CONVEX OPTIMIZATION BASED PLANNING AND CONTROL METHODS FOR SPACE-ROBOTIC SYSTEMS

BY

GAURAV MISRA

A dissertation submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Mechanical and Aerospace Engineering

Written under the direction of

Xiaoli Bai

and approved by

____

____

____

____

New Brunswick, New Jersey

October, 2019

ABSTRACT OF THE DISSERTATION

Convex Optimization based Planning and Control Methods for Space-robotic Systems

by

Gaurav Misra

Dissertation Director: Xiaoli Bai

Space-robotic systems are arguably the most promising technologies available currently for on-orbit satellite operations including docking, berthing, and repair, which have been demonstrated in typically manned and semi-autonomous missions. Another important application of space-robotic systems is space debris mitigation. Space debris are uncooperative space objects (i.e. without any internal actuation) including defunct satellites and spent rocket stages, all of which pose tremendous risk to current operational space assets. Autonomous robotic capture, control, and stabilization of such objects are becoming critical. However, space-robotic operations in proximity of such uncooperative object is challenging with large uncertainties.

As a result, optimality, robustness, and tractability constitute some of the desirable properties for any planning and control algorithm used for spacecraft guidance, control, and robotic operations.

Through the development of fast interior point methods for solving convex optimization problems with globally optimality, convex programming methods have been proposed and experimentally validated for real-time guidance and control of space systems. However, most current developments have explored solving locally optimal solutions to highly non-linear and constrained optimal control problems in real-time. The issues of robustness, tractability, and global optimality are still open problems.

To this end, this thesis investigates robust and optimal planning and control schemes for space-robotics that leverage convex programming. Primarily, four theoretical advances have been made: 1.) Exact reformulation for control of deterministic, nonlinear robotic systems as a convex program; 2.) Sequential, *iteratively feasible* convex relaxations leading to locally optimal solutions using difference of convex functions programming; 3.) Hierarchy of convex relaxations of systems formulated exactly or approximated with polynomial dynamics with global optimality certificates and guaranteed convergence; and 4.) Robust controller synthesis for nonlinear polynomial systems using polynomial optimization in the framework of nonlinear disturbance observers for both matched and mismatched uncertainties.

For applications, the thesis solves four challenging problems for trajectory planning and control during spacecraft proximity operation. First, quadratic programming based trajectory planning methods are formulated for free-floating space robots. Leveraging tools in analytical mechanics and differential geometry, a novel quadratic programming based trajectory planning scheme is developed for task-constrained end-effector motion which minimizes the base attitude disturbance, in addition to obstacle avoidance for both the unactuated base and manipulator. Second, the orbital station-keeping of spacecraft in the framework of the circular restricted three-body problem is solved using polynomial optimization in a receding horizon setting. Third, robust stabilization and tracking of spacecraft attitude motion in the presence of parametric uncertainties and external disturbances is explored in the framework of convex optimization based nonlinear disturbance observer synthesis. And fourth, an iteratively feasible convex programming based approach is proposed for solving optimal spacecraft guidance problems with non-convex constraints such as obstacle avoidance.

Acknowledgements

First and foremost, I would like to thank my advisor Prof. Xiaoli Bai for giving me the freedom to pursue challenging and interesting problems through out the course of my PhD. Thank you for your trust, patience, and guidance especially when I struggled with research. I am grateful for your mentorship. Your discipline and strong work ethics are truly inspirational and I hope I can inculcate these qualities in all of my future endeavors.

I would like to thank my committee members: Prof. Haym Benaroya, Prof. Qingze Zou, and Prof. Ping Lu for taking the time to review my disseration and providing valuable insights and feedback. I would like to thank all the professors who introduced me to the wonderful fields of control theory, optimization, and aerospace systems.

Thanks to all the past and present members of my research group, Liyang Wang, Hongwei Yang, Hao Peng, Tianyu Gao for providing a fun and friendly working enviroment. I enjoyed collaborating on research with you and hope for more such alliances in the future. A very special thanks to Jingren Wang. You have been a tremendous source of inspiration and support. Thanks for enriching my doctoral journey with compassion, love, and humor.

My friends have been extremely helpful and supportive, Shashank Thakkar, Saumil Patel, Birju Vacchani, Ajay Dusane, Shashank Gupta, Anujeema Saikia, Snehil Joshi, Kamalesh Kohli, TK Sriram, Kamakshi Kaul, Parul Bhatnagar, Chris Monaghan. I have enjoyed many 4 AM chats and memes with you guys. Thanks for putting up with my research frustrations!

Lastly, all of this would absolutely not be possible without the unconditional love and support of my parents and sister. The journey as a young student from Delhi to a PhD candidate has been arduous but immensely satisfying and my family has been there will me through both hardships and success.

Dedication

To my mother, Vinod Misra for her unwavering support and love.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

The space environment especially in the Low Earth Orbit is becoming increasingly cluttered every year. The access to space is now no longer limited to government agencies, with private companies, universities actively engaged in building and launching spacecraft. With the increasingly enhanced operational capabilities of relatively cheaper CubeSats, the threat posed from space debris is real and need for debris prevention, and mitigation is urgent. Over the years, many calls have been released by space agencies such as NASA, ESA [1], DARPA [59], UN COPUOS [106] to investigate the possibility of using passive/active debris removal techniques. Overall, it has been estimated that roughly five large objects would be required to be removed in order to counter the growing debris problem [92].

In addition, on-orbit satellite servicing and manipulation is also of interest. Robotic manipulation in space is an attractive technology to handle cooperative [127] and uncooperative operations. Cooperative space-robotic operations have seen tremendous success such as the Shuttle Remote Manipulator System (SRMS), better known as the Canada Arm [44]. Uncooperative space object capture and manipulations introduce new and unique challenges. There is seldom apriori information about the target parameters such as the mass, moment of inertia, and shape. In addition, collision avoidance, and environmental uncertainties such as attitude and orbital disturbances also are important factors to consider during manipulator operations. Real time autonomy is paramount for guidance and control, therefore developing control algorithms with on-board implementation capabilities is required.

In fact, the National Research Council (NRC) identified the top five challenges for space robotic and autonomous systems which include rendezvous, maneuvering, in situ analysis and sample return, hazard avoidance, and object recognition and manipulation [160]. The critical importance of these challenges is made obvious by past space mission failures and accidents. NASA's demonstration of autonomous rendezvous technology (DART) spacecraft [146] was launched in 2005. The mission was designed to rendezvous and perform autonomous close proximity maneuvers near the MUBLCOM satellite [22]. During approach, DART overshot one of its critical waypoints and eventually collided with the target spacecraft at a speed of 1.5 m/s. The key reasons attributed to this mishap was a combination of guidance and computer logic error which incorrectly estimated the amount of propellant left. The Japanese Aerospace Exploration Agency (JAXA)'s Hayabusa mission to asteroid Itokawa also suffered a number of anomalies. Two reaction wheels responsible for attitude control failed before touchdown [173] and the spacecraft also suffered challenges during descent due to fuel leakge [126]. A hopping mini-lander termed MINERVA failed to deploy correctly and was lost. Remarkably, with some timely interventions by ground control, ion engines were used to correct the spacecraft's attitude and conduct a successful sample return [191]. The Japanese Experimental Test Satellite, ETS-VII launched in 1997 demonstrated automated rendezvous, docking and tele-robot operations. The mission consisted of two satellites, a target and a chaser with a robotic arm teleoperated from ground. While the mission was widely successful and achieved significant milestones, the mission was not bereft of anomalies. The attitude control system malfunctioned due to thruster misfires attributed to a number of reasons including noisy gyro data and switching off of the Earth sensor [128]. While the above mentioned space mission anomalies are not exhaustive, it proves that autonomous spacecraft guidance, navigation, and control still has significant challenges ahead. Most importantly, there is an urgent need of real-time, robust, fault tolerant systems with provable safety guarantees.

The aim of this dissertation is to provide solutions towards such problems, particularly leveraging convex optimization. The motivation behind studying convex optimization based approaches comes from the need of real-time optimization which is critical in almost all aspects of space mission design. Convex programs typically have polynomial worst case complexity and upper bounds exist on the number of iterations required to solve them. Furthermore, under certain constraint satisfaction rules, the global optimal solutions can be recovered [25]. Lastly, due to self-dual embedding, convex programs do not require a user supplied initial guess. This especially comes in handy when dealing with systems with multiple constraints, uncertainties requiring on-board complex decision making. Moreover in this work, we will investigate both *guidance*, the process of planning and re-planning of kinodynamic trajectories of rigid and multi-link spacecraft under different environment settings, and *control*, specifically feedback control for stabilization and tracking of spacecraft trajectories.

1.1 Spacecraft proximity operations

Proximity operations typically involve relative spacecraft guidance, navigation, and control near target bodies including uncooperative objects such as debris [108, 109], small Solar system bodies such as asteroids and comets [111, 147, 113] and other cooperative spacecraft such as during docking or formation flying. Autonomous rendezvous and docking broadly includes both far-range and short-range approach with applications such as in-orbit assembly and manipulation and sample collection. For an uncooperative spacecraft manipulation and de-orbiting, Fig. 1.1 illustrates the main steps involved in the operation.

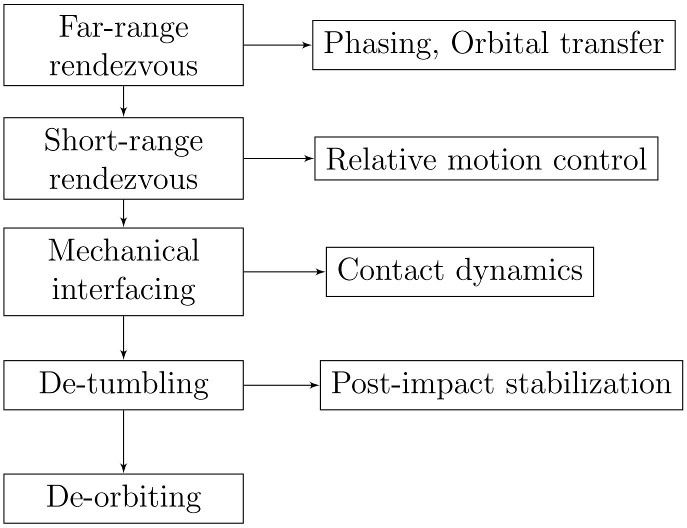


Figure 1.1: Operation flowchart for deorbiting spacecraft

Short-range rendezvous typically involves planning a safe trajectory in proximity of the target. This can be done under different dynamical regimes. Typically, linearized Clohessy-Hill-Wiltshire (CHW) equations of motion provide a convenient means to study relative motion between the two spacecraft [32]. The solutions to the CHW equations are characterized by decoupled inplane cycloidal motion and out-of-plane harmonic motion. Due to its simplicity, it has been widely used in practice, such as for Gemini [27] and Apollo [8] missions. However, the CHW equations assume that the two spacecraft are in circular orbit which can be restrictive. To accommodate for arbitrary eccentricity, Tschauner and Hempel [171] considered the linear spacecraft relative dynamics with true anomaly as the independent variable. While the linearized CHW and Tschauner-Hempel equations are expressed in Cartesian coordinates, alternate formulations also exist expressed in orbital elements [162].

The relative spacecraft equations discussed above only consider translational motion with the spacecraft assumed as a point mass. This assumption can in some cases be insufficient, especially if the target and chaser spacecrafts are sufficiently large and comparable in size. A motivating scenario where the point mass assumption is insufficient is spacecraft proximity operations near small Solar System bodies. This is due to gravitational coupled orbit-attitude dynamics which have significant impact on orbital motion of the spacecraft near small bodies [111, 114]. The effect of this coupling can be characterized by the ratio $\epsilon = \frac{\rho}{r}$, where ρ is the characteristic dimension of the spacecraft and r is the orbital radius which is quite small for proximity operations. The effect of rotational-translational coupling can also be significant during formation flying, rendezvous and docking. In addition, the coupled dynamics becomes important during vision-based relative attitude and position control, where arbitrary feature points on a target vehicle are to be tracked. Segal and Gurfil [154] provide an excellent analysis on the impact of coupling on spacecraft formation flying. They prove that neglecting the relative translation induced by the relative rotation can lead to considerable errors in proximity operations.

A different but related dynamical regime for proximity operations is that of a multi-body space system near a target. Typically, such multi-arm space robotic systems are employed for on-orbit satellite servicing and manipulation of space debris [44]. The space-robotic system typically consists of three major components: the base spacecraft or servicing satellite, an n-degree-of-freedom robot manipulator attached to the chaser or servicing satellite, and the target body. Space robotic operations for on-orbit servicing or de-tumbling occur during or after short-range rendezvous. During the process rendezvous, observation and planning is conducted for acquiring motion and physical properties information about the target. The second phase is to design and track a feasible trajectory of the robot's end-effector to interface/grasp with the target. This phase requires on-board capability to quickly design and re-design guidance laws in presence of uncertainties in real-time. Due to relatively short time-spans involved during robot motion near the target, the effect of orbital motion due to gravity can be neglected. Note that this is different from the rendezvous phase of the spacecraft which can last several thousand seconds and gravity influence must be included. Following end-effector approach to the target, the next phase consists of physical interception in which the robot physically grasps the capture point of the target. This requires maneuvering the space-robotic system to have zero relative velocity with respect to target to ensure a safe and successful capture. Following mechanical interfacing, control torques on the manipulator are applied to bring the entire captured system to rest following which de-orbiting can be performed.

A unifying feature of multi-body space-robotic system, point-mass or rigid body spacecraft motion during proximity operations are path constraints. Collision avoidance is necessary to ensure that the spacecraft trajectory is safe [142]. For a multi-arm robot, it is necessary that the links do not collide with each other or the base in addition to the target. The challenging aspect of collision avoidance stems from it's non-convexity which makes it harder to deal with in an optimal control setting. In addition, it is possible that the chaser spacecraft has constraints on it's attitude, to ensure that payload and on-board sensors point in the right direction. In addition, there can be attitude constraints to mitigate any thruster plume impingement on the spacecraft payload [151]. Furthermore, waypoint constraints are also a common feature where the spacecraft/space-robot end-effector must traverse through given waypoints. Such a strategy is common during spacecraft approach to the International Space Station (ISS) where a number of coasting arcs are designed prior to docking. During close approach, it may also be desirable to have the chaser's sensors in field-of-view of the target body. Due to such constraints, it is desirable to formulate the spacecraft guidance problem as an optimal control problem. The generic guidance problem is represented as

$$
\begin{aligned}
\underset{x(t),u(t),t}{\text{minimize}} \quad & \int_{t_0}^{t_f} \ell(x(t),u(t),t)dt + \ell_f(x_{t_f,t_f}) \\
\text{subject to} \quad & \dot{x} = f(x(t),u(t),t) \\
& g(x(t),u(t)) \leq 0 \\
& x(t) \in \mathcal{X} \\
& u(t) \in \mathcal{U} \\
& x(t_f) \in \mathcal{X}_f \\
& x(t_0) \in \mathcal{X}_0
\end{aligned}
\tag{1.1}
$$

where $f(x(t),u(t),t)$ are the system dynamics which are typically non-linear, $\mathcal{X}$ is the feasible region of the state, $g(x(t),u(t),t)$ are path constraints, $\mathcal{U}$ is the admissible control region, and $\mathcal{X}_0$, $\mathcal{X}_f$ denote the initial and final conditions.

1.2 Contributions and outline

This dissertation seeks to remedy some of the challenges discuss above in space-robotic system guidance and control. Particularly, the emphasis and contributions are on the following areas

- Real-time constrained motion planning of articulated space robots
- Optimal nonlinear receding horizon control for spacecraft control
- Robust optimization based nonlinear feedback control of spacecraft in presence of external disturbance and uncertainties
- Algorithmic and theoretical advances in direct transcription of non-convex optimal control problems.

Chapter 2 describes the optimal spacecraft guidance and control in more detail. The focus of this chapter is on recent advances on convex optimization based real-time control of different space systems including its use in present and future space mission design. In addition, key concepts of convex optimization are discussed with emphasis on sum-of-squares programming and polynomial optimization which forms a key component of this thesis.

Chapter 3 describes the real-time trajectory planning problem for free-floating space robotic manipulators. Particularly, we propose a quadratic programming based planner for the system under a range of constraints including task constraint end-effector motion, joint angle, velocity, and acceleration limits, and obstacle avoidance for both base and manipulator. Benchmark numerical simulations are also provided to compare the performance of the proposed approach with current non-convex planning methods.

Chapter 4 focuses on nonlinear model predictive control (MPC) for spacecraft trajectory tracking in the circular restricted three-body problem framework. Given a desired trajectory such as a Halo or Lissajous orbit, the objective of this work is to use truncated nonlinear dynamics upto second order which is then utilized for synthesis of MPC controller. The quadratic/polynomial spacecraft dynamics enable us to leverage global polynomial optimization to compute globally optimal solutions using moment-sum-of-squares methodology [15]. We compare the proposed polynomial MPC approach with linear and fully nonlinear MPC in terms of tractability and control costs.

Chapter 5 introduces the robust controller synthesis problem for polynomial systems. Nonlinear disturbance observer based control has seen significant success in motion control. However, the applicability of this approach is severely limited for complex nonlinear systems, where the disturbance-to-state mapping is also a function of the state. For polynomial systems, we introduce an algorithmic approach for synthesizing nonlinear disturbance observers by posing the problem as a matrix polynomial optimization which can be solved using semidefinite programming. In addition, numerical stability verification is provided using

Lyapunov analysis for systems with both matched and mismatched disturbances.

In Chapter 6, we extend the robust controller design to relative spacecraft attitude tracking. In a scenario where disturbance torques act on both the target and chaser spacecraft, we construct a disturbance observer using the sum-of-squares methodology. In addition, we provide comparisons with existing approaches.

Chapter 7 focuses on algorithmic improvements to existing direct transcription approaches to solving optimal control problems. Current approaches either use interior point methods to solve the resulting static optimization problems or leverage convex optimization to design faster iterative solutions. A common concern with both these approaches is that the intermediate iterations may not be a feasible or optimal solution to the problem. We leverage difference of convex function (DC) programming to decompose non-convex dynamics and provide an iteratively feasible convex approach to solving optimal control problems with linear dynamics and non-convex path constraints.

1.3 Relevant publications

Below is a list of published, accepted and in-preparation research pertaining to the subject matter of this dissertation.

- **Journal papers**

  1. **Misra, G.**, Bai, X. “Robust DOB based Control for Spacecraft Attitude Tracking using Sum-of-Squares Programming.” *Under review.*
  2. **Misra, G.**, Bai, X. “Output-feedback Stochastic Model Predictive Control for Glideslope Control during Aircraft Carrier Landing.” *Journal of Guidance, Control, and Dynamics*, (2019) *In press.*
  3. **Misra, G.**, Bai, X. “Task-Constrained Trajectory Planning of Space-Robotic Systems using Convex Optimization.” *Journal of Guidance,*

Control, and Dynamics, Vol. 40, No. 11 (2017), pp. 2857-2870.

4. **Misra, G.**, Bai, X. "Optimal Path Planning of Free-flying Space Manipulators using Sequential Convex Programming", *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 11 (2017), pp. 3026-3033.

- **Conference proceedings**

1. **Misra, G.**, Bai, X. 'Iteratively Feasible Optimal Spacecraft Guidance with Non-convex Path Constraints using Convex Optimization ", *AIAA Guidance, Navigation, and Control Conference, Florida, 2020. Under review*

2. **Misra, G.**, Bai, X. "Nonlinear Disturbance Observer based Control for Polynomial Systems with Mismatched Uncertainties using Sum-of-Squares Programming", *IEEE American Control Conference (ACC), Philadelphia, 2019.*

3. **Misra, G.**, Gao, T., and Bai, X. "Modeling and Simulation of UAV Carrier Landings", *AIAA Modeling and Simulation Technologies Conference, San Diego, 2019.*

4. **Misra, G.**, Bai, X. "Stochastic Model Predictive Control for Gust Alleviation during Aircraft Carrier Landing", *IEEE American Control Conference (ACC), Milwaukee, 2018.*

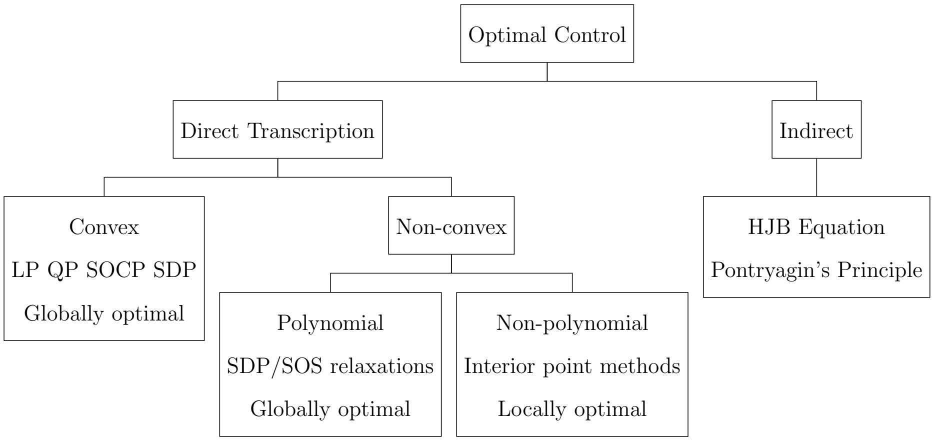
5. **Misra, G.**, Peng, H, and Bai, X. "Halo Orbit Station-keeping using Nonlinear MPC and Polynomial Optimization", *28th AIAA/AAS Spaceflight Mechanics Meeting, Kissimmee, FL, 2018.*

Chapter 2

Convex optimization based guidance and control

Aerospace guidance and control problems have typically been approached and solved in the framework of nonlinear optimal control. Existing approaches to solve optimal control problems can be broadly divided into two classes: indirect methods and direct methods. Indirect methods are based on variational calculus. The Pontryagin's minimum principle provides necessary conditions for optimality. Once the necessary conditions have been derived, the problem is reduced to a two-point boundary value problem amenable to methods such as multiple shooting. In order to alleviate the challenges posed by good initial guesses to solve boundary value problems, homotopy methods have been introduced. On the other hand, using time-marching or pseudo-spectral methods, direct methods aim at assuming a structure for state and control trajectories, such as piece-wise constant, or cubic. Thus, the continuous optimal control problem can be converted to a static optimization problem. Depending on the convexity of the resulting problem, different methods can be used to obtain either locally optimal or globally optimal solutions. In order to accommodate robustness and feedback into the system, such problems are solved in a receding horizon fashion, where a static optimization problem of fixed prediction horizon is solved at each step to obtain the necessary control inputs.

Due to fast interior point solvers available to solve convex programming problems with guarantees on time complexity, significant research has been devoted to investigate its use in solving highly complex, nonlinear aerospace control problems. In Table 2, some of the recent applications of convex optimization in guidance and control are provided. This includes both exact reformulation of the problem termed as lossless convexification and relaxed iterative approaches to solving optimal control problems in the framework of convex programming. THe flowchart below describes the different techniques used to solve optimal control problems.



Application	**Reference**
Planetary powered landing	[2], [3], [20], [167]
Spacecraft rendezvous	[101], [96]
Planetary entry	[97], [180]
Asteroid landing	[188], [136]
Fuel-optimal rocket landing	[95]
Space robot trajectory planning	[109], [108],[176]
Attitude control	[169],[28]
Missile guidance	[98]

Table 2.1: Compilation of convex optimization approaches proposed for different aerospace guidance and control problems

2.1 Convex optimization

This section gives an overview on convex optimization, a class of nonlinear optimization where the objective function and inequality constraints are convex and the equality constraints are linear. While several types of optimization problems span convex programming, here we discuss quadratic programming and semidefinite programming since they are extensively used in the next chapters.

2.1.1 Quadratic programming

A quadratic optimization problem can be defined as

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \frac{1}{2}x^{\mathrm{T}}Px + qx + r \\ \text{subject to} \quad & Ax = b \end{aligned} \tag{2.1}$$

where $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $q \in \mathbb{R}^{1\times n}$, $r \in \mathbb{R}$. If $P \in \mathbb{S}^n_+$, is positive semidefinite, then the problem in Eq. 2.2 is defined as convex quadratic program (QP) with a polyhedral feasible set. A variety of algorithms exist to solve QPs, including active-set, augmented Lagrangian, interior-point, and conjugate gradient methods. In case, the constraints are also quadratic, that is, of the form $\frac{1}{2}x^{\mathrm{T}}M_ix + n_i^{\mathrm{T}}x + l$, $i = 1, 2, ..., p$, the problem is termed as a quadratically-constrained quadratic program. For $M_i \in \mathbb{S}^n_{++}$, the feasible set is an intersection p ellipsoids and an affine set.

2.1.2 Semidefinite programming

Semi-definite programming (SDP) is one of the most broadest classes of convex optimization methods. It encompasses both linear programming and second-order cone programming. It can be seen as a natural extension of linear programming where the vector variables are replaced by matrices constrained to be positive semi-definite. Given matrices $C,\ A_1, ...A_m \in \mathbb{S}_n$, the primal form of SDPs is defined as

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & \text{tr}(CX) \\ \text{subject to} \quad & \text{tr}(A_jX) = b_j, j = 1, 2, ..., m \\ & X \succeq 0 \end{aligned} \tag{2.2}$$

where tr defines the trace of a matrix. The dual form for the SDP is expressed as

$$\begin{aligned} \underset{y}{\text{minimize}} \quad & b^{\mathrm{T}}y \\ \text{subject to} \quad & \sum_{j=1}^{m} y_jA_j - C \succeq 0 \end{aligned} \tag{2.3}$$

In the dual form, the linear matrix inequality (LMI) form of SDP is recovered. The feasibility set of an SDP is a spectahedron. For strong duality, that is, the dual and primal solutions are equivalent, the slater's condition is required to be satisfied. Note that however, unline linear programs, in general, the primal and dual SDP solutions may not be equal and only weak duality holds.

2.2 Sum-of-squares programming

The motivation behind sum-of-squares (SOS) programming stems from a basic question ubiquitous in operations research, control, and other areas: Given a polynomial $f(x_1, ..., x_n)$, for all $x \in \mathbb{R}^n$, is the function non-negative? It turns out that assessing this question is NP-hard for a function with degree ≥ 4. However, a sufficient and tractable condition to ascertain this is to check if the function is positive over the feasible set.

Definition 1. *A polynomial $f \in \mathcal{P}_{n,2d}$, that is, a multivariate function of n variables and maximum degree of $2d$ is called a sum-of-squares if*

$$f(x) = \sum_{i=1}^{k} p_i^2(x) \tag{2.4}$$

The SOS n-variate polynomials are denoted as Σ_n.

Theorem 1. *A polynomial $f \in \Sigma_{n,2d}$ if and only if*

$$p(x) = B_{n,d}(x)^T M B_{n,d}(x) \tag{2.5}$$

where $B_{n,d}(x)$ is any fixed basis for $\mathcal{P}_{n,d}$ and $M \succeq 0$.

Checking positivity of a polynomial $p(x)$ is akin to solving an SDP. Consider the example: $f(x) = 2x^4 + 5y^4 - 4x^2y^2$. Let the basis be $[x^2, y^2, xy]$. To find an SOS decomposition, a feasibility SDP problem can be solved. Denote $M \in \mathbb{R}^{3\times 3}$ where $M_{i,j}$ denotes each entry of matrix. This is expressed as

$$\begin{aligned} \text{maximize} \quad & - \\ \text{subject to} \quad & M_{11} = 2 \\ & M_{22} = 5 \\ & M_{33} + 2M_{12} = -4 \\ & M_{13} = M_{23} = 0 \\ & M \succeq 0 \end{aligned} \tag{2.6}$$

Note that however, a nonnegative polynomial is not necessarily SOS. For example, the Motzkin polynomial is non-negative but does not yield an SOS decomposition [132].

SOS programming finds applications in both unconstrained and constrained polynomial optimization. Consider the unconstrained case

$$\begin{aligned} \underset{x\in\mathbb{R}^n}{\text{maximize}} \quad & p(x) \\ \text{subject to} \quad & p(x) \in \mathcal{P}_{n,d} \end{aligned} \tag{2.7}$$

Using standard lifting techniques, we can define the optimal solution as ρ^*. Therefore, we have

$$\rho^* = \sup\{\rho : p(x) - \rho \geq 0\} \geq \sup\{\rho : p(x) - \rho \in \Sigma_{n,d}\} \tag{2.8}$$

From Eq. 2.8, the lower bound on the optimal value of the unconstrained polynomial optimization can be computed by solving an SDP [132]. Next, consider the general constrained polynomial optimization problem

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & p(x) \\ \text{subject to} \quad & g_i(x) \geq 0, i = 1, 2, ..., k. \\ & p(x), g_i(x) \in \mathcal{P}_{n,d} \end{aligned} \tag{2.9}$$

The set defined by $\{x \in \mathbb{R}^n | g_i(x) \geq 0 \forall i = 1, 2, ..., k\}$ is called a basic semi-algebraic set. Examples of basic semi-algebraic sets include the non-negative orthant $\mathbb{R}^n_+$ and the cone of positive semidefinite matrices. Similar to the unconstrained case, Eq. 2.9 can also be solved in the framework of SOS programming. This will be discussed in detail in Chapter 4.

Chapter 3

Real-time motion planning of space-robotic systems

3.1 Close-proximity trajectory planning

Trajectory planning techniques for multi-link space robots (i.e. a manipulator atop a base spacecraft) have received considerable attention over the last few decades [119, 39, 35, 130, 129, 76, 121], most of which fall into two categories: optimization techniques and sampling based planning. Due to the coupled dynamics between the base and the manipulator, trajectory planning schemes often employ optimization algorithms with the objective to minimize such coupling effects. In real applications, objectives can be divided into two types, namely, base motion (attitude) minimization and base reaction minimization. It is worth noting that the two objectives are in fact contradictory, i.e. minimizing the base attitude disturbance can result in undesirable base reactions and vice versa [187].

Variational techniques are proposed for optimal path planning of free-floating space manipulators while accounting for the holonomic and nonholonomic constraints in [5]. The resulting formulation is given as a system of differential-algebraic equations. Sequential Quadratic Programming methods for point-to-point motion planning problems for a free-floating space manipulator are employed in [153]. Redundancy resolution methods based on the reaction null space (RNS) approach are used to obtain minimum attitude disturbance trajectories in [123, 124, 179]. A motion planning technique for grasping and stabilization of an uncooperative space object is proposed wherein the planning task is formulated as a nonlinear optimization problem based on direct single shooting with obstacle avoidance in [75]. Pre- and post-capture motion planning to stabilize an object with uncertain dynamics is discussed in [4], where the optimal guidance law is framed directly in the end-effector task space. Using double integrator dynamics used to determine the motion of the end-effector, an optimal control law is found by applying Pontryagin's principle to minimize the total time of operation. A constrained least squares based redundancy resolution approach is used to minimize the base reactions with end-effector trajectory tracking constraints in [33], however, obstacle avoidance is not considered in the study. Particle swarm optimization (PSO) based planning combined with differential evolution is used in [94] to generate smooth paths for redundant free-floating space manipulators. Based on either nonlinear programming methods or evolutionary optimization methods such as PSO, most of these optimization techniques are slow, susceptible to initialization, and unsuitable for real-time implementation.

The key idea behind sampling based methods such as rapidly exploring random trees (RRT) [84] is to construct a configuration ($\mathcal{C}$) space roadmap by randomly placing configurations in the $\mathcal{C}$ space and connecting neighboring portions. A sampling-based A* algorithm is proposed for planning trajectories for a 7 Degree of Freedom (DOF) space manipulator with numerical and experimental validation in [135]. Another sampling-based approach for reaction-less motion of space robots has been recently proposed in [64], where the technique uses RRT based path planners to plan motions starting from rest for a dual arm non-redundant robot. The computation time required by the RRT planner to find a suitable path was reported to be approximately 81 s, which is not feasible for real-time applications. Although sampling based methods are efficient and probabilistically complete (i.e. if a feasible trajectory exists, the planner can compute it as time goes to infinity) for high dimensional $\mathcal{C}$ spaces, the resulting trajectory can have convergence issues especially for manipulators with end-effector task constraints and is often jerky requiring smoothing techniques [199], and therefore, not applicable for online autonomous planning.

