.

5.3 Prediction Step

In the prediction step, the Kalman filter projects the current state estimate forward in time. The prediction of the state is given by:

$$\hat{\mathbf{x}}_{-}(t+1) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t)$$

The predicted estimate covariance is:

$$\mathbf{P}_{-}(t+1) = \mathbf{A}\mathbf{P}(t)\mathbf{A}^T + \mathbf{Q}$$

Where: $\hat{\mathbf{x}}(t)$ is the state estimate at time t, $\mathbf{P}(t)$ is the error covariance matrix at time t, $\mathbf{Q}$ is the process noise covariance.

5.4 Correction Step

In the correction step, the filter incorporates the new measurement $\mathbf{z}(t)$ to update the state estimate. The innovation or measurement residual is:

$$\mathbf{y}(t) = \mathbf{z}(t) - \mathbf{C}\hat{\mathbf{x}}_{-}(t)$$

The Kalman gain is computed as:

$$\mathbf{K}(t) = \mathbf{P}_{-}(t)\mathbf{C}^T \left(\mathbf{C}\mathbf{P}_{-}(t)\mathbf{C}^T + \mathbf{R}\right)^{-1}$$

The updated state estimate is:

$$\hat{\mathbf{x}}(t) = \hat{\mathbf{x}}_{-}(t) + \mathbf{K}(t)\mathbf{y}(t)$$

The updated estimate covariance is:

$$\mathbf{P}(t) = \left(\mathbf{I} - \mathbf{K}(t)\mathbf{C}\right)\mathbf{P}_{-}(t)$$

Where: $\hat{\mathbf{x}}(t)$ is the updated state estimate, $\mathbf{K}(t)$ is the Kalman gain, $\mathbf{P}(t)$ is the updated error covariance matrix, $\mathbf{C}$ is the measurement matrix, $\mathbf{R}$ is the measurement noise covariance.

5.5 Kalman Filter Equations for Roll, Pitch, and Yaw

To apply the Kalman filter to the yaw, pitch, and roll system, we need to specify the measurement and system models for each of these components.

5.5.1 Roll Dynamics (ϕ)

For roll dynamics, the system equation becomes:

$$\dot{\phi}(t) = \omega_{\phi}(t)$$

The measurement equation could be:

$$z_{\phi}(t) = \phi(t) + v_{\phi}(t)$$

Where $z_{\phi}(t)$ is the measured roll angle, and $v_{\phi}(t)$ is the measurement noise for roll.

The Kalman filter update for roll is given by the equations:

$$\hat{\mathbf{x}}_\phi(t+1) = \mathbf{A}_\phi \hat{\mathbf{x}}_\phi(t) + \mathbf{B}u_\phi(t)$$

$$\mathbf{P}_\phi^-(t+1) = \mathbf{A}\mathbf{P}_\phi(t)\mathbf{A}^T + \mathbf{Q}$$

$$\mathbf{K}_\phi(t) = \mathbf{P}_\phi^-(t)\mathbf{C}_\phi^T \left(\mathbf{C}_\phi \mathbf{P}_\phi^-(t)\mathbf{C}_\phi^T + \mathbf{R}_\phi\right)^{-1}$$

$$\hat{\mathbf{x}}_\phi(t) = \hat{\mathbf{x}}_\phi^-(t) + \mathbf{K}_\phi(t)\left(z_\phi(t) - \mathbf{C}_\phi \hat{\mathbf{x}}_\phi^-(t)\right)$$

5.5.2 Pitch Dynamics (θ)

Similarly, for pitch dynamics, the system equation is:

$$\dot{\theta}(t) = \omega_\theta(t)$$

The measurement equation could be:

$$z_\theta(t) = \theta(t) + v_\theta(t)$$

The Kalman filter update for pitch is given by similar equations to those for roll:

$$\hat{\mathbf{x}}_\theta(t+1) = \mathbf{A}_\theta \hat{\mathbf{x}}_\theta(t) + \mathbf{B}u_\theta(t)$$

$$\mathbf{P}_\theta^-(t+1) = \mathbf{A}\mathbf{P}_\theta(t)\mathbf{A}^T + \mathbf{Q}$$

$$\mathbf{K}_\theta(t) = \mathbf{P}_\theta^-(t)\mathbf{C}_\theta^T \left(\mathbf{C}_\theta \mathbf{P}_\theta^-(t)\mathbf{C}_\theta^T + \mathbf{R}_\theta\right)^{-1}$$

$$\hat{\mathbf{x}}_\theta(t) = \hat{\mathbf{x}}_\theta^-(t) + \mathbf{K}_\theta(t)\left(z_\theta(t) - \mathbf{C}_\theta \hat{\mathbf{x}}_\theta^-(t)\right)$$

