Trajectory Optimization for Free Space Optical Communication

by

Dennis Wai

A report submitted in partial satisfaction of the
requirements for the degree of
Master of Science, Plan II

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Masayoshi Tomizuka, Chair
Professor J. Karl Hedrick

Fall 2016

The report of Dennis Wai, titled Trajectory Optimization for Free Space Optical Communication, is approved:

Chair ____ Date _______________

____ Date _______________

University of California, Berkeley

Trajectory Optimization for Free Space Optical Communication

Abstract

Trajectory Optimization for Free Space Optical Communication

by

Dennis Wai

Master of Science, Plan II in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Masayoshi Tomizuka, Chair

Communication in space is often done with the aid of large radio telescopes. Often, telescope size is correlated with the desired telecommunication distance (Ex. larger radio telescopes means farther distances). However, radio telescopy has two major limitations. First, the large physical size of radio telescopes places an upper bound on the number of onboard science instruments because there's a finite amount of volume in a rocket. Second, the communication throughput with radio telescopy is low and on the order of kilobytes per second.

Recent technological advances in manufacturing power efficient lasers have motivated research in free space optical communication, or the use of lasers as a carrier medium for communication in space. Compared to conventional radio frequency methods, free space optical communication can simplify spacecraft designs because of the laser's smaller form factor compared to traditional radio antennas. The fast data throughput enabled by free space optical communication can support advanced applications such as high definition planetary topographical maps.

In this report, I focus on free space optical communication links between two satellites in LEO (Low Earth orbit) and present the use of trajectory optimization as an offline tool to prescribe motion plans for both satellites. Then, the use of a time varying LQR to allow satellites to implement the motion plan online.

To my family

Contents

List of Figures

Acknowledgments

My time in Berkeley is finally coming to an end. Having spent both undergraduate and graduate at Berkeley, I have had the great opportunity to meet and work with many inspiring people.

First of all, I would like to extend my thanks to Professor Masayoshi Tomizuka, who since 2010 have been gracious enough to give me an opportunity to perform research in his laboratory.

Second, I would like to thank Professor Joonbum Bae, Professor Evan Chang-Siu, and Dr. Yizhou Wang for being the pivotal research mentors they were in my budding career as a researcher.

Third, I would like to thank Professor Oliver O'Reilly, Professor Pieter Abbeel, and Professor Packard for their advice and suggestions.

Fourth, I would like to thank Jasmine Thai on Euclid Ave for being a solid vegetarian Thai restaurant that largely subsidized my lunches on campus.

Finally, I would like to acknowledge my peers Te Tang, Hsien Chung Lin, Chung Yen Lin, Joseph Gonzales, Alyssa Novelia, Frank Chuang, Ben Yee, Stephen Chen, Jared Porter and Robert Luan for their support in academic and moral forms. I wish them all the best in their future endeavours.

Chapter 1

Introduction

Free space optical communication refers to the use of lasers as a medium for communication between two agents in free space. The two agents can both reside in earth orbit or one of the agent resides in orbit and the other on the Earth surface. In this report, both agents will reside in LEO (low Earth orbit) and will be required to maintain a line of sight lock to facilitate communication. Due to high satellite velocities and the narrow field of view of onboard optical sensors, the margin for imprecision is very low.

Conventional radio transceivers are used send commands to spacecrafts for mission tasks as well as to receive precious science data from spacecrafts. However, advanced applications such as telerobotics, streaming videos, or high definition science data like topographical maps require high bandwidth communication channels between the two terminals.

Free space optical communication, which uses lasers instead of radio waves as the medium for communication, is an attractive alternative to radio transceivers because it can provide a high data throughput on the order of mega/gigabytes per second whereas conventional methods can only support throughput on the order of kilobytes per second. In the case of communication between two satellites in LEO (Low Earth Orbit), a motion plan is required for at least one of the satellite so that a line of sight is maintained between the two terminals. In Ch. 2, I propose trajectory optimization, an optimal control technique, to generate motion plans for this system. The plant dynamics and the assumptions made will be presented in Ch. 4.

However, the generated motion plan must also comply with several performance require-

Figure 1.1: Artist rendering of satellite achieving link with a receiver

ments. The benefit of higher data throughput comes at the cost of a stricter pointing accuracy requirement. Whereas traditional radio frequency methods require a pointing accuracy of 12 milliradians (approximately .5 degrees), free space optical communication require a pointing accuracy of 500 microradians (approximately .03 degrees) because of the lack of diffusivity in laser beams. Moreover, the spacecrafts' nadir point will periodically need to point towards earth in order to downlink onboard data to an Earth station. Both of these requirements can be formulated as constraints and included in our optimization problem. They will be derived and described in Ch. 3.

Next, I will discuss the use of time-varying LQR on the solution of trajectory optimization in order to generate a controller that can be implemented online in Ch. 5. Simulation results of running that controller online will be presented in Ch. 6.

In Ch. 7, I will discuss future directions to explore in order to greater understand the limitations and feasibility of using trajectory optimization for free space optical communication.

Chapter 2

Trajectory Optimization

In [11], trajectory optimization was used to find a collision-free path for an agent navigating through a known map world. Trajectory optimization is a nonlinear optimization technique that takes in tentative trajectories and returns suboptimal trajectories that obey user-inputted constraints. It is a technique that can handle nonconvex constraints and objectives, which are often encountered in real-world problems.

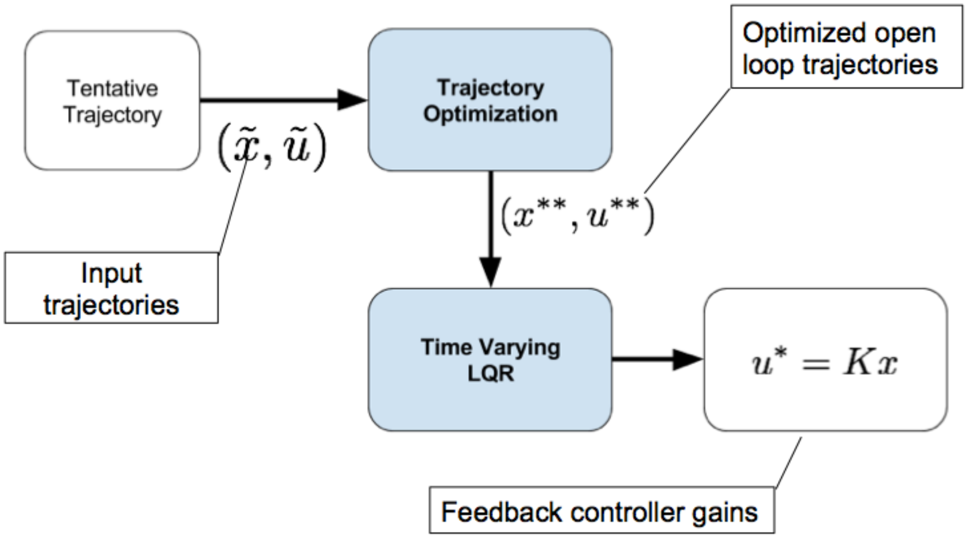


Figure 2.1: Flowchart overview of trajectory optimization

The problem is formulated as follows:

$$\begin{aligned}\min_{x,u}\ &\mathcal{T}(\tilde{x},\tilde{u},x,u)\\ s.t\ \ &\text{initial state}\\ &\text{constraints}\end{aligned}$$

where x and u are the state and control inputs and $\tilde{x}$ and $\tilde{u}$ are tentative state and control. Together, x and u form a trajectory.

Underneathe the hood, trajectory optimization is based on the idea of sequential convex programming, where a nonconvex optimization problem is solved by breaking it down into a series of solvable convex approximations at each sequential step. Given an initial guess, sequential convex programming will create a convex approximation to the problem around the guess and, within a box constraint, return a candidate solution to the convex problem. Then, the candidate solution is evaluated against the original nonconvex problem and if the improvements is justified, the candidate solution is reused as an initializer in subsequent convex approximation problem.

In order to ensure that constraints (possibly nonconvex) are met, the inequality and equality constraints are included in the convex approximation by first convexifying them with a hinge function and absolute value function, respectively. The hinge function is defined as $\|x\|^+ = \max(x, 0)$. At constraint satisfaction, both inequality and equality constraints are driven to zero. The convex approximation incentivizes constraint satisfaction by increasing a multiplicative penalty coefficient to the constraint at each sequential step. The algorithm is presented as follows.

The algorithm **??** was reproduced from [11] and more details can be found there.

Previously in free space optical communication, scientists would precompute the pose of the emitter and location of the receiver satellite in LEO and compile a look-up table that will be referenced at run time by the emitter. In other words, at time t, the emitter onboard controller will reference the location of the other satellite at time t and then reorient itself to point in that general direction.

Often times, this will be sufficient to establish and maintain a lock. Although this method has been met with moderate success in OPALS (Optical PAyload for Lasercomm