In this study, the trajectory planning problem is posed as a convex quadratic optimization problem with the goal to solve it in real-time. Convex quadratic programs (QP) are desirable from a computational standpoint since such problems can be solved in polynomial time with the worst case complexity for a convex QP with linear constraints being $\mathcal{O}(n^3L)$ where n is the size of the decision variable and L is the program input size [50]. In general, convex QP based planning methods are faster than techniques leveraging disjunctive programming [21, 37] (based on mixed integer linear or quadratic programs) which are NP-complete and have exponential worst case complexity. Recent advances in numerical techniques such as interior point methods allow solving convex optimization problems quickly and efficiently [25], making these methods desirable for on-board implementation without the use of look-up tables. Convex optimization based methods have been successfully demonstrated for UAV trajectory generation [11], spacecraft swarm control [116], constrained attitude control [89], planetary [20] and asteroid [188] landing, rendezvous and proximity operations [102], and robot motion planning [152]. In fact, custom real-time Interior-Point Method solvers have been proposed and experimentally validated on radiation hardened flight processors for pin-point planetary landing [40, 148]. Previous studies on convex optimization applications in robotics have primarily looked at path tracking which assumes that a high level planner provides an obstacle free geometric path [196, 174]. In contrast, here we focus on trajectory planning of redundant manipulators on a free-floating spacecraft platform (i.e. without any translational and rotational actuation on the spacecraft) with the manipulator's end-effector subjected to task constraints. Although quadratic programming based approaches have been proposed for fixed-base [197] and also wheeled- and legged-robotic systems [155], we have found no studies using convex optimization techniques for free-floating mobile base robotic platforms. Furthermore, the motion of free-floating space-robotic systems differs significantly from fixed-base ground robots because of the micro-gravity environment, the nonlinear dynamic coupling between the base and the manipulator, and the nonholonomic constraints due to the conservation of angular momentum [109]. Furthermore, planning and control for space robots becomes even more challenging and complicated in the presence of initial non-zero momentum [120] as the resulting system is affine with a drift term.

The main contributions of this approach lie in four aspects

- A convex quadratic programming based approach is developed in robot joint space for path planning of kinematically redundant free-floating space-robotic systems with non-zero initial momentum. It is through this novel formulation in terms of QP that the real-time planning becomes possible.

- This technique can incorporate goals such as base attitude minimization, physical constraints including joint angle, joint velocity, and acceleration bounds, and obstacle avoidance. Given a desired trajectory of the end-effector (which evolves on the non-Euclidean space SE(3)), we plan kinematically optimal paths in the joint space to achieve the desired primary (e.g. end-effector tracking) and secondary tasks (e.g. obstacle avoidance). Two primary task cases are discussed, when the end-effector is required to track a given trajectory over the given time horizon and when the end-effector is constrained to move through a finite set of way-points during the operation. Additionally, the task formulation is carried out directly on the Lie group of rigid body motions SE(3) to avoid algorithmic singularity issues.

- Only forward kinematics are employed in the formulation, thereby avoiding the explicit calculation of the pseudo-inverse of the Generalized Jacobian

Matrix (GJM) which can lead to dynamic singularities.

- A novel obstacle avoidance scheme is also proposed for the free-floating base. The obstacle avoidance formulation is based on the dynamical coupling between the base and the manipulator due to the conservation of total system momentum.

3.1.1 Mathematical Preliminaries

Consider a kinematically redundant n-dof revolute jointed, free-floating space manipulator under no external forces and no base actuation as shown in Fig. 3.1. The joints are assumed to be rigid with no joint friction. Let $\theta \in \mathbb{R}^n$ be the joint angles for the manipulator. Let the position vector of the base spacecraft centroid with respect to an inertial frame, denoted as $\sum_I$ in Fig. 3.1 be given as $r_b \in \mathbb{R}^3$ and the attitude of the base be given by $Q_b \in \mathrm{SO}(3)$, where $\mathrm{SO}(3) = \{Q = \mathbb{R}^{3\times 3} : Q^{\mathrm{T}} Q = \mathrm{E},\ \det(Q) = 1\}$ is the Lie group of rotations of a rigid body. The kinematics of the base spacecraft is given as

$$\dot{Q}_b = \omega_b^{\times} Q_b \tag{3.1}$$

$$\dot{r}_b = v_b \tag{3.2}$$

where $v_b, \omega_b \in \mathbb{R}^3$ are the linear and angular velocities of the base spacecraft in the inertial frame, respectively, and $(.)^{\times} : \mathbb{R}^3 \to \mathfrak{so}(3) \subset \mathbb{R}^{3\times 3}$ denotes the skew-symmetric operator which is the vector space isomorphism between $\mathbb{R}^3$ and $\mathfrak{so}(3)$, with $\mathfrak{so}(3)$ being the Lie algebra of SO(3). Let the position vector of the link centroid be $r_i \in \mathbb{R}^3$ with respect to the inertial frame and $p_i \in \mathbb{R}^3$ with respect to the base expressed in the inertial frame.

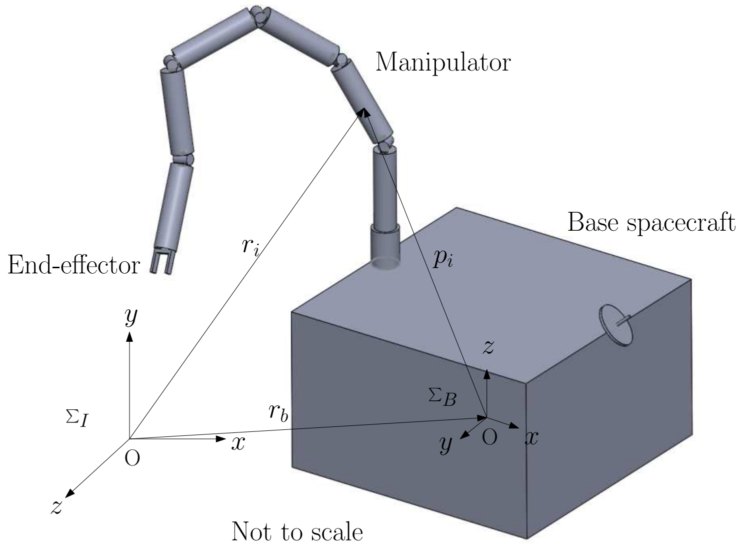


Figure 3.1: Spatial model for a free-floating space manipulator. $\sum_I$ and $\sum_B$ denote the inertial and base spacecraft body frame, respectively.

We have

$$r_i = r_b + p_i \tag{3.3}$$

The linear and angular velocity of the link i in the inertial frame is

$$\begin{aligned} v_i &= v_b + \omega_b^{\times} p_i + \nu_i \\ \Omega_i &= \omega_b + \omega_i \end{aligned} \tag{3.4}$$

where $\nu_i, \omega_i \in \mathbb{R}^3$ are the linear and angular velocity of the link relative to the base spacecraft and expressed in the inertial frame. The total linear and angular momentum of the space-robotic system is given as [186]

$$\sum_{i=0}^{n} m_i v_i = \mathrm{P} \tag{3.5}$$

$$\sum_{i=0}^{n} I_i \Omega_i + m_i r_i^{\times} v_i = \mathrm{H} \tag{3.6}$$

where $i = 0$ denotes the base spacecraft, $I_i \in \mathbb{R}^{3\times 3}$ denotes the inertia matrix of body i in the inertial frame, $\mathrm{P}, \mathrm{H} \in \mathbb{R}^3$ denote the linear and angular momentum of the system. Furthermore, we observe that Eq. (3.5) is an integrable Pfaffian constraint. Integrating the constraint, we find the mass centroid of the entire system either remains stationary or translates with a constant velocity depending on the magnitude of the linear momentum. Contrary to Eq. (3.5), Eq. (3.6) cannot be directly integrated analytically and hence is a nonholonomic constraint. The forward kinematics of link i at velocity level is expressed as

$$\xi_i = \begin{bmatrix} \nu_i \\ \omega_i \end{bmatrix} = \begin{bmatrix} \mathrm{J}_{\mathrm{Ti}} \\ \mathrm{J}_{\mathrm{Ri}} \end{bmatrix} \dot{\theta} \tag{3.7}$$

where $\xi_i \in \mathbb{R}^6$ includes both the linear and the angular velocity and $\mathrm{J}_{\mathrm{Ti}}, \mathrm{J}_{\mathrm{Ri}} \in \mathbb{R}^{3\times n}$ denote the translational and rotational Jacobian matrices of link i, respectively [158]. Using Eqs. (3.5) and (3.7), we get

$$\begin{bmatrix} \mathrm{G}_v\mathrm{E} & \mathrm{G}_{v\Omega} \\ \mathrm{G}_{v\Omega}^{\mathrm{T}} & \mathrm{G}_{\Omega} \end{bmatrix} \begin{bmatrix} v_b \\ \omega_b \end{bmatrix} + \begin{bmatrix} \mathrm{G}_{v\theta} \\ \mathrm{G}_{\Omega\theta} \end{bmatrix} \dot{\theta} + \begin{bmatrix} \mathrm{O}_{3\times 1} \\ r_b^{\times}\mathrm{P} \end{bmatrix} = \begin{bmatrix} \mathrm{P} \\ \mathrm{H} \end{bmatrix}, \tag{3.8}$$

where $\mathrm{O}_{3\times 1}$ represents a three-dimensional null vector and,

$$\mathrm{G}_v = \sum_{i=0}^{n} m_i \in \mathbb{R} \tag{3.9}$$

$$\mathrm{G}_{v\Omega} = -\sum_{i=0}^{n} m_i p_i^{\times} \in \mathbb{R}^{3\times 3} \tag{3.10}$$

$$\mathrm{G}_{\Omega} = I_b + \sum_{i=1}^{n} I_i + m_i (p_i^{\times})^{\mathrm{T}} p_i^{\times} \in \mathbb{R}^{3\times 3} \tag{3.11}$$

$$\mathrm{G}_{v\theta} = \sum_{i=0}^{n} m_i \mathrm{J}_{\mathrm{Ti}} \in \mathbb{R}^{3\times n} \tag{3.12}$$

$$\mathrm{G}_{\Omega\theta} = \sum_{i=1}^{n} I_i \mathrm{J}_{\mathrm{Ri}} + m_i p_i^{\times} \mathrm{J}_{\mathrm{Ti}} \in \mathbb{R}^{3\times n} \tag{3.13}$$

where $\mathrm{E} \in \mathbb{R}^{3\times3}$ denotes an identity matrix and $I_b \in \mathbb{R}^{3\times3}$ denotes the inertia matrix of the base spacecraft. Under the assumption of a free-floating system which involves no internal actuation on the base spacecraft and no external forces, conservation of momentum holds. A majority of the previous literature on space-robotic systems have assumed the initial momentum to be zero to simplify analysis. However, during on-orbit operations, momentum can accumulate in the system. In addition, some detumbling approaches such as the bias momentum method [190] propose using initial stored momentum to facilitate post-capture control of a target spacecraft. Therefore, we take a more general approach in this paper. Let $\mathrm{P}_0, \mathrm{H}_0$, which are not necessarily zero be the initial linear and angular momentum, respectively. We can then derive relations between base velocities, joint rates and the initial momentum.

$$\mathrm{G}_v v_b + \mathrm{G}_{v\Omega}\omega_b + \mathrm{G}_{v\theta}\dot{\theta} = \mathrm{P}_0 \tag{3.14}$$

$$\mathrm{G}_{v\Omega}^{\mathrm{T}} v_b + \mathrm{G}_{\Omega}\omega_b + \mathrm{G}_{\Omega\theta}\dot{\theta} + r_b \times \mathrm{P}_0 = \mathrm{H}_0 \tag{3.15}$$

Using Eq. (3.14), the linear velocity of the base spacecraft can be expressed as

$$v_b = \frac{\mathrm{P}_0 - \mathrm{G}_{v\Omega}\omega_b - \mathrm{G}_{v\theta}\dot{\theta}}{\mathrm{G}_v} \tag{3.16}$$

Substituting Eq. (3.16) into Eq. (3.15), we get

$$\frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}}{\mathrm{G}_v}\left(\mathrm{P}_0 - \mathrm{G}_{v\Omega}\omega_b - \mathrm{G}_{v\theta}\dot{\theta}\right) + \mathrm{G}_{\Omega}\omega_b + \mathrm{G}_{\Omega\theta}\dot{\theta} + r_b^{\times}\mathrm{P}_0 = \mathrm{H}_0 \tag{3.17}$$

Rearranging Eq. (3.17), we obtain

$$\left(\frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}}{\mathrm{G}_v} + r_b^{\times}\right)\mathrm{P}_0 + \left(\mathrm{G}_{\Omega} - \frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}\mathrm{G}_{v\Omega}}{\mathrm{G}_v}\right)\omega_b + \left(\mathrm{G}_{\Omega\theta} - \frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}\mathrm{G}_{v\theta}}{\mathrm{G}_v}\right)\dot{\theta} = \mathrm{H}_0 \tag{3.18}$$

Therefore, the base angular velocity is represented as a function of the joint rates and momentum as

$$\omega_b = \big(\mathrm{G}_\Omega - \frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}\mathrm{G}_{v\Omega}}{\mathrm{G}_v}\big)^{-1}\Big(\mathrm{H}_0 - \big(\frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}}{\mathrm{G}_v} + r_b^{\times}\big)\mathrm{P}_0 + \big(\frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}\mathrm{G}_{v\theta}}{\mathrm{G}_v} - \mathrm{G}_{\Omega\theta}\big)\dot{\theta}\Big) \tag{3.19}$$

Using auxiliary variables, this equation can be written as

$$\omega_b = \hat{\mathrm{J}}_\omega\dot{\theta} + \mathrm{G}_1\mathrm{H}_0 + \mathrm{G}_2\mathrm{P}_0 \tag{3.20}$$

where

$$\begin{aligned}
\mathrm{G}_1 &= \big(\mathrm{G}_\Omega - \frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}\mathrm{G}_{v\Omega}}{\mathrm{G}_v}\big)^{-1} \\
\hat{\mathrm{J}}_\omega &= \mathrm{G}_1\big(\frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}\mathrm{G}_{v\theta}}{\mathrm{G}_v} - \mathrm{G}_{\Omega\theta}\big) \\
\mathrm{G}_2 &= -\mathrm{G}_1\big(\frac{\mathrm{G}_{v\Omega}^{\mathrm{T}}}{\mathrm{G}_v} + r_b^{\times}\big)
\end{aligned} \tag{3.21}$$

Finally, using Eqs. (3.16) and (3.20), the linear velocity for the base spacecraft is expressed as

$$v_b = \hat{\mathrm{J}}_v\dot{\theta} + \mathrm{G}_3\mathrm{H}_0 + \mathrm{G}_4\mathrm{P}_0 \tag{3.22}$$

where

$$\begin{aligned}
\hat{\mathrm{J}}_v &= -\Big(\frac{\mathrm{G}_{v\Omega}\hat{\mathrm{J}}_\omega + \mathrm{G}_{v\theta}}{\mathrm{G}_v}\Big) \\
\mathrm{G}_3 &= -\frac{\mathrm{G}_{v\Omega}}{\mathrm{G}_v}\mathrm{G}_1 \\
\mathrm{G}_4 &= \frac{\mathrm{E} - \mathrm{G}_{v\Omega}\mathrm{G}_2}{\mathrm{G}_v}
\end{aligned} \tag{3.23}$$

Next, we look at the end-effector kinematics. Let $g_e \in \mathrm{SE}(3)$ be the end-effector pose in the inertial frame. The state of the end-effector is described as

$$g_e = \begin{bmatrix} Q_e & r_e \\ 0 & 1 \end{bmatrix} = \mathcal{F}(\theta, r_b, Q_b) \tag{3.24}$$

where $Q_e \in \mathrm{SO}(3)$ is the attitude of the end-effector; $r_e \in \mathbb{R}^3$ is the position vector of the end-effector in the inertial frame; and $\mathcal{F} : \mathrm{SE}(3) \times \mathbb{R}^n \to \mathrm{SE}(3)$, is the differentiable nonlinear mapping from the joint space to the task space of the manipulator, commonly known as the forward kinematics map. The end-effector kinematics is represented as

$$\dot{g}_e = g_e(\mathrm{N}_e\xi_e)^\vee \tag{3.25}$$

where

$$(\xi_e)^\vee = \begin{bmatrix} (\omega_e)^\times & \nu_e \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3), \quad \mathrm{N}_e = \begin{bmatrix} Q_e & 0 \\ 0 & Q_e \end{bmatrix} \tag{3.26}$$

where $\xi_e = \begin{bmatrix} \nu_e & \omega_e \end{bmatrix}^\mathrm{T} \in \mathbb{R}^6$ is the end-effector velocity in the inertial frame and N_e is the transformation matrix from body to inertial frame. The end-effector velocity can be mapped to alternate coordinate frames using the adjoint map, $\mathrm{Ad}_g : \mathfrak{se}(3) \to \mathfrak{se}(3)$, given by [118]

$$\mathrm{Ad}_{g_e} = \begin{bmatrix} Q_e & r_e^\times Q_e \\ 0 & Q_e \end{bmatrix} \in \mathbb{R}^{6\times 6}, \text{ s.t. } \mathrm{Ad}_{g_e} X^\wedge = \left(g_e X g_e^{-1}\right)^\wedge \tag{3.27}$$

where $X \in \mathfrak{se}(3)$, $(\cdot)^\wedge : \mathfrak{se}(3) \to \mathbb{R}^6$ is the inverse of the vector space isomorphism $(\cdot)^\vee : \mathbb{R}^6 \to \mathfrak{se}(3)$. The velocity (twist) of the end-effector, $\xi_e \in \mathbb{R}^6$ is expressed in terms of base and joint variables as

$$\xi_e = \mathrm{J}_s \begin{bmatrix} \upsilon_b \\ \omega_b \end{bmatrix} + \mathrm{J}_m \dot{\theta} \tag{3.28}$$

where $\mathrm{J}_m = \begin{bmatrix} \mathrm{J_T} & \mathrm{J_R} \end{bmatrix}^\mathrm{T} \in \mathbb{R}^{6\times n}$ is similar to the conventional Jacobian for a ground manipulator [158] and the spacecraft Jacobian matrix is expressed as

$$\mathrm{J}_s = \begin{bmatrix} \mathrm{E} & -(r_e - r_b)^{\times} \\ 0 & \mathrm{E} \end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{3.29}$$

where $r_e \in \mathbb{R}^3$ is the position vector of the end-effector in the inertial frame. Using Eqs. (3.20) and (3.22), the base velocity can be eliminated to obtain the relationship between the end-effector velocity and joint velocities. This is given as

$$\xi_e = \mathrm{J}_g \dot{\theta} + \mathrm{J}_s \begin{bmatrix} G_4 & G_3 \\ G_2 & G_1 \end{bmatrix} \begin{bmatrix} \mathrm{P}_0 \\ \mathrm{H}_0 \end{bmatrix} \tag{3.30}$$

where $\mathrm{J}_g = \mathrm{J}_s \begin{bmatrix} \hat{\mathrm{J}}_v \\ \hat{\mathrm{J}}_\omega \end{bmatrix} + \mathrm{J}_m \in \mathbb{R}^{6\times n}$ is the Generalized Jacobian Matrix (GJM) [157].

It is important to recognize that even in the presence of non-zero initial momentum, both the base spacecraft and the end-effector velocities are affine functions of the joint velocities as shown in Eqs. (3.22) and (3.30). This property is crucial in formulating the trajectory planning as a convex optimization problem, which is presented in the next section.

3.2 Problem Formulation

Given a desired end-effector trajectory/task in the inertial space, the planning problem considered here requires solving for a feasible joint path that satisfies the task-compliance constraints, physical constraints such as joint limits, and obstacle avoidance, while optimizing a given cost function. At the kinematic level, the manipulator motion can be expressed in the joint space by simple double integrator dynamics where the joint acceleration is considered as the control input. With a predefined cost function $\Gamma(\theta(t)$, the trajectory planning problem in continuous time denoted as Problem-1 is stated as

$$\begin{aligned}
\text{Problem-1} \quad \underset{\theta(t),\dot{\theta},u(t)}{\text{minimize}} \quad & \Gamma\big(\theta(t),\dot{\theta}(t),u(t)\big) \\
\text{subject to} \quad & \ddot{\theta}(t) = u(t) \\
& m_i\big(\theta(t),\dot{\theta}(t)\big) \le 0 \ , \forall t \in [0, t_f] \\
& g_e^d(t) = \mathcal{F}\big(\theta(t), r_b, Q_b\big) \\
& \theta(t=0) = \theta_0, \dot{\theta}(t=0) = \dot{\theta}_0 \\
& r_b(t=0) = r_0, Q_b = Q_0, \xi_b(t=0) = \xi_b^0
\end{aligned} \tag{3.31}$$

where, $\Gamma\big(\theta(t),\dot{\theta}(t),u(t)\big) : \mathbb{R}^{3n} \to \mathbb{R}$ is the cost function, $u(t)$ is the control input, t_f is the predefined final time, $m_i(\theta(t),\dot{\theta}(t)) \le 0$ (interpreted as element wise inequality) include collision avoidance constraints and limitations on joint angles, velocities, and controls. $g_e^d(t) : [0 \ \ t] \to \text{SE}(3)$ denotes the desired end-effector trajectory, $\mathcal{F}\big(\theta(t), r_b, Q_b\big) : \mathbb{R}^n \times \text{SE}(3) \to \text{SE}(3)$ denotes the forward kinematics map, $\theta_0, \dot{\theta}_0$ are the joint angles and velocities at the initial time, and r_0, Q_0, and ξ_b^0 are the initial position, attitude, and velocity for the base spacecraft. The above planning problem is generally a non-linear, non-convex optimization problem due to the nonlinear equality constraints pertaining to the end-effector pose trajectory and the obstacle avoidance constraints.

Discrete Time Formulation

Let Δt denote the time step and N be the total number of nodes, leading to the final time $t_f = N\Delta t$. We propose the trajectory planning at the joint velocity level, such that the discrete-time variables are given as $\dot{\theta}[k] \triangleq \dot{\theta}(k\Delta t)$. The joint accelerations given as $\ddot{\theta}[k] \in \mathbb{R}^n$ are considered as the control input, i.e. $u[k] \triangleq \ddot{\theta}(k\Delta t)$. Furthermore, the control input $u[k]$ is considered to be of a zero-order hold type [19] such that

$$u(t) = u[k],\ \forall t \in [k\Delta t, (k+1)\Delta t] \tag{3.32}$$

The discrete state equation using a trapezoidal discretization is written as

$$\dot{\theta}[k] = \dot{\theta}[k-1] + \frac{\Delta t}{2}(u[k-1] + u[k]) \tag{3.33}$$

A critical aspect during manipulator operations is to ensure that the rotational disturbance on the base is minimized. This requirement stems from communication, power constraints as well as obstacle avoidance. Therefore, the objective of the planning problem is chosen to minimize the base attitude disturbance and generate feasible smooth paths in the joint space. The base attitude described by Q_b is related to the base angular velocity through Eq. (3.1), therefore, a suitable objective function is the 2-norm of the angular velocity. Using Eq. (3.20), we get

$$\|\omega_b\|_2^2 = \left\|\hat{\mathrm{J}}_\omega \dot{\theta} + b\right\|_2^2 \tag{3.34}$$

where $b = \mathrm{G}_1\mathrm{H}_0 + \mathrm{G}_2\mathrm{P}_0$.

The cost function is chosen to minimize the base attitude and the control effort, represented as

$$\Gamma(\dot{\theta}, u) = \left\|\hat{\mathrm{J}}_\omega \dot{\theta} + b\right\|_2^2 + u^{\mathrm{T}}\mathrm{S}u \tag{3.35}$$

where $\mathrm{S} \in \mathbb{R}^{n\times n}$ is a positive definite weight matrix that can be chosen based on the relative priority of base angular motion and optimal joint motions for the optimization problem. Importantly, notice since we use the joint velocities as the design variable for the optimization problem, the first term of the objective function is a quadratic form of the joint velocity. This is critical to formulate the problem as a QP which leads to the possibility of real-time implementation.

Problem-1 defined in Eq. (4.16) is rewritten in the discrete setting as

$$\begin{aligned}
\text{Problem-2} \quad & \underset{\dot{\theta}[k],u[k]}{\text{minimize}} && \left\| \hat{\mathrm{J}}_\omega[k]\dot{\theta}[k] + b[k] \right\|_2^2 + u[k]^{\mathrm{T}}\mathrm{S}u[k] \\
& \text{subject to} && \dot{\theta}[k] = \dot{\theta}[k-1] + \frac{\Delta t}{2}(u[k-1] + u[k]) \\
& && \dot{\theta}^{min} \leq \dot{\theta}[k] \leq \dot{\theta}^{max} \\
& && u^{min} \leq u[k] \leq u^{max} \\
& && h(\dot{\theta}[k]) \leq 0 \\
& && \theta[0] = \theta_0, \dot{\theta}[0] = \dot{\theta}_0 \\
& && r_b[0] = r_0, Q_b[0] = Q_0 \\
& && \xi_b[0] = \xi_b^0 \\
& && g_e^d[k] = \mathcal{F}\big(\theta[k], r_b, Q_b\big)
\end{aligned} \tag{3.36}$$

where $\xi_b = [v_b, \omega_b]^{\mathrm{T}} \in \mathbb{R}^6$ is the base velocity. Note that Problem-2 aims at minimizing the *instantaneous* angular velocity of the base rather than its sum over the given horizon. Furthermore, although the objective function is in QP form, Problem-2 is non-convex with obstacle avoidance and end-effector constraints.

3.3 Obstacle Avoidance

The kinematic redundancy allows the planner to satisfy secondary tasks such as obstacle avoidance. Some of the common approaches to the obstacle avoidance include the artificial potential field method [71], the pseudo-inverse method [103], and the Jacobian transpose method [88]. In this paper, an inequality-based obstacle avoidance is used as proposed in [68].

Let the configuration manifold of the robot be given as $\mathcal{Q} \in \mathrm{SE}(3) \times \mathbb{R}^n$, the tangent space of $\mathcal{Q}$ as $T_q\mathcal{Q}$ at some configuration q, and the robot state space is defined as $\mathcal{X} = \mathcal{Q} \times T_q\mathcal{Q}$. Defining the state time space as $\mathcal{S} = \mathcal{X} \times [0, t_f]$, we can describe the feasible motion as

$$\mathcal{S}_{feasible} = \{(r_b, Q_b, \xi_b, \theta, \dot{\theta}, t) \in \mathcal{S} : \mathcal{R}(q) \cap \mathcal{O} = \emptyset \quad , \forall t \in [0, t_f]\} \tag{3.37}$$

where $\mathcal{R}(q)$ is the volume occupied by the robot at configuration q, $\mathcal{O}$ is defined as the known obstacle region, and $\emptyset$ denotes a null set. Whenever a collision is detected between the robot and the obstacle (modeled as convex polygons), an escape velocity is designed to drive the robot away from the obstacle. This is mathematically formulated as

$$\mathrm{J_T}(\theta)\dot{\theta} = \dot{x}_0 \tag{3.38}$$

where $\mathrm{J_T} \in \mathbb{R}^{3\times n}$ is the translational Jacobian of the link and $\dot{x}_0$ is the escape velocity which is defined as a function of the link-obstacle distance as shown in Fig. 3.2.

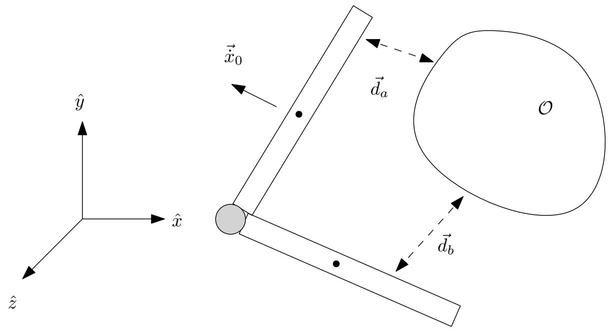


Figure 3.2: Induced escape velocity for obstacle avoidance. Links and obstacles are assumed as convex polyhedra. $\vec{d}_a$ and $\vec{d}_b$ are the minimum distances between the links and obstacle. If the minimum instantaneous distance between the link and obstacle is less or equal to some threshold, an escape velocity shown as $\dot{x}_o$ is imparted.

Several algorithms for collision detection between convex objects exist in literature [14, 18] and will not be discussed here for brevity. Using the method proposed in [68], obstacle avoidance strategy is expressed via the minimum distance d between two convex objects as

$$\dot{d} \geq -\epsilon \frac{d - d_2}{d_1 - d_2} \quad \text{if} \quad d < d_1 \tag{3.39}$$

where $\epsilon > 0$ is a tuning parameter which governs the convergence speed and d_1, d_2 denote the outer and inner safety threshold distance, respectively. A key advantage of this strategy is that it provides a lower bound on the minimum distance between the objects, which is expressed as [68]

$$\forall t > 0, \quad d(t) \geq d_2 + (d(0) - d_2) \exp(-\frac{\epsilon}{d_1 - d_2} t) > d_2, \quad \text{if} \quad d(0) \geq d_2 \tag{3.40}$$

Furthermore, $\dot{d}$ is formulated in the joint velocity space as

$$\dot{d} = \mathrm{J}_{\mathrm{T}}(\theta)\dot{\theta} \cdot \gamma \tag{3.41}$$

where $(\cdot)$ denotes the dot product operator and $\gamma \in \mathbb{R}^3$ is the unit vector between the closest points of the two convex polyhedras. Finally, the obstacle avoidance constraint is expressed via a linear inequality as

$$\dot{\theta} \cdot \mathrm{J}_{\mathrm{T}}(\theta)^{\mathrm{T}} \gamma \geq b_G \tag{3.42}$$

where, $b_G = -\epsilon \frac{d - d_2}{d_1 - d_2}$. In the presence of a field of obstacles, the avoidance constraint becomes $d_{ij} > d_2$, $\forall i = 1.2..N_{\text{links}}$ and $j = 1.2...M_{\text{obstacles}}$.

Base-obstacle avoidance We recall the relationship between the base translational velocity and the joint velocities derived via the conservation of momentum, originally formulated in Eq. (3.22).

$$v_b = \hat{\mathrm{J}}_v \dot{\theta} + \mathrm{G}_3 \mathrm{H}_0 + \mathrm{G}_4 \mathrm{P}_0 \tag{3.43}$$

Similar to the approach discussed above for link-link and link-obstacle avoidance, we can frame the obstacle avoidance for the base spacecraft in terms of a linear inequality constraint using the joint velocity as

$$\dot{\theta} \cdot \hat{\mathrm{J}}_{\mathrm{v}}^{\mathrm{T}} \gamma \geq b_B \tag{3.44}$$

where $b_B = (b_G - (\mathrm{G}_3\mathrm{H}_0 - \mathrm{G}_4\mathrm{P}_0 \cdot \gamma)$.

We note that the optimization framework along with the constraint formulation in Problem-2 is formulated in a discrete setting. Therefore, it becomes paramount to ensure that the obstacle avoidance constraints are not violated between the two adjacent discrete nodes. Next, we provide a condition that guarantees the obstacle avoidance constraints between the discrete nodes.

Proposition 1. *The obstacle avoidance constraints are not violated between discrete nodes if* $\epsilon \leq \frac{d_1 - d_2}{\Delta t}$.

Proof. Under the assumption that the time step chosen is small, we can assume that the closest points between the two convex objects do not change due to rotation or translation between the nodes. The minimum distance requirement is expressed as

$$d(\tau) > d_2 \;\; \forall \tau \in (t, t + \Delta t] \tag{3.45}$$

Linearizing Eq. (3.45), we obtain

$$d(\tau) = d(t) + \dot{d}(t)(\tau - t) + \mathcal{O}(\Delta t^2) \tag{3.46}$$

Ignoring higher order terms corresponding to $\mathcal{O}(\Delta t^2)$, we consider the limiting case where the minimum distance approaches the inner safety distance, d_2 at time

t, i.e. $d = d_2 + \delta_1$ for some $\delta_1 > 0$. We can express Eq. (3.46) as

$$d(\tau) = d_2 + \delta_1 + \dot{d}(t)(\tau - t) \tag{3.47}$$

Reformulating, Eq. (3.39) using a slack variable δ_2, we have

$$\dot{d}(t) = -\epsilon \frac{\delta_1}{d_1 - d_2} + \delta_2, \quad \delta_2 \geq 0 \tag{3.48}$$

Substituting Eq. (3.48) into Eq. (3.47), we get

$$d(\tau) = d_2 + \delta_1 (1 - \frac{\epsilon}{d_1 - d_2}(\tau - t)) + \delta_2(\tau - t) \tag{3.49}$$

We require that the minimum distance $d(\tau) > d_2$ to ensure obstacle avoidance constraints are not violated. This leads to

$$\delta_1 (1 - \frac{\epsilon(\tau - t)}{d_1 - d_2}) + \delta_2(\tau - t) > 0 \tag{3.50}$$

Since δ_1, δ_2, although unknown, are positive and non-negative, a conservative least upper bound can be found for ϵ that guarantees collision avoidance between adjacent nodes. This is expressed as

$$\epsilon \leq \frac{d_1 - d_2}{\Delta t} \tag{3.51}$$

Thus, if the chosen convergence parameter, $\epsilon \leq \frac{d_1 - d_2}{\Delta t}$, collision avoidance is guaranteed $\forall \tau \in (t, t + \Delta t]$. □

3.4 End-effector Task Compliance

Pre-grasping tasks for the space robot can include target observation, safe approach to the grasp locations without collisions, pose synchronization to ensure that the grasp locations is directed towards the manipulator, and relative velocity minimization between the target and the end-effector. We consider two types of task-induced constraints for the end-effector. i.e. end-effector task compliance, here.