5.5.3 Yaw Dynamics (ψ)

For yaw dynamics, the system equation is:

$$\dot{\psi}(t) = \omega_\psi(t)$$

The measurement equation could be:

$$z_\psi(t) = \psi(t) + v_\psi(t)$$

The Kalman filter update for yaw is:

$$\hat{\mathbf{x}}_\psi(t+1) = \mathbf{A}_\psi \hat{\mathbf{x}}_\psi(t) + \mathbf{B}u_\psi(t)$$

$$\mathbf{P}_\psi^-(t+1) = \mathbf{A}\mathbf{P}_\psi(t)\mathbf{A}^T + \mathbf{Q}$$

$$\mathbf{K}_\psi(t) = \mathbf{P}_\psi^-(t)\mathbf{C}_\psi^T \left(\mathbf{C}_\psi \mathbf{P}_\psi^-(t)\mathbf{C}_\psi^T + \mathbf{R}_\psi\right)^{-1}$$

$$\hat{\mathbf{x}}_\psi(t) = \hat{\mathbf{x}}_\psi^-(t) + \mathbf{K}_\psi(t)\left(z_\psi(t) - \mathbf{C}_\psi \hat{\mathbf{x}}_\psi^-(t)\right)$$

5.6 Conclusion

This Kalman filter process will recursively estimate the attitude states (roll, pitch, and yaw) over time based on system dynamics and noisy measurements. Proper tuning of the process noise covariance matrix ($\mathbf{Q}$) and the measurement noise covariance matrix ($\mathbf{R}$) is critical for optimal performance of the Kalman filter.

6 Simulation Results

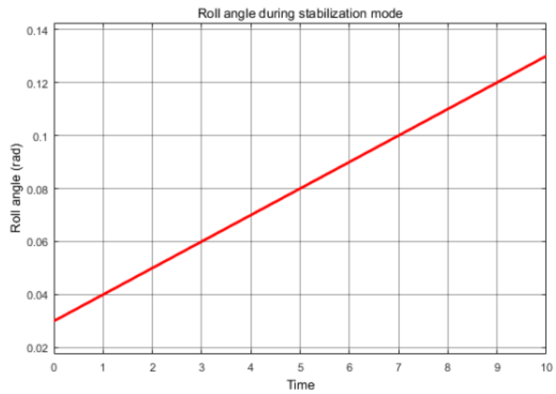

Figure 4: Roll angle

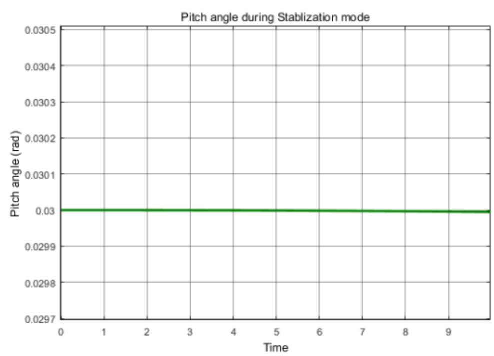

Figure 5: Pitch angle

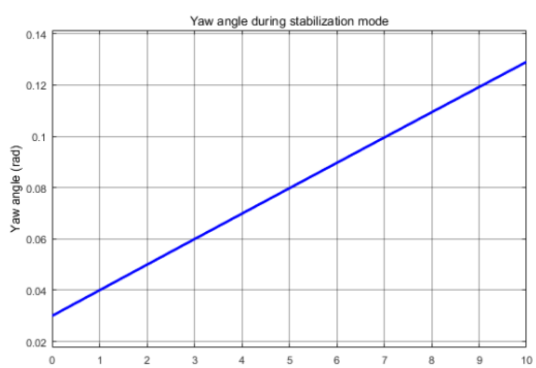

Figure 6: Yaw angle

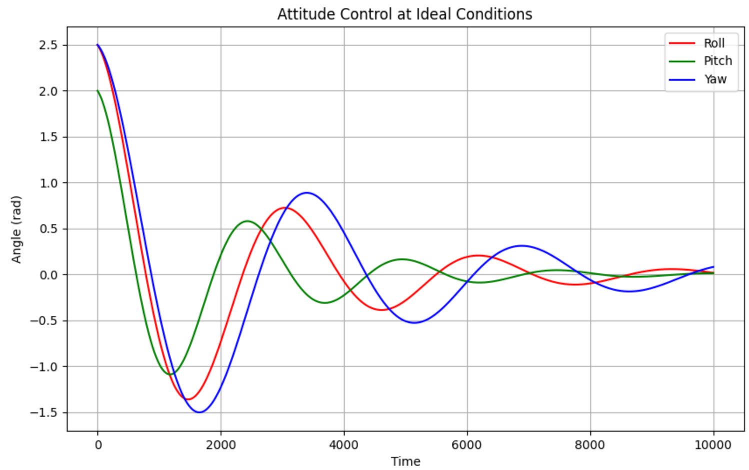

Figure 7: Attitude Control at ideal conditions

6.1 Explanation of Simulation Results

The axes in the simulation results are as follows:

- **x-axis:** Time in arbitrary units (likely seconds or simulation steps).
- **y-axis:** Angle in radians for the Roll, Pitch, and Yaw components of the satellite.

6.1.1 Key Observations

Initial Overshoot All three curves—Roll (red), Pitch (green), and Yaw (blue)—exhibit a sharp increase at the beginning of the simulation. This represents the system's initial response to an initial condition or disturbance, where the system temporarily overshoots the desired angles. This behavior is typical for dynamic control systems that react to sudden changes or initial disturbances. The overshoot indicates that the control system must apply corrective actions to bring the angles back toward their desired values.

Damping Behavior After the initial overshoot, the system experiences damped oscillations as it stabilizes. The damping rate is evident from how quickly the peaks of the oscillations reduce over time. The system gradually moves towards its steady-state position, with the oscillations becoming smaller and smaller. This damping is a result of the control system's feedback mechanism, which reduces the error between the desired and actual angles.

Stabilization Over time, the Roll, Pitch, and Yaw angles converge to steady-state values. The angles approximately settle to 1.5 radians for Roll and Yaw, and around 1.2 radians for Pitch. This steady-state convergence indicates that the control system has effectively stabilized the satellite's attitude. The stabilization is achieved after a few oscillations and with minimal steady-state error, demonstrating the effectiveness of the control algorithm used.

The simulation results demonstrate how the attitude controller compensates for disturbances or initial conditions. The system stabilizes as a result of the feedback control mechanism. The feedback loop, possibly implemented using Proportional-Derivative (PD) control, aims to minimize the error between the desired and actual angles. The oscillatory nature of the response, followed by steady-state convergence, suggests that the controller has been designed to achieve critical damping or slight underdamping for faster stabilization.

6.1.2 Differences Among Roll, Pitch, and Yaw

Yaw (Blue Curve) Yaw (represented by the blue curve) exhibits the highest initial overshoot and takes longer to stabilize compared to Roll and Pitch. This suggests that the yaw dynamics are more challenging to control, possibly due to higher inertia or greater disturbance in the yaw axis. The longer settling time could also reflect the presence of a larger control effort needed to stabilize the yaw angle.

Roll (Red Curve) and Pitch (Green Curve) Roll (red) and Pitch (green) angles stabilize faster and exhibit less overshoot compared to Yaw. This could be due to more favorable system dynamics or better tuning of the controller for these axes. The smaller overshoot and faster stabilization imply that the Roll and Pitch axes are easier to control or less susceptible to disturbances, requiring less corrective action from the control system.

6.1.3 Relevance to Satellite Attitude Control

The results demonstrate the satellite's ability to achieve a desired attitude orientation despite initial deviations. Several performance metrics can be derived from the results:

- **Overshoot:** This indicates how far the angles deviate from the target initially. A higher overshoot suggests a more aggressive response to disturbances.
- **Settling Time:** The time required for the system to stabilize within a tolerance band around the steady-state value. Faster settling time indicates a more responsive control system.
- **Steady-State Error:** This is the difference between the final angle and the desired angle. A minimal steady-state error shows that the controller is performing well and driving the satellite's attitude to the target with high accuracy.

These metrics are essential for evaluating the performance of attitude control systems in satellite dynamics, especially in scenarios involving high-precision alignment and stabilization.

7 References

- Research Paper Used, "Implementation of a Communication Satellite Orbit Controller Design Using State Space Techniques " click here
- Research Paper Used, "DESIGN AND DEVELOPMENT OF A MOMENTUM WHEEL-BASED AOCS MODEL OF A GEOSTATIONARY SATELLITE " click here
- Roll, Pitch, Yaw, "Real Time Simulator of Roll, Pitch, Yaw" click here
- Youtube, "Roll-Pitch-Yaw Angles in Robotics" click here
- Research Paper Used for Simulation, "Stabilizing Roll, Pitch and Yaw Angles for Attitude Control System (ACS)" click here
- Simulink File Used for Simulation, "Satellite Attitude Dynamics simulink" click here
- My Github Repository, "Satellite Attitude Control System" click here