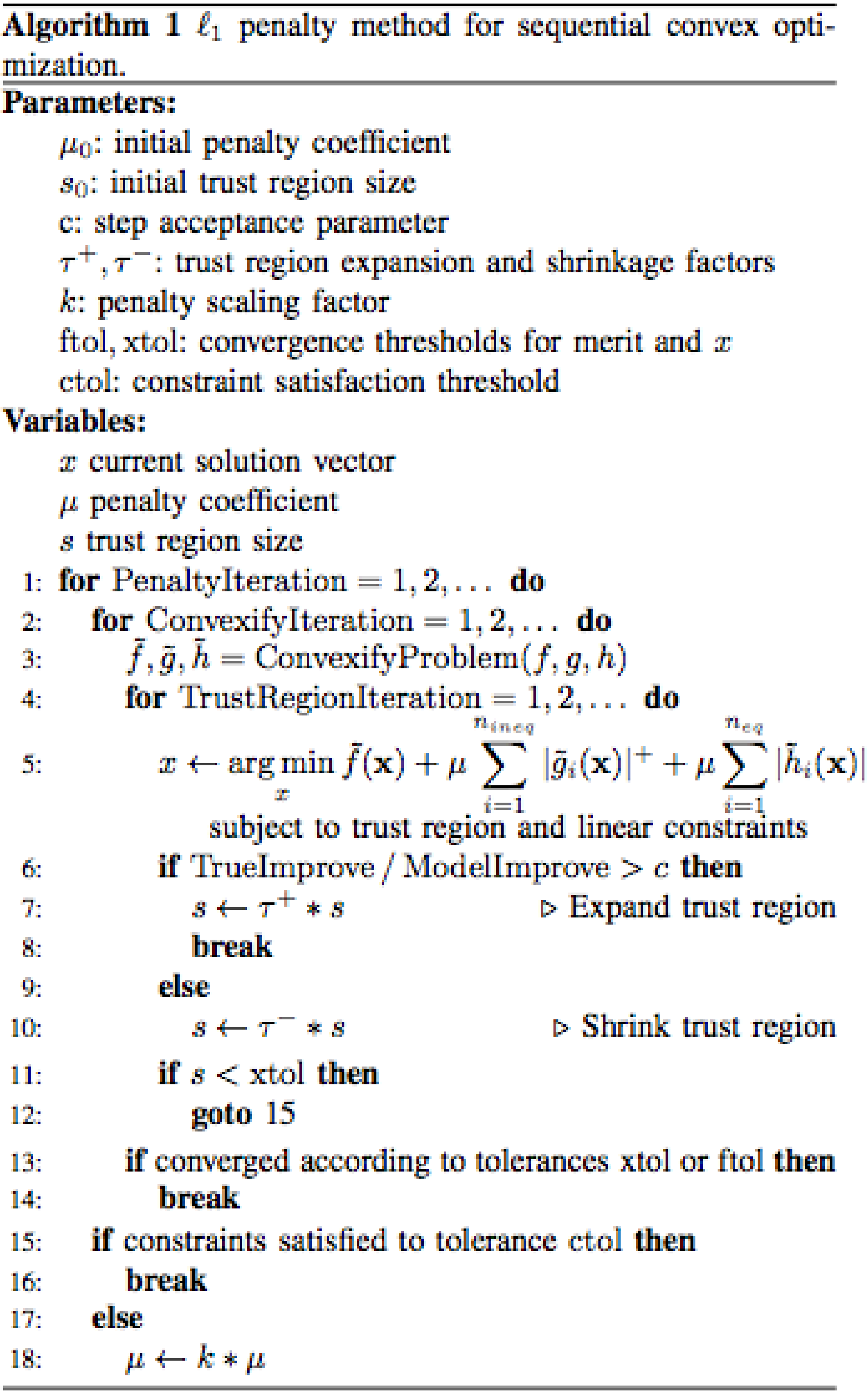
Science) (**??** and **??**) and OICETS (Optical Inter-orbit Communications Engineering Test) (**??**), it denies the control engineer from taking advantage of the pre-computed knowledge of each satellite's trajectory, even if we only know it accurately over a finite horizon. For sufficiently high orbits (160km to 2000km), the effects of atmospheric friction and drag on the satellites are negligible. However, there also exists high fidelity models that can accurately compensate for these friction effects if they had a significant effect on the satellites. These higher fidelity models can then be in turned used to estimate the satellites' poses. In this work, the assumption is made that the effects of friction and drag are negligible.

In addition, the current method cannot handle including constraints on control inputs, constraints on satellite orientation, such as the aforementioned nadir constraint, or constraints on the pointing accuracy for optical communication. The inability to include constraints in current methods makes it difficult to parametrize and tune the performance of the controller. Moreover, the current method only prescribes a control strategy for one of the two agents in the communication link. Either the emitter reorients itself and accomodates the receiever or the receiver reorients itself and accomodate the emitter.

Trajectory optimization is an attractive technique for the problem of motion planning in free space optical communication because it can easily handle including constraints in the problem formulation and allows the control engineer to tune for desired performance. Moreover, trajectory optimization's framework extends easily for multi-agent control because the control engineer can simply extend the state vector.

Summary

In this chapter, trajectory optimization, a technique based on sequential convex programming, is proposed to tackle the problem of motion plan generation in free space optical communication. It is an attractive technique because it can include constraints as well as easily handle multiple agents in the system.



Algorithm 1 ℓ_1 penalty method for sequential convex optimization.

```
Parameters:
    $\mu_0$: initial penalty coefficient
    $s_0$: initial trust region size
    c: step acceptance parameter
    $\tau^+, \tau^-$: trust region expansion and shrinkage factors
    $k$: penalty scaling factor
    ftol, xtol: convergence thresholds for merit and $x$
    ctol: constraint satisfaction threshold
Variables:
    $x$ current solution vector
    $\mu$ penalty coefficient
    $s$ trust region size
 1: for PenaltyIteration = 1, 2, ... do
 2:   for ConvexifyIteration = 1, 2, ... do
 3:     $\bar{f}, \bar{g}, \bar{h}$ = ConvexifyProblem($f, g, h$)
 4:     for TrustRegionIteration = 1, 2, ... do
 5:       $x \leftarrow \arg\min_{x} \bar{f}(\mathbf{x}) + \mu \sum_{i=1}^{n_{ineq}} |\bar{g}_i(\mathbf{x})|^+ + \mu \sum_{i=1}^{n_{eq}} |\bar{h}_i(\mathbf{x})|$
            subject to trust region and linear constraints
 6:       if TrueImprove / ModelImprove > $c$ then
 7:         $s \leftarrow \tau^+ * s$                ▷ Expand trust region
 8:         break
 9:       else
10:         $s \leftarrow \tau^- * s$                ▷ Shrink trust region
11:       if $s <$ xtol then
12:         goto 15
13:     if converged according to tolerances xtol or ftol then
14:       break
15:   if constraints satisfied to tolerance ctol then
16:     break
17:   else
18:     $\mu \leftarrow k * \mu$
```

Figure 2.2: Algorithm for trajectory optimization

Chapter 3

Constraint Formulation

Dynamic and Input Constraint

For either satellites, fuel constraints or hardware limits might impose a cap on the control effort that can be exerted by the craft. We can formulate this as a standard constraint in the framework for trajectory optimization as follows:

$$\begin{aligned} u_t &\in U \ , \ \forall t \in T \\ u_t &\leq U_{max} \\ -u_t &\leq U_{min} \end{aligned}$$

In addition, the satellites have to obey dynamical constraints from one time step to the next. Therefore, the dynamics are formed into the standard equality constraint. In other words,

$$x_{t+1} = f(x_t, u_t) \ , \ \forall t \in T$$

Nadir Constraint

In a satellite system, the nadir point is the point on the craft that must be pointed towards the source of gravity. In this case, it would be earth. The nadir constraint is important because it allows satellites to maintain in communication contact with earth through traditional radio means. In free space optical communication, this nadir constraint can be enforced during a optical communication session or before and after.

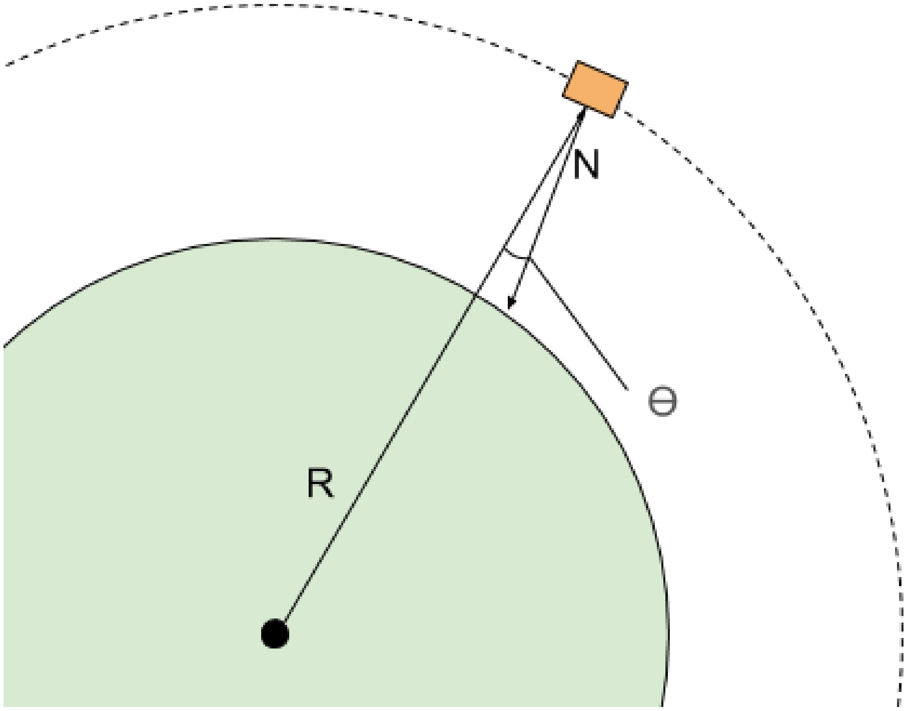


Figure 3.1: Schematic drawing of the nadir constraint

In Fig. 3.1, the satellite is shown in orange and the planet in green. The vector R represents the radial vector pointing to the nadir point of the satellite, which is a point of the earth-pointing satellite face, the vector N is the normal vector of the earth-pointing satellite face, and θ (not to be confused for the satellite pitch) is the angle between these two vectors. If perfectly aligned, vector R and N would be collinear. In other words, $\theta = 0$. Therefore, the nadir constraint is prescribed to be:

$$\bar{R} \cdot \bar{N} = \cos\theta < \delta$$

where $\bar{R}$ and $\bar{N}$ are normalized vectors. Therefore, we can prescribe a 5 degree nadir pointing constraint by setting δ to be $\cos(5^o) \approx .996$.

In standard optimization form, the nadir constraint is rewritten as:

$$g(x) \leq 0$$
$$\mathbf{R_s} \cdot \mathbf{N} + \delta \leq 0$$

and δ is an adjustable parameter.

Pointing Constraint

The pointing constraint refers to the line of sight lock that must be established between the emitter and receiver satellite. In Fig. 3.2, the emitter satellite is in orange and to the left while the receiver satellite is in orange and to the right. The planet is once again in green. Ideally, the vector B will coincide with the origin of vector A, the receiver satellite's reception area. If that is not the case, then Δ is the 2-norm distance between the point of intersection between vector B and the plane, defined by the normal vector A.