Waypoint guidance

Here we consider the case that the end-effector is required to go through a certain finite set of waypoints during the operation. For example, during target observation, the end-effector is often guided such that the target remains in the robot's field of view. Let the waypoints be denoted by $g^l \in \mathrm{SE}(3), l = 1, 2, 3..r$, to be traversed in a given sequence of time. We would like to find a continuous trajectory $g_e(t) \in \mathrm{SE}(3)$ that satisfies the task requirements. Interpolation techniques for motions evolving on SE(3) are not straightforward as compared to Euclidean spaces due to the lack of a bi-invariant Riemannian metric. However, research has been done on generating and interpolating trajectories on SE(3) based on invariant screw motions and by decomposing motion on SE(3) and considering rotations and translations separately [34, 194, 13, 9].

Consider a point to point planning example as shown in Fig. 3.3. Let $\big[g_e[0], \xi_e^\vee[0]\big] \in \mathrm{TSE}(3)$, be the initial end-effector state, where $\mathrm{TSE}(3) \simeq \mathrm{SE}(3) \times \mathfrak{se}(3)$ is the tangent bundle of SE(3). The final end-effector state is given as $\big[g_e[f], \xi_e^\vee[f]\big]$. The goal is to find a $\mathcal{C}^2$ trajectory $g_e(.) : [0, t_f] \to \mathrm{SE}(3)$ that satisfies the initial and final end-effector states. We use the De Casteljau algorithm on SE(3) [9] which is based on an Ad-invariant pseudo Riemannian metric (a non-degenerate metric tensor which need not be positive definite). The basic idea behind the algorithm is to generate a smooth curve by generating 'control' points using the boundary conditions and interpolating via a combination of matrix exponential $(\exp_{\mathrm{SO}(3)}, \exp_{\mathrm{SE}(3)})$ and logarithm $(\log_{\mathrm{SO}(3)}, \log_{\mathrm{SE}(3)})$ maps. We provide the basics of Logarithm and Exponential maps in the appendix while detailed discussions

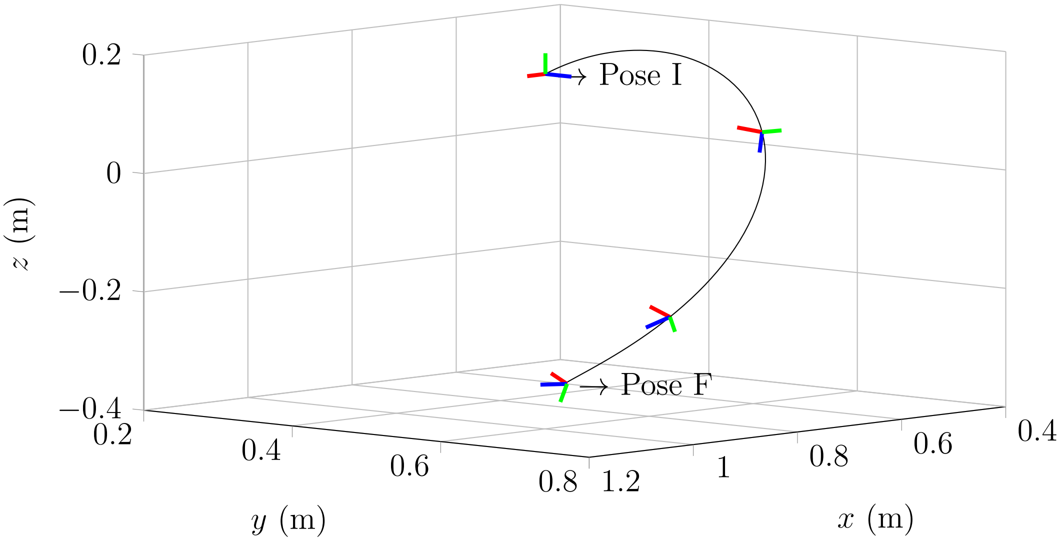


Figure 3.3: End-effector trajectory generated by the De Casteljau algorithm based on the ad-invariant pseudo Riemannian metric. The initial pose and final pose are shown as Pose I and pose F, respectively.

on the Decasteljau algorithm can be found in [9, 10].

Continuous tracking

It is also possible that the desired trajectory is given in a continuous form. Let the desired trajectory be given as $g_e^d(t) \in \mathrm{SE}(3)$. The pose in discrete time can then be expressed as

$$g_e^d[k] \triangleq g_e^d(k\Delta t) = \begin{bmatrix} Q_e[k] & r_e[k] \\ 0 & 1 \end{bmatrix}, \forall k = [0, N] \tag{3.52}$$

Given an end-effector pose trajectory in discrete time, the associated desired twists are related to the pose via the logarithm map, $\log_{\mathrm{SE}(3)} : \mathrm{SE}(3) \rightarrow \mathfrak{se}(3)$, represented as

$$\xi_e^d[k] = \mathrm{N}_e[k] \frac{1}{\Delta t} \log_{\mathrm{SE}(3)}({g_e^d}^{-1}[k] g_e^d[k+1])^{\wedge} \tag{3.53}$$

After computing the discrete twists for either waypoint guidance using the Decasteljau algorithm or continuous tracking in Eq. (3.52), the twists can be mapped back to the joint space using the Jacobian matrix given by Eq. (3.30), which is expressed in discrete time as

$$\xi_e^d[k] = \mathrm{J}_g[k]\dot{\theta}[k] + \mathrm{J}_s[k]\begin{bmatrix} G_4[k] & G_3[k] \\ G_2[k] & G_1[k] \end{bmatrix}\begin{bmatrix} \mathrm{P}_0 \\ \mathrm{H}_0 \end{bmatrix} \tag{3.54}$$

With Eq. (3.54), the end-effector pose constraints is expressed as linear equality constraints given as

$$\tilde{\xi}_e[k] = \mathrm{J}_g[k]\dot{\theta}[k] \equiv \xi_e^d[k] - \mathrm{J}_s[k]\begin{bmatrix} G_4[k] & G_3[k] \\ G_2[k] & G_1[k] \end{bmatrix}\begin{bmatrix} \mathrm{P}_0 \\ \mathrm{H}_0 \end{bmatrix} \tag{3.55}$$

Finally, the task constrained trajectory planning problem is expressed as Problem-3, given as

$$\begin{aligned} \text{Problem-3} \quad & \underset{\dot{\theta}[k],u[k]}{\text{minimize}} \quad && \left\|\hat{\mathrm{J}}_\omega[k]\dot{\theta}[k] + b[k]\right\|_2^2 + u[k]^{\mathrm{T}}\mathrm{S}u[k] \\ & \text{subject to} && \dot{\theta}[k] = \dot{\theta}[k-1] + \frac{\Delta t}{2}(u[k-1] + u[k]) \\ & && \dot{\theta}^{min} \leq \dot{\theta}[k] \leq \dot{\theta}^{max} \\ & && u^{min} \leq u[k] \leq u^{max} \\ & && \gamma^{\mathrm{T}}\mathrm{J}_{\mathrm{T}}(\theta[k])\dot{\theta}[k] \geq b_G[k] \qquad (a) \\ & && \gamma^{\mathrm{T}}\hat{\mathrm{J}}_{\mathrm{v}}[\mathrm{k}]\dot{\theta}[k] \geq b_B[k] \qquad (b) \\ & && \theta[0] = \theta_0, \dot{\theta}[0] = \dot{\theta}_0 \\ & && r_b[0] = r_0, Q_b[0] = Q_0 \\ & && \xi_b[0] = \xi_b^0 \\ & && \mathrm{J}_g[k]\dot{\theta}[k] = \tilde{\xi}_e[k] \end{aligned} \tag{3.56}$$

Note the inequality constraints a, b in Problem-3 denote the obstacle avoidance constraints for the manipulator and the base, respectively. Next, we prove that Problem-3 is a convex quadratic program with linear constraints.

Proposition 2. *The constrained optimization represented in Problem-3 is a convex quadratic program.*

Proof. Let $z[k] = \begin{bmatrix} \dot{\theta}[k] & u[k] \end{bmatrix}^{\mathrm{T}}$ be the new optimization variable. Problem-3 is expressed as

$$\begin{aligned} \underset{z[k]}{\text{minimize}} \quad & z[k]^{\mathrm{T}}\mathrm{M}[k]z[k] + n^{\mathrm{T}}z + l \\ \text{subject to} \quad & Cz[k] = d, \quad \forall k = 1, 2..N \\ & Hz[k] \leq f \\ & z[0] = \begin{bmatrix} \dot{\theta}_0 & 0 \end{bmatrix}^{\mathrm{T}} \\ & r_b[0] = r_0, Q_b[0] = Q_0 \\ & \theta[0] = \theta_0, \xi_b[0] = \xi_b^0 \end{aligned} \tag{3.57}$$

where

$$\mathrm{M}[k] = \begin{bmatrix} \hat{\mathrm{J}}_\omega[k]^{\mathrm{T}}\hat{\mathrm{J}}_\omega[k] & 0 \\ 0 & \mathrm{S} \end{bmatrix} \in \mathbb{R}^{2n \times 2n}, \quad n = \begin{bmatrix} 2\hat{\mathrm{J}}_\omega[k]^{\mathrm{T}}b[k] & 0 \end{bmatrix} \in \mathbb{R}^{2n}, \quad l = b[k]^{\mathrm{T}}b[k] \in \mathbb{R} \tag{3.58}$$

$$C = \begin{bmatrix} \mathrm{J}_g[k] & 0 \\ \mathrm{E} & -\frac{\Delta t}{2}\mathrm{E} \end{bmatrix} \in \mathbb{R}^{(6+n) \times 2n}, \quad d = \begin{bmatrix} \tilde{\xi}_e[k] \\ \begin{bmatrix} \mathrm{E} & \frac{\Delta t}{2} \end{bmatrix} z[k-1] \end{bmatrix} \in \mathbb{R}^{6+n} \tag{3.59}$$

$$H = \begin{bmatrix} \gamma^{\mathrm{T}}\mathrm{J}_{\mathrm{T}}(\theta[k]) & 0 \\ \mathrm{E} & \mathrm{E} \\ \gamma^{\mathrm{T}}\hat{\mathrm{J}}_{\mathrm{v}}[\mathrm{k}] & 0 \\ \mathrm{E} & \mathrm{E} \end{bmatrix} \in \mathbb{R}^{(2n+2) \times 2n}, \quad f = \begin{bmatrix} b_G \\ -\begin{bmatrix} \dot{\theta}^{min} & u^{min} \end{bmatrix}^{\mathrm{T}} \\ b_B \\ \begin{bmatrix} \dot{\theta}^{max} & u^{max} \end{bmatrix}^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{2n+2} \tag{3.60}$$

We first prove that the matrix $\mathrm{M}[k]$ is generally atleast positive semidefinite.

For a symmetric block matrix $X = \begin{bmatrix} X_1 & X_2 \\ X_2^{\mathrm{T}} & X_4 \end{bmatrix}$, we have the following properties

- If X_4 is invertible and $X_4 \succ 0$, then $X \succeq 0$ if and only if the schur complement of block X_4, $\mathrm{schur}(X_4) = X_1 - X_2^{\mathrm{T}} X_4^{-1} X_2 \succeq 0$ [46].

- For the case when X_1 is rank-deficient or singular, we have the following equivalence relation: $X \succeq 0$ if and only if $X_1 \succeq 0$, $(\mathrm{E} - X_1 X_1^{\dagger}) X_2 = 0$, and $X_4 - X_2^{\mathrm{T}} X_1^{\dagger} X_2 \succeq 0$, where † denotes the pseudo-inverse [46].

- Furthermore, if $X_1 \in \mathbb{R}^{n \times n}$ is a symmetric matrix, then X_1 is positive semi-definite if there exists a matrix $\mathrm{F} \in \mathbb{R}^{m \times n}$ such that $X_1 = \mathrm{F}^{\mathrm{T}}\mathrm{F}$ [195].

For the considered case, we know that matrix S which is block X_4 is positive definite. The positive semi-definiteness of $\mathrm{M}[k]$ is determined solely by the matrix X_1 which is given as

$$X_1 = (\hat{J}_\omega)[k]^{\mathrm{T}} \hat{J}_\omega[k] \tag{3.61}$$

This leads to

$$z[k]^{\mathrm{T}} \mathrm{M}[k] z[k] = \sum_{i=1}^{n} (\mathrm{M}[k] z[k])^2 \geq 0 \tag{3.62}$$

At configurations where X_1 loses rank, we still obtain the positive semi-definiteness of matrix $\mathrm{M}[k]$ based on the second property discussed above since $X_2 = 0$ and $X_4 \succ 0$. Thus, $\mathrm{M}[k]$ is proven to be positive semi-definite, and therefore, we conclude that Problem-3 is a convex quadratic program. □

However, there may exist situations where at some node k, the GJM $\mathrm{J}_g[k] \in \mathbb{R}^{6 \times n}$ is rank-deficient, which leads to dynamic singularities. At such singular configurations, the rows of the equality constraint matrix C cease to be independent. However, the proposed formulation can accommodate such situations by decomposing $\mathrm{J}_g[k]$ as follows [31]

$$\mathrm{J}_g = \begin{bmatrix} \mathrm{J}_1 & \mathrm{J}_2 \end{bmatrix}^{\mathrm{T}} \tag{3.63}$$

where $\mathrm{J}_1 \in \mathbb{R}^{r\times n}$ has full row rank and $\mathrm{J}_2 \in \mathbb{R}^{6-r\times n}$. The end-effector twists are correspondingly prescribed as $\tilde{\xi}_e[k] = \begin{bmatrix} \tilde{\xi}_1[k] & \tilde{\xi}_2[k] \end{bmatrix}^{\mathrm{T}}$ with the new equality constraint

$$\mathrm{J}_1[k]\dot{\theta}[k] = \tilde{\xi}_1[k] \tag{3.64}$$

where $\tilde{\xi}_1[k]$ denotes the direction of end-effector twists for which there are no singular configurations. The objective function can then be augmented with a 2-norm of the end-effector pose tracking error in the singular direction, leading to the new cost function given as $\left\|\hat{\mathrm{J}}_\omega[k]\dot{\theta}[k] + b[k]\right\|_2^2 + u[k]^{\mathrm{T}}\mathrm{S}u[k] + \left\|\mathrm{J}_2[k]\dot{\theta}[k] - \tilde{\xi}_2[k]\right\|_2$, with $\tilde{\xi}_2[k]$ denoting the twists in singular direction. Compared to inverse kinematics based approaches requiring explicit inversion of $\mathrm{J}_g[k]$, therefore demanding arbitrarily large velocities in the neighbourhood of the singularity, the QP formulation can handle such constraints more optimally.

The algorithm for solving Problem-3 in the convex programming framework is displayed in Algorithm 1, where the function *De Casteljau* is used to interpolate the end-effector trajectory if waypoint guidance is required and *SolveQP* is used to solve the convex program formulated in Eq. (4.21).

Algorithm 1 Convex Planning Algorithm

```
Initialize manipulator configuration $\in \mathrm{SE}(3) \times \mathbb{R}^n$
Initialize desired end-effector trajectory $g_e(t)$
if $g_e \in$ waypoints then
    De Casteljau $(g_e, \xi_e)$
end if
Choose discretization step $h = \Delta t$
Compute discrete time horizon, $N = \frac{t_f}{\Delta t}$
Discretize end-effector pose $g_e^d[k]$
Get $\xi_e^d[k]$ via $\log_{\mathrm{SE}(3)}$
for k = 0,1 ... N do
    $SolveQP(\dot{\theta}[k], u[k])$
end for
```

Numerical Results

Table 3.1: Manipulator physical properties

	Base	Link 1	Link 2	Link 3	Link 4	Link 5	Link 6	Link 7	Link 8	Link 9	Link 10
Mass, kg	800	5	5	5	5	5	5	5	5	5	5
I_x, kgm^2	100.4	0.01	0.01	0.21	0.01	0.21	0.01	0.21	0.21	0.01	0.01
I_y, kgm^2	75	0.21	0.21	0.01	0.21	0.01	0.21	0.01	0.01	0.21	0.21
I_z, kgm^2	100.4	0.21	0.21	0.21	0.21	0.21	0.21	0.21	0.21	0.21	0.21
Length, m	[1, 0.75, 1]	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2

In this section, the proposed convex optimization scheme is numerically validated using a continuous end-effector tracking example. The space-robotic system consists of a free-floating base spacecraft and a 10-dof manipulator. Table 3.1 presents the physical properties of the system. Notice the mass ratio of the manipulator to base is $\frac{1}{16}$, hence significant coupling between the base and the manipulator can be expected. Three obstacles modeled as convex polyhedrons are assumed in the proximity of the manipulator. A Gilbert-Johnson-Keerthi (GJK) algorithm [49] is implemented on MATLAB for collision queries between the obstacles and the manipulator. The inner and outer safety threshold distances for obstacle avoidance are chosen as 0.05 and 0.1 m while the convergence parameter ϵ is set as 0.5 m$\cdot$ s^{-1}. The goal of the planning algorithm is to minimize the base attitude disturbance and control effort while adhering to the end-effector task constraints, joint limits, and ensure a collision free motion.

Table 3.2: Numerical simulation parameters

	Symbols, unit	Initial Conditions (t=0)
	r_b, m	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^{\mathrm{T}}$
Base	Q_b	Identity matrix
	ξ_b, m$\cdot$ s^{-1} and rad$\cdot$s^{-1}	$\begin{bmatrix}0.005 & 0.005 & -0.005 & 0.003 & -0.0001 & 0.008\end{bmatrix}^{\mathrm{T}}$
Manipulator	θ, rad $\dot{\theta}$, rad$\cdot$s^{-1}	$\begin{bmatrix}\frac{\pi}{5} & -\frac{2\pi}{3} & 0 & \frac{\pi}{4} & -\frac{\pi}{4} & \frac{\pi}{2} & \frac{\pi}{2} & -\frac{\pi}{2} & 0 & 0\end{bmatrix}^{\mathrm{T}}$ $\begin{bmatrix}0.23 & 0.08 & 0.06 & 0.24 & 0.05 & 0.19 & -0.15 & -0.11 & 0.03 & 0.17\end{bmatrix}^{\mathrm{T}}$
Initial Momentum	P_0, kg $\cdot$ m $\cdot$ s^{-1} H_0, kg $\cdot$ m^2 $\cdot$ s^{-1}	$\begin{bmatrix}5.08 & 6.81 & 5.81\end{bmatrix}^{\mathrm{T}}$ $\begin{bmatrix}-2.64 & 0.54 & -0.55\end{bmatrix}^{\mathrm{T}}$
		Constraints
State bounds	θ, rad $\dot{\theta}$, rad$\cdot$ s^{-1}	$\theta \in \begin{bmatrix}-\frac{2\pi}{3} & \frac{2\pi}{3}\end{bmatrix}$ $\dot{\theta} \in \begin{bmatrix}-\frac{\pi}{2} & \frac{\pi}{2}\end{bmatrix}$
Control bounds	u, rad$\cdot$s^{-2}	$\ddot{\theta} \in \begin{bmatrix}-\frac{2\pi}{3} & \frac{2\pi}{3}\end{bmatrix}$
		Desired Trajectory
End-effector pose	$r_e(t)$, m	$\begin{bmatrix}-0.35 + 0.5\sin\frac{\pi t}{10} & -0.81 + 0.5\sin\frac{\pi t}{10} & 0.97 - 0.3\sin\frac{\pi t}{6}\end{bmatrix}^{\mathrm{T}}$
	$Q_e(t)$	$Q_e[0]\exp_{\mathrm{SO(3)}}\left(\begin{bmatrix}0.005\sin\frac{t}{5} \\ 0 \\ 0.5\sin\frac{t}{5}\end{bmatrix}^{\times}\right)$

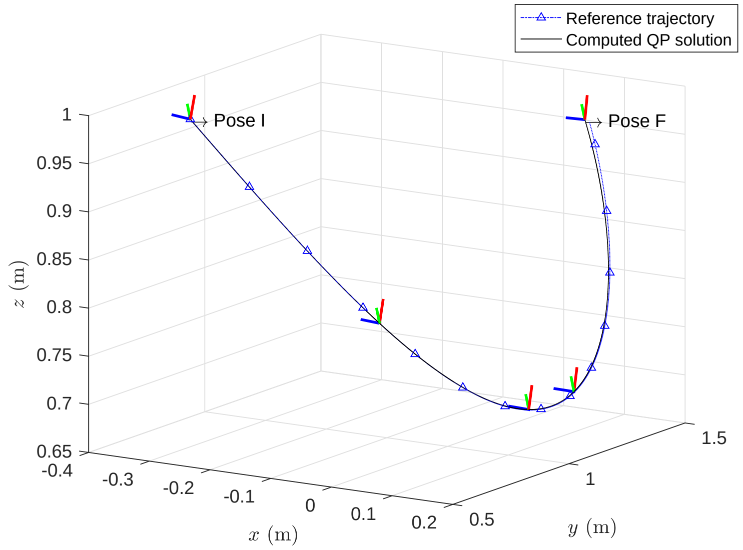
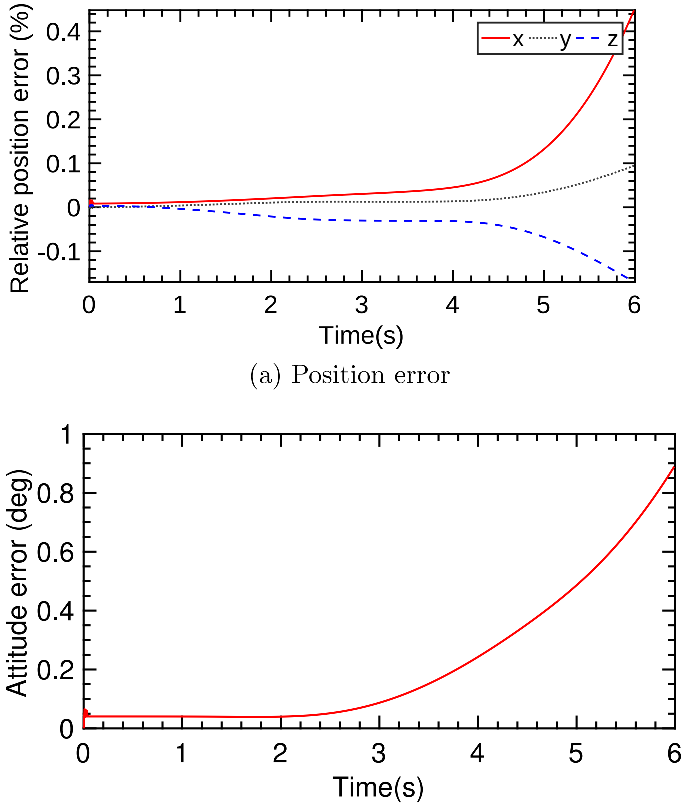


Figure 3.4: End-effector pose trajectory. End-effector attitude is displayed at finite intervals. Pose I and F denote initial and final pose, respectively.

The total time of operation is specified as $t_f = 6$ s with a step size of 30 ms, leading to a total of 200 discrete nodes. The initial conditions are presented in Table 3.2 and notice these conditions are chosen such that the total initial momentum is non-zero for the system.



(a) Position error

(b) Attitude error

Figure 3.5: End-effector trajectory error. Attitude error is computed via the logarithm map, $\left\|\log_{\mathrm{SO}(3)}\left(Q_e^d(t)^{\mathrm{T}} Q_e(t)\right)\right\|$, where $Q_e^d(t) \in \mathrm{SO}(3)$ is the desired attitude trajectory.

Fig. 3.4 illustrates the manipulator end-effector pose trajectory obtained from the planning algorithm in the inertial frame. The attitude of the end-effector is also displayed at discrete intervals. The relative end-effector position and attitude errors normalized to the reference trajectory are displayed in Figs 3.5. Some deviation is observed in the computed pose, which is attributed to the numerical error since a first order Euler integration scheme has been used to compute the joint angles. However, from Figs. 3.5, we observe that the relative position error is less than 0.4% for a 6 s operation and the attitude error expressed via the exponential coordinates is less than one degree.

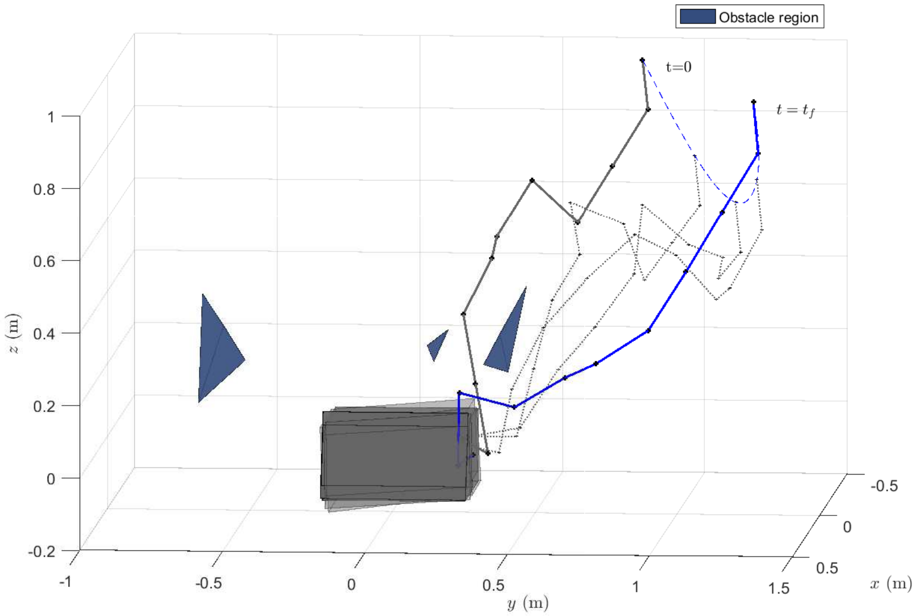


Figure 3.6: Three-dimensional manipulator motion. Obstacles are shown in blue.

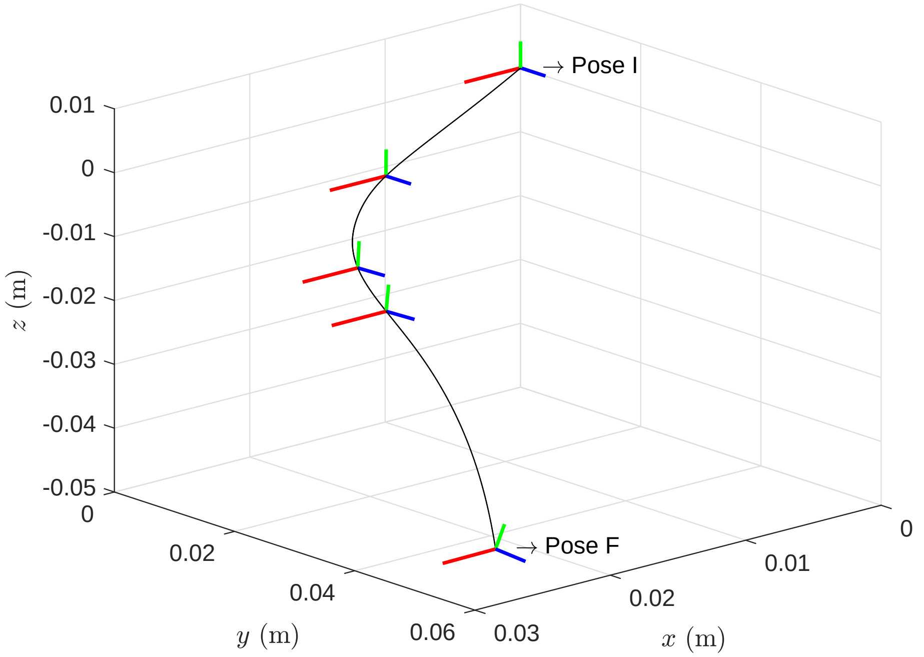


Figure 3.7: Base spacecraft pose trajectory. Base attitude is displayed at finite intervals. Pose I and F denote initial and final pose, respectively.

The corresponding three-dimensional trajectory of the robotic system along with the obstacle regions relative to the inertial frame is shown in Fig. 3.6. The base spacecraft motion is illustrated in Figs. 3.7 and 3.8. Although it may seem that the base rotation is large, this is largely due to the initial non-zero base angular velocity. If on the other hand, only the control effort is minimized in Problem-3, the base rotates by approximately 12 deg. In summary, the computed QP solution allows for smooth tracking of the desired trajectory in the presence of obstacles, state and control bounds, and even as the base rotates by approximately 8 deg and translates by approximately 6 cm during the motion. Following end-effector approach to the target, the next phase consists of physical interception in which the robot physically grasps the capture point of the target. This requires manuerving the space-robotic system to have zero relative velocity with respect to target to ensure a safe and successful capture. Following mechanical interfacing, control torques on the manipulator are applied to bring the entire capture

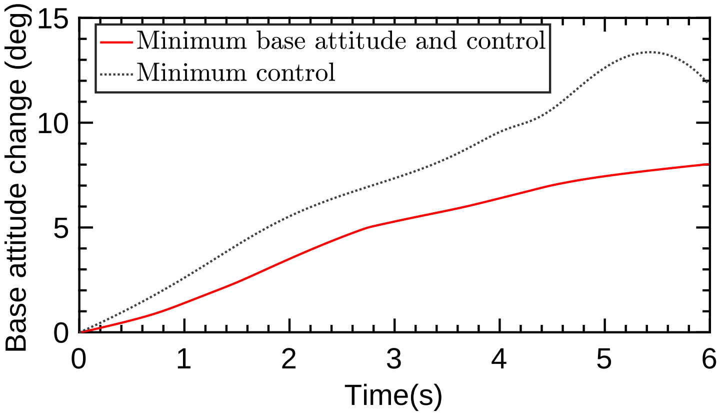
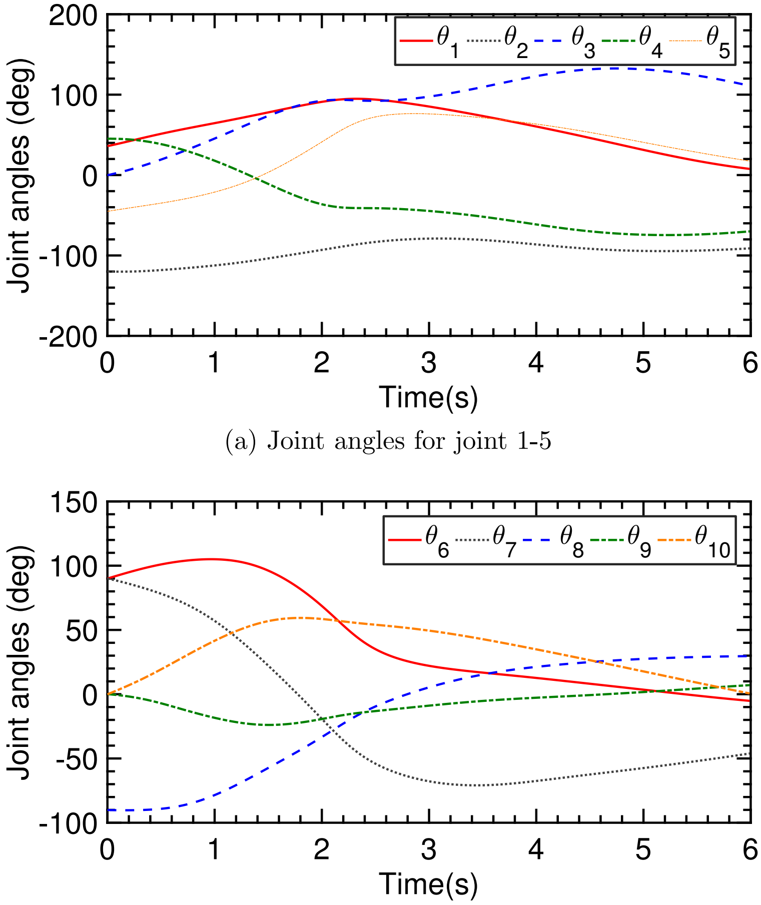


Figure 3.8: Base attitude change during manipulator operation. Attitude change is computed as $\left\|\log_{\mathrm{SO}(3)}\left(Q_b(t=0)^{\mathrm{T}} Q_b(t)\right)\right\|$, where $Q_b(t) \in \mathrm{SO}(3)$ is the attitude trajectory of the base spacecraft.



