Control of a Satellite with a 2-Link Manipulator

Naia Lum (A20461257)

Course: MMAE 540
Professor: Matthew Spenko
Date: November 26, 2024

Contents

1 Introduction **1**

2 Question 1: Defining System Dynamics and Controller **1**
2.1 System Dynamics 1
2.2 System Control 2
2.2.1 PD Controller 2
2.2.2 Adaptive Controller 2
2.2.3 Model Predictive Controller (MPC) 3

3 Question 2: Frenet-Serret Path Following **4**
3.1 Satellite Bus Path Following 4
3.2 Satellite with 2-Link Manipulator Path Following 5

4 Discussion **5**

1 Introduction

In the space environment, precision navigation and control are crucial for executing high-precision maneuvers such as positioning in orbit, docking, or flybys near a target. This project investigates two key questions related to controlling a 2-link manipulator attached to a satellite bus in a zero-gravity environment:

1. How can such a system be simulated and controlled effectively in zero-gravity conditions?

2. Can the system use Frenet-Serret Path Following for maneuvering around obstacles?

MATLAB and ChatGPT [1] are used for developing the simulation, where the reaction torques and forces generated by the manipulator affect the motion of the satellite bus. The following sections outline the system dynamics, control strategies, and simulation results.

2 Question 1: Defining System Dynamics and Controller

In order to proceed to path following or designing guidance solutions, the first step is defining the system and the best fit controller to study those dynamics. Using prior knowledge [2] of 2-link manipulator dynamics with a few adjustments, it is possible to design linear controllers for the system.

2.1 System Dynamics

The system investigated is defined as a satellite bus modeled as a point mass and a 2-link manipulator with rotational joints. The manipulator dynamics influence the satellite's linear and angular motion through reaction forces and torques. Figure 1 illustrates the system configuration.

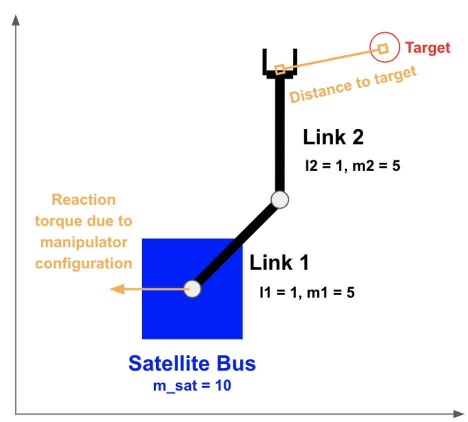


Figure 1: Satellite system with a 2-Link Manipulator under study

The dynamics of the satellite-manipulator system are described by the equation:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} = \boldsymbol{\tau}, \tag{1}$$

where $\mathbf{H}(\mathbf{q})$ is the inertia matrix representing the system's resistance to changes in motion, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ captures the Coriolis and centripetal forces that arise from velocity-dependent effects, and $\boldsymbol{\tau}$ represents the control torques applied at the manipulator's joints.

The main change in dynamics is the removal of the gravitational matrix due to the zero-gravity environment. Additionally, the satellite bus rotational dynamics are defined by the reaction torque and manipulator arm inertia, and the translational dynamics are defined by the acceleration of the satellite and the thrust force.

2.2 System Control

After defining the base dynamics used for the project, now it is important to define a controller for that system. Using end effector PD control will allow for intersection with a target point. This control input requires using a virtual spring and damper at the end effector. The goal of the controller is to minimize errors in position and velocity, gradually forcing the end effector to converge to the target point with some set tolerance. This control law is described by:

$$\tau_{\text{manipulator}} = -J^{\top}\left(K_p\tilde{x} + K_d\dot{\tilde{x}}\right), \tag{2}$$

Where J is the manipulator's Jacobian matrix relating joint velocities to end-effector velocity, K_p and K_d are the proportional and derivative gains respectively, $\tilde{x}$ is the position error, and $\dot{\tilde{x}}$ is the velocity error.

2.2.1 PD Controller

For simplicity, a PD controller was attempted. However, due to the unknown nature of the system parameters, such as mass and inertia affected by changing manipulator configurations in zero-gravity, this controller required significant tuning and still led to some instability in certain configurations.