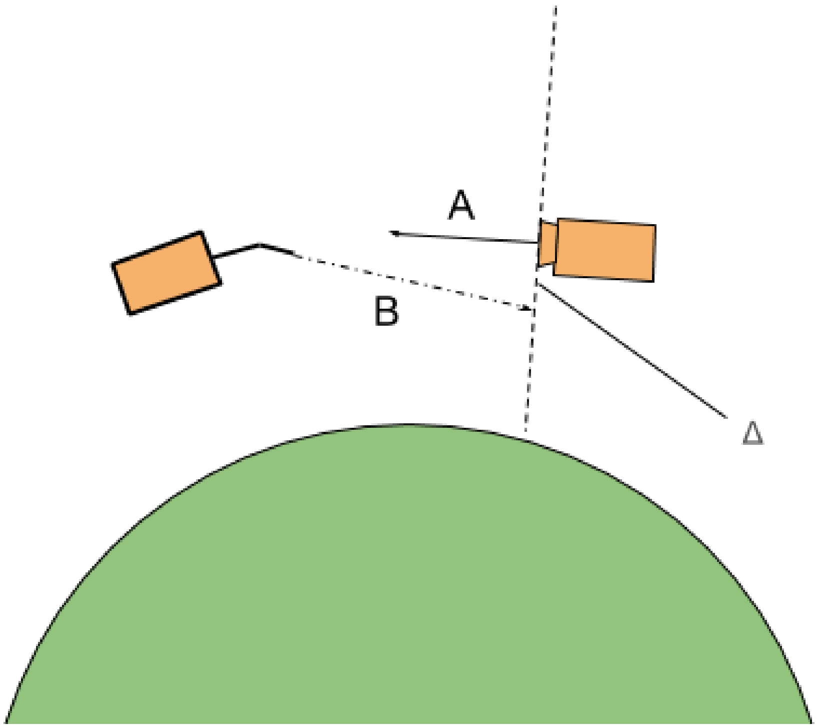


Figure 3.2: Schematic drawing of the pointing constraint

The plane defined by vector A can also be defined by two basis vectors v_1 and v_2. Subsequently, the intersection (henceforth referred to as w) between vector B and plane defined by vector A can be parametrized by the basis vectors.

$$p_2 + \alpha v_1 + \beta v_2 = w$$

where p_2 is the origin of vector A in Fig. 3.2 on the receiver spacecraft.

That same point w can also be described by extending vector B from p_1, the point from which the laser emanates from the emitter spacecraft.

$$p_1 + \gamma B = w$$

Therefore, the coefficents can be determined by solving the system of linear equations. Once the coefficients are determined, γ is used to determine w. At which point, the error Δ can be defined as:

$$\Delta = \|w - p_2\|_2$$

Defining $\eta > 0$ as the acceptable Euclidean distance of error, we can normalize the pointing error Δ by using the standard optimization form as follows:

$$\begin{aligned} g(x) &\leq 0 \\ \frac{\Delta}{\eta} - 1 &\leq 0 \end{aligned}$$

Summary

The performance requirements for free space optical communication is recasted as constraints in order to fit them into the framework of trajectory optimization. The requirement that both satellites have to point towards earth and the requirement that both satellites have to maintain a pointing lock, referred to as the nadir and pointing requirement, respectively, are geometrically derived and formed as inequality constraints. The dynamic constraint is an equality constraint and the input constraint is a series of inequality constraints placed on the input.

Chapter 4

System Dynamics

Satellite Design

As a a baseline spacecraft, I chose to use NASDA's (National Space Development Agency of Japan, formerly JAXA) OICETS satellite as a benchmark satellite design. The emitter satellite uses a two axis gimbal-actuated turret to point the laser towards the target satellite with joint constraints. The azimuth joint is constrained between -190 to 190 degrees whereas the pitch joint is constrained between 0 and 120 degrees. Both satellites have thrusters that allow them to exert pure moments about their center of masses and in turn, change their respective yaw, pitch and roll directly. A beacon on the receiver satellite will be tracked by a CCD sensor on the emitter satellite with a narrow field of view. This sensor typically would provide the closed loop feedback for point error. For the purpose of this paper, I will assume a similarly spacecraft design (thrusters that exert pure moments and mass properties) for the receiver as well. More specific information can be found in [8].

Dynamics Model

In this problem, we use the standard orbital dynamics model for a point mass in space. The equations of motion for such a system is

$$\ddot{r} - \frac{\hat{h}^2}{m^2 r^3} = -\frac{GMm}{r^2}$$

$$m\dot{r^2}\dot{\sigma} = 0$$

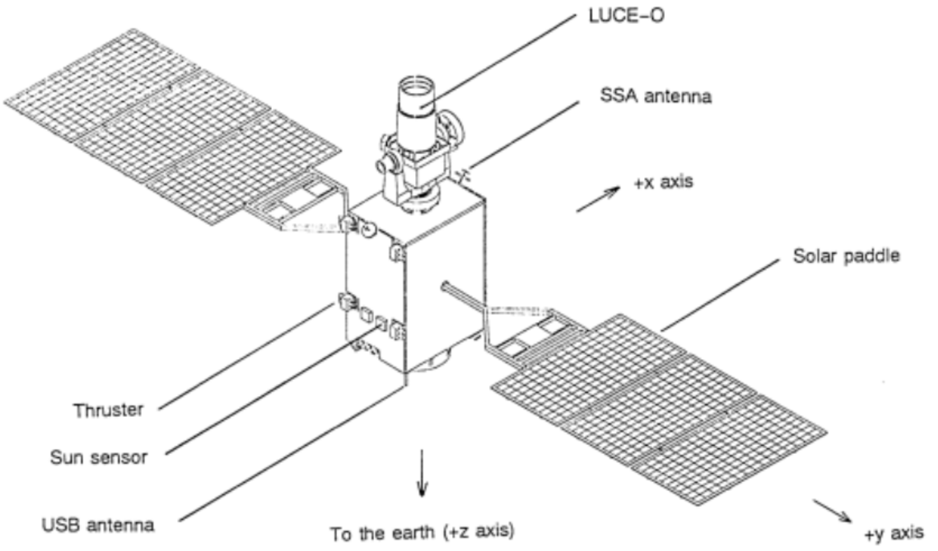


Figure 4.1: OICETS satellite with a two axis robot arm mounted on top

where the angular momentum is conserved (ignoring friction effects) and first equation deals with the radial distance of the satellite from the earth. Note that we use a cylindrical polar system here to reduce the number of states, and subsequently the optimization problem computation time, required for our system. An additional implementation detail is that satellite orbits tend to have an inclination (α) with respect to the equatorial plane. Although it is not shown here, the orbits generated by the equation of motion above can be transformed by applying a rotation of α.

However, the large timescale and distances for this orbital satellite system will tend to cause numerical issues. For this reason, we use a nondimensionalized form of the equations.

$$\frac{\partial^2 \rho}{\partial \tau^2} = \frac{1}{\rho^3} - \frac{1}{\rho^2}$$
$$\frac{\partial \sigma}{\partial \tau} = \frac{1}{\rho^2}$$

where $\tau = \omega_k t$ is the nondimensionalized time and ω_k the Kepler frequency. $\rho = \frac{r}{r_0}$ is the nondimensionalized radial vector and r_0 is the normalized radius. For more information, see

[10]

The satellite, both emitter and receiver, can reorient in space. To keep the problem tractable, we assume that both spacecrafts can exert pure moments about its body axis and the spacecraft has the inertial matrix of a massive box.

$$\tau = \mathbf{J}\alpha + \omega \times \mathbf{J}\omega$$

$$\mathbf{J} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}$$

$$\lambda_1\ddot{\psi} = (\lambda_2 - \lambda_3)\dot{\phi}\dot{\theta} + M_1$$

$$\lambda_2\ddot{\theta} = (\lambda_3 - \lambda_1)\dot{\psi}\dot{\phi} + M_2$$

$$\lambda_3\ddot{\phi} = (\lambda_1 - \lambda_2)\dot{\psi}\dot{\theta} + M_3$$

where $\alpha = \begin{bmatrix}\psi, \theta, \phi\end{bmatrix}$ are the yaw, pitch and roll rates, respectively.

With these models, the state representation for a two satellite system at time step t is

$$\begin{aligned} x_t = [&\rho_1, \dot{\rho}_1, \sigma_1, \psi_1, \theta_1, \phi_1, \dot{\psi}_1, \dot{\theta}_1, \dot{\phi}_1, \\ &\rho_2, \dot{\rho}_2, \sigma, \psi_2, \theta_2, \phi_2, \dot{\psi}_2, \dot{\theta}_2, \dot{\phi}_2] \end{aligned}$$

where the subscript represents the satellite index and 1 being the emitter satellite.

The control inputs for the system at a time step t:

$$u_t = [M_{1x}, M_{1y}, M_{1z}, M_{2x}, M_{2y}, M_{2z}]$$

where M_{ij} are moments in the ith spacecraft's principal directions for the j^{th} axis. Note that spacecraft 1 is assumed to be the emitter.

Therefore, for a $t = 1 \ldots T$ horizon, the state trajectory will have dimensions $(18+6) \times T$ by 1. In other words, a state trajectory will look like:

$$(x, u) = [x_0 \; u_0 \; x_1 \; u_1 \ldots x_T \; u_T]$$

Recalling that trajectory optimization has the following formulation:

$$
\begin{aligned}
\min_{x,u}\; & \mathcal{T}(\tilde{x},\tilde{u},x,u) \\
s.t\;\; & \text{initial state} \\
& \text{constraints}
\end{aligned}
$$

we can now fill in the blanks and arrive at a more complete formulation.

$$\min_{\tilde{x},\tilde{u}} \quad \sum_{t=1}^{T}(x_t-\tilde{x_t})^TQ(x_t-\tilde{x_t})+\sum_{t=1}^{T}(u_t-\tilde{u_t})^TR(u_t-\tilde{u_t}) \tag{4.1}$$

$$s.t. \quad \tilde{x_0}=x_{init} \tag{4.2}$$

$$\forall t\in 1\ldots T\ :\ x_{t+1}=f(x_t,u_t) \tag{4.3}$$

$$\forall t\in 1\ldots T\ :u_t\in U \tag{4.4}$$

$$\forall t\in 1\ldots T\ :\mathbf{R_s}\cdot\mathbf{N}+\delta<0,\quad \delta<0 \tag{4.5}$$

$$\forall t\in 1\ldots T\ :\frac{\Delta}{\eta}-1<0,\quad \eta>0 \tag{4.6}$$

In Eq. 4.1, we chose a simple linear quadratic cost to penalize solutions that deviate too far from the tentative trajectory. In Eq. 4.2, the initial condition constraint is enforced whereas the dynamics constraint, as described earlier in this chapter, is enforced in Eq. 4.3.In Eq. 4.4, we enforce control input constraints at each time step, if there are any, and Eq. 4.5 and 4.6 are the performance constraints that we want to enforce in free space optical communication.

To prepare the results in this work, each of the aforementioned constraints is formulated in MATLAB and then inputted as arguments into MATLAB's FMINCON, which will then return an optimized trajectory as a solution.