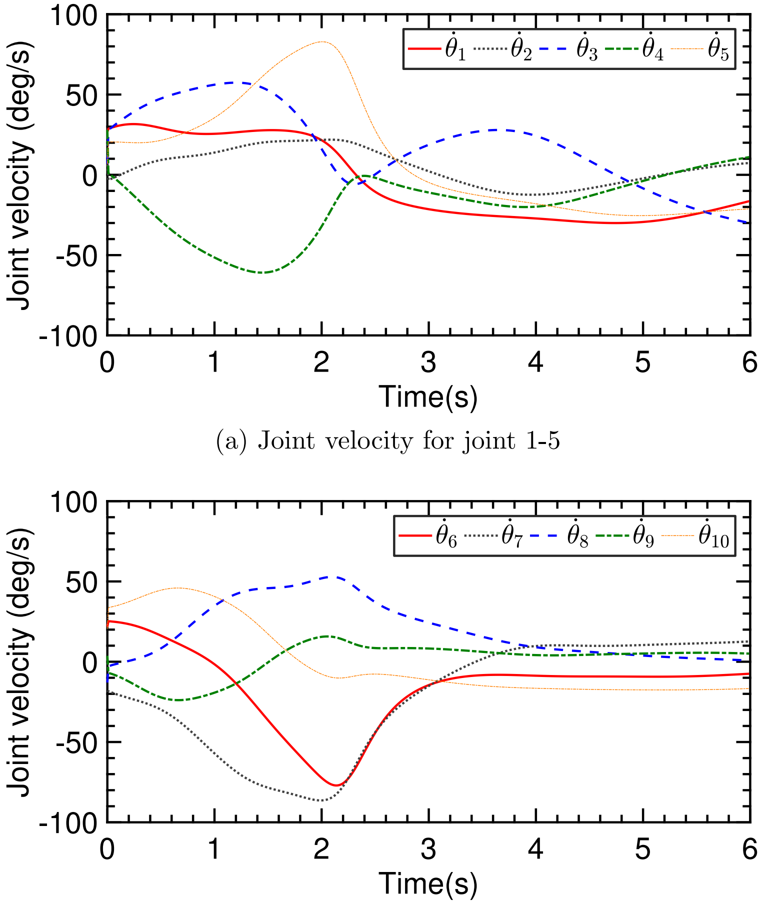
(a) Joint angles for joint 1-5

(b) Joint angles for joint 6-10

Figure 3.9: Joint angle trajectory.

The resulting joint angle trajectories obtained from the solution of the QP problem are shown in Fig. 3.11; the joint velocities are shown in Fig. 3.10; and the joint control inputs are shown in Fig. 3.11. All state and control constraints have been satisfied as observed from these figures. Furthermore, the magnitude of the total system momentum remains constant as illustrated in Fig. 3.12.

The planning algorithm is implemented on a PC running Windows 10, with an Intel Core i5 CPU @ 2.90 GHz and 8 GB RAM. We use the YALMIP [100] optimization environment on MATLAB with the Gurobi [51] QP solver for the numerical simulations. In order to appreciate the advantage in formulating the motion in the joint velocity space which leads to a QP, we compared our approach with an alternate discrete formulation which solves Problem-1, where the decision variables are chosen as both joint angles and velocities.

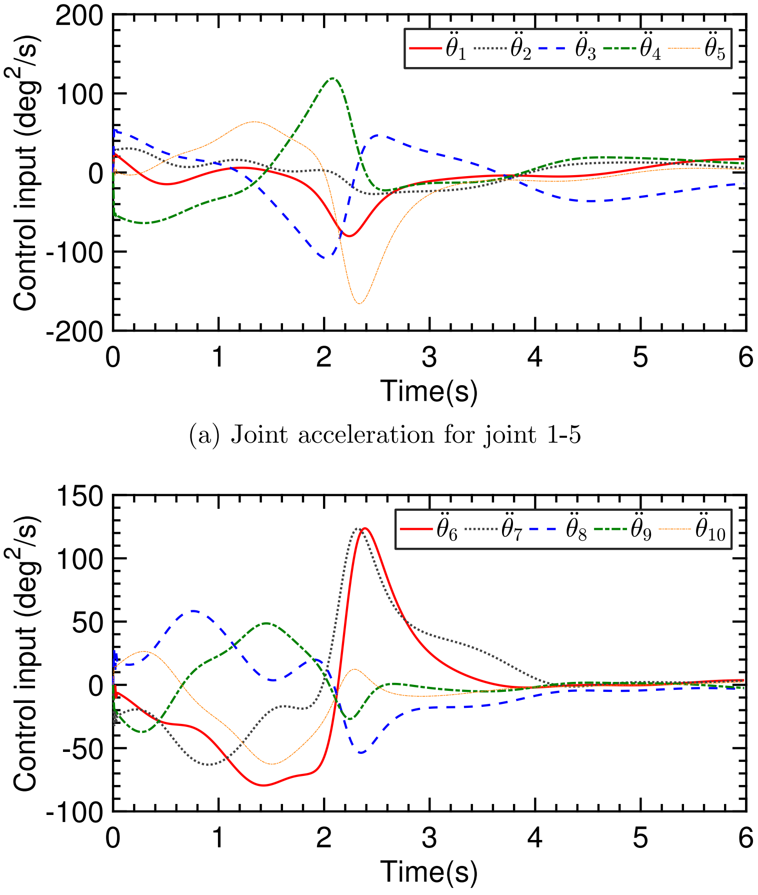


(a) Joint velocity for joint 1-5

(b) Joint velocity for joint 6-10

Figure 3.10: Joint velocity trajectory.

Note that with this approach, we end up with a non-convex, nonlinear program. Fig. 3.13 illustrates the computation time comparison between our proposed formulation and the alternate nonlinear program along with the given stepsize for the problem. We test both the Barrier (B) algorithm and the Dual-simplex (DS) algorithm within the Gurobi solver for the proposed QP formulation. A sequential quadratic programming (SQP) solver in the Knitro optimization package [26] is used to solve the nonlinear program. The SQP solver requires an initial guess which is used and iteratively refined to linearize and convexify the equality, inequality constraints, and objective, respectively. We use the joint states at the previous node [k-1] as the initial guess for the problem to be solved at node [k].



(a) Joint acceleration for joint 1-5

(b) Joint acceleration for joint 6-10

Figure 3.11: Joint acceleration trajectory.

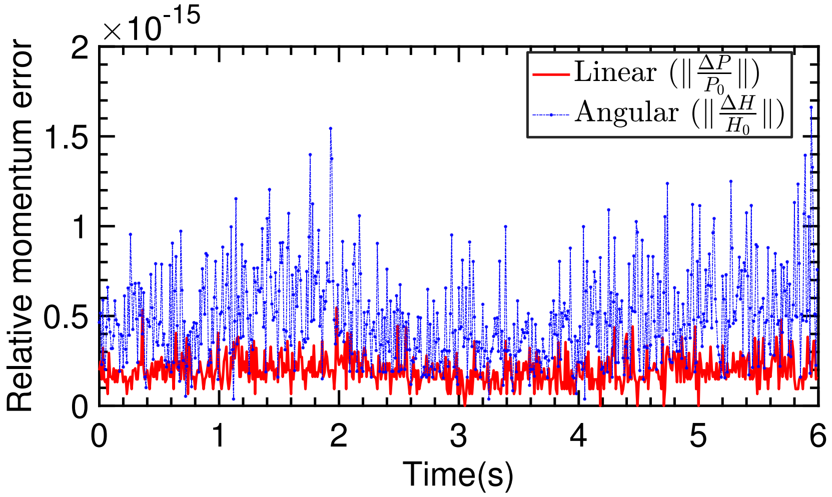


Figure 3.12: Relative momentum error during manipulator motion.

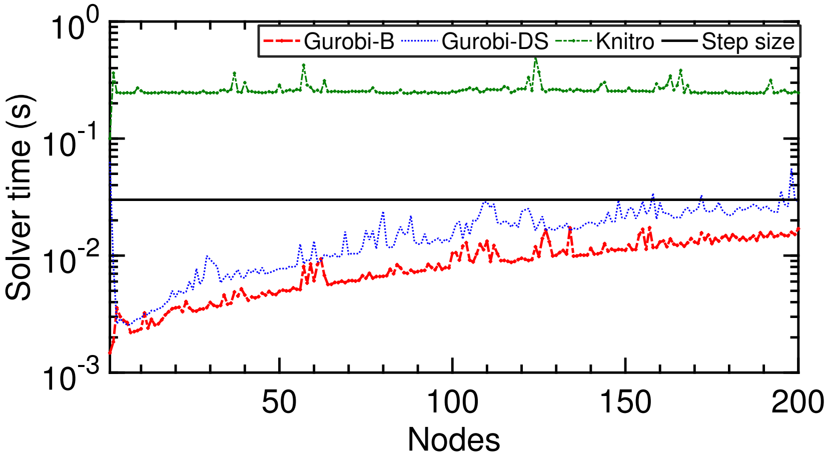


Figure 3.13: Computation time comparison of Gurobi QP solvers- Barrier (B) and Dual-simplex (DS) with non-convex SQP based Knitro solver. Computation times for the QP solvers- B and DS are well below the chosen constant step size.

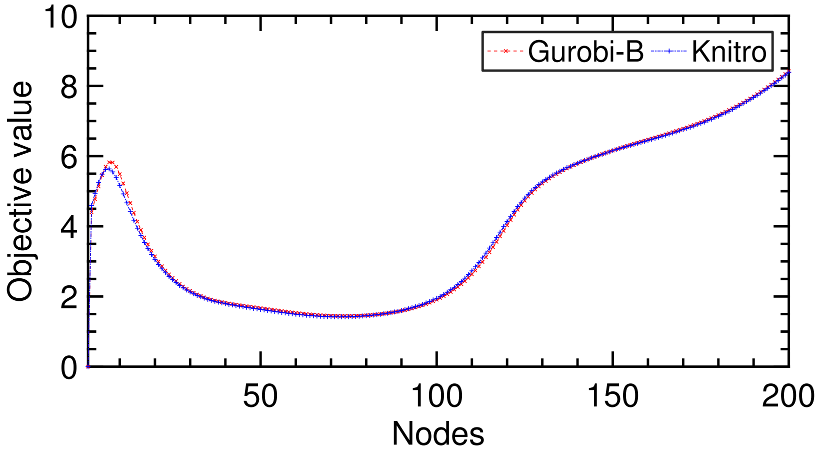


Figure 3.14: Objective value for Gurobi QP solver and non-convex SQP Knitro solver.

This greatly speeds up the computation time for the SQP solver. We observe that the Barrier method for solving the QP is approximately two orders of magnitude faster than the corresponding nonlinear program solver. Furthermore, with an average computation time of 8 ms per node where the given step size is 0.03 s per node, real-time planning and replanning can indeed be achieved. We also compare the optimal solution obtained by the QP and the nonlinear program. Their objective values are shown in Fig. 3.14. It is observed that the objective value is slightly higher for the QP initially, however overall the difference is small.

Chapter 4

Globally optimal model predictive control for spacecraft trajectory tracking

4.1 Introduction

Autonomous orbit control in challenging dynamical regimes is a significant challenge in spacecraft guidance and control. Particularly, orbit station-keeping neaar Libration points originating from the solution of the restricted three body problem is of particular interest. These points provide an ideal location for deep space robotic and human explorations as well as monitoring and studying the solar activity. A comprehensive architecture of human exploration missions to such points can be found in Farquhar et al [43]. The dynamics in proximity of these libration points is well investigated [47]. The phase space around the collinear libration points is a saddle×center×center type. Among the infinitely many periodic and quasi-periodic solutions, the most well-known include the halo, vertical, and planar Lyapunov orbits. A key feature of such orbits is the inherent instability associated with the motion. Therefore, an active station-keeping control strategy is often required to keep the spacecraft close to the desired reference orbit.

Several station-keeping strategies have been proposed in the literature; see Shirobokov et al. [156] for a comprehensive survey. Time-varying linear quadratic regulator (LQR) techniques and backstepping control with time-invariant LQR have been used to design optimal feedback control strategies for halo orbit station-keeping for the Earth-moon L1 point [122]. LQR based control is also studied in

Ghorbani and Assadian [48] for the Earth-moon system but the model takes into account the gravitational perturbations from the Sun, planets and eccentricity effects. Nonlinear programming based control strategies have been explored in Ulybyshev [172] for long term station-keeping for lunar halo orbits. $\mathcal{H}_\infty$ based control has been applied for asymptotic stabilization of spacecraft motion in a halo orbit by considering a linear time-varying approximation of the nonlinear dynamics [74]. Impulsive control methods such as discrete LQR and sliding mode control have been studied for station-keeping for halo orbits at the L2 Lagrange point in the Earth-moon system [91]. Bai and Junkins [12] apply the modified Chebyshev-Picard iteration method for halo orbit station-keeping by considering the trajectory in the form of orthogonal Chebyshev polynomials. By constraining the positions at two points, the station-keeping problem is formulated as a two-point boundary value problem which is solved to find discrete control impulses.

Although the literature on spacecraft station-keeping in libration point orbits is rich, several control issues exist. Some significant challenges include state and actuator constraints on the system, optimality in the presence of model uncertainty, and on-board feedback among others. Model predictive control (MPC) is an interesting solution for the challenges listed above. Due to MPC's inherent capability to account for state and control bounds while optimizing the desired performance criteria, in addition to real-time computational capability for liner and (some) nonlinear systems, it has become a very popular control methodology in aerospace applications such as relative spacecraft motion control [184], rendezvous and proximity operations [36, 55, 54, 183, 42]. Peng et al. [134] have demonstrates that the receding horizon control method is feasible to maintain periodic libration point orbits in the more complex Sun-Mercury elliptic restricted three-body problem using a linearized model. Infact, most of the literature consider linear dynamics for MPC control design due to the convenient convexity of the optimization problem resulting from the simplified/linearized dynamics (as long as other constraints are also convex). This allows such problems to be solved via fast and reliable interior point methods.

However, the dynamics describing the motion of the spacecraft in a three body regime is highly nonlinear. In this chapter, we consider a nonlinear model predictive control strategy for station-keeping of libration point orbits. Although MPC strategies have been considered in prior studies [66], the emphasis has been either on obtaining solution using linearized dynamics, or locally optimal solutions using nonlinear MPC. In contrast, we propose a nonlinear dynamical model based on a polynomial approximation of the nonlinear terms. Apart from capturing higher order nonlinearities, the optimal control problem associated with this dynamical model with a polynomial cost function can be transcribed as a static polynomial optimization. Interestingly, such problems can be reduced to semidefinite programs (SDP) via theory of moments and non-negative polynomials [78]. This allows the solution for the original non-convex polynomial MPC problems to be found by solving a series of convex SDPs [53]. While the series of SDPs provide a lower bound on the optimal value of the problem, stronger theoretical guarantees exist which prove that the method is monotonically convergent to a global optimum in certain cases. This is appealing for applications where control performance degradation is critical as the globally optimal solutions may be more desirable. Furthermore, this approach does not require any initial guess as opposed to local nonlinear descent methods. The remainder of the chapter is organized as follows. In section 4.2, the dynamical model for the halo orbit station-keeping is developed. Section 4.3 presents the background behind polynomial optimization; Section 4.4 focuses on the formulation of the original problem as a non-convex polynomial MPC problem. Numerical results for station-keeping of Halo and Lissajous orbits using the polynomial MPC approach are shown in Section 4.5.

4.2 Equations of Motion

The circular restricted three body problem (CRTBP) can be stated as evaluating the motion of a negligible mass m_3 under the gravitational influence of the two primaries m_1 and m_2 that move in circular Keplerian orbit owing to their gravitational interaction. Let the position vector of m_3 be defined as $r = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathrm{T}}$ in the synodic frame with the barycenter as the origin and let r_1, r_2 be the position vectors of the spacecraft from the Sun and Earth respectively. Fig. 4.1 illustrates the geometry of the Sun-Earth three body system. The dimensionless equations of motion are expressed as [166]

$$\ddot{x} = 2\dot{y} + x - (1-\mu)\frac{x+\mu}{r_1^3} - \mu\frac{x-1+\mu}{r_2^3} \tag{4.1}$$

$$\ddot{y} = -2\dot{x} + y - (1-\mu)\frac{y}{r_1^3} - \mu\frac{y}{r_2^3} \tag{4.2}$$

$$\ddot{z} = -(1-\mu)\frac{z}{r_1^3} - \mu\frac{z}{r_2^3} \tag{4.3}$$

Here,

$$r_1^2 = (x+\mu)^2 + y^2 + z^2, \tag{4.4}$$

$$r_2^2 = (x-1+\mu)^2 + y^2 + z^2 \tag{4.5}$$

and where the gravitational parameter $\mu = \frac{m_2}{m_1+m_2}$, the distance between the primaries is normalized to 1, while the new time unit is taken as the product of the angular velocity and time. To address the nonlinear terms in the equations above, we translate the origin to the libration point and expand the nonlinear terms as functions of Legendre polynomials P_n which is defined as

$$\frac{1}{\sqrt{(x-a)^2 + (y-b)^2 + (z-c)^2}} = \frac{1}{D}\sum_{n=0}^{\infty}(\frac{\rho}{D})^n P_n(\frac{ax+by+cz}{D\rho}) \tag{4.6}$$

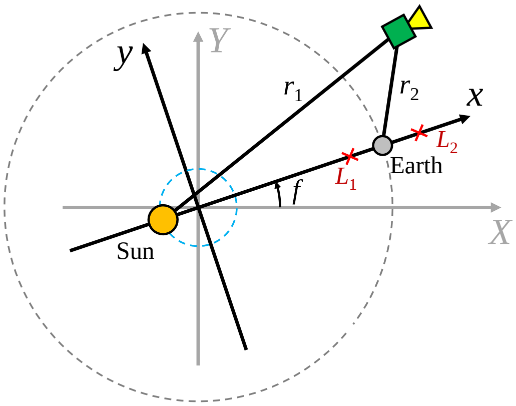


Figure 4.1: Geometry of the three body system

where $D^2 = a^2+b^2+c^2$ and $\rho^2 = x^2+y^2+z^2$. Define γ as the distance between L1 and the smaller primary. We can express the CRTBP equations in a coordinate frame centered at L1 using Legendre polynomials as [72]

$$\ddot{x} - 2\dot{y} - (1+2c_2)x = \frac{\partial}{\partial x}\sum_{n\geq 3} c_n\rho^n P_n(\frac{x}{\rho}) \tag{4.7}$$

$$\ddot{y} + 2\dot{x} + (c_2-1)y = \frac{\partial}{\partial y}\sum_{n\geq 3} c_n\rho^n P_n(\frac{x}{\rho}) \tag{4.8}$$

$$\ddot{z} + c_2 z = \frac{\partial}{\partial z}\sum_{n\geq 3} c_n\rho^n P_n(\frac{x}{\rho}) \tag{4.9}$$

where $c_n = \gamma^{-3}(\mu + (-1)^n(1-\mu)(\frac{\gamma}{1-\gamma})^{n+1})$. A third order approximation was used by Richardson [143] to compute periodic orbit solutions in the vicinity of collinear libration points. The approximated nonlinear equations are expressed as

$$\ddot{x} - 2\dot{y} - (1+2c_2)x = \frac{3}{2}c_3(2x^2-y^2-z^2) + 2c_4x(2x^2-3y^2-3z^2) + \mathcal{O}(4) \tag{4.10}$$

$$\ddot{y} + 2\dot{x} + (c_2-1)y = -3c_3xy - \frac{3}{2}c_4(4x^2-y^2-z^2)y + \mathcal{O}(4) \tag{4.11}$$

$$\ddot{z} + c_2 z = -3c_3xz - \frac{3}{2}c_4z(4x^2-y^2-z^2) + \mathcal{O}(4) \tag{4.12}$$

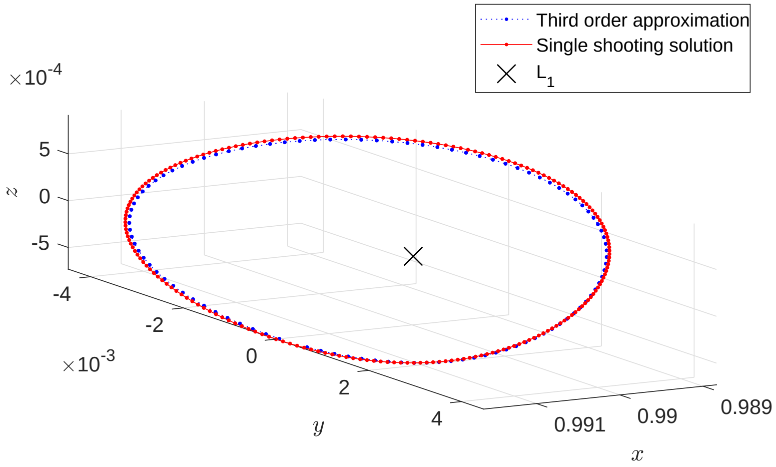


Figure 4.2: Nominal orbit obtained using single shooting method and analytical third order approximation

Using the equations shown above, Richardson developed a third-order periodic solution using the Lindstedt-Poincarè perturbation method. While such perturbation techniques prove to be qualitatively insightful, they are insufficient for dynamical analysis near the libration points. Analytical methods such as the one developed by Richardson are then combined by numerical techniques such as differential corrections to iteratively produce a periodic orbit. Fig. 4.2 illustrates a nominal Halo orbit for the Sun-Earth restricted three body system in dimension-less coordinates obtained using the single shooting method.

For the purpose of this study, Taylor expansions are used to approximate the nonlinear CRTBP model instead of Legendre expansions. This is done for two reasons: 1. Taylor expansions are more general and therefore applicable to other similar dynamical systems such as relative spacecraft dynamics. 2. In addition, it was found that Taylor expansions resulted in more faithful representation of the nonlinear model as compared to Legendre polynomial expansions. A second order Taylor expansion model is used in this work which is expressed as

$$\ddot{x} = 2\dot{y} + \left(\frac{3\mu}{2\gamma_L^4} - \frac{3\mu - 1}{2(\gamma_L - 1)^4}\right)\left(2x^2 - y^2 - z^2\right) + \left(\frac{2\mu}{\gamma_L^3} - \frac{2\mu - 1}{(\gamma_L - 1)^3}\right)x - \mu - \gamma_L \tag{4.13}$$

$$+ \frac{\mu}{\gamma_L^2} - \frac{\mu - 1}{(\gamma_L - 1)^2} + 1$$

$$\ddot{y} = -2\dot{x} - \left(\frac{3\mu}{\gamma_L^3} - \frac{3\mu - 1}{(\gamma_L - 1)^3}\right)xy + \left(1 - \frac{\mu}{\gamma_L^3} + \frac{\mu - 1}{(\gamma_L - 1)^3}\right)y \tag{4.14}$$

$$\ddot{z} = -\left(\frac{3\mu}{\gamma_L^3} - \frac{3\mu - 1}{(\gamma_L - 1)^3}\right)xz + \left(\frac{\mu - 1}{(\gamma_L - 1)^3} - \frac{\mu}{\gamma_L^3}\right)z \tag{4.15}$$

Due to the inherent instability of these orbits in a realistic n-body scenario with perturbations such as solar radiation pressure, active station-keeping in the form of continuous or impulsive control is often needed. Once a nominal orbit has been specified, the controller ensures that the spacecraft trajectory remains sufficiently close to the nominal orbit. In addition, instead of mitigating the effects of system nonlinearity by control, it may be economical to use a more accurate nonlinear model. In the next section, we propose the nonlinear polynomial model given by Eq. 13-15 (truncated at order two) for design of a station-keeping control strategy based on polynomial optimization based nonlinear MPC which are discussed in the next sections.

4.3 Polynomial Optimization

In Chapter 2, the methodology of sum-of-squares and semidefinite programming was introduced. In this section, the discussion on polynomial optimization is studied further. Consider the following optimization problem

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & x \in \mathcal{K} \end{aligned} \tag{4.16}$$

where $\mathcal{K} \coloneqq \{x : g_j(x) <= 0, j = 1, 2, 3..\}$ is a basic semi-algebraic set, i.e. a set formed by the intersection of finitely many polynomial inequalities and $g_j(x), f(x)$ are real multivariate polynomials. We define a polynomial $p(x)$ in $x \in \mathbb{R}^n$ as a finite combination of monomials, given by [131]

$$p(x) = \sum_{\alpha} c_\alpha x^\alpha = \sum_{\alpha} c_\alpha x_1^{\alpha_1} ... x_n^{\alpha_n}, \quad c_\alpha \in \mathbb{R}, \quad \alpha = (\alpha_1 ... \alpha_n), \quad \alpha_i \in \mathbb{N}_0. \tag{4.17}$$

A polynomial is denoted as sum of squares (SOS) if it can be represented as

$$p(x) = \sum_{j \in \mathcal{J}} \big(g_j(x)\big)^2 \tag{4.18}$$

Alternatively, a multivariate polynomial $p(x)$ with degree $2d$ and in n variables is a SOS if it can be represented as

$$p(x) = z^{\mathrm{T}} Q z \tag{4.19}$$

where $Q \succ 0$ and $z = [1, x_1, x_2 ... x_n, x_1 x_2 ... x_n^d]$ is the vector of monomials upto degree d. The optimization problem represented in Eq. 4.16 is non-convex and NP complete. The problem can be rewritten with lifted variables as

$$\begin{aligned} \underset{x \in \mathcal{K}, \lambda \in \mathbb{R}}{\text{maximize}} \quad & \lambda \\ \text{subject to} \quad & f(x) - \lambda \geq 0 \end{aligned} \tag{4.20}$$

Thus, we require the largest λ such that $f(x) - \lambda$ is non-negative over a semi-algebraic set $\mathcal{K}$. By replacing the non-negativity condition by positivity, tractable positivity certificates can be leveraged to solve this problem.

Therefore, a weaker formulation of Eq. 4.21 in terms of positivity, and in particular SOS (if $\mathcal{K}$ is compact) is described as

$$\begin{aligned} \underset{x\in\mathcal{K},\lambda\in\mathbb{R}}{\text{maximize}} \quad & \lambda \\ \text{subject to} \quad & f(x) - \lambda = s_0(x) + \sum_{j=1}^{m} s_j(x) g_j(x) \end{aligned} \tag{4.21}$$

where $s_0(x), s_j(x)$ are SOS with deg(s_0), deg$(s_j g_j) \leq 2t$, for any $t \geq$ deg$(p(x)/2)$, deg$(g_{j(x)}/2)$. The constraint in Eq. 4.21 is obtained from the positivity certificate proposed by Putinar given below

Lemma 1. *(Putinar's positivstellensatz [82]) Define the quadratic module generated by the family of polynomials g_j as Q_g expressed as*

$$Q(g) := \sigma_0 + \sum_{j=1}^{m} \sigma_j g_j \tag{4.22}$$

where $(\sigma)_{j=0}^{m}$ are SOS. Assume there exists $u \in Q_g$ such that the level set $\{x \in \mathbb{R}^n : u(x) \geq 0$ is compact. If $p(x) > 0$ on the set $\mathcal{K}$, then $p(x) \in Q_g$. Therefore, we can express

$$p(x) = s_0 + \sum_{j=0}^{m} s_j g_j \tag{4.23}$$

for some SOS polynomials $(s(x))_{j=0}^{m}$.

Denoting the problem in Eq. 4.21 as P$_{SOS}$, we note that this problem is in fact a semidefinite program which can be solved upto global optimality and that P$_{SOS}$ provides a lower bound on the original problem. Furthermore the bounds can be strengthened by increasing t. This is to say

$$P_{SOS}(t) \leq P_{SOS}(t+1) \leq P^*$$

where P^* is the optimal value of Eq. 4.21. Furthermore, the polynomial optimization problem shown above can also be formulated as an infinite dimensional linear optimization problem over the space of measures. Let $\mathbb{K}$ be a Borel subset of $\mathbb{R}^n$ and let $\mathcal{M}(\mathbb{K})$ be the space of finite signed Borel measures on $\mathbb{K}$, with the positive cone $\mathcal{M}(\mathbb{K})_+$ being the space of finite Borel measures μ on $\mathbb{K}$.

$$P* = \underset{x \in \mathbb{K}}{\text{minimize}} \quad p(x) = \underset{\mu \in \mathcal{M}(\mathbb{K})_+}{\text{minimize}} \quad \int p(x) d\mu \quad \text{subject to} \int_{\mathbb{K}} d\mu = 1 \tag{4.24}$$

The proof of Eq. 4.24 can be found in the book by Lasserre [82] and is not provided here. Denote $p(x) = \sum_\alpha p_\alpha x^\alpha = c^{\mathrm{T}}[x(2t)]$ such that $\alpha \leq 2t$, where $[x(2t)]$ is now a monomial vector. Each monomial $x^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} .. x_n^{\alpha_n}$ is modeled by a new variable y_α and $[x(2t)]$ is modeled as $(y_\alpha)_{\|\alpha\| \leq 2t}$.

The original polynomial optimization problem is then written in terms of the sequence of moments of measures as

$$\begin{aligned} \underset{x \in \mathcal{K}}{\text{minimize}} \quad & c^{\mathrm{T}} y \\ \text{subject to} \quad & y \in \text{conv}([x(2t)] : x \in \mathcal{K}) \end{aligned} \tag{4.25}$$

We model the positive semi-definite constraint $[x(t)][x(t)]^{\mathrm{T}} \succeq 0$ by $M_t(y) = (y_{\alpha+\beta})_{\|\alpha\|, \|\beta\| \leq t}$.

If $x \in K$, then the constraint $g_j(x)[x(t_j)][x(t_j)]^{\mathrm{T}} \geq 0$ is expressed by $M_{t_j}(y) = \sum_\gamma (g_{j\gamma} y_{\alpha+\beta+\gamma})_{\|\alpha\|, \|\beta\| \leq t_j})$ and where $t_j = t - \deg(g_j)$.

Finally, the dual-moment problem is expressed as [83]

$$\begin{aligned} \underset{y}{\text{maximize}} \quad & c^{\mathrm{T}} y \\ \text{subject to} \quad & M_t(y) \succeq 0 \\ & M_{t_j}(g_j y) \succeq 0, j = 1, 2..m \\ & y_0 = 1 \end{aligned} \tag{4.26}$$

The optimization problem expressed in Eq. 4.26 is solved sequentially with increasing t. Based on the rank equality condition for the moment matrices in the equation, the global optimality of the solution can be concluded. In the next section, we show how the optimal tracking control problem can be cast as a polynomial optimization and then solved in a receding horizon fashion as is done for MPC.

4.4 MPC Approach

MPC is a control methodology where a constrained open-loop optimal control problem is solved at each sampling instant over a finite time horizon [73]. However, only the first control action is implemented and the horizon is shifted with the new state used as the initial condition. This provides a feedback action, making the system robust to disturbances. Systems with linear dynamics and convex inequality constraints and performance index can be solved efficiently using standard convex optimization solvers. For nonlinear dynamical systems, MPC is more challenging since the resulting problem is non-convex and local optimum solutions are generally found. However, for the class of systems with polynomial vector fields, the global optimum of the non-convex MPC can be obtained [140].

Consider a constrained discrete-time nonlinear system given by

$$x_{k+1} = f(x_k, u_k), \quad x_k \in \mathcal{X}, u_k \in \mathcal{U} \tag{4.27}$$

where $x \in \mathbb{R}^n$ are the system states and $u \in \mathbb{R}^m$ are the control inputs. The function f is a real polynomial function in x and u with the equilibrium point at the origin. A polynomial vector field $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $f(x) = [f_1(x), ..., f_n(x)]^{\mathrm{T}}$ is a vector field with $f_i(x) \in \mathbb{R}[x]$ [139]. The sets $\mathcal{X}$ and $\mathcal{U}$ are assumed as semi-algebraic sets given by a union of a finite number of real polynomial equations and inequalities in x and u, respectively. The open-loop optimal control problem which is required to be solved at each instant k in a receding horizon manner is then expressed as

$$\begin{aligned} \underset{u_k}{\text{minimize}} \quad & J_N(x_k, u_k) \\ \text{subject to} \quad & x_{i+1|k} = f(x_{i|k}, u_{i|k}), \quad x_{k|k} = x_k \\ & u_{i|k} \in \mathcal{U}, \quad x_{i|k} \in \mathcal{X} \\ & x_{k+N} \in \mathcal{X}_f, \quad i = k, k+1....k+N-1 \end{aligned} \tag{4.28}$$

where, $x_{i|k}$ is the predicted state at instant $i > k$, the cost function $J_N(x_k, u_k) = \sum_{i=k}^{k+N-1} l(x_{i|k}, u_{i|k}) + F(x_{k+N|k})$ and $N > 0$ is the control horizon. Under certain conditions on the terminal cost $F(x_{k+N|k})$, and terminal set $\mathcal{X}_f$, the asymptotic stability and recursive feasibility of the closed-loop system can be determined. This is not discussed here for brevity.