2.2.2 Adaptive Controller

The adaptive controller updates the system's mass estimates based on real-time position and velocity errors. This approach significantly improves stability, as shown in the following mass update laws:

$$\dot{\hat{m}}_1 = -\gamma\ell_1\cos(q_1)\tilde{x} \tag{3}$$

$$\dot{\hat{m}}_2 = -\gamma\ell_2\cos(q_1+q_2)\tilde{x} \tag{4}$$

where $\hat{m}_1, \hat{m}_2$ are the estimated masses of the manipulator links, γ is the adaptation rate, ℓ_1, ℓ_2 are the lengths of the links, q_1, q_2 are the joint angles, and $\tilde{x}$ is the position error.

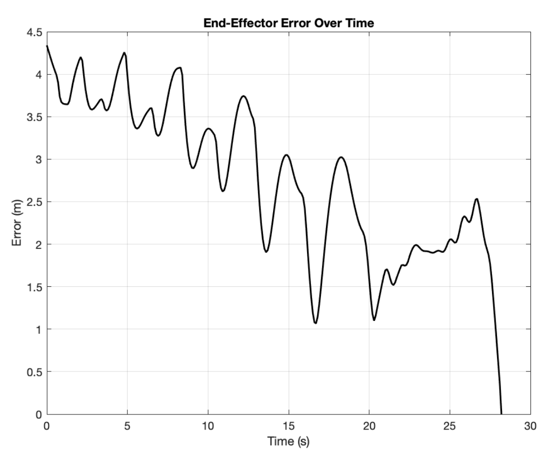


Figure 2: Adaptive Controller End Effector Error to Convergence

2.2.3 Model Predictive Controller (MPC)

An MPC uses a receding horizon optimization approach. [3] Costs were defined for position error, control effort, and a thruster penalty. This controller optimizes the control sequence over a finite horizon to minimize a defined cost function while satisfying system constraints, such as limited thrust capability and joint torque limits.

The controller minimizes a cost function over a prediction horizon N:

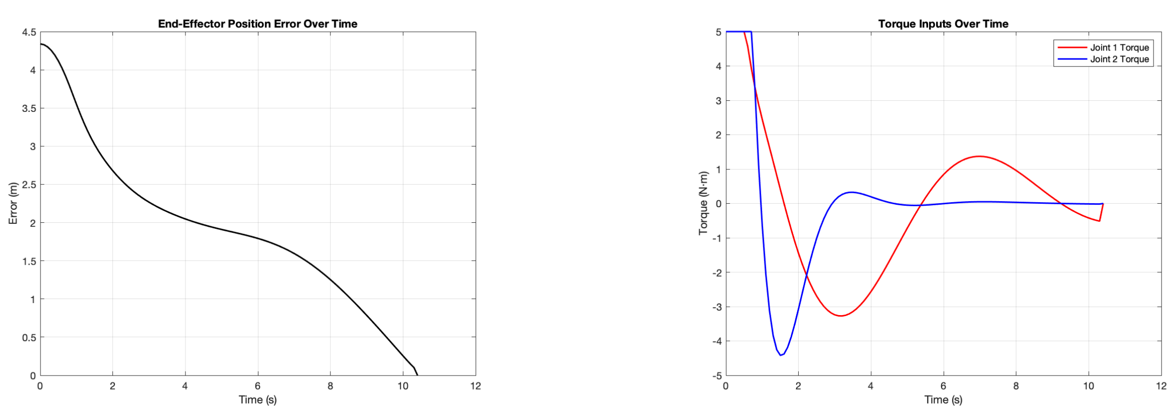
$$J = \sum_{t=0}^{N-1} \left[(\mathbf{x}_t - \mathbf{x}_{\text{target}})^\top \mathbf{Q} (\mathbf{x}_t - \mathbf{x}_{\text{target}}) + \mathbf{u}_t^\top \mathbf{R} \mathbf{u}_t + c_{\text{thruster}} \right],$$

where:

- $(\mathbf{x}_t - \mathbf{x}_{\text{target}})^\top \mathbf{Q} (\mathbf{x}_t - \mathbf{x}_{\text{target}})$: Penalizes deviation from the target state, weighted by $\mathbf{Q}$, encouraging the system to reach the desired position.
- $\mathbf{u}_t^\top \mathbf{R} \mathbf{u}_t$: Penalizes control effort (joint torques), weighted by $\mathbf{R}$, ensuring efficient use of actuation.
- c_{thruster}: Represents the penalty for thruster usage, discouraging excessive reliance on thrusters and promoting fuel efficiency and effective spacecraft dynamics.