Summary

In this chapter, we summarized the dynamics of the satellite system. The assumption was made that friction forces would be negligible and pure moments around the principal axes can be exerted around the spacecraft. Under those assumptions, the standard orbital dynamics model for a point mass in space was chosen for the satellite's Cartesian motion whereas the standard rigid body rotational dynamics was chosen for the satellite's rotational motion.

Chapter 5

Time Varying LQR

In standard LQR (linear quadratic regulator), the system is linear and can be expressed in the form:

$$x_{t+1} = Ax_t + Bu_t$$

Given a linear quadratic cost,

$$\text{cost} = x_t^T Q x_t + u_t^T R u_t$$

a Ricatti equation can be solved to determine K, the optimal feedback gain that will minimize the quadratic cost.

However, in order for it to be useful in trajectory optimization, where the system can potentially be nonlinear, we extend LQR to nonlinear systems by taking advantage of linearization.

Given a nonlinear system,

$$x_{t+1} = f(x_t, u_t)$$

we make the first order linearization around an equilibrium point (x^*, u^*).

$$\begin{aligned} x_{t+1} &\approx f(x^*, u^*) + \frac{\partial f}{\partial x}(x^*, u^*)(x_t - x^*) + \frac{\partial f}{\partial u}(x^*, u^*)(u_t - u^*) \\ x_{t+1} - f(x^*, u^*) &\approx A(x_t - x^*) + B(u_t - u^*) \\ x_{t+1} - x^* &\approx A(x_t - x^*) + B(u_t - u^*) \end{aligned}$$

where $\frac{\partial f}{\partial x}(x^*, u^*) = A$ and $\frac{\partial f}{\partial u}(x^*, u^*) = B$ and $x^* = f(x^*, u^*)$. Thus, the nonlinear system is transformed into a standard linear system form that we can easily apply LQR to.

In trajectory optimization, we have a returned optimized trajectory (a state sequence) of feasible points. We linearize the system around each of these feasible points (as described above) and compute via LQR the optimal feedback gain at that time step. However, since the nonlinear system is linearized at each time step of the state sequence, the time-varying LQR gain must be computed instead. The time-varying gain is computed once a horizon H is defined.

$$\begin{aligned}
P_0 &= 0 \\
K_i &= -(R_{H-i} + B_{H-i}^T P_{i-1} B_{H-i})^{-1} B_{H-i}^T P_{i-1} A_{H-i} \\
P_i &= Q_{H-i} + K_i^T R_{H-i} K_i + (A_{H-i} + B_{H-i} K_i)^T P_{i-1} (A_{H-i} + B_{H-i} K_i)
\end{aligned}$$

Therefore, in time-varying LQR, the optimal control at the i^{th} step from the end of the state sequence is

$$u_{H-i} - u_{H-i}^* = K_i(x_{H-i} - x_{H-i}^*)$$

Summary

In this chapter, we presented the standard LQR and the linearized extension to LQR as important steps to arrive at the time varying LQR that is used in trajectory optimization. Time varying LQR is used to synthesize a controller over the entire optimized trajectory that can then be used to implement online.

Chapter 6

Simulation Results

Simulation Parameters

Henceforth, tentative trajectory is the initial trajectory given to trajectory optimization to begin the search for an optimal trajectory.

The target trajectory is the trajectory returned by trajectory optimization. It is an open loop trajectory and might not entirely feasible.

The simulated trajectory is created when a time-varying LQR controller is synthesized over the target trajectory. Therefore, the simulated trajectory takes the model for the system plant and propagates it forward using the a set of linear feedback gains. It is considered to be a closed loop trajectory. A visualization of a simulated trajectory is shown in Fig. 6.1.

The simulations were performed with the following initial conditions:

- Simulation horizon of $(\sigma_f - \sigma_0) = \frac{\pi}{8}$ (approx. 2500 km apart)
- Satellite mass of 600 kg
- Satellite height of 2000000 meters
- Orbit inclination $\alpha = 5$ degrees
- Orbital period of 127 minutes
- Time step dT of 1 second

Note that originally, I prescribed a time step dT of .1 which generated a large trajectory that took very long to compute. Choosing $dT = 1$ offered the same fidelity of results but with less computation time.

For both the emitter and receiver satellite, the initial yaw ψ was chosen such that they nadir constraint is met. In addition, both satellites are initialized to start in different phases (σ_0) of their respective orbit, but on the same orbital plane. However, they are no more than $\frac{\pi}{8}$ in phase apart from each another.

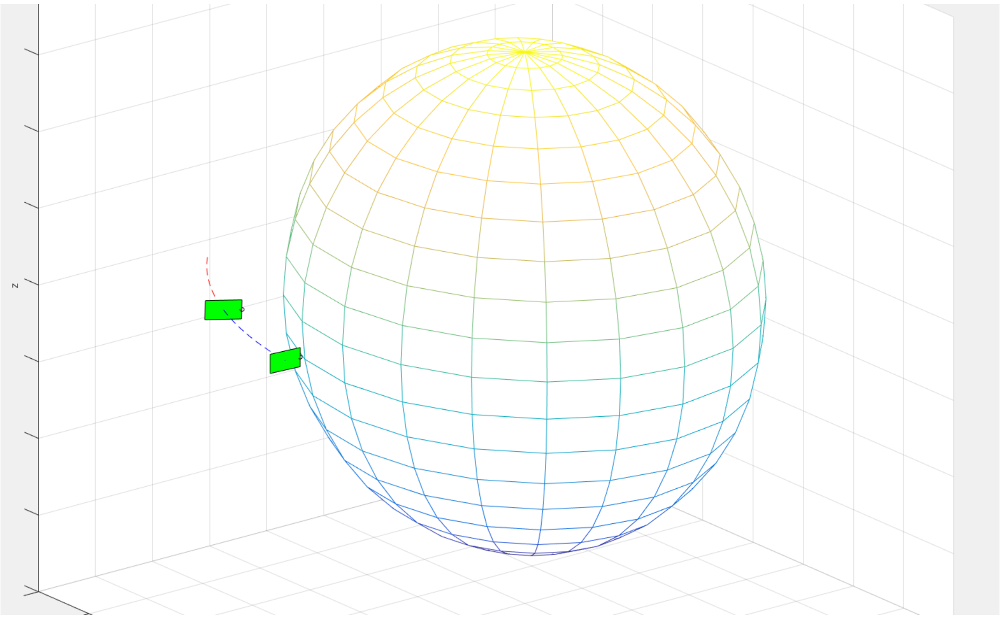


Figure 6.1: Side view of the orbit visualizer for debug purposes

In Fig. 6.1, a side view of the orbit visualizer is shown. The two green rectangles represent not-to-scale satellite bodies and the black circles are the spacecrafts' nadir points. In this time step, the satellites fulfill the nadir constraint because each of them point towards the planet. I built this simulation environment in MATLAB and it is used to visualize my work.

Tentative Trajectory

In this section, I demonstrate the tentative trajectory I provided to start off the optimization. These plots will be useful to contrast the results I develop later. In Fig. 6.2, the body angles

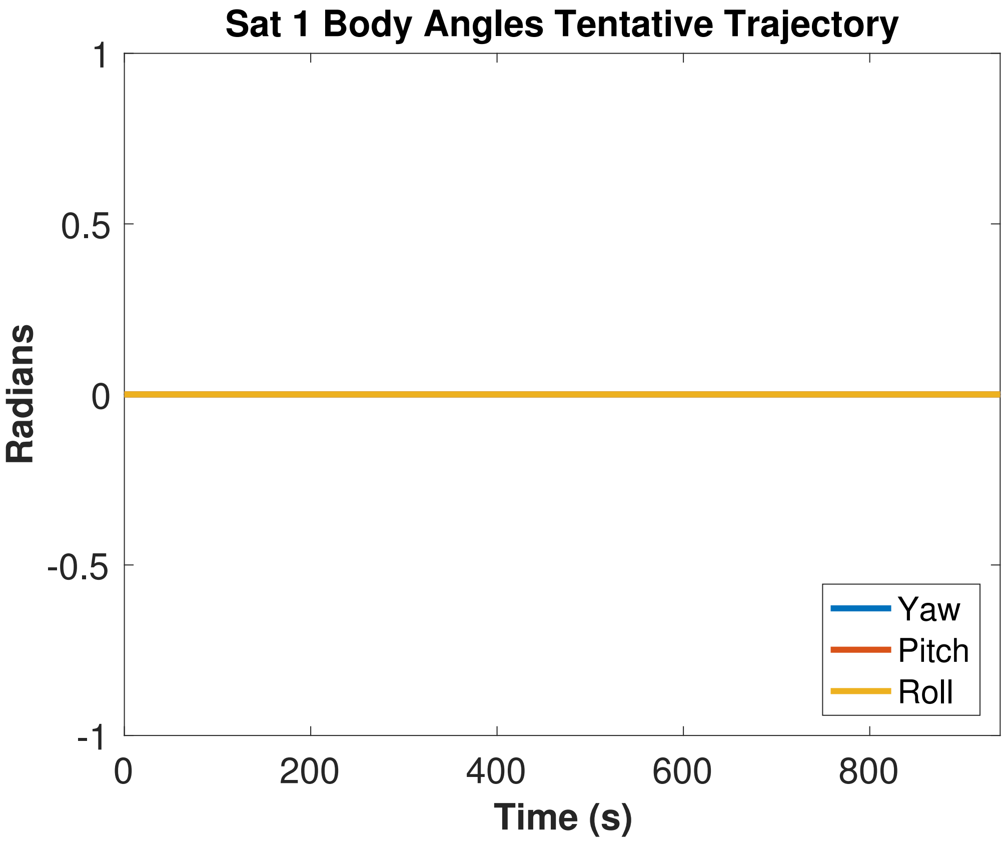


Figure 6.2: Plot of the emitter satellite's body angles in tentative trajectory

of the emitter satellite are shown. Other states within the target trajectory are omitted for clarity. The y axis has units of radians and the x axis has units of time in seconds. Note that essentially, the body angle stays the same throughout the entire tentative trajectory.

A plot for the tentative control trajectory is not provided because it is initialized to be all zeros.