Under the assumption that the cost function described above can be expressed as a polynomial map and the terminal set is described via a polynomial inequality, we now reformulate the nonlinear MPC problem as a polynomial optimization. The predicted states are expressed in compact form as

$$\zeta = [x_{k+1|k}, x_{k+2|k}, ...x_{k+N|k}]^{\mathrm{T}} \tag{4.29}$$

Using a recursive relation, the system is then described via the control $\nu = [u_{k|k}, ...u_{k+N-1|k}]^{\mathrm{T}}$ and the state x_k as

$$\zeta = F(\nu, x_k) \tag{4.30}$$

Note that $F_i, i = 1...N$, are now multivariate polynomial functions in ν. Since we already define the inequality constraints as polynomial and assume a polynomial cost function, we obtain the following theorem

Theorem 2. *[139] The finite horizon optimal control problem defined in Eq. 15* *can be formulated as a polynomial optimization problem of the form*

$$\underset{\nu\in\mathcal{K}}{\mathit{minimize}} \quad p_0(\nu) \tag{4.31}$$

for discrete time polynomial systems, if $\mathcal{K} = \{\nu \in \mathbb{R}^{m.N} : p_i(\nu) \geq 0, i = 1, 2...2(n+m)N+1\}$*, is a compact set described by multivariate polynomial inequalities* $p_i(\nu) \in \mathbb{R}[\nu], i = 1, 2...2(n+m)N+1$.

Having cast the open-loop optimal control problem as a polynomial optimization, relaxation methods based on the moment-SOS approach can be used to solve the MPC problem. The tracking MPC problem studied here uses quadratic objective for the tracking error and the control effort. The formulation is expressed as [112]

$$\begin{aligned}
\underset{u}{\text{minimize}} \quad & \sum_{i=1}^{N_p-1} (x_{i|k} - x^d_{i|k})^{\mathrm{T}} Q (x_{i|k} - x^d_{i|k}) + u^{\mathrm{T}}_{i|k} R u_{i|k} + e^{\mathrm{T}}_{N_p|k} P e_{N_p|k} \\
\text{subject to} \quad & x_{i+1|k} = f(x_{i|k}, u_{i|k}) \\
& u_{i|k} \in \mathcal{U} \\
& x_{i|k} \in \mathcal{X} \\
& x_{k+N} \in \mathcal{X}_f, \quad i = k, k+1....k+N-1
\end{aligned} \tag{4.32}$$

where $x^d_{i|k}$ is the desired trajectory at instant $i > k$ and $e_{N_p|k} = x_{N_p|k} - x^d_{N_p|k}$. Both terminal constraint set $\mathcal{X}_f$ and terminal cost P are used in the numerical simulations to ensure recursive feasibility and stability. The equations of motion are obtained by truncating the dynamics near L1 upto second order and discretized using a fourth order Adam-Bashworth method. Once the control input is computed using the formulation expressed in Eq. 4.32, a change of coordinates is carried out to express the acceleration in the barycenteric frame, wherein the first control input is implemented on the nonlinear CRTBP equations given in Eqs.

Parameter	Value
Q	$\text{diag}(\begin{bmatrix}10^6 & 10^6 & 10^6 & 1 & 1 & 1\end{bmatrix})$
P	Discrete algebraic Riccati solution
R	$\text{diag}(\begin{bmatrix}10^3 & 10^3 & 10^3\end{bmatrix}$
N_p	15
N	60

Table 4.1: Polynomial MPC parameters

1-3 and the corresponding new state is obtained, which after mapping backing to the L1 coordinate frame becomes to new initial state and the problem is solved in a receding horizon fashion. The performance of the approach is examined in the next section.

4.5 Numerical results

In this section, the proposed polynomial MPC approach is evaluated for station-keeping of a Halo orbit near the L1 libration point in the Sun-Earth restricted three body system. The amplitude (A_z) of the halo orbit is taken as $150,000$ km. The nominal orbit is generated using the third order approximation and refined using the single shooting method as discussed in Section 4.1.

The total horizon length is approximately taken as one revolution for the Halo orbit, where it is parameterized in N equal spacings of true anomaly with a step size of 0.051. A key issue with polynomial MPC is the choice of the prediction horizon N_p. If N_p is taken as too large, the number of SDP variables grow exponentially, thereby rendering the problem computationally intractable. The parameters used in the tracking MPC problem are provided in Table 4.1. A very large initial orbital insertion error of 40000 km is considered in the x direction Fig. 4.3 shows the controlled orbit obtained in the coordinate frame centered at L1. The projection of the computed trajectory in XY and XZ plane is displayed in Fig. 4.4. Good tracking accuracy can be observed from the trajectory computed

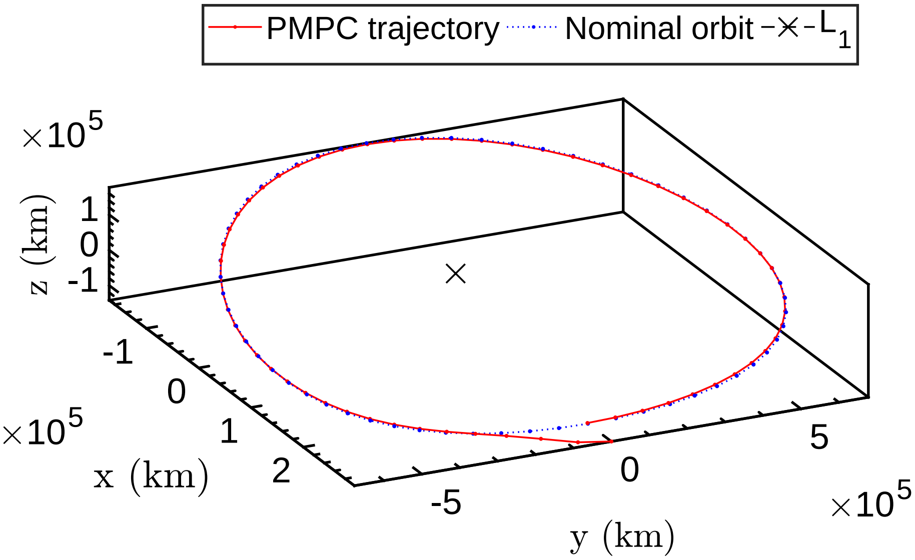


Figure 4.3: Three-dimensional tracking trajectory

using the polynomial MPC. The corresponding control input is illustrated in Fig. 4.5.

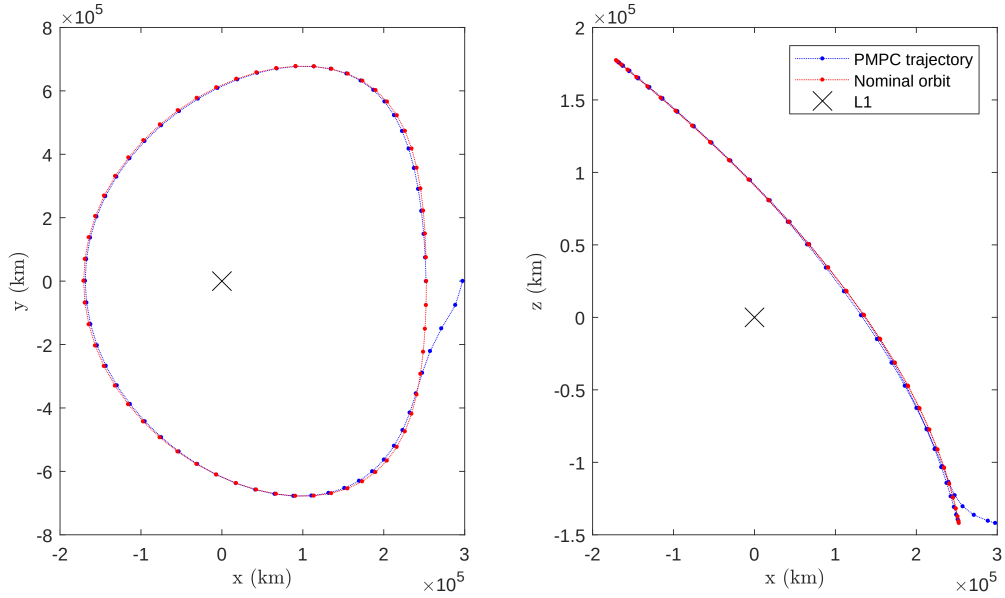


Figure 4.4: Two dimensional project of the tracking trajectory in XY and XZ plane

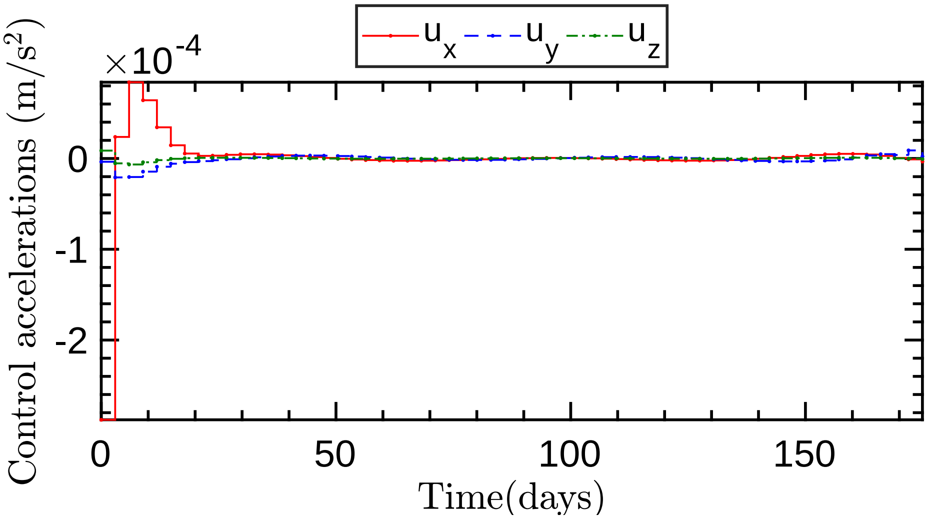


Figure 4.5: Control accelerations obtained using polynomial MPC

As a comparison, the nominal nonlinear CRTBP model was used in the MPC framework with the same parameter. In addition, linear quadratic control (LQR) and linear MPC models were also used and the results are compared with the polynomial MPC (PMPC) scheme, solved globally using Gloptipoly and locally using IPOPT in Table 4.2

Scheme	ΔV (ms^{-2})	Solver
PMPC (global)	7.94×10^{-4}	Gloptipoly with Mosek
Nominal NMPC	6.81×10^{-4}	IPOPT
PMPC (local)	9.02×10^{-4}	IPOPT
LQR	7.01×10^{-4}	NA
LMPC	9.78×10^{-4}	Gurobi

Table 4.2: Comparison of ΔV required for tracking using different control schemes

It is observed that nominal nonlinear MPC has lower ΔV compared to that of global PMPC. This is expected as the PMPC uses an approximate model to compute the control inputs. LQR is also shown to have a lower ΔV, however, LQR cannot accommodate state and control constraints.

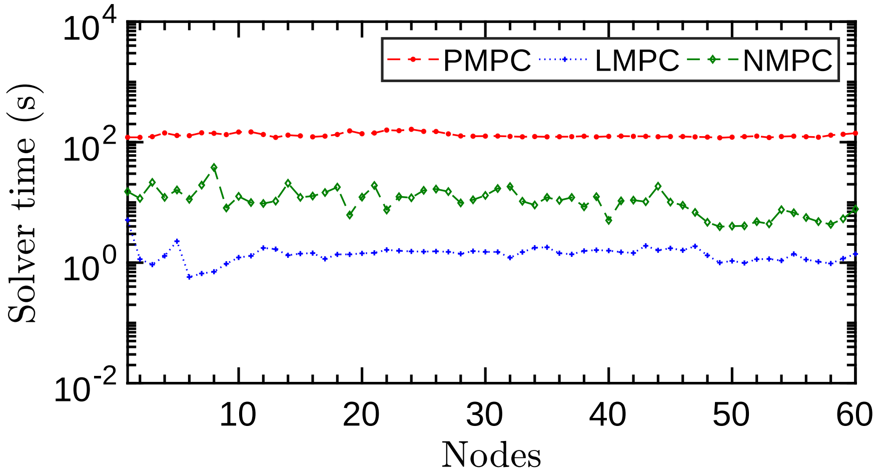


Figure 4.6: Computation time taken per MPC step

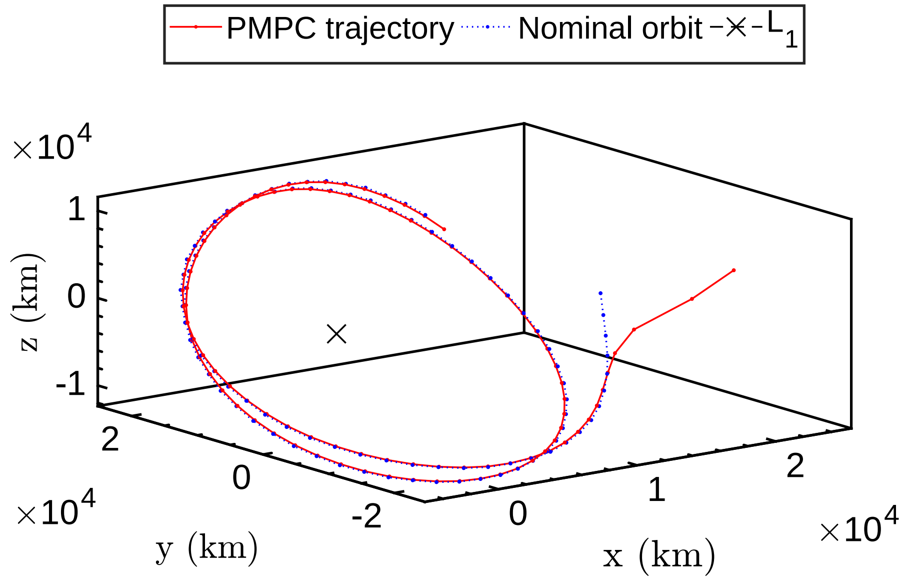


Figure 4.7: Three-dimensional Lissajous trajectory

The polynomial MPC method implemented on a PC running Windows 10, with an Intel Core i5 CPU @ 2.90 GHz and 8 GB RAM. We use the YALMIP [100] optimization environment on MATLAB with the Gloptipoly [58] toolbox for global polynomial optimization which in turn uses the Mosek [117] SDP solver. The computation time by these methods is also recorded and is illustrated in Fig. 4.6

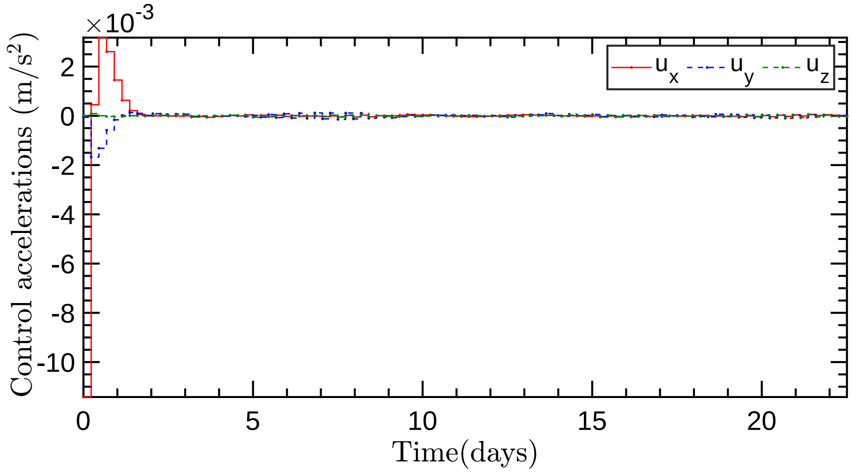


Figure 4.8: Control input generated using PMPC

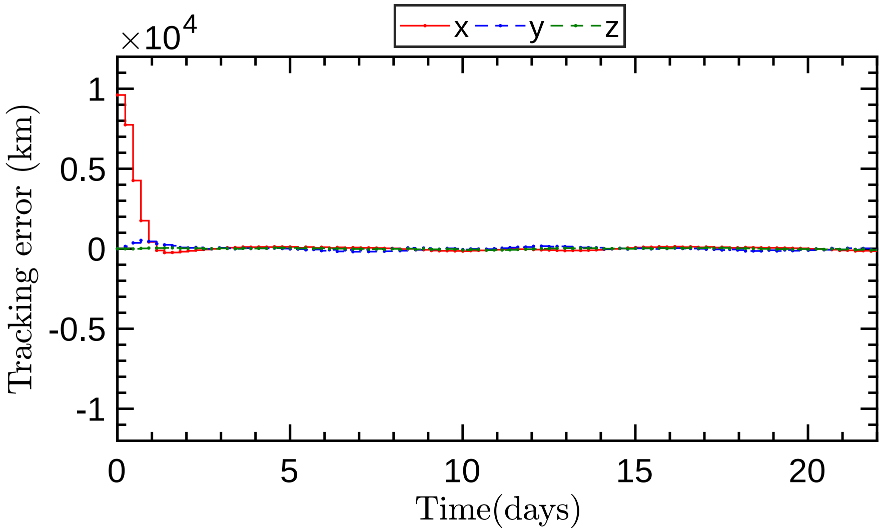


Figure 4.9: Tracking error profile

Next, the PMPC method is studied for trajectory tracking of Lissajous orbit in the Earth-Moon restricted three body problem. A nominal orbit is generated and corrected using multiple shooting. An insertion error of 9500 km is considered in the x-direction. The total horizon length is taken as 100 which corresponds to 22 days. The prediction horizon considered is 10, corresponding to 5.45 hours. Fig. 4.7 shows the three-dimensional tracking trajectory generated with the PMPC method. The control inputs generated by PMPC is shown in Fig 4.8. The tracking error is illustrated in Fig. 4.9. The tracking error is seen to converge within three days.

N_p	2	6	10	15
PMPC ΔV	0.059	0.03212	0.03206	0.0318
NMPC ΔV	DNC	0.03219	0.03212	0.032
LMPC ΔV	0.117	0.0475	0.047	0.0465
LQR ΔV	0.12	0.12	0.12	0.12

Table 4.3: Control comparisons for Lissajous orbit tracking

In Table 4.3, the ΔV required for tracking is computed for a range of different prediction horizons. The PMPC approach is shown to have lower $DeltaV$ values than linear MPC, LQR and nonlinear MPC. Moreover, for a very small prediction horizon, the nonlinear MPC failed to converge to a feasible solution. Although PMPC obtains similar control budget as compared to nonlinear MPC, it does not require any warm start and initial guesses. Furthermore, while the tracking performance of polynomial MPC is good, its applicability is still limited to small prediction horizons. For larger horizons, the problem size increases exponentially and while theoretically the optimization problem is solvable in polynomial time, it is computationally demanding. In cases where the problem size is big, parametric explicit MPC formulations using algebraic geometry maybe more desirable [45].

Chapter 5

Robust controller synthesis for polynomial systems using convex optimization

In the preceding chapters, the problem of real-time, 'globally' optimal trajectory planning and control has been discussed. However, a key aspect of robustness to disturbances is not considered. In this chapter, a convex optimization based approach is studied for the synthesis of robust disturbance observer based control. Convex optimization based stability analysis and control methods have received widespread attention, especially for polynomial systems. In particular, the moment based Lasserre hierarchy [83] and its dual the sums-of-squares method for solving global non-convex polynomial optimization problems using convex programming methods have been applied to a wide range of problems such as polynomial model predictive control [53, 112], occupation measure based optimal control [81], and robust stability analysis [67]. Robust control of linear and nonlinear systems has also been addressed using Linear Matrix Inequalities (LMI) based approaches [149]. More recently, input constrained robust control of polynomial systems using polynomial optimization is addressed in [138]. While several robust control methods exist, most of them are based on the worst case approach which leads to a conservative control design. Moreover, for robust optimization based control, constraints on the convexity of the disturbance set may be typically required.

Disturbance observer based control [30] enables a more flexible approach in dealing with disturbance attenuation. In this framework, a nominal controller is first designed under the assumption that no disturbances act on the system, then disturbance compensation is added to counteract the influence of the perturbations acting on the system. The feed-forward compensation requires an estimate of the disturbance which is provided through a suitably designed linear or nonlinear disturbance observer (NDOB). A distinct advantage of this approach is the so called 'patch'feature [90] wherein after the observer synthesis for the given system, it can be added to different control systems without requiring a completely new control design and formal verification. Furthermore, no assumptions such as convexity are needed for the disturbance input apart from its boundedness.

The design of the NDOB depends on the dynamics of the nonlinear system studied. In particular, the NDOB gain matrix explicitly depends on the time-varying disturbance-to-state mapping to ensure convergence of the disturbance estimation error. Due to nonlinearity of the mapping, design of such DOBs is non-trivial and indeed can be extremely challenging. For systems perturbed by harmonic disturbances with known frequency, [29] provides a DOB synthesis solution using differential geometric methods. However, most of the current methods for DOB synthesis are tailored to particular problems. In [87] and [165], the DOB synthesis is made simpler by considering the upper bound of the mapping with the disturbance. However, this approach is sub-optimal and in general may not work for general nonlinear systems.

For polynomial systems, the disturbance to state mapping is typically a polynomial matrix or vector field. Thus, it is possible to formulate the DOB synthesis problem as a polynomial optimization and leverage sum-of-squares programming for its solution. Although physical systems in general are not polynomial,as a consequence of the Weierstrass approximation theorem, any continuous function in a closed domain can be uniformly approximated using polynomial functions [145]. Therefore, we focus on the DOB synthesis for general polynomial systems in this work. Our main contributions of the proposed approach are two fold: A systematic framework is provided for design of NDOBs for systems with polynomial vector or matrix fields. Using sum-of-squares programming, it is demonstrated how to design exponentially convergent NDOBs in addition to providing solutions to constrain the gain magnitudes so that chattering in the control input can be avoided. Secondly, for systems with mismatched disturbances where asymptotic stabilization is typically not possible, a suitable compensation gain matrix is proposed and the input-to-state stability of the system is verified using sum-of-squares (SOS) programming.

5.1 Background

Consider the class of affine nonlinear polynomial systems defined by

$$\dot{x} = f(x) + g_1(x)u + g_2(x)d \tag{5.1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $d \in \mathbb{R}^m$. d is the disturbance vector which may be constant or time-varying. It is assumed that $f(x)$, $g_1(x)$, and $g_2(x)$ are polynomial vector or matrix fields and hence are smooth functions of x. A general design framework for NDOB based control is to separately design a nonlinear feedback controller and an observer to estimate the disturbance. Finally, the NDOB is integrated as a feed-forward to the controller to attenuate the disturbances. Suppose, that the disturbance is bounded, slowly-varying but unknown. A suitable structure of NDOB is expressed as [29]

$$\dot{h} = -l(x)g_2(x)h - l(x)[g_2(x)p(x) + f(x) + g_1(x)u] \tag{5.2}$$

$$\hat{d} = h + p(x) \tag{5.3}$$

where $h \in \mathbb{R}^m$ is the internal state of the observer, $\hat{d}$ is the estimated disturbance, and $p(x)$ is a nonlinear function to be designed. The observer gain matrix $l(x)$ is defined as

$$l(x) = \frac{\partial p(x)}{\partial x} \tag{5.4}$$

Let the disturbance estimation error be denoted as e_d, using (5.2) and (5.3), the error dynamics is given by

$$\dot{e}_d = \dot{\hat{d}} - \dot{d} = -l(x)g_2(x)e_d, \text{ for } \dot{d} \approx 0 \tag{5.5}$$

It is seen that for a suitable choice of $l(x)$, the estimation error dynamics can be made asymptotically stable. For exponential stability, the design of $l(x)$ should satisfy

$$-e_d^{\mathrm{T}} l(x)g_2(x)e_d \leq -\gamma \left\| e_d \right\|_2^2 \tag{5.6}$$

where γ is a positive constant. In most applications studied under DOB based control, the choice of $l(x)$ is made simple since the disturbance to state mapping given by $g_2(x)$ is assumed to be an identity matrix. Therefore, finding $l(x)$ which is positive-definite renders the error dynamics stable. For problems with special structure, such as Euler-Lagrange systems, a systematic choice of $l(x)$ was proposed in [115]. Assuming full column rank for $g_2(x)$, one suitable parameterization for $l(x)$ is

$$l(x) = \alpha g_2^{\dagger}(x) \tag{5.7}$$

where $(.)^{\dagger}$ denotes the left inverse of the matrix and $\alpha > 0$. Note, however, that this choice may not be optimal and as such, depends on the system in question which may not satisfy the rank condition imposed on $g_2(x)$. The constraints originating in (5.6), restrict the usage of NDOBs. It also paves the way for investigating alternate methods for design of NDOBs that are general enough for a class of nonlinear systems. Another significant challenge which restricts the usage of NDOBs is the so called 'matching'condition. Under the matching condition, for nonlinear systems, the disturbance and control must enter the system through the same channel, i.e. $g_1(x) = g_2(x)$. In such cases, the controller takes the form $u(x) = u_n(x) - \hat{d}$, where u_n is the feedback controller for the nominal unperturbed system. For systems not satisfying this requirement, differential geometric techniques are proposed in [189] to attenuate disturbance effects from system output channels. However, in this work, we focus on finding an exponentially convergent NDOB and a disturbance compensation gain function that renders the state and disturbance estimation error dynamics input-to-state stable (ISS) using SOS programming.

5.2 Disturbance Observer and Control Design

Assumption 1. *The external disturbance d is unknown, slowly-varying, $\dot{d} \approx 0$, but bounded.*

For a general nonlinear function $g_2(x)$, the design of $p(x)$ satisfying(5.5) is computational challenging and often depends on the system studied. Except for special cases such as linear systems or when $g_2(x)$ does not depend on the state, NDOB synthesis becomes problem dependent.

Proposition 3. *Consider the class of nonlinear matrix function $g_2(x) \in \mathbb{R}^{n\times m}$ that are restricted to be polynomial functions of the state x. The NDOB gain matrix $l(x)$ and auxiliary vector $p(x)$ synthesis can be posed as a polynomial optimization problem and solved using SOS relaxations [110].*

Proof: Define $p_i(x) = \Sigma_{0\leq j+k\leq d} c_{ijk} x^j x^k$, $\forall i = 1, 2, 3..l$ as a generic d-order polynomial to be designed such that $l(x) = \frac{\partial p(x)}{\partial x}$. The synthesis of $p(x)$ is cast as a polynomial optimization problem given by

$$\begin{aligned} \underset{c_{ijk}}{\text{minimize}} \quad & \|\mathbf{c}\|_1 \\ \text{subject to} \quad & l(x)g_2(x) - \upsilon\mathbf{I} \succeq 0 \end{aligned} \tag{5.8}$$

where $\mathbf{I}$ is an identity matrix of appropriate size, υ is a small but positive scalar, and $l(x)g_2(x) : \mathbb{R}^n \to \mathcal{S}_m$, is a polynomial mapping, i.e., each entry $l(x)g_2(x)[ij]$ of the $m \times m$ symmetric matrix is a polynomial in the indeterminate x, $\mathbf{c}$ is the vector of coefficients c_{ijk}. This is indeed a specific instance of the broader class of polynomial optimization problems where the matrix constraint in Eq. (5.8) defines a basic semi-algebraic set $\mathcal{K} := \{x : q_i(x) > 0, i = 1, 2, 3..\}$ i.e., a set formed by the intersection of finitely many polynomial inequalities. Let $r(t, x) = \det\big(t\mathbf{I} - (l(x)g_2(x) - \upsilon\mathbf{I})\big)$ be the characteristic polynomial of $l(x)g_2(x) - \upsilon\mathbf{I}$. Rewriting $r(t, x)$ in the form [57]

$$r(t, x) = t^l + \Sigma_{i=1}^{l}(-1)^i q_i(x)t^{m-i} \tag{5.9}$$

It is proved in [79, 77] that all the roots of $r(t, x)$ are positive, i.e, $l(x)g_2(x) - \upsilon\mathbf{I} \succeq 0$, if and only if $q_i(x) \geq 0, \forall i = 1, 2, 3..l$. Having converted Eq. (5.8) into a scalar polynomial optimization problem, results from SOS relaxations [83] can be applied by restricting $q_i(x) \in \Sigma_x$ and iteratively solving the resulting semidefinite programs (SDP) to compute the observer gain matrix $l(x)$ and the corresponding decision variables c_{ijk}. □ Notice that $\upsilon\mathbf{I}$ is added to ensure that the eigenvalues of $l(x)g_2(x)$ are strictly positive. The problem described in Eq. (5.8) aims at finding the smallest gain, and thereby the smallest coefficients such that the NDOB is exponentially convergent. This is desirable since arbitrarily large gains may result in undesirable chattering in the control inputs. Using the standard lifting methods, the ℓ_1 norm problem described above can be converted to the standard SDP formulation using lifted coefficients [25].

Theorem 3. *Consider the system expressed in (5.5). If the observer gain matrix is designed using (Eq. 5.8) and the disturbance is assumed to be slowly varying such that $\dot{d} \approx 0$, the NDOB can track the disturbance d and the estimation error converges to the origin*

Proof: Consider the Lyapunov function

$$V = \frac{1}{2} e_d^{\mathrm{T}} e_d \tag{5.10}$$

where $e_d = \hat{d} - d$ is the disturbance tracking error. The time derivative of V is

$$\dot{V} = e_d^{\mathrm{T}} (\dot{e}_d) \approx -e_d^{\mathrm{T}} \dot{\hat{d}} = -e_d^{\mathrm{T}} (\dot{h} + \frac{\partial p(x)}{\partial x} \dot{x}) \tag{5.11}$$

$$\begin{aligned} \dot{V} = -e_d^{\mathrm{T}} \Big(& - l(x) g_2(x) h - l(x) [g_2(x) p(x) + \\ & f(x) + g_1(x) u] + l(x) [f(x) + g_1(x) u + g_2(x) d] \Big) \end{aligned} \tag{5.12}$$

Using (5.3), and substituting the disturbance term d, we get

$$\dot{V} = -e_d^{\mathrm{T}} \Big(l(x) g_2(x) \hat{d} - l(x) (\dot{x} - f(x) - g_1(x) u) \Big) \tag{5.13}$$

$$\dot{V} = -e_d^{\mathrm{T}} \Big(l(x) g_2(x) \Big) e_d < 0 \text{ since } l(x) g_2(x) \succ 0 \tag{5.14}$$

which implies that the disturbance estimate approaches to disturbance d globally exponentially. □ Although the NDOB synthesized from (Eq. 5.8) is exponentially convergent, it cannot directly be applied to the system since $g_1(x) \neq g_2(x)$, i.e, the uncertainties are mismatched. Similar to the approach in [30], we seek a composite control law given by

$$u(x) = \chi(x) + \zeta(x)\hat{d} \tag{5.15}$$

where $\chi(x) : \mathbb{R}^n \to \mathbb{R}^m$ is the feedback control law without considering disturbances on the system, and $\zeta(x) : \mathbb{R}^n \to \mathbb{R}^{m\times m}$ is the disturbance compensation gain matrix. Using the composite control law, (5.15) in (5.1), along with the error dynamics, we get

$$\dot{x} = f(x) + g_1(x)\chi(x) + g_1(x)\zeta(x)(d + e_d) + g_2(x)d \tag{5.16}$$

$$\dot{e}_d = -l(x)g_2(x)e_d \tag{5.17}$$

Consider the closed loop system with the augmented state $z = [e_d, x]^{\mathrm{T}}$, the augmented state dynamics are expressed as

$$\begin{aligned}\dot{z} = &\begin{bmatrix} -l(x)g_2(x)e_d \\ f(x) + g_1(x)\chi(x) + g_1(x)\zeta(x)e_d \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{0} \\ g_1(x)\zeta(x) + g_2(x) \end{bmatrix} d \end{aligned} \tag{5.18}$$

where $\mathbf{0}$ is a matrix of zeros of appropriate dimensions.

Assumption 2. *The origin of the augmented closed loop system is globally asymptotically stable in the absence of disturbances.*

Due to the uncertainty mismatch, it is unlikely to achieve asymptotic stability in presence of disturbances. Therefore, we seek a disturbance compensation gain $\zeta(x)$ that renders the closed loop system (5.18) input-to-state stable (ISS). Let (5.18) can be represented as $\dot{z} = \vartheta(z, d)$ with $\vartheta(0, 0) = 0$. This system is said to be ISS if there exists a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ function γ such that for any initial state z_0 and bounded input d, the solution $z(t)$ exists for all

$t \in [0, \infty]$ and satisfies [52]

$$\|z(t)\| \leq \max\{\beta(\|z_0\|, t), \gamma(\sup_{\tau \in [0,t]} \|d(\tau)\|)\} \tag{5.19}$$

Definition 2. *[159] A continuously differentiable function $V : \mathbb{R}^n \to [0, \infty]$ is called an ISS-Lyapunov function for the system if there exists $\mathcal{K}_\infty$ functions α_1, α_2, α_3, and σ such that*

$$\alpha_1(\|z\|) \leq V(z) \leq \alpha_2(\|z\|), \quad \textit{for any } z \in \mathbb{R}^{n+m} \tag{5.20}$$

and

$$\frac{\partial V(z)}{\partial z}\vartheta(z, d) \leq -\alpha_3(\|z\|) + \sigma(\|d\|), \ \forall (z, d) \in \mathbb{R}^{n+m} \times \mathbb{R}^m \tag{5.21}$$

Numerically tractable ISS conditions using SOS programming for general polynomial systems, cascade connections, and feedback interconnected systems is proposed in [62, 63].