At each timestep, the controller solves this optimization problem to compute the optimal control sequence $\mathbf{u} = [\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{N-1}]$ over the prediction horizon. The first control input, $\mathbf{u}_0$, is the joint torques $[\tau_1, \tau_2]^\top$, which are applied to the system. The optimization process then repeats iteratively with updated state information, ensuring that the controller dynamically adapts while balancing accuracy, efficiency, and resource constraints.

The controller minimizes a cost function over a prediction horizon N:



(a) MPC end effector error to convergence

(b) Joint torques converging to stability

Figure 3: MPC controller performance

3 Question 2: Frenet-Serret Path Following

3.1 Satellite Bus Path Following

Once it was proven that a satellite bus could be controlled to follow a path, the system was adapted to the original problem of the addition of a 2-Link Manipulator. As MPC is often used for more complicated systems when optimization is necessary, the adaptive controller is used for simplicity in future simulations.

Before adding the manipulator arm to the system, a more realistic satellite bus with intermittent thrust boosts for velocity and orientation control was implemented. The system control is defined using the Frenet-Serret Coordinate System, which aims to guide the satellite along a sinusoidal trajectory. In this context, the states $\dot{s}$, $\dot{l}$, and $\dot{\theta}_t$ represent the satellite's motion along the Frenet-Serret frame: $\dot{s}$ is the tangential velocity, $\dot{l}$ is the lateral deviation rate, and $\dot{\theta}_t$ is the angular velocity of the tangent vector relative to the trajectory. Additionally, the satellite bus dynamics include velocity $\dot{v}$ and angular velocity $\dot{\omega}$, defined as:

$$\dot{v} = \frac{F_{\text{thrust}}}{m} - \text{damping} \cdot v, \qquad \dot{\omega} = \frac{\tau_{\text{torque}} + \tau_{\text{reaction}}}{I} - \text{damping} \cdot \omega, \tag{5}$$

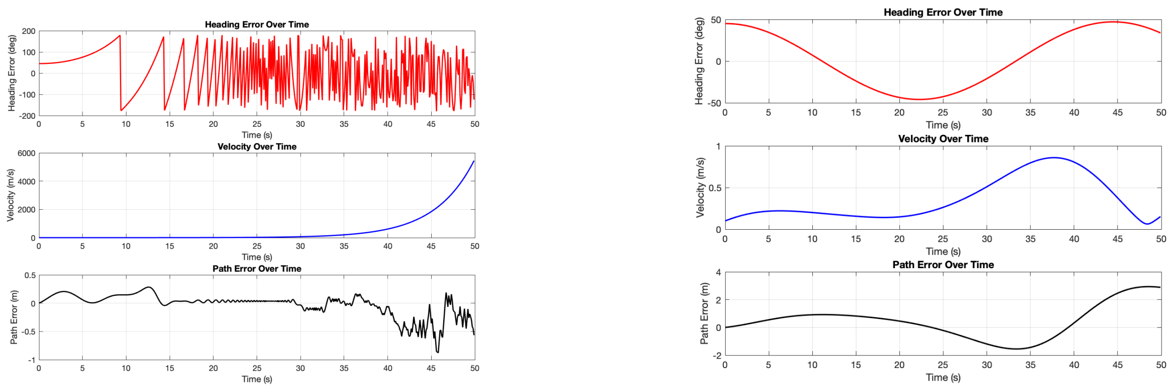
where F_{thrust} is the applied thrust force, m is the satellite mass, τ_{torque} and τ_{reaction} are the applied and reaction torques, respectively, and I is the moment of inertia. Damping terms represent the system's resistance to motion in both translational and rotational dynamics.

The reaction torque from the manipulator is applied to stabilize the satellite. Thrust is initially applied in any direction opposite to the desired path. These simulations were simple to perform, however in a realistic case the thrusters would not be able to simultaneously act in any direction; there would be constraints on the system based off mechanical design and a limited thruster count. Therefore to control the satellite in this way required for a more defined control laws for the translational force and reaction torque force and how they related to the system. The system is defined by the following dynamics and controls:

$$F_{\text{thrust}} = k_v \cdot (v_{\text{target}} - v) - k_l \cdot l \tag{6}$$

$$\tau_{\text{torque}} = -k_\theta \cdot \theta_t \tag{7}$$

In the first attempts to implement this the system was unstable. There were no velocity constraints and because of this it allowed its rotational and translational velocities to overshoot, leading to an unstable path following system. Once those constraints were defined and the respective velocity and orientation proportional gains were lowered, the system was able to reach stability.