Target Trajectory

In contrast to Fig. 6.2, the yaw ψ in Fig. 6.3 decreases to compensate for the satellite going around the planet. If the satellite did not start decreasing its yaw, then the nadir constraint will be unsatisfied. Fig. 6.3 plots the emitter satellite's body angles. Other states within the target trajectory are omitted for clarity. The receiver target trajectory are very similar and also omitted for brevity. The y axis has units of radians and the x axis has units of time

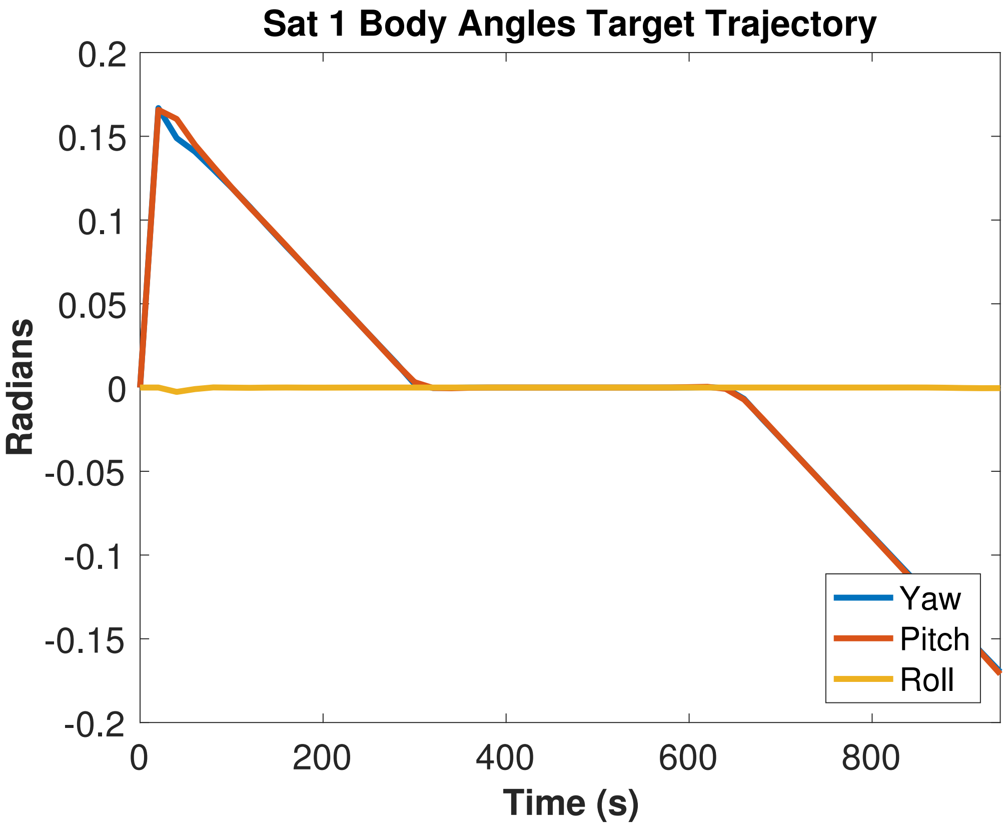


Figure 6.3: Plot of the emitter satellite's body angles as part of the target trajectory

in seconds

Fig. 6.4 and 6.5 are plots of the two satellites' control inputs, which largely reflect changes being made to the yaw angle of the spacecrafts. The y axis has units of kgm^2/s^2 and the x axis has units of time in seconds

Simulated Trajectory

In contrast to Fig. 6.2 and 6.3, the yaw ψ (Fig. 6.6) not only decreased but decreased earlier in the simulated trajectory. Compared to the target trajectory, the simulated trajectory is feasible and smoother. In Fig. 6.6, the emitter satellite's body angle is plotted. Other states within the simulated trajectory are omitted for clarity. The receiver target trajectory are very similar and also omitted for brevity. The y axis has units of radians and the x axis has units of time in seconds

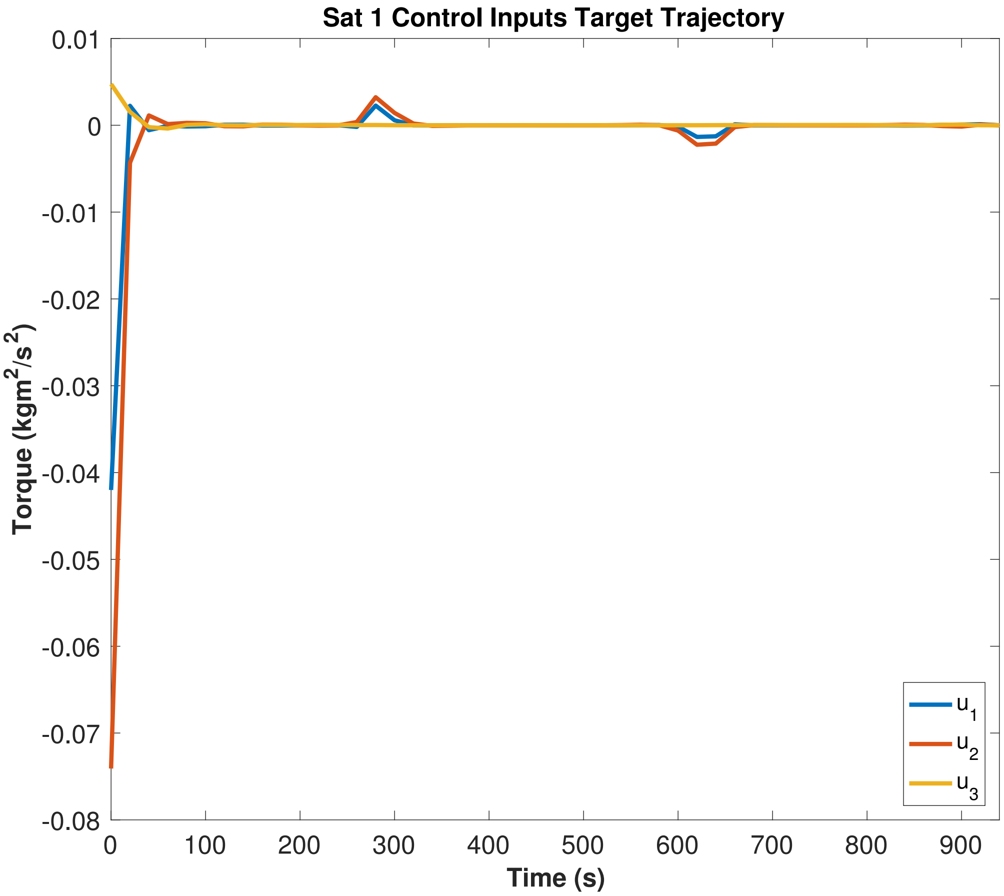


Figure 6.4: Plot of the emitter satellite's control inputs as part of the target trajectory.

The control trajectory in Fig. 6.7, derived from a set of feedback gains, is also much smoother compared to the target control trajectory (Fig. 6.4 and 6.5). The y axis has units of kgm^2/s^2 and the x axis has units of time in seconds. The receiver satellite's control inputs are very similar and are omitted for brevity.

Nadir Constraint

In Fig. 6.8, the nadir score for the tentative trajectory is shown to contrast the results developed later. It is clear that the nadir constraint is not satisfied because the score is positive. Having a negative score means that the constraint is satisfied. Because this is the tentative trajectory, it is clear that the nadir constraint is not satisfied.

In Fig. 6.9 and 6.10, the nadir score for the target and simulated trajectory is shown.

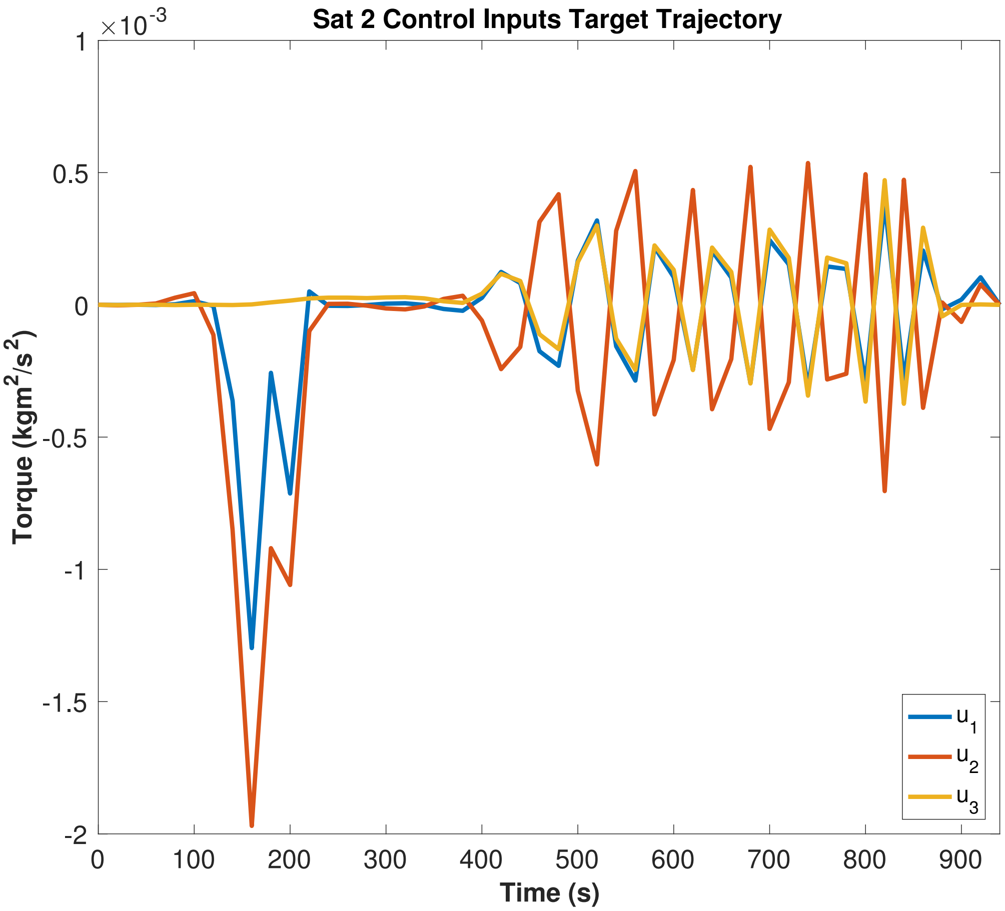


Figure 6.5: Plot of the receiver satellite's control inputs as part of the target trajectory.

The nadir constraint is satisfied in both the target and simulated case. The characteristic for the simulated nadir score is different from the target trajectory can probably be attributed to the linearization in when computing the LQR feedback gains. The linearization might have contributed to inaccuracies that resulted in the results in Fig. 6.10. Despite the difference, the nadir score remains wholly negative for the entire duration.