Proposition 4. *Given the feedback control $\chi(x)$, the NDOB gain matrix $l(x)$, and corresponding auxiliary vector $p(x)$, computed using (Eq. 5.8), for a suitable polynomial matrix $\zeta(x)$, the closed loop system (5.18) can be rendered ISS and the property can be verified using SOS programming.*

Proof: We seek a polynomial matrix $\zeta(x)$ such that the system (5.18) is ISS with input d. For a matrix $\zeta(x) \in \mathbb{R}^{m \times m}$ that is non-singular for all x, if there exists polynomials $\alpha_i \in \mathcal{K}_\infty$, Lyapunov function $V \in \mathcal{K}_\infty$, SOS polynomials Σ_x

such that

$$V(z) - \alpha_2(\|z\|) = \Sigma_z \tag{5.22}$$

$$- \alpha_1(\|z\|) + V(z) = \Sigma_z \tag{5.23}$$

$$- \frac{\partial V(z)}{\partial z}\vartheta(z,d) - \alpha_3(\|z\|) + \sigma(\|d\|) = \Sigma_{z,d} \tag{5.24}$$

then the system is ISS [62]. This is due to the well-known property that the system $\dot{z} = h(z,d)$ is ISS if and only if it admits an ISS-Lyapunov function [159], we can conclude that the system is ISS for a given $\zeta(x)$ such that (5.22-5.24) are satisfied. □ An immediate consequence of the proposition is the structure of the gain matrix $\zeta(x)$ and the construction of the $\mathcal{K}_\infty$ polynomials. Similar to the approach described in [62], it is sufficient to consider real univariate polynomials of even degree given by

$$\alpha(s) = \sum_{i=1}^{M} c_i s^{2i} \tag{5.25}$$

where c_i are the coefficients. The polynomial $\alpha(s) \in \mathcal{K}_\infty$ if and only if [63]

$$s\frac{d\alpha(s)}{ds} \geq 0 \; \forall s \in \mathbb{R} \tag{5.26}$$

For numerical ISS-Lyapunov analysis, (5.26) must be imposed for all $\alpha_i(s), \sigma(s)$. The compensation gain matrix $\zeta(x)$ should be designed such that it is nonsingular for all x and that the ISS conditions for the augmented system are satisfied. To ensure non-singularity, it is sufficient to consider a strictly positive definite compensation gain. A suitable form for $\zeta(x)$ can be

$$\zeta(x) = \text{diag}\Bigg([\sum_{i=0}^{m} c_i x^{2i}, \sum_{j=0}^{m} c_j x^{2j} ... \sum_{k=0}^{m} c_k x^{2k}]\Bigg) \tag{5.27}$$

Note that the degree of the monomial starts from 0 to ensure that $\zeta(x)$ is positive definite and thus non-singular even if $x = 0$.

5.3 Numerical Results

5.3.1 Disturbance observer design for matched disturbances

Consider the system

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_1 x_2 \\ -x_1^2 \end{bmatrix} + \begin{bmatrix} 1 & x_1 \\ x_2 & x_2^2 + 1 \end{bmatrix} (u + d) \tag{5.28}$$

where $u \in \mathbb{R}^2$ is the control input and $d \in \mathbb{R}^2$ is the disturbance. Since $g_2(x) = g_1(x)$, the disturbances act in the same channel as the control inputs. We seek a disturbance gain matrix such that the disturbance estimation error is exponentially convergent. The nonlinearity in the dynamics is quadratic. Therefore, a cubic monomial is proposed for the design of auxiliary vector $p_i(x) = \mathbf{c}_i^{\mathrm{T}}[1, x_1, x_1 x_2 .... x_2^3]$ for $i = 1, 2$ so that the observer gain is an even degree polynomial matrix. The decision variables are the coefficient vectors $c_i \in \mathbb{R}^{10}$. The gain synthesis is posed as an SOS feasibility problem and solved using the MATLAB toolbox SOSTOOLS [137] with Sedumi [161] as the underlying SDP solver. Throughout this section, monomials with coefficients less than 10^{-5} have been omitted. The auxiliary vector and the gain matrix are obtained as

$$p(x) = \begin{bmatrix} 0.203x_2^2 + 0.603x_1 \\ 0.301x_1^2 + 0.135x_2^3 + 0.406x_2 \end{bmatrix} \tag{5.29}$$

$$l(x) = \begin{bmatrix} 0.603 & 0.406x_2 \\ 0.603x_1 & 0.406x_2^2 + 0.406 \end{bmatrix} \tag{5.30}$$

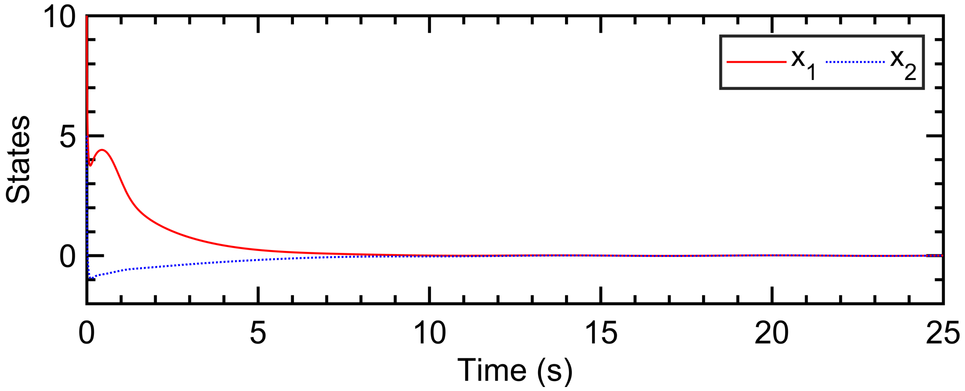


Figure 5.1: State trajectory.

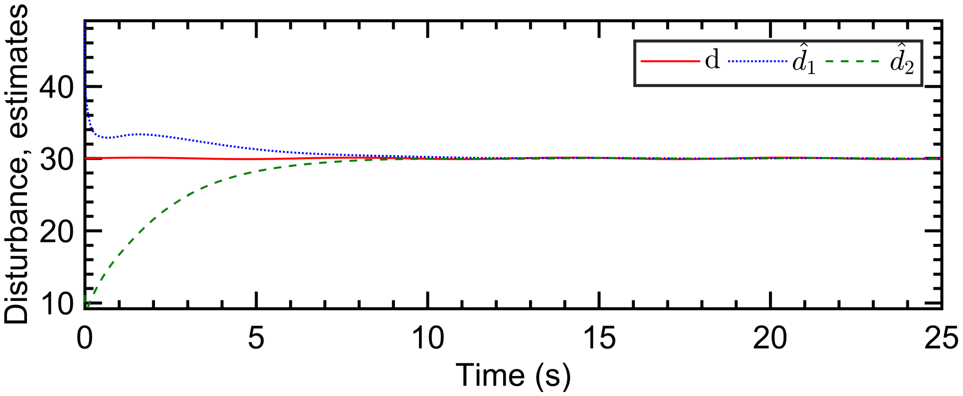


Figure 5.2: Disturbance and its estimation profiles.

A large but slowly varying disturbance profile is assumed, given as $d_i(t) = 30 + 0.5\sin(t)$ for $i = 1, 2$. A regulation problem is studied using the computed NDOB gains and a simple proportional controller given as $u = -kx$, with $k = \text{diag}([10, 10])$. The use of this simple controller is deliberate, since it allows us to assess the performance of the NDOB in presence of large and potentially destabilizing perturbations especially when the nominal controller has no inherent robustness capabilities.

The initial state is taken as $x_0 = [10, 5]$, while the initial value for the internal state of the observer given by $z_0 = [0, 0]$. Fig. 5.1 shows the profiles of the state with time. Rapid convergence to the origin is observed for both states. In Figs. 5.2

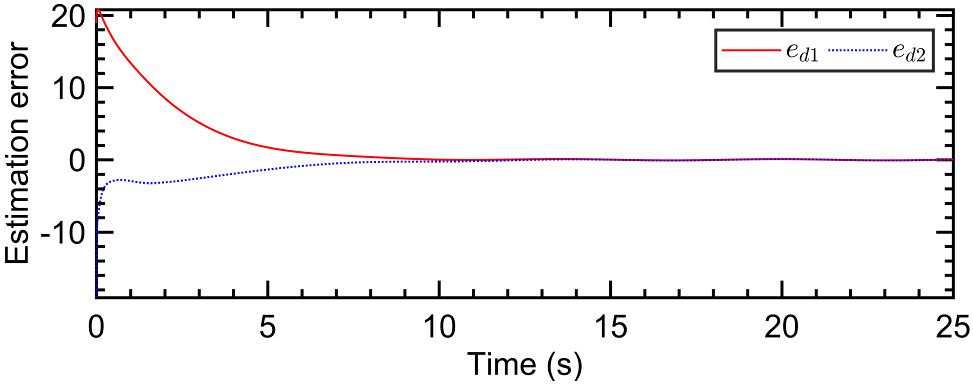


Figure 5.3: Disturbance estimation error profiles. Both estimates converge to the actual disturbance state within the first 10 s.

and 5.3, the disturbance and its estimates, and the estimation error is illustrated. Both $\hat{d}_1$ and $\hat{d}_2$ are able to track and compensate for the disturbance.

5.3.2 Disturbance observer design for mismatched disturbances

Consider a system with similar dynamics as the first case but with mismatched disturbances.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_1 x_2 \\ -x_1^2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} u + \begin{bmatrix} 1 & x_1 \\ x_2 & x_2^2 + 1 \end{bmatrix} d \tag{5.31}$$

For such systems, guaranteeing asymptotic stability in the presence of disturbances using an NDOB based control is generally not possible. We seek a compensation gain matrix $\zeta(x)$, a nominal controller $\chi(x)$ which asymptotically stabilizes the system without disturbances, and an NDOB such that with the augmented system is ISS.

The closed loop system (5.31) without disturbances is asymptotically stable for the nominal controller

$$\chi(x) = \begin{bmatrix} x_2 x_1 - x_1 \\ x_1^2 - x_2 \end{bmatrix} \tag{5.32}$$

Consider the Lyapunov function

$$V = \frac{1}{2} x^{\mathrm{T}} x > 0 \; \forall x \neq 0 \tag{5.33}$$

Taking the time derivative, we get

$$\dot{V} = x_1(x_1 x_2 + \chi_1(x)) + x_2(-x_1^2 + \chi_2(x)) \tag{5.34}$$

where $\chi(x) = [\chi_1(x), \chi_2(x)]^{\mathrm{T}}$. Substituting for $\chi(x)$,

$$\dot{V} = -(x_1^2 + x_2^2) < 0, \; \forall x \neq 0 \tag{5.35}$$

Therefore, the closed loop system without disturbances is asymptotically stable under the control law $\chi(x)$. Note that since $g_2(x)$ is the same mapping considered in the previous example, the observer design remains the same. Next, we prove that the system is ISS for a given compensation matrix $\zeta(x) \succ 0 \; \forall x$. Let the disturbance compensation gain be given as

$$\zeta(x) = \begin{bmatrix} 1 + x_1^2 + x_2^2 & 0 \\ 0 & 0.5 + (x_1 + x_2)^2 \end{bmatrix} \tag{5.36}$$

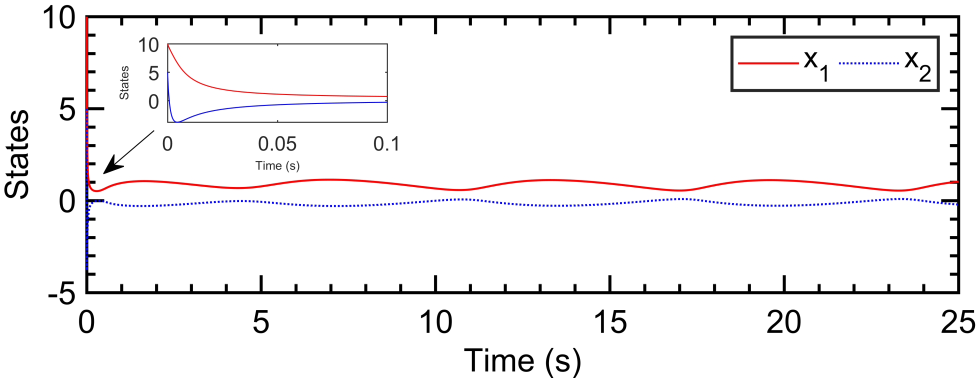


Figure 5.4: State trajectory. The trajectories rapidly converge to the neighborhood of origin within 0.25 s and remain bounded.

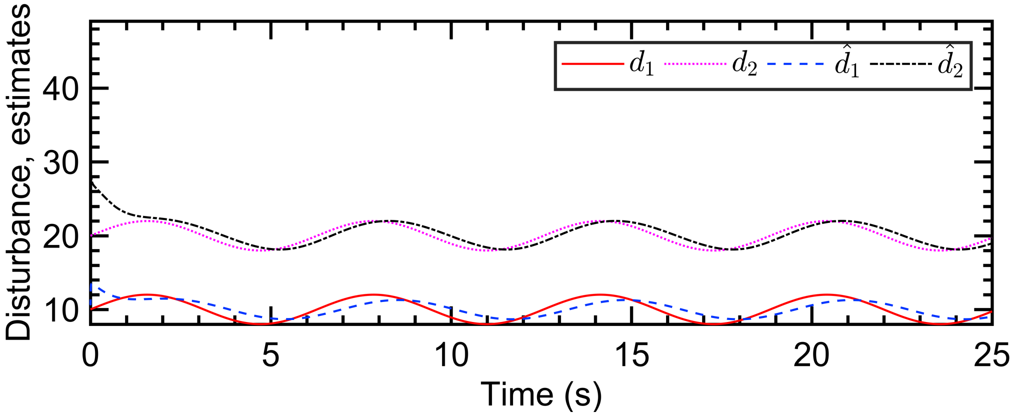


Figure 5.5: Disturbance and its estimation profiles.

For the given non-singular $\zeta(x)$, a fourth degree polynomial ISS-Lyapunov function is considered. The $\mathcal{K}_\infty$ functions $\alpha_1(s)$,$\alpha_2(s)$, $\alpha_3(s)$, and $\sigma(s)$ are parameterized as

$$\alpha_1(s) = c_1 s^2 \tag{5.37}$$

$$\alpha_2(s) = c_2 s^4 + c_3 s^2 \tag{5.38}$$

$$\alpha_3(s) = c_4 s^2 \tag{5.39}$$

$$\sigma(s) = c_5 s^4 + c_6 s^2 \tag{5.40}$$

where the coefficients $c_i, i = 1..6$ are the decision variables. Using $\chi(x)$ as shown before, $\zeta(x)$ given in (5.36),and the NDOB gain (5.29), the feasibility problem given by (5.22-5.24) and (5.26) is solved to find the ISS-Lyapunov function and the corresponding $\mathcal{K}_\infty$ functions. Solving the SDP feasibility problem, we obtain

$$\begin{aligned} V(x,e_d) = {} & 0.0018x_1^2e_{d2}^2 - 0.00094x_1x_2e_{d1}e_{d2} + 0.00015x_1x_2e_{d2}^2 \\ & + 0.0051x_1e_{d1}^3 - 0.00033x_1e_{d1}^2e_{d2} + 0.015x_1e_{d1}e_{d2}^2 \\ & + 0.00025x_1e_{d1} - 0.0073x_1e_{d2}^3 - 0.00016x_1e_{d2} \\ & + 0.0006x_2^2e_{d2}^2 + 0.000558x_2e_{d1}^3 + 0.0083x_2e_{d1}e_{d2}^2 \\ & + 0.0213x_2e_{d2}^3 + 0.00011x_2e_{d2} + 0.8045e_{d1}^4 \\ & + 1.579e_{d1}^2e_{d2}^2 + 0.7937e_{d1}^2 + 0.1084e_{d1}e_{d2}^3 + 0.0011e_{d1}e_{d2} \\ & + 0.9725e_{d2}^4 + 0.7683e_{d2}^2 \end{aligned} \tag{5.41}$$

where $e_d = [e_{d1}, e_{d2}]^{\mathrm{T}}$ is the disturbance estimation error.

$$\alpha_1(\|x,e_d\|) = 2.098 \times 10^{-6}(x_1^2 + x_2^2 + e_{d1}^2 + e_{d2}^2) \tag{5.42}$$

$$\begin{aligned} \alpha_2(\|x,e_d\|) = {} & 2.271(x_1^4 + x_2^4 + e_{d1}^4 + x4^4) \\ & + 1.136(x_1^2 + x_2^2 + e_{d1}^2 + +x4^2) \end{aligned} \tag{5.43}$$

$$\alpha_3(\|x,e_d\|) = 1.942 \times 10^{-6}(x_1^2 + x_2^2 + e_{d1}^2 + e_{d2}^2) \tag{5.44}$$

$$\sigma(\|d\|) = 4.151(d_1^4 + d_2^4) + 1.351(d_1^2 + d_2^2) \tag{5.45}$$

where $d = [d_1, d_2]^{\mathrm{T}}$ is the disturbance. Since the SDP problem is feasible, the system (5.31) with disturbance estimation error dynamics is ISS with ISS-Lyapunov function given by (5.41).

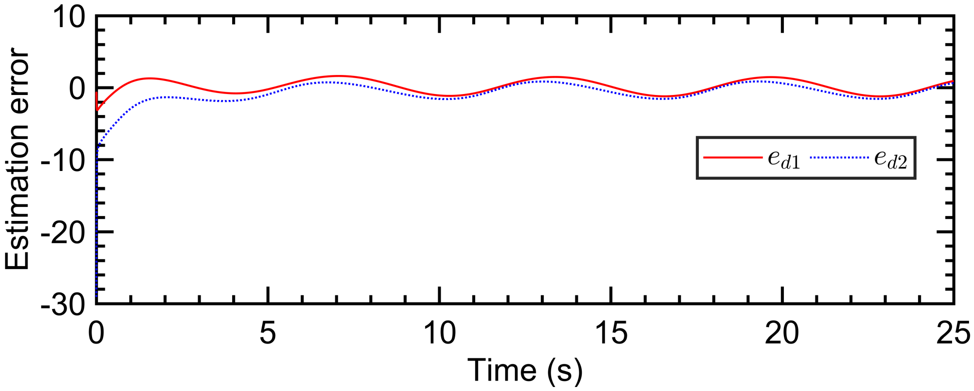


Figure 5.6: Disturbance estimation error profiles. Estimation errors remain bounded in the neighborhood of origin.

The ISS property of the closed-loop system is numerically validated with the disturbance $d(t) = \begin{bmatrix} 10 + 2\sin(t) \\ 20 + 2\sin(t) \end{bmatrix}$ using the control law (5.15) with $\chi(x)$ and $\zeta(x)$ given in (5.32) and (5.36), respectively and NDOB gain (5.29). The initial state of the system is taken as $x_0 = [10, 5]^{\mathrm{T}}$ while the initial internal state of the observer is $h_0 = [0, 0]^{\mathrm{T}}$. Fig 5.4 shows the state trajectory for the given time-varying disturbances. Both the states converge rapidly towards origin and remain bounded which proves that the system is ISS. The disturbance estimates are able to track the periodic disturbance input as shown in Fig 5.5. The disturbance error is also bounded and close to the origin as seen from Fig 5.6. In the next chapter, the relative attitude tracking problem of spacecraft will be investigated using nonlinear disturbance observer based control.

Chapter 6

Robust disturbance observer based relative attitude tracking control of spacecraft using sum-of-squares programming

Attitude control of a spacecraft is critical to ensure the success of many space missions. During proximity operations, a chaser spacecraft is typically required to synchronize it's attitude and angular velocity to those of the target to enable safe inspection and manipulation. During formation flying, attitude synchronization of multiple spacecraft is also required. The attitude synchronization or the tracking problem has been studied extensively under a wide range of control methodologies, including, hybrid attitude control [150], adaptive control [7], finite-time feedback control [38], Lyapunov based control [185], disturbance observer based sliding mode control [178] amongst others. During proximity operations, it is possible that the chaser spacecraft experiences disturbances such as gravity gradient torque. Furthermore, the target may be uncooperative and freely tumbling due to attitude control loss. The problem of attitude tracking with disturbances for an aircraft is studied in [178]. To ensure robustness against system uncertainties, a linear disturbance observer is combined with a nonlinear sliding mode controller to guarantee the stability of the system.

For spacecraft proximity operations, nonlinear disturbance observers (DOBs) based robust control has been proposed to compensate for model uncertainty and external disturbances [61, 87, 163, 99, 164, 200]. Disturbance observers are desirable since the compensation control based on the estimates of the disturbance obtained from the observer can be directly added as a patch to a nominal feedback control scheme [30]. For the purpose of this study, the focus is primarily on relative attitude tracking with disturbance torques acting on both the target and chaser spacecraft. It is assumed that the magnitude of target inertia uncertainties is smaller than the disturbance torque on the target. The goal of the robust control design is to track the attitude and angular velocity of the target and attenuate disturbances acting on the system.

A challenging feature of the attitude tracking error dynamics is that the external disturbance or uncertainty is typically coupled to the angular velocity and attitude error states which makes the synthesis of the DOB gain matrix challenging. Furthermore, the disturbances enters the system through a different channel resulting in a 'mismatch' condition. Existing literatures on DOB based control for relative position and attitude control of spacecraft make simplifying assumptions by considering the upper bounds on the disturbance to state mappings which makes the synthesis of DOBs straightforward [87, 164].

Differently, in this chapter, the proposed method leverages the polynomial nature of the tracking error dynamics to design an exponentially convergent DOB to compensate for the uncertainties and disturbances for general polynomial systems as discussed in Chapter 5. The observer gain synthesis is posed as a matrix sum-of-squares problem which can be be solved efficiently using semidefinite programming. The main contributions of this method are as follows: 1.) A systematic, computationally tractable approach leveraging sum-of-squares optimization is proposed for synthesis of nonlinear DOBs for relative attitude tracking of rigid spacecraft; 2.) The approach does not consider any approximation or upper bounds of the underlying dynamics; 3.) The proposed method is compared to conventional control schemes including sliding mode control and DOBs with simplified dynamics and shows superior control and tracking performance. The remainder of the brief is as follows: Section 6.1 introduces the attitude tracking problem along with the DOB design, the tracking control scheme is introduced in

Section 6.2, Section 6.3 presents the numerical validation of the proposed method.

6.1 Relative attitude tracking and DOB design

Consider a relative proximity operation scenario where a chaser spacecraft is required to track the attitude of an uncooperative target spacecraft. The variables with subscript $(.)_c$ denote chaser variables, $(.)_t$ denote target variables, and $(.)_e$ denote the error variables. The attitude parameterization chosen are the classical Rodrigues parameters (CRP) denoted by σ while the angular velocity is denoted by ω. The CRP attitude parameterization is related to the rotation matrix as [65]

$$R(\sigma) = \frac{1}{1+\sigma^{\mathrm{T}}\sigma}\big((1-\sigma^{\mathrm{T}}\sigma)\mathbf{E}_{3\times3} - 2\sigma^{\times} + 2\sigma\sigma^{\mathrm{T}}\big) \tag{6.1}$$

where $R(\sigma) \in \mathrm{SO}(3)$ is the rotation matrix. For simplicity, it is assumed that the moment of inertia of the chaser spacecraft is known and the uncertainty in the target moment of inertia (if any) is significantly smaller than the disturbance torques on the target. However, both the chaser and target are assumed to have time-varying disturbance torques acting on them. The chaser rotational kinematics and dynamics are expressed as

$$\dot{\sigma}_c = G(\sigma_c)\omega_c \tag{6.2}$$

$$J_c\dot{\omega}_c = -\omega_c^{\times}J_c\omega_c + \tau_c + d_c \tag{6.3}$$

where

$$G(\sigma) = \frac{1}{2}\big(\mathbf{E}_{3\times3} + \sigma^{\times} + \sigma\sigma^{\mathrm{T}}\big) \tag{6.4}$$

and where $J_c \in \mathbb{R}^{3\times3}$ is the chaser moment of inertia, $\tau_c \in \mathbb{R}^3$ are the control torques and $d_c \in \mathbb{R}^3$ are the disturbance torques. Similarly, the target dynamics are expressed as

$$\dot{\sigma}_t = G(\sigma_t)\omega_t \tag{6.5}$$

$$J_t\dot{\omega}_t = -\omega_t^\times J_t\omega_t + d_\tau \tag{6.6}$$

where $J_t \in \mathbb{R}^{3\times 3}$ is the target moment of inertia and $d_\tau \in \mathbb{R}^3$ are the disturbance torques on the target. The relative error between the chaser and target variables can then be expressed as [65]

$$\sigma_e = \frac{\sigma_c - \sigma_t + \sigma_c^\times \sigma_t}{1 + \sigma_c^{\mathrm{T}}\sigma_t} \tag{6.7}$$

$$\omega_e = \omega_c - R\omega_t \tag{6.8}$$

Taking time derivative of the relative errors, we get

$$\dot{\sigma}_e = G(\sigma_e)\omega_e \tag{6.9}$$

$$J_c\dot{\omega}_e = J_c\dot{\omega}_c - J_cR\dot{\omega}_t - J_c\dot{R}\omega_t \tag{6.10}$$

where R denotes the attitude error expressed using the rotation matrix. Substituting for the chaser and target angular accelerations

$$\begin{gathered} J_c\dot{\omega}_e = \tau - \omega_c^\times J_c\omega_c + d_c - J_cR\omega_e^\times\omega_t + \\ J_cRJ_t^{-1}\omega_t^\times J_t\omega_t - J_cRJ_t^{-1}d_\tau \end{gathered} \tag{6.11}$$

The error dynamics can then be expressed by substituting ω_t using (6.8)

$$\begin{gathered} J_c\dot{\omega}_e = \tau + d_c - \omega_c^\times J_c\omega_c - J_cR(\omega_e)^\times R^{\mathrm{T}}(\omega_c - \omega_e) + \\ J_cR\Bigg(J_t^{-1}(R^{\mathrm{T}}(\omega_c - \omega_e))^\times J_tR^{\mathrm{T}}(\omega_c - \omega_e) - J_t^{-1}d_\tau) \end{gathered} \tag{6.12}$$

Remark 1. *The uncertainty in the target parameters enters the error dynamics* *through the term* $\begin{bmatrix} -J_c R J_t^{-1} & \mathbf{E}_{3\times3} \end{bmatrix} \begin{bmatrix} d_\tau \\ d_c \end{bmatrix}$. *Therefore, the disturbance to state mapping explicitly depends on the rotation matrix error* $R(\sigma_e)$.

Remark 2. *The rotation matrix error* $R(\sigma_e) \in \mathrm{SO}(3)$ *is a compact manifold and can be upper bounded. Previous literature uses this property to design DOB for attitude tracking by considering an upper bound of the attitude error resulting in the state-disturbance mapping being an identity matrix [163, 87, 164]. The upper bound* $\max \|R\| = \mathbf{E}_{3\times3}$ *results in the mapping* $\begin{bmatrix} -J_c J_t^{-1} & \mathbf{E}_{3\times3} \end{bmatrix} \begin{bmatrix} d_\tau \\ d_c \end{bmatrix}$. *Since the mapping is independent of states, the DOB design is simplified.*

Remark 3. *The disturbance and input enter through different channels, therefore there exist a mismatching condition in the system dynamics.*

If the rotation matrix error is upper bounded as described in Remark 7.21, the DOB design is straightforward. Since $\begin{bmatrix} \mathbf{E}_{3\times3} & -J_c J_t^{-1} \end{bmatrix}$ is constant, full-rank and invertible, a suitable choice for the observer gain matrix can be $\begin{bmatrix} -J_c J_t^{-1} & \mathbf{E}_{3\times3} \end{bmatrix}^{\mathrm{T}}$, which simplifies the observer design process. This approach commonly used in literature will be compared with our proposed polynomial DOB method in Section 6.3.

For the attitude tracking problem introduced in (6.12), we first note that the disturbance acts on the dynamics of ω_e and ω_c and is dependent on the rotation matrix through σ_e. Thus, the attitude kinematics for σ_c can be ignored from the observed design. From the relative error dynamics equations, the disturbance to state mapping term $g_2(x)$ in (5.1) is expressed as

$$g_2(\sigma_e) = \begin{bmatrix} 0_{3\times6} \\ \begin{bmatrix} 0_{3\times3} & \mathbf{E}_{3\times3} \end{bmatrix} \\ \begin{bmatrix} -J_c R J_t^{-1} & \mathbf{E}_{3\times3} \end{bmatrix} \end{bmatrix} \tag{6.13}$$

We seek a DOB gain matrix such that the disturbance estimation error is exponentially convergent. Using the expression for the rotation matrix in (6.1), this is mathematically expressed as

$$\frac{l(\sigma_e, \omega_e, \omega_c)}{1+\sigma_e^{\mathrm{T}}\sigma_e} \begin{bmatrix} 0_{3\times 6} \\ \begin{bmatrix} 0_{3\times 3} & (1+\sigma_e^{\mathrm{T}}\sigma_e)\mathbf{E}_{3\times 3} \end{bmatrix} \\ \begin{bmatrix} \Upsilon(\sigma_e) & (1+\sigma_e^{\mathrm{T}}\sigma_e)\mathbf{E}_{3\times 3} \end{bmatrix} \end{bmatrix} \succ 0 \tag{6.14}$$

where $\Upsilon(\sigma_e) = -J_c\bigg((1-\sigma_e^{\mathrm{T}}\sigma_e)\mathbf{E}_{3\times 3}) + 2\sigma_e\sigma_e^{\mathrm{T}} - 2\sigma_e^{\times}\bigg)J_t^{-1}$, Note that the scalar term $\frac{1}{1+\sigma_e^{\mathrm{T}}\sigma_e}$ is always positive and can be ignored from the equation above. For brevity, consider the auxiliary term $\bar{g}_2(\sigma_e, \omega_e, \omega_c)$ given by

$$\bar{g}_2(\sigma_e) = \begin{bmatrix} 0_{3\times 6} \\ \begin{bmatrix} 0_{3\times 3} & (1+\sigma_e^{\mathrm{T}}\sigma_e)\mathbf{E}_{3\times 3} \end{bmatrix} \\ \begin{bmatrix} \Upsilon(\sigma_e) & (1+\sigma_e^{\mathrm{T}}\sigma_e)\mathbf{E}_{3\times 3} \end{bmatrix} \end{bmatrix} \tag{6.15}$$

Thus, the following feasibility polynomial matrix sum-of-squares problem is solved using SOS programming

$$\begin{aligned} &\text{minimize} && - \\ &\text{subject to} && l(\sigma_e, \omega_e, \omega_c)\bar{g}_2(\sigma_e) - \epsilon\mathbf{E} \succeq 0 \end{aligned} \tag{6.16}$$

where $\epsilon > 0$. The choice of ϵ directly impacts the resulting eigenvalues of the matrix in (6.14) and can be tuned to improve estimate convergence. Note that solving (6.16) can be computationally challenging. While posing the problem as an SOS problem in a computational optimization environment, it is not only required that the eigenvalues of the matrix are non-negative but symmetricity is also desired. Thus additional constraints have to be imposed to ensure that the matrix remains positive-semidefinite. This constraint can be alleviated using the following proposition.