(a) Unstable path following control

(b) Stable path following control

Figure 4: Comparison of path following control: unstable (left) vs. stable (right).

3.2 Satellite with 2-Link Manipulator Path Following

Once it was proven that a satellite bus could be controlled to follow a path, the system was modified to address the original problem by incorporating a 2-Link Manipulator. The addition of the manipulator introduced complexity to the dynamics, as the reaction forces and torques from the manipulator's motion affected the satellite's stability and path-following capabilities.

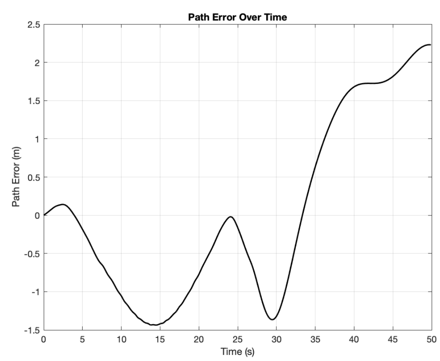
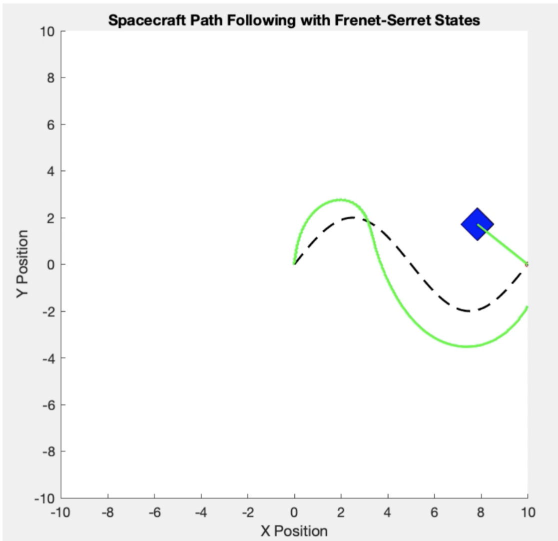


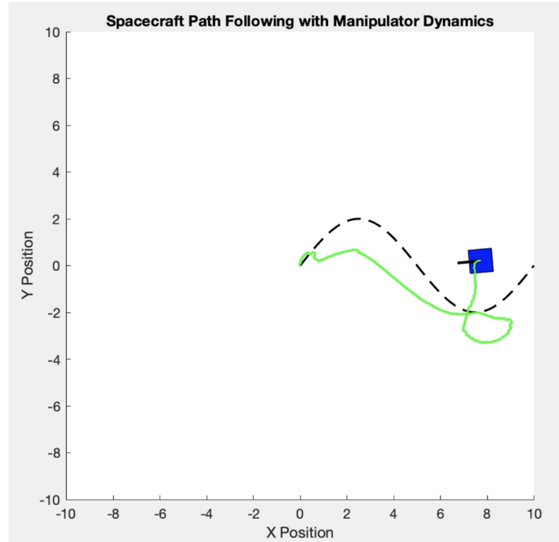
Figure 5: 2-Link Manipulator Path Error

4 Discussion

The results of this project demonstrate that the adaptive controller is the most effective for this system, providing stable and reliable performance in handling the nonlinear dynamics and parameter uncertainties introduced by the 2-link manipulator. While the MPC offered precise control and optimization, its complexity and computational cost made it less practical for real-time applications compared to the adaptive approach.



(a) Without 2-Link Manipulator Dynamics



(b) With 2-Link Manipulator Dynamics

Figure 6: Comparison of path following control: wihtout (left) vs. with (right) manipulator

The path-following capability of the system, by the Frenet-Serret coordinate system, was successfully demonstrated. However, the precision of the path following could be further improved by fine-tuning control gains and incorporating robust strategies to account for external disturbances or unmodeled dynamics.

Overall, this work establishes a foundation for controlling a satellite with a 2-link manipulator, while future developments can focus on enhancing robustness and adaptability to more realistic space scenarios.

References

1. OpenAI. *ChatGPT* `https://openai.com/chatgpt`.
2. Spenko, M. *Course Notes for MMAE 540: Robotics* Illinois Institute of Technology. 2024.
3. MathWorks. *What Is Model Predictive Control (MPC)?* `https://www.mathworks.com/help/mpc/gs/what-is-mpc.html` (2024). `https://www.mathworks.com/help/mpc/gs/what-is-mpc.html`.