Pointing Constraint

In Fig. 6.11, a plot of the pointing score throughout the session is shown. A negative score means that the constraint was satisfied. A score of 1000 means that the laser was 1000 times away from an acceptable performance (i.e. 1000 meters away from where the laser should be pointing). Observe that the optimized, constraint-satisfied target trajectory is still infeasible

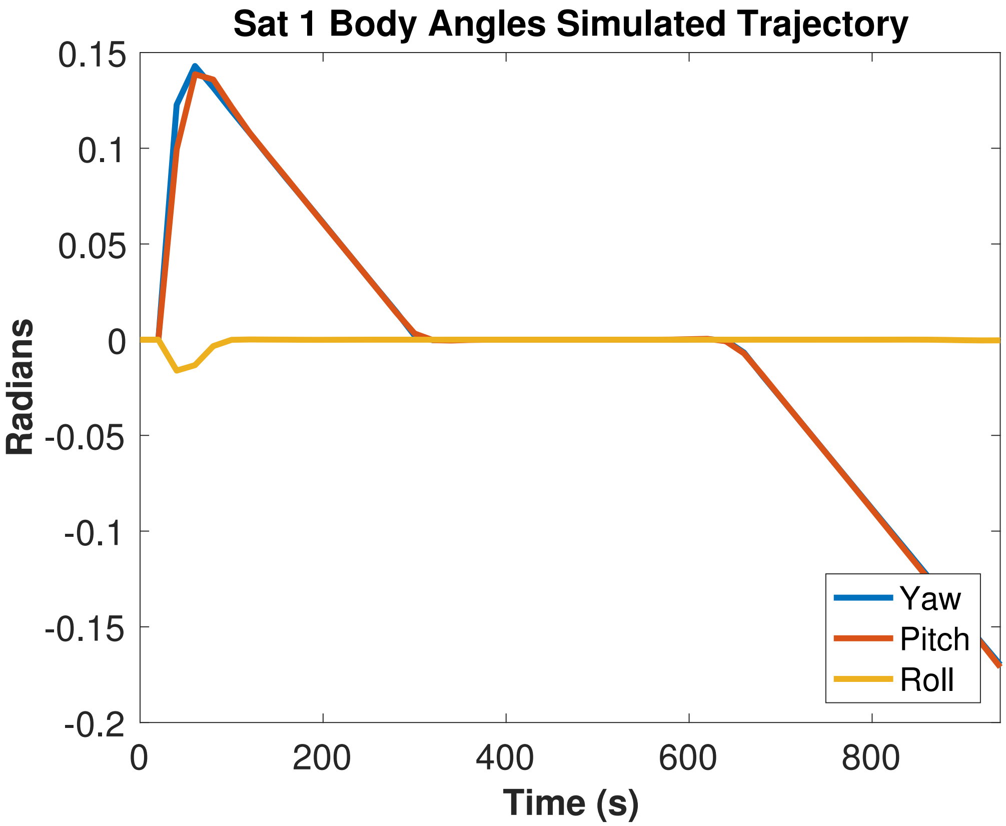


Figure 6.6: Plot of the emitter satellite's body angles as part of the simulated trajectory

with respect to achieving our performance criteria. Contrasted to Fig. 6.13, the somewhat oscillatory high and low score behavior is repeated, although smaller is magnitude.

In Fig. 6.12 and Fig. 6.14, the simulated results from synthesizing time varying LQR over the aforementioned target trajectory (Fig. 6.11, 6.13). It is clear that the simulated runs also failed to satisfy the performance requirements for laser pointing. This can possibly be attributed to errors in the linearization or FMINCON's numerical inaccuracies. Further work should be done to explore whether this is consistent behavior with this formulation or if a different formulation is needed for better performance. One possible extension might be to translate the euclidean error to a measure in radians, which might be better behaved.

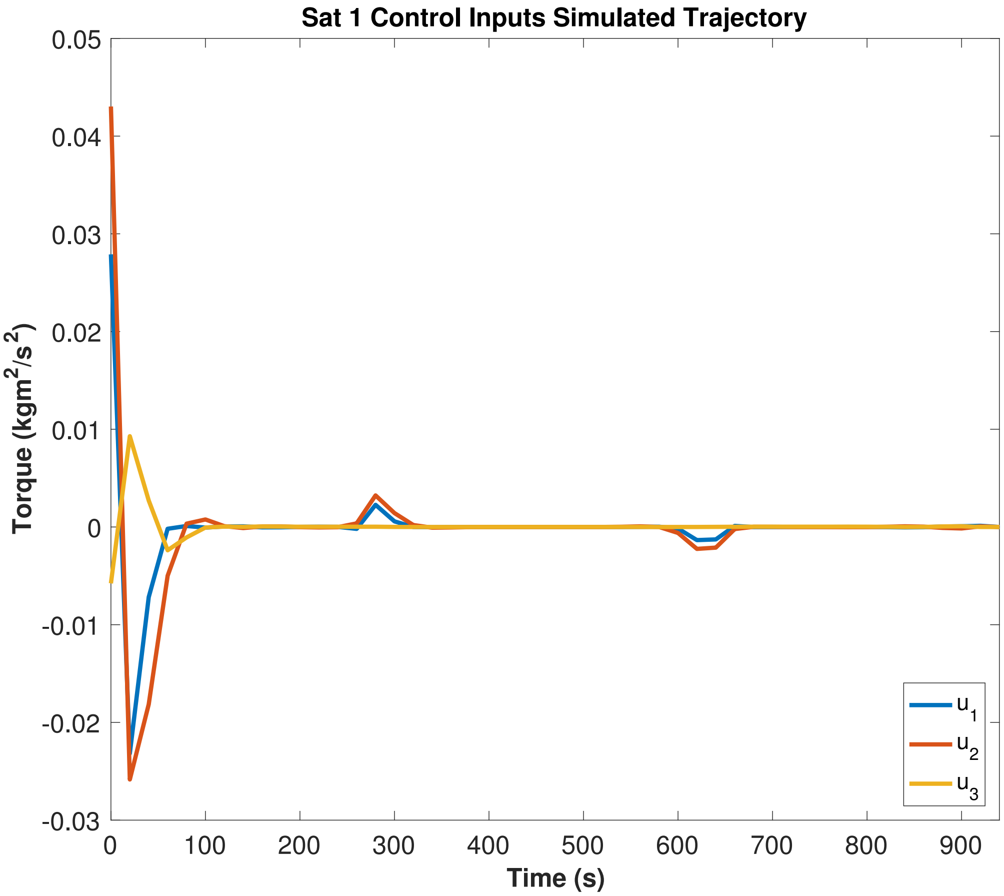


Figure 6.7: Plot of the emitter satellite's control inputs as part of the simulated trajectory.

Summary

In this chapter, simulation results from trajectory optimization are presented. The tentative trajectory, which are initializer trajectories for the optimization problem, are shown to not obey performance requirements (i.e. the nadir and pointing constraints). However, trajectory optimization returns target trajectories that are feasible with respect to all of the constraints. Taking these target trajectories, we generated simulation trajectories, which are created by implementing online with a synthesized time varying LQR feedback controller, that would still obey performance requirements.