Proposition 5. *The synthesis of the DOB design proposed in (6.16) can be computationally improved by solving the feasibility problem with the constraint*

$$l(\sigma_e,\omega_e,\omega_c)\bar{g}_2(\sigma_e) + \big(l(\sigma_e,\omega_e,\omega_c)\bar{g}_2(\sigma_e)\big)^T - \epsilon\mathbf{E} \succeq 0 \tag{6.17}$$

Proof. The proof of the proposition is straightforward. The stability of the disturbance estimate error which is a time-varying system cannot be directly ascertained by the eigenvalues of the matrix $A(t) = l(\sigma_e,\omega_e,\omega_c)\bar{g}_2(\sigma_e)$. However, a sufficient condition to conclude stability is $A(t)+A(t)^{\mathrm{T}} \succ 0$. Consider a radially unbounded Lyapunov function

$$V(e_d) = e_d^{\mathrm{T}} e_d \tag{6.18}$$

Taking the time derivative of V

$$\dot{V}(e_d) = -e_d^{\mathrm{T}}(A(t) + A(t)^{\mathrm{T}})e_d < 0 \tag{6.19}$$

Furthermore,

$$\dot{V}(e_d) \leq -\lambda_{max}(A(t) + A(t)^{\mathrm{T}})V(e_d) \tag{6.20}$$

$$\dot{V}(e_d) \leq -\lambda_{max}(A(t) + A(t)^{\mathrm{T}})\|e_d\|^2 \leq -\epsilon\|e_d\|^2 \tag{6.21}$$

where λ_{max} is the maximum eigenvalue of the matrix $(A(t) + A(t)^{\mathrm{T}})$. Since $\lambda_{max}, \epsilon > 0$, the time-varying system $\dot{e}_d = A(t)e_d$ is globally exponentially stable [70]. □

The nonlinearity in the disturbance mapping $\bar{g}_2(\sigma_e)$ is quadratic in σ_e. In case alternate attitude parameterizations such as the modified Rodrigues parameters are chosen, the mapping would be quartic and therefore more computational challenging. Although a cubic monomial is ideal for the design of auxiliary vector

$p_i(x) = \mathbf{c}_i^{\mathrm{T}}[1, \sigma_{e1}, \sigma_{e1}^2, \sigma_{e1}^3, ..., \sigma_{e2}..., \omega_{e1}, ..., \omega_{e3}^3]$ for $i = 1, 2, ..., 6$ so that the observer gain is an even degree polynomial matrix, we use a quadratic monomial for $p_i(x)$ which is computationally more tractable and still satisfies Proposition 5. The decision variables are the coefficient vectors $c_i \in \mathbb{R}^{324}$. Given the chaser moment of inertia matrix $J_c = \text{diag}[100, 50, 100]$ and the chaser moment of inertia $J_t = 2J_c$, the gain synthesis is posed as an SOS feasibility problem and solved using the MATLAB toolbox SOSTOOLS [137] with Sedumi [161] as the underlying SDP solver. Throughout this section, monomials with coefficients less than 10^{-5} have been omitted. The auxiliary vector $p(\sigma_e, \omega_e) \in \mathbb{R}^6$ are obtained as

$$\begin{aligned} p_1(\sigma_e, \omega_e) &= 1488\omega_{e1} + 3411\omega_{c1} - 118\sigma_{e2}\omega_{e3} + 118\sigma_{e3}\omega_{e2} \\ &\quad + 93.4\sigma_{e3}\omega_{c2} \end{aligned} \tag{6.22}$$

$$\begin{aligned} p_2(\sigma_e, \omega_e) &= 1177\omega_{e2} + 3611\omega_{c2} + 281\sigma_{e1}\omega_{e3} - 281\sigma_{e3}\omega_{e1} \\ &\quad - 140\sigma_{e1}\omega_{c3} + 140\sigma_{e3}\omega_{c1} \end{aligned} \tag{6.23}$$

$$\begin{aligned} p_3(\sigma_e, \omega_e) &= 1488\omega_{e3} + 3411\omega_{c3} - 118\sigma_{e1}\omega_{e2} + 118\sigma_{e2}\omega_{e1} \\ &\quad - 93.4\sigma_{e1}\omega_{c2} \end{aligned} \tag{6.24}$$

$$\begin{aligned} p_4(\sigma_e, \omega_e) &= 1255\omega_{c1} - 1766\omega_{e1} - 2066\sigma_{e2}\omega_{e3} + 2066\sigma_{e3}\omega_{e2} \\ &\quad + 2011\sigma_{e2}\omega_{c3} - 1855\sigma_{e3}\omega_{c2} \end{aligned} \tag{6.25}$$

$$\begin{aligned} p_5(\sigma_e, \omega_e) &= 1000\omega_{c2} - 1333\omega_{e2} + 1388\sigma_{e1}\omega_{e3} - 1388\sigma_{e3}\omega_{e1} \\ &\quad - 1344\sigma_{e1}\omega_{c3} + 1344\sigma_{e3}\omega_{c1} \end{aligned} \tag{6.26}$$

$$p_6(\sigma_e, \omega_e) = 1255\omega_{c3} - 1766\omega_{e3} - 2066\sigma_{e1}\omega_{e2} + 2066\sigma_{e2}\omega_{e1} \\ + 1855\sigma_{e1}\omega_{c2} - 2011\sigma_{e2}\omega_{c1} \tag{6.27}$$

The gain matrix can then be computed as $l(x) = \frac{\partial p(x)}{\partial x} \in \mathbb{R}^{6\times 9}$.

6.2 Tracking control scheme

Theorem 4. *Let the sliding surface be defined as*

$$s = \omega_e + \lambda\sigma_e \tag{6.28}$$

The goal of the composite controller is to track the disturbances and state error such that $\omega_e \to 0$, $\sigma_e \to 0$, and $e_d \to 0$ as $t \to \infty$. Consider the relative kinematics and dynamics for tracking expressed in (6.12), the following control law guarantees asymptotic convergence of the state and disturbance estimate error

$$\tau = \tau_{smc} - J_c\Big(\Big[\mathbf{E}_{3\times 3} \quad -J_c R J_t^{-1}\Big]\, \hat{d}\Big) \tag{6.29}$$

where τ_{smc} is the nominal SMC controller given by

$$\tau_{smc} = J_c\big(- \lambda G(\sigma_e)\omega_e - Ks - \\ (-\omega_c^{\times} J_c\omega_c - J_c R(\omega_e)^{\times} R^{T}(\omega_c - \omega_e)) + \\ J_c R(J_t^{-1}(R^{T}(\omega_c - \omega_e))^{\times} J_t R^{T}(\omega_c - \omega_e))\big) \tag{6.30}$$

where $\lambda, K \succ 0$.

Proof. Consider the Lyapunov function

$$V = \frac{1}{2}s^{\mathrm{T}}s + e_d^{\mathrm{T}}e_d \tag{6.31}$$

The time derivative of the Lyapunov function is given as

$$\dot{V} = s^{\mathrm{T}}\dot{s} - e_d^{\mathrm{T}}(A(t) + A(t)^{\mathrm{T}})e_d \tag{6.32}$$

$$\dot{V} = s^{\mathrm{T}}(\dot{\omega}_e + \lambda G(\sigma_e)\omega_e) - e_d^{\mathrm{T}}(A(t) + A(t)^{\mathrm{T}})e_d \tag{6.33}$$

Using the control law defined in (6.29) in (6.33)

$$\begin{aligned} \dot{V} &= s^{\mathrm{T}}(-Ks - \begin{bmatrix}\mathbf{E}_{3\times3} & -J_cRJ_t^{-1}\end{bmatrix} e_d) \\ &- e_d^{\mathrm{T}}(A(t) + A(t)^{\mathrm{T}})e_d \end{aligned} \tag{6.34}$$

Remark 4. *Matrices K and $\begin{bmatrix}\mathbf{E}_{3\times3} & -J_cRJ_t^{-1}\end{bmatrix}$ admit constant upper bounds.*

Consider the disturbance estimate error system where

$$\|\dot{e}_d\| = \sqrt{\langle\, A(t)e_d, e_d\rangle + \langle e_d, A(t)e_d\rangle} \leq \sqrt{\lambda_{max}}\|e_d\| \tag{6.35}$$

Thus, an upper bound on the time derivative of the Lyapunov function is given by

$$\dot{V} \leq -\gamma_1\|s\|^2 - \gamma_2\|s\|\|e_d\| - \gamma_3\|e_d\|^2 \tag{6.36}$$

where γ_1, γ_2, and γ_3 denote the upper bounds on terms in (6.34). (6.36) can then be transformed as

$$\dot{V} \leq -(\gamma_1 - \frac{\gamma_3}{4\gamma_2})\,\|\omega_e\|^2 - \big(\sqrt{\gamma_2}\,\|e_d\| - \frac{\gamma_3}{2\sqrt{\gamma_2}}\,\|\omega_e\|\,\big)^2 \tag{6.37}$$

For $\dot{V} \leq 0$, $\gamma_1 \geq \frac{\gamma_3}{4\gamma_2}$. Note that γ_1 denotes the maximum eigenvalue of the symmetric matrix K. Therefore, with a sufficiently large gain, $\dot{V} \leq 0$ can be guaranteed. This implies $V(t) \leq V(0)$. Since e_d is bounded by the design of the DOB, from (6.37), ω_e, σ_e, and correspondingly R are bounded as well. The closed-loop dynamics in presence of disturbances is

$$J_c\dot{\omega}_e = -K(\omega_e + \lambda\sigma_e) - \begin{bmatrix}\mathbf{E}_{3\times3} & -J_cRJ_t^{-1}\end{bmatrix} e_d \tag{6.38}$$

Thus, $\dot{\omega}_e$ is bounded as well. Taking the time derivative of $\dot{V}$

$$\begin{aligned}\ddot{V} &= \big(\dot{\omega}_e + \lambda G(\sigma_e)\omega_e\big)^{\mathrm{T}}\big(\dot{\omega}_e + \lambda G(\sigma_e)\omega_e\big)\\ &\quad + (\omega_e + \lambda\sigma_e)^{\mathrm{T}}(\ddot{\omega}_e + \lambda\dot{G}(\sigma_e)G(\sigma_e)\omega_e^2 + \lambda G(\sigma_e)\dot{\omega}_e)\\ &\quad + \dot{e}_d^{\mathrm{T}}\dot{e}_d + e_d^{\mathrm{T}}\ddot{e}_d\end{aligned} \tag{6.39}$$

Since $\dot{e}_d$ converges exponentially to zero and ω_e and σ_e are bounded, $\ddot{e}_d$ is bounded as well. Thus, from (6.39), we conclude that $\ddot{V}$ is bounded as well. Therefore, using Barbalat's lemma, we obtain that $\omega_e \to 0$, $\sigma_e \to 0$, and $e_d \to 0$ as $t \to \infty$. Thus, the composite control law in (6.29) yields global asymptotic stability. □

6.3 Numerical results

To numerically validate the efficacy of the proposed observer and controller design, a robust tracking example is considered. Table 6.1 provides the parameters used in the study. Where $\sqcap\!\sqcup$ denotes a square wave of given magnitude. The magnitude of disturbance torques on target is assumed to be larger than that on the chaser since the target does not have any on-board actuation. For a realistic tracking scenario, the effects of gravity gradient torque and torque produced via the solar radiation pressure and drag are added to the periodic disturbance torque expressed in Table 6.1. Although the resulting disturbances are fairly large in magnitude, these values are deliberately chosen to ascertain the performance of the proposed DOB design. The maximum gravity gradient torque, $\tau_g = [\tau_{g1}, \tau_{g2}, \tau_{g3}]^{\mathrm{T}}$ is modeled via

Parameter	Values
J_c	diag[100,150,100]
J_t	diag[200,300,200]
$\sigma_c(t=0)$	$\begin{bmatrix} 0 & 0 & 0.1 \end{bmatrix}^{\mathrm{T}}$
$\omega_c(t=0)$	$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$
$\sigma_e(t=0)$	$\begin{bmatrix} \pi/3 & -\pi/2 & \pi/2 \end{bmatrix}^{\mathrm{T}}$
$\omega_e(t=0)$	$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$
$d_c(t)$	$\begin{bmatrix} 3\cos(\pi t/5) \\ 2\cos(\pi t/10) \\ 9\cos(\pi t/10) \end{bmatrix} \times 10^{-3}$
$d_\tau(t)$	$\begin{bmatrix} 3\sin(\pi t/10) \\ 2\sin(\pi t/10) + 0.1\sqcap\!\sqcup(\pi t/5) \\ 2\sin(\pi t/10) + 0.3\sqcap\!\sqcup(\pi t/10) \end{bmatrix} \times 10^{-3}$
λ	$0.125\mathbf{E}_{3\times3}$
K	$0.2\mathbf{E}_{3\times3}$

Table 6.1: Parameters used in the numerical simulation for relative attitude tracking

the following expression [193]

$$\tau_{g1} = \frac{3\mu}{r^5}(I_3 - I_2) \tag{6.40}$$

$$\tau_{g2} = \frac{3\mu}{r^5}(I_1 - I_3) \tag{6.41}$$

$$\tau_{g3} = \frac{3\mu}{r^5}(I_2 - I_1) \tag{6.42}$$

where μ is the gravitational parameter, r is the radius of the spacecraft orbit, and I_1, I_2, I_3 are the components of the moment of inertia of the spacecraft. The maximum torque due to solar radiation pressure is computed as [193]

$$\tau_{srp} = \frac{F_s}{c}A_s(1+q)(C_p - C_g) \tag{6.43}$$

where $F_s = 1367\ \mathrm{W/m^2}$ is the solar constant, c is the speed of light, A_s is the surface area of the spacecraft facing the Sun, C_p and C_g denote the center of solar pressure and gravity, respectively, and q is the reflectance factor. Finally

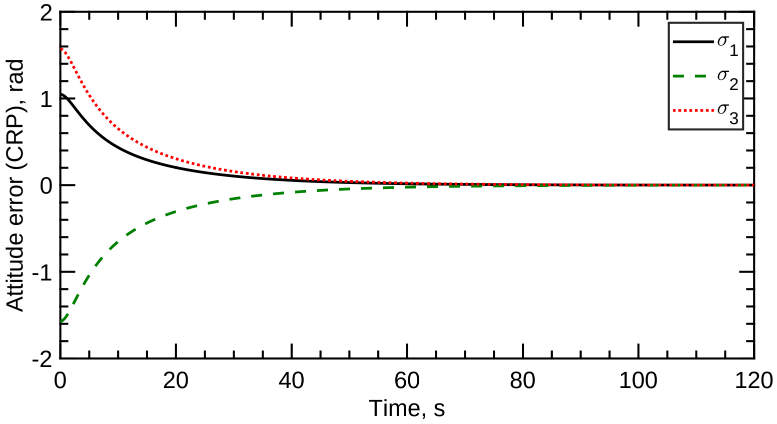


Figure 6.1: Attitude error trajectory in terms of CRP

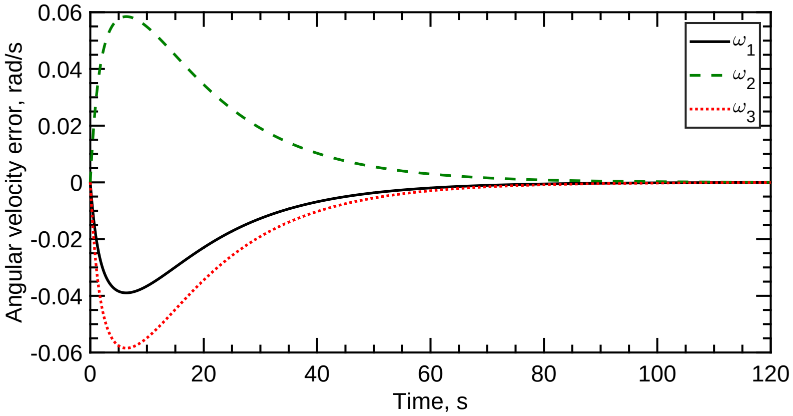


Figure 6.2: Angular velocity error trajectory

the torque produced due to atmospheric drag is given as [193]

$$\tau_d = \frac{1}{2}\rho C_D A V^2 (C_{pa} - C_g) \tag{6.44}$$

where ρ is the atmospheric density, C_D is the drag coefficient, $V = \sqrt{\frac{\mu}{r}}$ is the orbital velocity of the spacecraft, A is the surface area normal to the drag, and C_{pa} denotes the center of air pressure. Note that however, the magnitude of these torques are of the order of 10^{-4} Nm and hence are quite small. Fig. 6.1 shows the time profile of the relative attitude error in terms of the classic Rodrigues parameters.The errors converge within 70 s. Fig. 6.2 illustrates the relative angular velocity trajectory. Although the initial error is close to zero, there is an increase

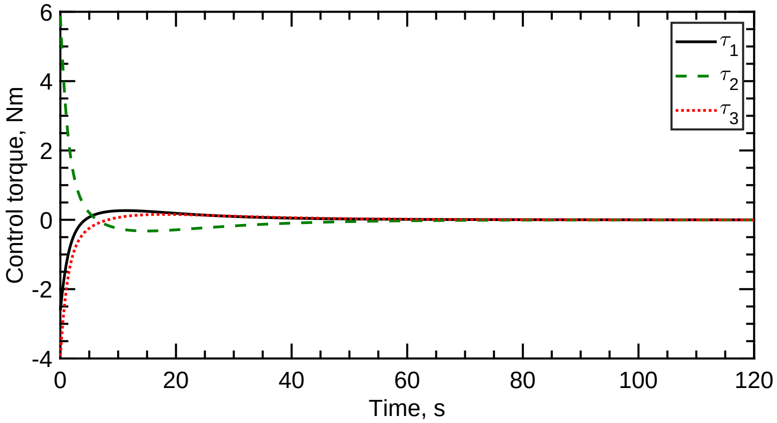


Figure 6.3: Control torque profiles

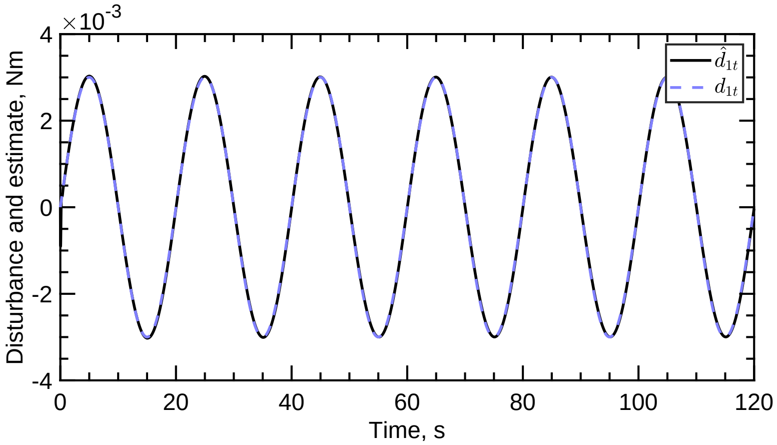


Figure 6.4: Disturbance torque on target spacecraft and it's estimate in body frame x axis

in the angular velocity within the first 5 s after which the angular velocities decay to zero. Fig. 6.3 shows the torque profiles. The control torques converge to the vicinity of origin in 60 s.

In Figs. 6.4, 6.5, and 6.6, the time profile of the disturbance torques on the target spacecraft and it's estimate is provided. It is observed that within the first one second, the estimates are able to track the time varying torques with very low error. Fig. 6.7, 6.8, and 6.9 illustrates the disturbance and estimates on the chaser spacecraft. The deviations observed in the estimates is higher within the first 1 s but they rapidly decay and good tracking performance is observed. Overall, the performance of the DOB with a nominal sliding mode

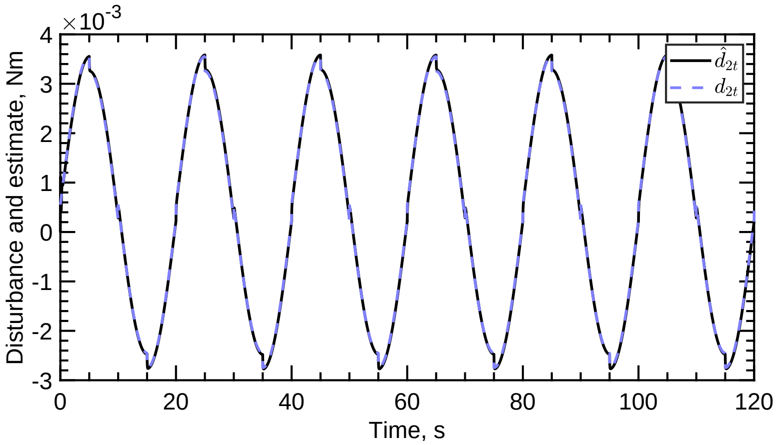


Figure 6.5: Disturbance torque on target spacecraft and it's estimate in body frame y axis

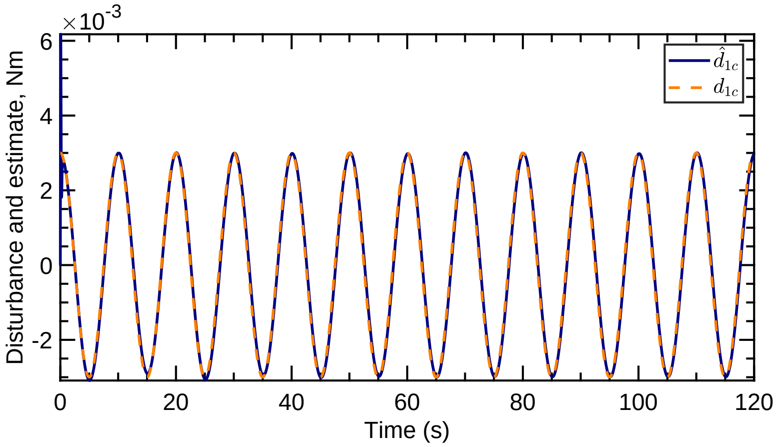


Figure 6.6: Disturbance torque on target spacecraft and it's estimate in body frame z axis

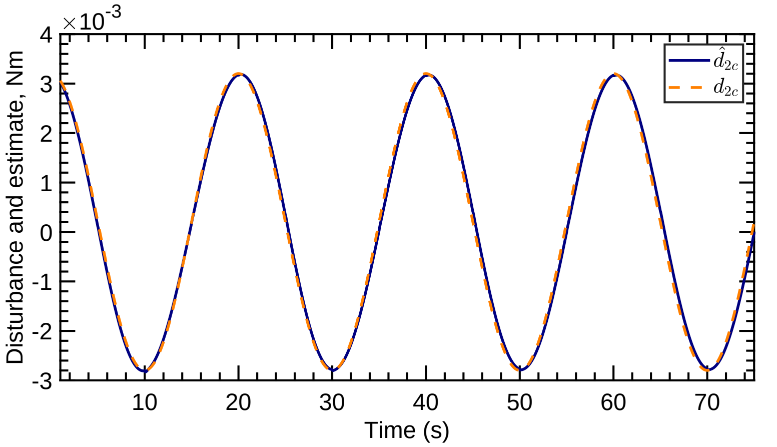


Figure 6.7: Disturbance torque on chaser spacecraft and it's estimate in body frame x axis

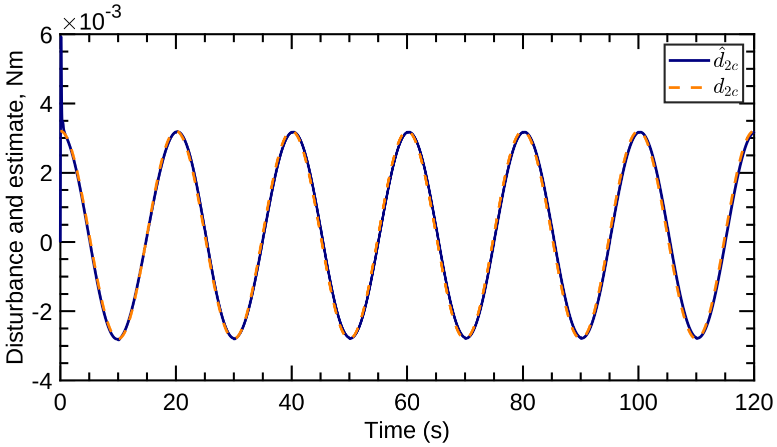


Figure 6.8: Disturbance torque on chaser spacecraft and it's estimate in body frame y axis

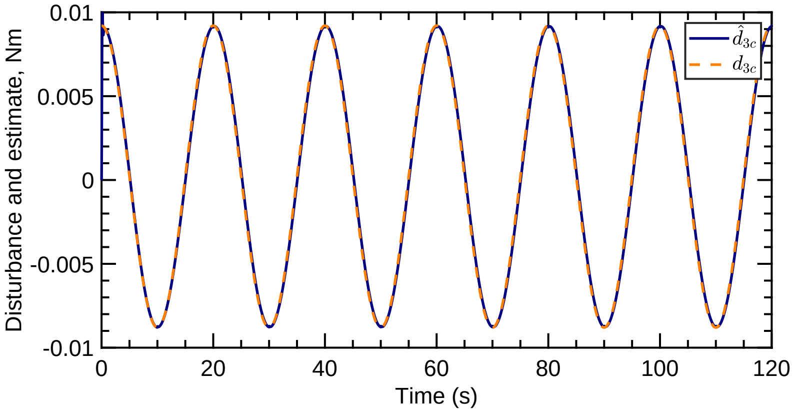


Figure 6.9: Disturbance torque on chaser spacecraft and it's estimate in body frame z axis

controller is very promising. The speed of convergence for the observer depends on the coefficients of the observer gain matrix which are tuned using the term ϵ in (6.17). However, from Lyapunov analysis, we find that coefficients with higher magnitude in the observer gain matrix $l(\sigma_e, \omega_e, \omega_c)$ affect the term γ_2 that need to be compensated with a sliding mode gain values as seen from (6.37) to ensure that Theorem 4 holds. Overall, the results illustrate that the proposed polynomial NDOB based controller can not only guarantee the stability of system but in addition, accurately estimate the external disturbances, such that the relative attitude tracking can be successfully executed in the presence of large disturbance torque.

The superior performance of the DOB based control is illustrated in Fig. 6.10 via a net torque magnitude comparison with a nominal sliding mode control (SMC) scheme without disturbance observation expressed in (6.30) and a SMC with the disturbance observer obtained from approximating the system dynamics as discussed in Remark 7.21 and using the same composite control law in (6.29). Compared to the nominal SMC scheme and the approximate NDOB based controller, the polynomial NDOB requires less control effort to stabilize the system, as shown in Table 6.2. In comparison, the approximate NDOB augmented SMC method is observed to have higher torque magnitude than both controllers. In addition, the performance of the polynomial and approximate NDOB methods is analyzed through the disturbance estimate error, e_d. In Fig. 6.11, the disturbance error estimates are illustrated. The polynomial NDOB shows significantly better disturbance tracking performance, where the magnitude of the estimate error is approximately 10^{-4} at most times. On the other hand, the approximate NDOB based controller fails to optimally track the disturbance and the error magnitude is above 10^{-3} Nm. Furthermore, large observer gains are required to track the disturbance for the approximate NDOB which in turns degrades the controller performance as observed in Fig. 6.10.

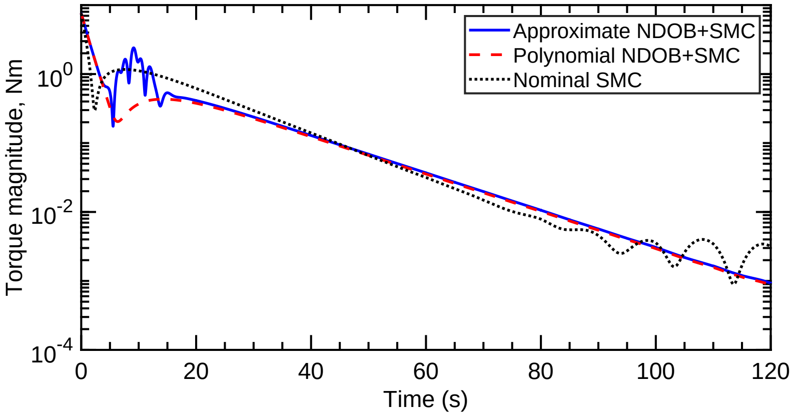


Figure 6.10: Control magnitude comparison between nominal SMC ,SMC augmented with the polynomial NDOB, and SMC augmented with approximate NDOB

Controller	Time integral of torque magnitude (Nm)
Nominal SMC	8.87×10^3
Approximate NDOB with SMC	8.98×10^3
Polynomial NDOB with SMC	7.62×10^3

Table 6.2: Comparison of controller performance

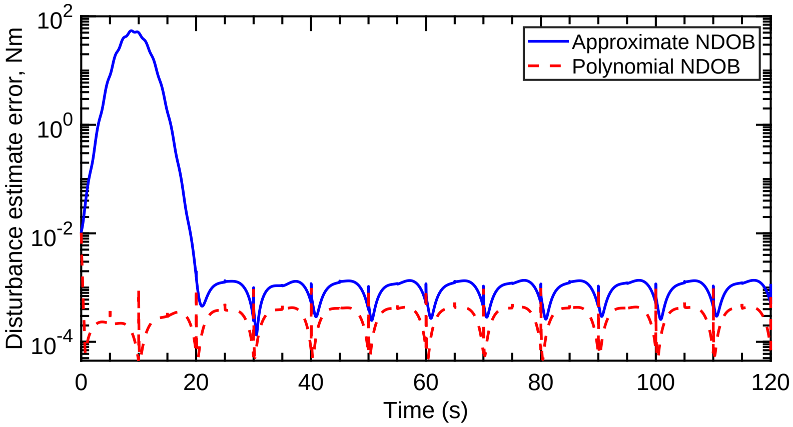


Figure 6.11: Disturbance error magnitude comparison between polynomial NDOB and NDOB with approximate dynamics

Chapter 7

Iterative feasible solutions for nonlinear optimal control problems using DC programming and polynomial optimization

In Chapter 2, convex optimization based approaches for solving nonlinear optimal control problems are briefly discussed. Using direct methods, the optimal control is transcribed and solved as a static optimization. Although time marching and pseudo-spectral methods have been practically implemented [69] and commercial solvers such as GPOPS [133] and DIDO [144] exist, two significant challenges are required to be tackled in aerospace guidance and control.

- First, there are often no bounds available on the computational time and effort required.
- Tractability can also be challenging in the sense that a bad initial guess can lead to numerical divergence of the solution.

These drawbacks make the applicability of nonlinear programming based transcription methods in real-time scenarios challenging. On the other hand, convex optimization can be reliably be solved in polynomial time. Iterative or sequential convex programming based approaches to solve optimal control problems involve solving a series of simpler convex programs till convergence is achieved. Such methods have been applied to a range of challenging guidance and control problems including planetary entry [97, 182], rocket landing [95], orbit transfer [181], proximity operations [101], space robot motion planning [108] among others. The main operating principle behind most approaches leveraging convex programming is linearizing the state dynamics and convexifying the non-convex inequality constraints. This results in a convex subproblem assuming a convex cost function. To ensure that linearization does not result in an infeasible convex problem, virtual control type terms can be introduced [105]. To mitigate the risk of unboundedness, a trust region constraint commonly seen in sequential quadratic programming methods is typically imposed. Using a penalty based approach, the problem is iteratively solved until convergence is achieved. Such class of methods have been numerically validated for quadrotor maneuvers in presence of obstacles [168], and model predictive control [104]. Wang et al. [177] combine the advantages of pseudo-spectral discretization and sequential convex optimization to design optimal guidance laws for fuel-optimal rocket landing problems in a receding horizon setting. This method also implements linearized dynamics and constraints in a trust region framework.

A disadvantage of the linearized trust region, penalty based methods proposed above is the feasibility of the iterates. Most methods only guarantee feasibility of the solutions when they converge. For highly constrained, challenging problems, the mission may require early termination of the optimal control algorithm and recover a sub-optimal but feasible solution. Using general pseudo-spectral based methods or iterative relaxed convex solutions, this is not possible as the iterates are often in-admissible and convergence is non-monotonic, meaning, that the objective function cost may increase between iterations. Zhang et al. [198] address the iterative feasibility challenge by representing the optimization problem in the framework of the Convex-Concave procedure [192], a specialized method in Difference of Convex (DC) programming [60]. Under the assumption that the initial guess is feasible, the admissibility of the iterates is guaranteed in this method. Furthermore, the global convergence of the algorithm is proved under mild assumptions. Although the method proposed is general in [198], only concave inequality constraints pertaining to obstacle avoidance are studied. More importantly, how to tractably numerically obtain difference of convex functions decomposition is not discussed. More recently, Virgili-Llop and Romano [175] also propose another variant of the recursively feasibly sequential convex programming method with linear dynamics and apply it to the spacecraft reorientation problem. The method proposed is similar in spirit to the technique discussed and studied in this paper. However, contrary to our approach, the key idea behind maintaining recursive feasibility of the iterates is the inner convex approximation of the non-convex constraints using Taylor expansions. This can be understood as a higher order regularization based on the convexification of the higher order (greater or equal to order 3) terms of a Taylor series expansion.

In this chapter, we propose an iterative method for solving nonlinear optimal control problems using convex optimization assuming that the state dynamics is linear. The method works for general non-convex constraints without restriction on the type of path constraints. By representing the inequality path constraints as difference of convex functions, the convex-concave procedure is implemented to iteratively solve the transcribed optimization problem. The main contributions of the method are: 1.) We propose a sequential convex approach to solve optimal control problems with non-convex path constraints with guaranteed feasibility assuming a feasible initial guess without requiring trust region constraints; 2.) Using recent advances in decomposition of polynomials as the difference of convex sum-of-squares (SOS) polynomials [6], we provide a systematic and tractable way to deal with non-convex path constraints such as obstacle avoidance; 3.) We provide convergence proofs and solve an *anytime* feasible and sub-optimal problem for a spacecraft relative guidance scenario in cluttered environments. The remainder of the chapter is as follows, Section 7.1 introduces the general nonlinear optimal control problem and it's transcription along with the DC programming approach to solve the optimal control iteratively. Theoretical proofs for convergence is provided in Section 7.2 with theoretical proofs on convergence; Section7.3 introduces the optimal relative spacecraft guidance problem with numerical validation of the approach.