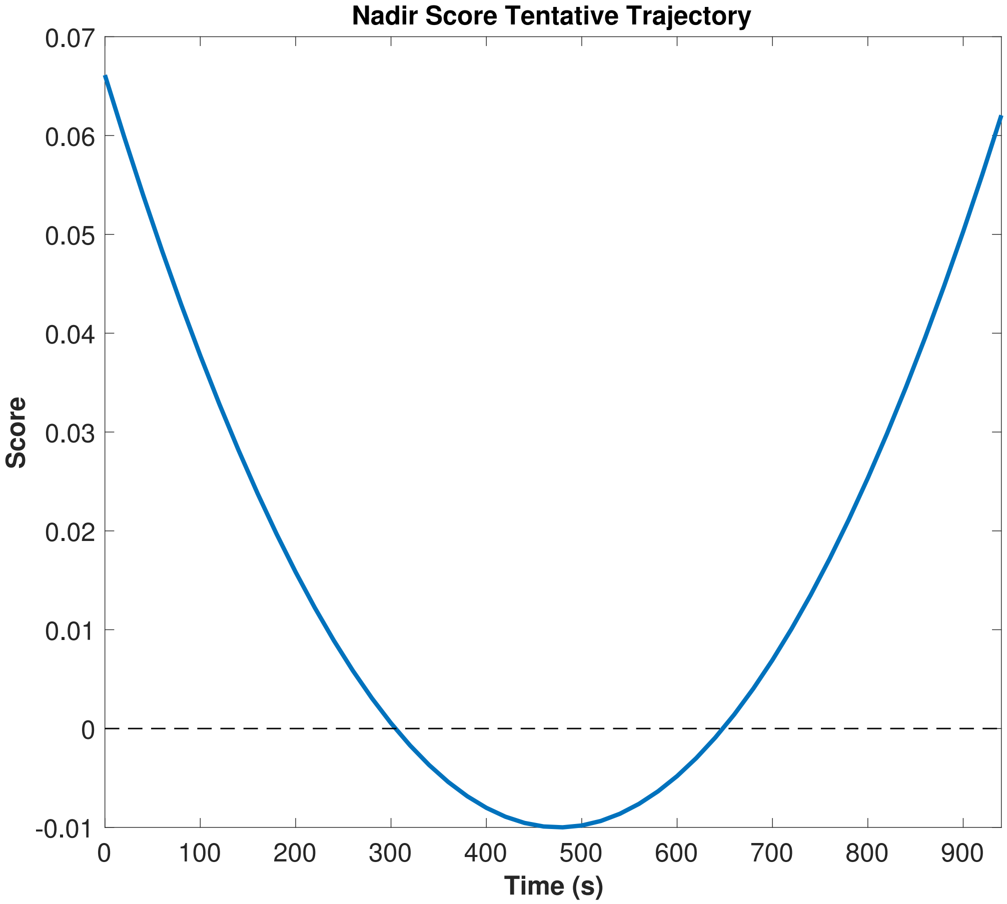


Figure 6.8: Plot of the nadir constraint score for the tentative trajectory

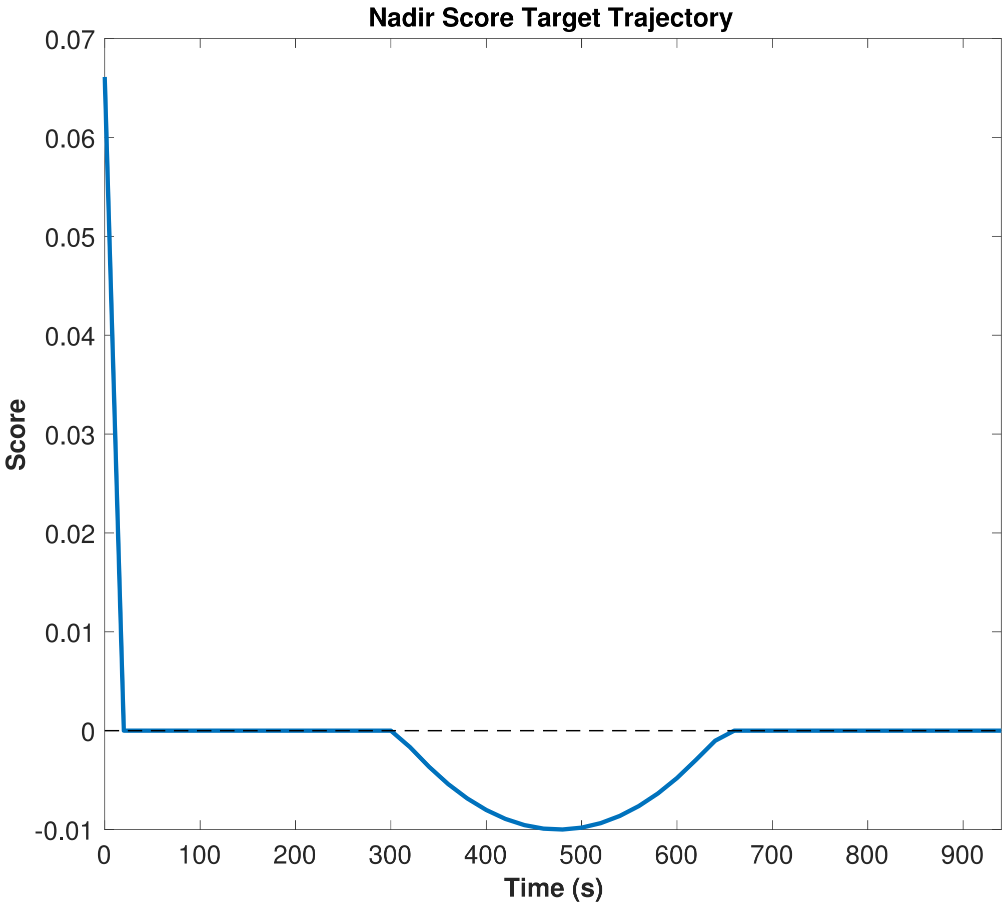


Figure 6.9: Plot of the nadir constraint score for the target trajectory.

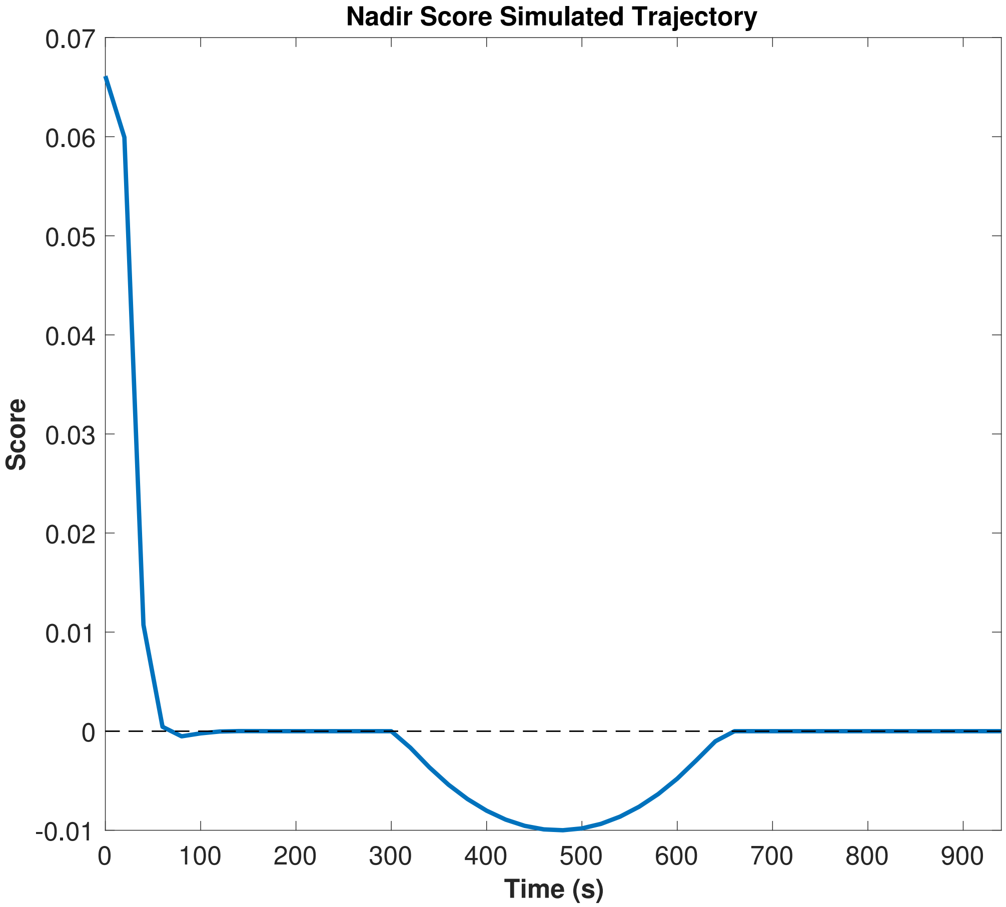


Figure 6.10: Plot of the nadir constraint score for the simulated trajectory.

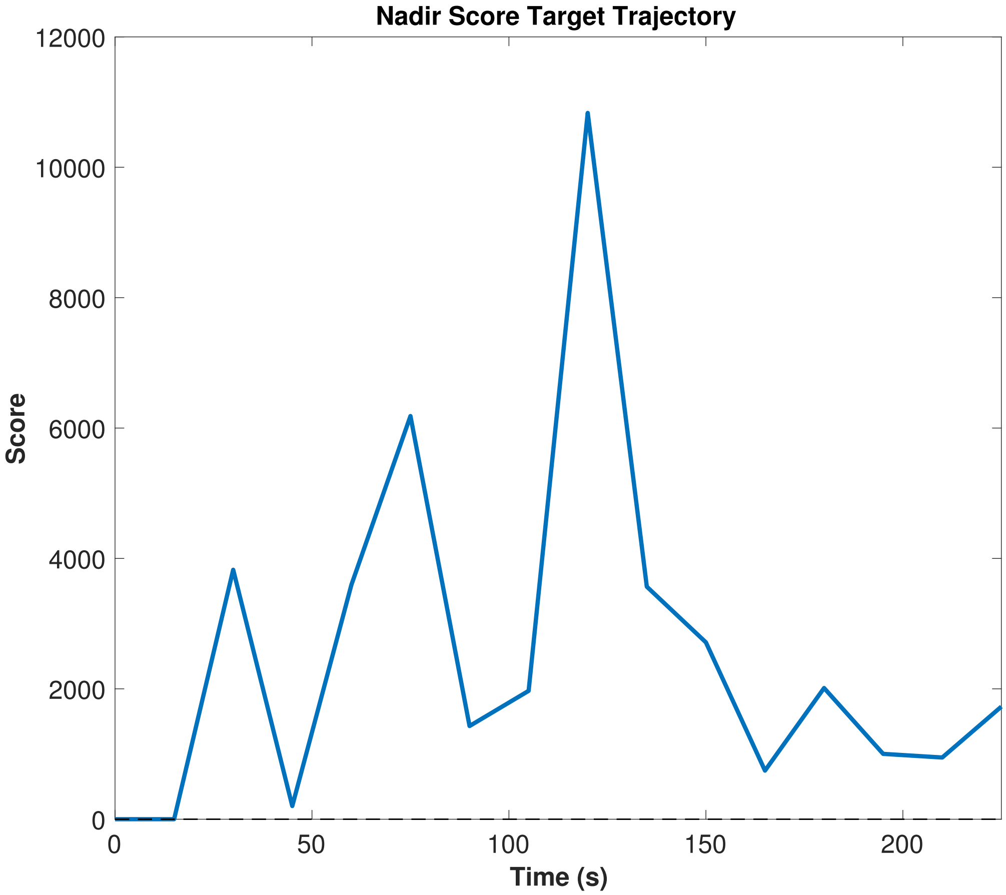


Figure 6.11: Plot of the pointing constraint score for the target trajectory at 2500km apart.

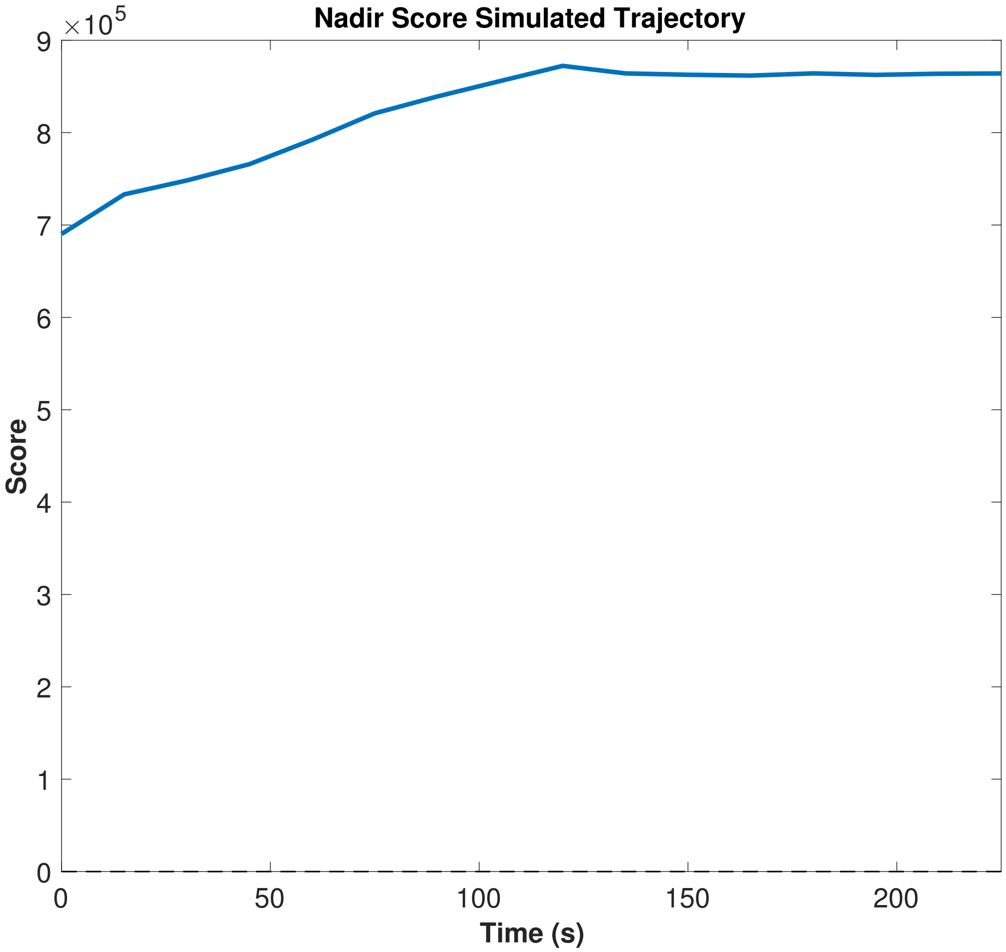


Figure 6.12: Plot of the pointing constraint score for the simulated trajectory at 2500km apart.

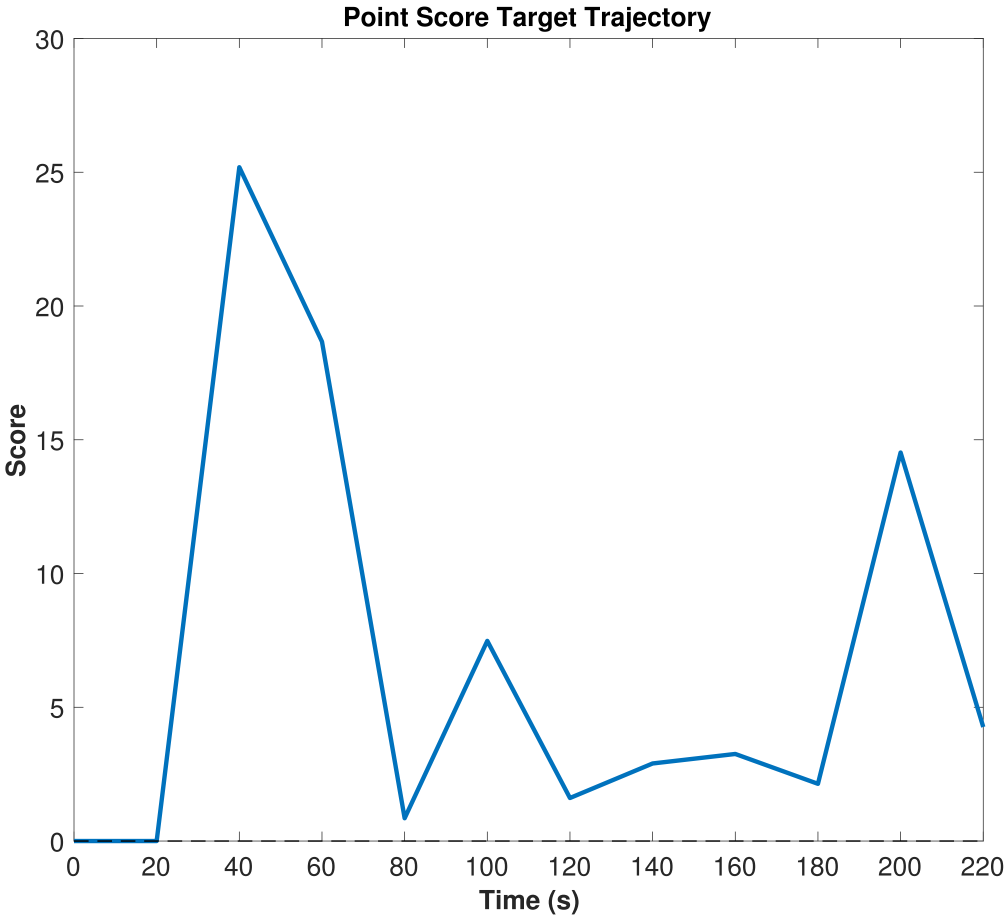


Figure 6.13: Plot of the pointing constraint score for the target trajectory at 120km apart.

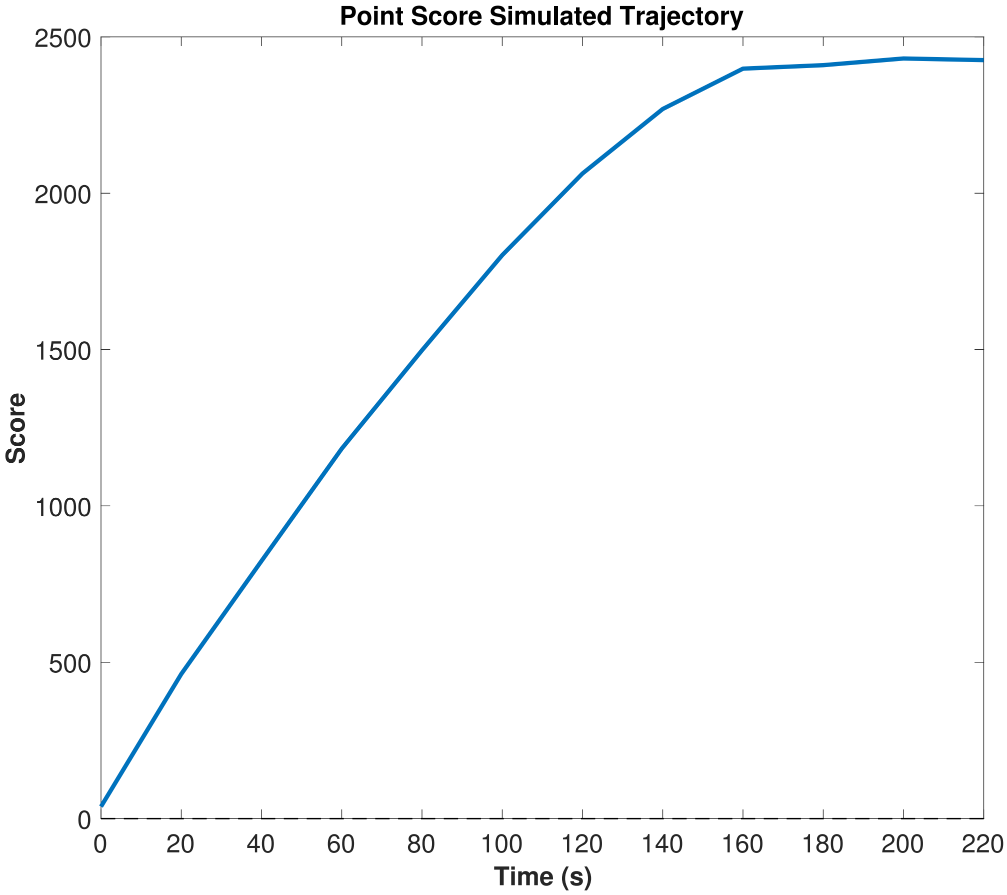


Figure 6.14: Plot of the pointing constraint score for the simulated trajectory at 120km apart.

Chapter 7

Future Work

Although the application of trajectory optimization shows promises in free space optical communication, there are limitations that informed future directions to explore.

First, the use of MATLAB's build-in FMINCON tends to be slow and at times, return inaccurate results. Almost all of the results generated here are done with FMINCON. Instead, a more specific package for trajectory optimization should be explored to see if it can return faster and more consistently accurate results.

Second, the simulated test cases only include cases where the satellites are at a known distance apart and performing an equatorial orbit. Although promising results were generated, future test cases should evaluate whether the algorithm will work for polar orbits or when the satellites have more eccentric orbits.

Third, baseline satellites used in this work also assumed the satellite were able to generate pure moments around its principal axis. Moreover, most satellite designs typically included a robot turret arm to steer the laser, which can absolve the need for the satellite to constantly reorient itself. The robot turret arm should be included in the model in the future to more accurately depict satellite designs.

Finally, LEO-LEO optical communication links are of interest because they represent a first stepping stone towards practice usages of free space optical communication. The next step towards realizing trajectory optimization for free space optical communication is to test the technique against a situation where the two terminals are on different planetary body (i.e. moon and earth).

References

[1] P. Abbeel, A. Coates, and A. Y. Ng. “Autonomous Helicopter Aerobatics through Apprenticeship Learning”. In: *The International Journal of Robotics Research* 29.13 (2010), 16081639. DOI: 10.1177/0278364910371999.

[2] Pieter Abbeel. *CS287 Course Notes*. 2015.

[3] Matthew Abrahamson. *Optical PAyload for Lasercomm Science (OPALS)*. 2014. URL: http://phaeton.jpl.nasa.gov/external/projects/optical.cfm.

[4] Matthew J. Abrahamson et al. “Achieving operational two-way laser acquisition for OPALS payload on the International Space Station”. In: *Free-Space Laser Communication and Atmospheric Propagation XXVII* (2015). DOI: 10.1117/12.2182473.

[5] Takashi Jono et al. “OICETS on-orbit laser communication experiments”. In: *Free-Space Laser Communication Technologies XVIII* (2006). DOI: 10.1117/12.673751.

[6] Hemani Kaushal and Georges Kaddoum. “Optical Communication in Space: Challenges and Mitigation Techniques”. In: *IEEE Communications Surveys and Tutorials* (2016), 11. DOI: 10.1109/comst.2016.2603518.

[7] Soung Sub. Lee. “Dynamics and control of satellite relative motion: designs and applications”. PhD thesis. 2009.

[8] Keizo Nakagawa and Akio Yamamoto. “Preliminary design of Laser Utilizing Communications Equipment (LUCE) installed on Optical Inter-Orbit Communications Engineering Test Satellite (OICETS)”. In: *Free-Space Laser Communication Technologies VII* (1995). DOI: 10.1117/12.207415.

[9] B. V. Oaida et al. “OPALS: An optical communications technology demonstration from the International Space Station”. In: *2013 IEEE Aerospace Conference* (2013). DOI: 10.1109/aero.2013.6497167.

[10] Oliver M. O’Reilly. *Intermediate dynamics for engineers: a unified treatment of Newton-Euler and Lagrangian mechanics*. Cambridge University Press, 2008.

[11] John Schulman et al. “Finding Locally Optimal, Collision-Free Trajectories with Sequential Convex Optimization”. In: *Robotics: Science and Systems IX* (2013). DOI: 10.15607/rss.2013.ix.031.