7.1 Problem formulation

Consider a generic continuous optimal control problem given below

$$\begin{aligned} \underset{u(t),t_0,t_f}{\text{minimize}} \quad & \int_{t_0}^{t_f} \mathbf{H}(t, x(t), u(t)) + \mathbf{M}(t_0, t_f, x(t_f)) \\ \text{subject to} \quad & \dot{x}(t) = f(t, x(t), u(t)) \\ & C(t, x(t), u(t) \leq 0 \\ & \Phi(x(t_f), u(t_f))) \leq 0 \end{aligned} \tag{7.1}$$

where, the cost function is expressed in the Bolza form, and subjected to dynamics represented by $f(t, x(t), u(t)$, path constraints $C(t, x(t), u(t)$ and terminal constraints $\Phi(x(t_f), u(t_f))$. The problem expressed in Eq. 7.1 is infinite-dimensional. To practically solve this problem, we can transcribe or convert the infinite-dimensional problem into a finite-dimensional approximation. This can be undertaken in three major steps [16]:

1. Convert the dynamical system into a problem with finite number of variables. For example: using pseudospectral [41] or time-marching methods [141].

2. Solve the finite-dimensional problem using nonlinear optimization.

3. Assess the accuracy and repeat the transcription process to improve performance if necessary.

Due to challenges in obtaining optimal solutions in near-real time, methods based on convex relaxations have gained attention. Sequential convex programming (SCP) methods have been proposed as a viable alternative to transcription based nonlinear programming approach in solving optimal control problems. While several variants of SCP have been proposed, the approach typically solves a sequence of relaxed convex problems iteratively till convergence is achieved. Penalty methods allow constrained optimization problem to be solved as unconstrained optimization problem by augmenting the objective function with a penalty function with an associated penalty parameter. This ensures that the problem remains feasible as long as the joint state and control bounds are respected. A simple penalty method is expressed as

$$\begin{aligned} \underset{y \in \mathbb{R}^n}{\text{minimize}} \quad & J(y) + \lambda\big(\upsilon(\Gamma(y))\big) \\ \text{subject to} \quad & |y - y_k| \leq \rho \end{aligned} \tag{7.2}$$

where $J(y) : \mathbb{R}^n \to \mathbb{R}$ is the original objective function of the constrained optimal control problem in discrete form, $\lambda \in \mathbb{R}_{>0}$ and $\upsilon(.) : \mathbb{R}^n \to \mathbb{R}$ are the penalty parameter and penalty function, respectively, and $\Gamma(y) : \mathbb{R}^m \to \mathbb{R}^n$ denote the linearized equality and convexified inequality constraints which include the state dynamics and path constraints. For a sufficiently large value of λ, the solution of the unconstrained optimization problem is equal to the constrained problem. To ensure that the solution does not diverge, a trust region constraint parametrized by radius ρ is used, where y_k denotes the solution at the previous iteration. Note that due to linearization of dynamics and path constraints with the penalty implementation, the solution at intermediate iterations may not be feasible. Furthermore, the objective function may converge non-monotonically to the local optimum solution. For highly constrained scenarios where near real-time solutions are desirable, it is beneficial to use an approach that allows an early termination of the iterative algorithm where the solution is at least feasible and naturally sub-optimal. To this end, a particularly promising approach is the DC programming method which is discussed below

7.1.1 DC programming

The Difference of convex function (DC) algorithm is one of several methods to solve non-convex optimization problems. A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be a difference of convex (DC) function if there exists convex functions g and h $: \mathbb{R}^n \to \mathbb{R}$ such that $f = g - h$. DC problems have the form

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & g_0(x) - h_0(x) \\ \text{subject to} \quad & g_i(x) - h_i(x) \leq 0, \quad , i = 1, ..., m \end{aligned} \tag{7.3}$$

The class of DC functions is quite broad, which includes every twice differentiable function with a bounded Hessian [56]. DC problems arise in diverse applications and areas including feedback control [24], machine learning [86] and statistics [170]. Traditionally, the approach to solve DC programs can be broadly classified into global and local methods. Global methods such as branch and bound [85] are quite effective but can be slow in practice. On the other hand, locally optimal solutions to DC problems can be obtained using heuristic algorithms such as the convex-concave procedure (CCP) [192]. In the basic version of the CCP procedure, the concave part of the objective and inequality constraints are replaced by convex upper bounds and the resulting problem is solved iteratively until convergence is achieved. This is shown in the algorithm below Note that

Algorithm 2 Convex-Concave Procedure

```
Given: Initial feasible point $x_0$, $k := 0$
while convergence not achieved do
    Convexify $\hat{h}_i(x, x_k) \triangleq h_i(x_k) + \nabla h_i(x_k)^{\mathrm{T}}(x - x_k)$ for $i = 0, 1, ...m$
    Solve
        $\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad g_0(x) - \hat{h}_0(x, x_k)$
        $\text{subject to} \quad g_i(x) - \hat{h}_i(x, x_k) \leq 0, \quad , i = 1, ..., m$
    Update iteration $k := k + 1$
end while
```

Algorithm 2 requires a feasible initial guess. In [93], the infeasibility of the initial guess is addressed in a penalty framework. The problem is relaxed by adding slack variables to the constraints, $g_i(x) - \hat{h}_i(x, x_k) \leq s_i$ and the sum of the constraint violations are penalized by augmenting the objective function. A distinct advantage of the CCP is that it does not require trust region constraints typically seen in other methods such as sequential quadratic programming. In addition, in the SQP method, a quadratic program is solved with linear constraints. Therefore, all the information above the second order is lost during the iterations. On the contrary, the CCP retains the convexity of the first part and only linearizes the concave portion of the objective and inequality constraints.

7.1.2 Optimal control with linear dynamics and non-convex path constraints

Consider the continuous optimal control problem shown in Eq. 7.1. Using either time-marching or pseudo-spectral discretization, the continuous optimal control can be transcribed into a static optimization problem. Let the underlying system dynamics be linear. In concise form, this is expressed as

$$\begin{aligned} \underset{y}{\text{minimize}} \quad & f_0(y) \\ \text{subject to} \quad & f_i(y) \leq 0, \quad , i = 1, ..., m \\ & py + s = 0 \end{aligned} \tag{7.4}$$

where $y = [x, u]$ contains both the state and control as the decision variables, $f_0(y)$ is the discretized objective function, the linear equality constraints denote the discretized state dynamics, and the inequality path constraints are expressed via $f_i(y)$. The equality constraint represents the linear dynamics and is linear. In these cases, the main source of non-convexity arises from the non-convex path constraints. For spacecraft proximity operations, such constraints can include keep-out zones, obstacle avoidance among others. While DC decomposition of functions is a known result, specific results on how to obtain such decompositions have been limited. A *universal* DC decomposition technique exploits the bounded Hessian property of the function and proposes the following DC decomposition [125]

$$f(x) = (\frac{1}{2}\rho^* \|x\|^2) - (\frac{1}{2}\rho^* \|x\|^2 - f(x)) \tag{7.5}$$

where ρ^* is a constant satisfying $\rho^* \geq \max_x \rho(\nabla^2 f(x))$. However, this decomposition may not be optimal. Furthermore, in order to calculate ρ^*, the spectral radius of the hessian of the function needs to be computed which can be computationally challenging for highly non-convex functions. In Eq. 7.5, a possible decomposition is shown based on the spectral radius of the hessian of the function $f(x)$. Naturally, the decomposition is not unique. For any decomposition, $f = g - h$, $f = g - p - h + p$ is another equivalent decomposition. This paves the way for the question: how to obtain an optimal decomposition of a given non-convex function?.

Definition 3. *Let $y = [y_1, ..., y_n]^T$ be a vector of variables. A polynomial $b(x)$ is said to be SOS-convex if $y^T H_{b(x)} y$ is convex for all x and y where $H_{b(x)}$ denotes the Hessian of $b(x)$ [80].*

Instead of computing convex functions g and h such that $f = g - h$, it is possible to make the problem computationally tractable by looking at the subset of convex functions g and h that are also sum-of-squares. This is made possible by posing the problem as a sum-of-squares problem which can be solved efficiently using semidefinite programming. The feasibility problem to find the decomposition of $f(x)$ is given by [6]

$$\begin{aligned} \underset{g,h}{\text{minimize}} \quad & - \\ \text{subject to} \quad & f = g - h \\ & z^{\mathrm{T}} H_{g(x)} z \succeq 0 \\ & z^{\mathrm{T}} H_{h(x))} z \succeq 0 \end{aligned} \tag{7.6}$$

where $z \in \mathbb{R}^n$. Note that the non-convex constraint $f(x)$ is convexified at a given iterate x_k as $g(x) - \big(h(x_k) + \nabla h(x_k)^{\mathrm{T}}(x - x_k)\big)$. An optimal decomposition for the polynomial $f(x)$ is to compute $h(x)$ such that it is as close to an affine function as possible around the iterate x_k. This is mathematically expressed as [6]

$$\begin{aligned} \underset{g,h}{\text{minimize}} \quad & \mathrm{Tr} H_h(x_k) \\ \text{subject to} \quad & f = g - h \\ & z^{\mathrm{T}} H_{g(x)} z \succeq 0 \\ & z^{\mathrm{T}} H_{h(x))} z \succeq 0 \end{aligned} \tag{7.7}$$

where Tr denotes the trace. Eq. 7.7 computes g and h such that $h(x)$ minimum average curvature around the point x_k. Alternatively, it is possible to minimize the worst case curvature by minimizing $\lambda_{max} H_h(x_k)$, where λ_{max} is the maximum eigenvalue of the Hessian of $h(x)$ evaluated at point x_k. Another notion in DC decomposition is that of undominated decomposition [23].

Definition 4. *A g be a DC decomposition of f, that is, g is convex and $f - g$ is convex. A function $\bar{g}$ is said to dominate g if $g - \bar{g}$ is convex and non-affine. A DC decomposition g of f is undominated if no other DC decomposition of f dominates g.*

The undominated decomposition problem is expressed as

$$\begin{aligned} \underset{g,h}{\text{minimize}} \quad & \frac{1}{\mathcal{A}_n}\int_{\mathcal{S}^{n-1}} \text{Tr}H_g d\sigma \\ \text{subject to} \quad & f = g - h \\ & z^{\mathrm{T}} H_{g(x)} z \succeq 0 \\ & z^{\mathrm{T}} H_{h(x))} z \succeq 0 \end{aligned} \tag{7.8}$$

where $\mathcal{A}_n = \frac{2\pi^{n/2}}{\Gamma(n/2)}$ is a normalization constant, $\Gamma(n/2)$ is the gamma function, and $\mathcal{S}^{n-1}$ denotes the unit sphere in $\mathbb{R}^n$.

The promising feature with the decomposition methods shown in Eqs. 7.6-7.8 is that they can be posed as sum-of-squares problems and solved using semidefinite programming.

7.2 Optimal control algorithm and convergence proof

Using the DC decomposition techniques mentioned above, an iterative scheme to solve the transcribed optimal control problem in Eq. 7.4 is solved via the following algorithm

Algorithm 3 Iterative optimal control algorithm

```
**Given:** Initial feasible trajectory $y_0 = [x_0, u_0]$, $k \coloneqq 0$
**while** $\|f_0(y_k) - f_0(y_{k-1})\| \geq \beta)$ **do**
    Decompose non-convex inequality constraint and linearize concave part in Eq. 7.4
        $\hat{f}_i(y, y_k) \triangleq g_i(y) - \big(h_i(y_k) + \nabla h_i(y_k)^{\mathrm{T}}(y - y_k)\big)\text{for}i = 0, 1, ...m \quad (7.9)$
    Solve
        $\underset{y}{\text{minimize}} \quad f_0(y)$
        $\text{subject to} \quad py + s = 0 \quad (7.10)$
        $\qquad g_i(y) - \big(h_i(y_k) + \nabla h_i(y_k)^{\mathrm{T}}(y - y_k)\big) \leq 0, \quad , i = 1, ..., m$
    Update iteration $k \coloneqq k + 1$
**end while**
```

Lemma 2. *The function $\hat{f}_i(y, y_k)$ is an inner convex approximation of the original non-convex constraint function $f_i(x)$, for $i = 0, 1, ..., m$ and satisfies*

$$f_i(y) \leq \hat{f}_i(y, y_k) \tag{7.11}$$

$$f_i(y_k) = \hat{f}_i(y_k, y_k) \tag{7.12}$$

$$\frac{\partial f_i(y_k)}{\partial y_j} = \frac{\partial \hat{f}_i(y_k, y_k)}{\partial y_j} \tag{7.13}$$

Proof: Let the solution to the convexified problem in Eq. 7.10 be denoted as y_{k+1}. Since y_{k+1} is an optimal solution to the convexified problem, it satisfies the inequality constraint

$$g_i(y_{k+1}) - h_i^l(y_{k+1}, y_k) \leq 0, \quad \forall i = 1, 2..., n \tag{7.14}$$

where $h^l(y_{k+1}, y_k) = h(y_{k+1}) + \nabla h_i(y_k)^{\mathrm{T}}(y_{k+1} - y_k\big)$. Evaluating the original non-convex constraint in Eq. 7.4 at y_{k+1}

$$g_i(y_{k+1}) - h_i(y_{k+1}), \quad \forall i = 1, 2..., n \tag{7.15}$$

Taking second order Taylor expansion of $h(y_{k+1})$ in Eq. 7.15

$$g_i(y_{k+1}) - h^l(y_{k+1}, y_k) - \ddot{h}(y_{k+1}, y_k) \tag{7.16}$$

Since $h_i(y)$ is a convex function, $H_{h_i}(y) \succ 0$ and therefore $\ddot{h}(y_{k+1}, y_k) \geq 0$. This results in the following inequality

$$g_i(y_{k+1}) - h_i(y_{k+1}) \leq g_i(y_{k+1}) - h^l(y_{k+1}, y_k) - \ddot{h}(y_{k+1}, y_k) \leq 0 \tag{7.17}$$

Thus, we have

$$f_i(y_{k+1}) \leq \hat{f}(y_{k+1}, y_k) - \ddot{h}(y_{k+1}, y_k) \tag{7.18}$$

This completes the proof for Eq. 7.11 in the lemma. The proof for Eqs. 7.12-7.13 is trivial. Evaluating $\hat{f}(y, y_k)$ and $\frac{\partial \hat{f}_i(y,y_k)}{\partial y_j}$ at point y_k returns the original nonlinear function and its gradient respectively. □

Lemma 3. *The optimal solution y_{k+1} at iteration $k+1$ for the convexified problem in Eq. 4 is feasible for the original non-convex problem expressed in Eq. 7.4*

Proof: To ensure dynamics and path constraint feasibility in Eq. 7.4, we require

$$f_i(y_{k+1}) = g_i(y_{k+1}) - h_i(y_{k+1}) \leq 0, \quad , i = 1, ..., m \tag{7.19}$$

$$py_{k+1} + s = 0 \tag{7.20}$$

Eq. 7.20 refers to the linear dynamics and is automatically satisfied for Eq. 7.4. For Eq. 7.19, we use the following property: Considering that $h_i(y)$ is a twice differentiable convex function, a quadratic upper bound exists and is expressed as [17]

$$h_i(y) \leq h_i(y_k) + \nabla h_i(y_k)^{\mathrm{T}}(y - y_k) + \frac{M}{2}\|y - y_k\|^2 \tag{7.21}$$

where the above expression holds for any y_k and $M \geq \max_y \nabla^2 h(y)$. Using the upper bound of $h_i(y)$ at point y_{k+1} in Eq. 7.19

$$g_i(y_{k+1}) - h_i(y_{k+1}) \leq g_i(y_{k+1}) - h_i(y_k) - \nabla h_i(y_k)^{\mathrm{T}}(y_{k+1} - y_k) - \frac{M}{2}\|y_{k+1} - y_k\|^2 \tag{7.22}$$

$$g_i(y_{k+1}) - h_i(y_{k+1}) \leq g_i(y_{k+1}) - \hat{f}(y_{k+1}) - \frac{M}{2}\|y_{k+1} - y_k\|^2 \leq 0 \tag{7.23}$$

Note that $\hat{f}(y_{k+1}) \leq 0$ are the convexified path constraints and are always satisfied since the point y_{k+1} is feasible/optimal. Furthermore $M > 0$, which implies that the nonconvex path constraints expressed in Eq. 7.4 are satisfied. Note that we can also directly prove Eq. 7.19 using the inner convex approximation argument in Lemma 2. This completes the proof. □

Lemma 4. *The iterative Algorithm 3 is a descent (monotone) algorithm.*

Proof: Consider $f_0(y)$ as an arbitrary non-convex cost function. The DC decomposition of $f_0(y)$ is given as $f_0(y) = g_0(y) - h_0(y)$.

Let the solution to the convexified problem be given by y_{k+1}. The objective function in Eq. 4 is then expressed as

$$g_0(y_{k+1}) - h_{l0}(y_{k+1}, y_k) \tag{7.24}$$

where $h_{l0}(y_{k+1}, y_k) = h_0(y_{k+1}) + \nabla h_0(y_k)^{\mathrm{T}}(y_{k+1} - y_k\big)$. Thus,

$$g_0(y_{k+1}) - h_{l0}(y_{k+1}, y_k) \leq g_0(y_{k+1}) - h_{l0}(y_k, y_k) \tag{7.25}$$

Thus, for the non-convex objective function, the following inequality holds

$$\begin{aligned} g_0(y_{k+1}) - h_0(y_{k+1}) &\leq g_0(y_{k+1}) - h_{l0}(y_{k+1}, y_k) \\ &\leq g_0(y_k) - h_0(y_k) \leq g_0(y_k) - h_{l0}(y_{k+1}, y_k) \end{aligned} \tag{7.26}$$

This completes the proof. □

Theorem 5. *The fixed point of the sequence generated by the iterative algorithm is a KKT point to the original non-convex problem.*

proof: Suppose y^* is the optimal solution obtained from the converged sequence in Algorithm 3. The Lagrangian for the convexified problem is given as

$$L(\lambda_i, \mu) = f_0(y) + \sum_{i=1}^{m} \lambda_i \big(g_i(y) - h_l(y, y_k)\big) + \mu^{\mathrm{T}}(py + s) \tag{7.27}$$

where λ, μ are the Lagrange multipliers. The KKT conditions for the convexified problem is then expressed as

$$\nabla L(\lambda_i, \mu) = \nabla f_0 + \sum_{i=1}^{m} \lambda_i \big(\nabla g_i(y) - \nabla h_l(y, y_k)\big) \tag{7.28}$$

$$\lambda_i (g_i(y) - h_l(y, y_k) = 0 \tag{7.29}$$

$$\lambda_i \geq 0 \tag{7.30}$$

Since, y^* is the optimal solution for the problem, subsituting for y^* in y and y_k, we obtain

$$\nabla L(\lambda_i, \mu) = \nabla f_0 + \sum_{i=1}^{m} \lambda_i \big(\nabla g_i(y^*) - \nabla h_l(y^*, y^*)\big) \tag{7.31}$$

$$\lambda_i (g_i(y^*) - h_l(y^*, y^*) = 0 \tag{7.32}$$

$$\lambda_i \geq 0 \tag{7.33}$$

Note that since $\nabla h_l(y^*, y^*) = \nabla h(y^*)$ and $h_l(y^*, y^*) = h(y^*)$ from Lemma 2, the point y^* satisfies the KKT conditions for the original problem in Eq. 7.4 as well. This completes the proof. □

7.3 Numerical demonstration of optimal relative spacecraft guidance

The iteratively feasible sequential convex programming approach is studied using an optimal spacecraft guidance problem. The dynamics of the chaser spacecraft in a circular orbit in proximity of a target in a circular orbit is expressed using the Clohessy-Hill-Wiltshire equations. The dynamics are given as

$$\ddot{x} = 3n^2x + 2x\dot{y} \tag{7.34}$$

$$\ddot{y} = -2n\dot{x} \tag{7.35}$$

$$\ddot{z} - -n^2z \tag{7.36}$$

where $n = \sqrt{\frac{\mu}{a^3}}$ is the orbital rate, μ is the gravitational parameter, and a is the radius of the circular orbit. The key challenge in finding the optimal guidance law is the non-convex obstacle avoidance constraint. Using simplified circular/spherical shape of the obstacle, concave inequality constraints are used in [198] to account for obstacle avoidance. However, this assumption does not work with non-convex obstacles and convex hull approximations of the obstacles may not be optimal. The optimal guidance problem in discrete domain is expressed as

$$\begin{aligned}
\underset{x,u}{\text{minimize}} \quad & \sum_{k=0}^{N} \eta_k \\
\text{subject to} \quad & X_{k+1} = AX_k + BU_k \\
& \|U_k\| \leq \eta_k \\
& d_k^{\mathrm{T}} U_k \leq \|U_k\| \cos\theta \\
& \|U_k\| \leq U_{max} \\
& -x_k^4 - y_k^4 - z_k^4 + x_k y_k z_k^2 - x_k^2 y_k z_k + 200 \leq 0 \qquad (a) \\
& -(x_k^2 - y_k^2)^2 - z_k^4 + 75 \leq 0 \qquad (b) \\
& -(z_k^2 - y_k^2)^2 - x_k^4 + 1 \leq 0 \qquad (c) \\
& x_0 = x0, x_N = xf
\end{aligned} \tag{7.37}$$

The objective of the guidance problem is to minimize the total control norm which in discrete domain is expressed as $\sum_{k=0}^{N} \|U_k\|$. Although the objective is

Table 7.1: Non-convex constraint decomposition

Constraint	Undominated solution
$\mathrm{a} := -x_k^4 - y_k^4 - z_k^4 + x_k y_k z_k^2$ $-x_k^2 y_k z_k + 200$	$\mathrm{g} := 0.583\mathrm{x}_k^4 + 0.5y_k^4 + 0.75y_k^2 x_k^2 + 0.583z_k^4$ $+0.5z_k^2 x_k^2 + 0.75z_k^2 y_k^2 - 0.5z_k y_k x_k^2 + 0.5z_k^2 y_k x_k$ $\mathrm{h} := \text{-}200 + 1.5833\mathrm{x}_k^4 + 1.5y_k^4 - 0.75y_k^2 x_k^2$ $+1.588z_k^4 + 0.5z_k y_k x_k^2 - 0.5z_k^2 y_k x_k$
$b := -(x_k^2 - y_k^2)^2 - z_k^4 + 75$	$\mathrm{g} := 0.33x_k^4 + 0.33y_k^4 + 2y_k^2 x_k^2$ $h := -75 + 1.33x_k^4 + 1.33y_k^4$
$c := -(z_k^2 - y_k^2)^2 - x_k^4 + 1$	$\mathrm{g} := 0.33\mathrm{z}_k^4 + 0.33y_k^4 + 2z_k^2 y_k^2$ $h := -1 + 1.33y_k^4 + 1.33y_k^4$

non-differentiable, using slack variables, the objective has been converted to a second order cone constraint. Plume impingement constraints are also implemented where $d_k \in \mathbb{R}^3$ denotes the docking axis fixed on the target. In order to solve Eq. 7.37, the non-convex obstacle avoidance constraints expressed in (a-c) are decomposed by using the notion of undominated decompositions. The obstacle avoidance constraints $(a-c)$ are created to model a fairly complex, realistic target spacecraft structure with multiple appendages. In particular, constraint a models a central non-convex shaped body resembling a spacecraft bus, and constraints b and c model appendages such as external solar panels and radiator shades. Table 7.1 provides the non-convex constraints $a - c$ as well as their decomposition in terms of SOS-convex DC functions. The decomposition is carried out via the method of undominated decompositions discussed in Eq 7.8 and using the SPOT toolbox on MATLAB [107]. The problem studied here can be considered as a near-field rendezvous where a chaser is required to be optimal guided in close proximity to a target spacecraft while ensuring that the chaser remains safe from any collisions.

A circular orbit with radius 800 km is considered. The initial position and velocity of the spacecraft is taken as $x_0 = [-40, 15, -15, 0, 0, 0]^{\mathrm{T}}$ and the final position and velocity is taken as $x_f = [7, 0, 2, 0, 0, 0]$. The problem is initialized by solving Eq. 7.37 without the objective, i.e, a feasible initial guess is generated

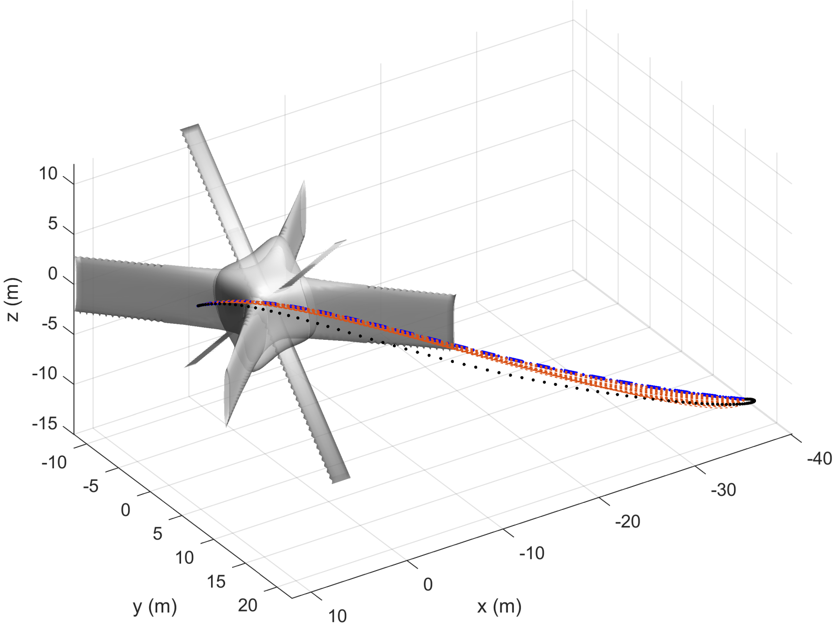


Figure 7.1: Three dimensional spacecraft trajectory around a non-convex shaped target shown in gray. Orange trajectories denote intermediate sub-optimal and feasible iterates. The black and blue trajectories denote the initial guess and converged solution, respectively.

using the interior point solver IPOPT. The problem in Eq. 7.37 is then convexified by linearizing the concave part of the decomposed non-convex obstacle avoidance constraints. This problem is iteratively solved until convergence is achieved.

Figs. 7.1 and 7.2 illustrate alternate views of the three-dimensional spacecraft trajectories in presence of a non-convex target. The XY projection of the trajectory is shown in Fig. 7.3. It is observed that all the iterates shown in orange remain feasible and away from the target body. Fig. 7.4 shows the velocity profile of the spacecraft. The control profiles are shown in Fig. 7.5. The control constraint ensures that the accelerations do not exceed the maximum threshold taken as 1 m/s^2. The variation of the objective value is shown in Fig. 7.6 where convergence is achieved in approximately 32 iterations. Monotonic convergence of the objective is observed which is expected from the convex-concave procedure.

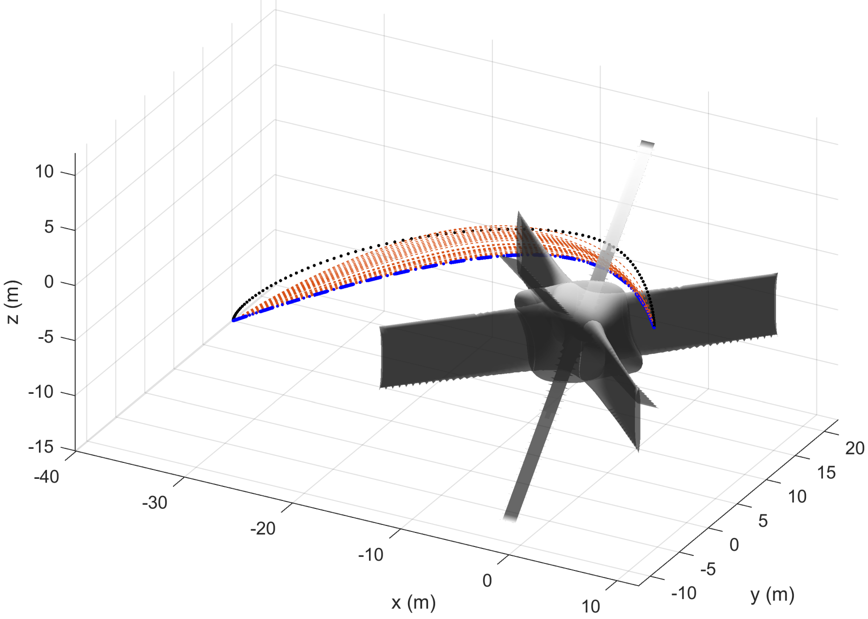


Figure 7.2: Alternate view of three dimensional spacecraft trajectory

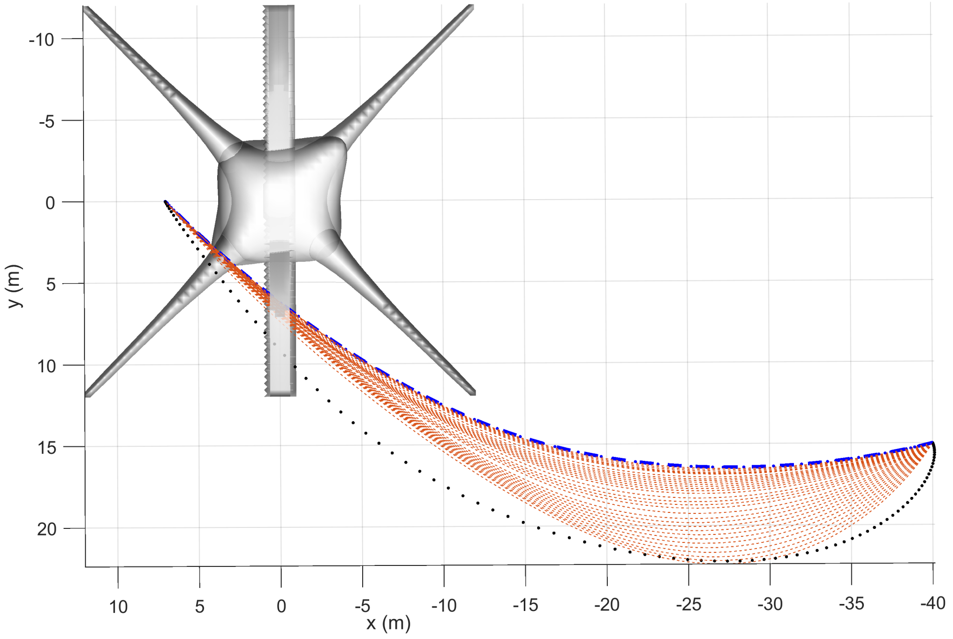


Figure 7.3: XY projection of spacecraft trajectory

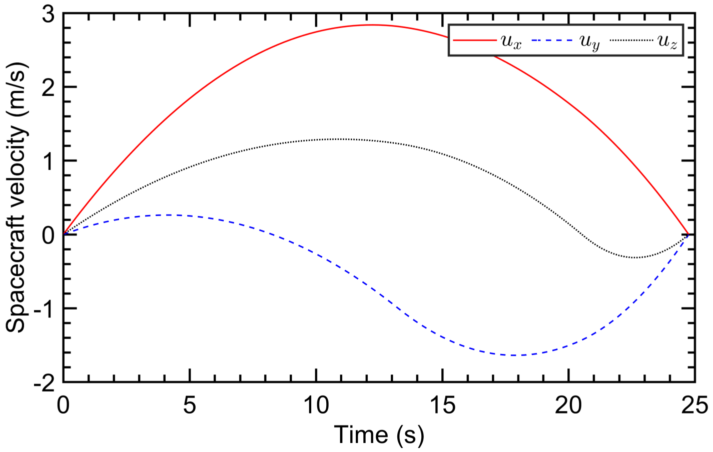


Figure 7.4: Spacecraft velocity profiles

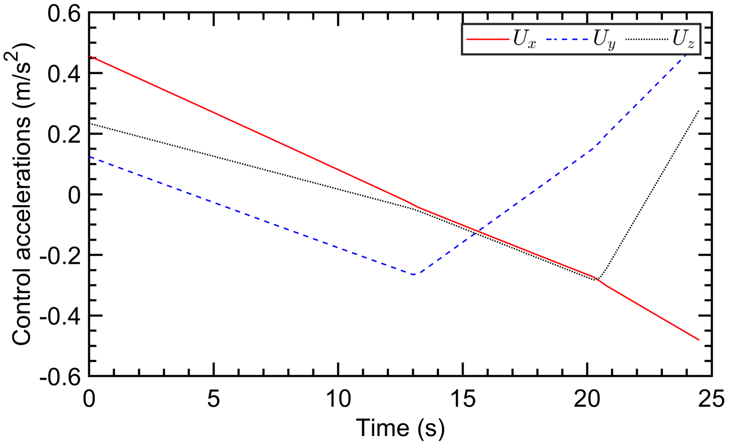


Figure 7.5: Spacecraft control profiles

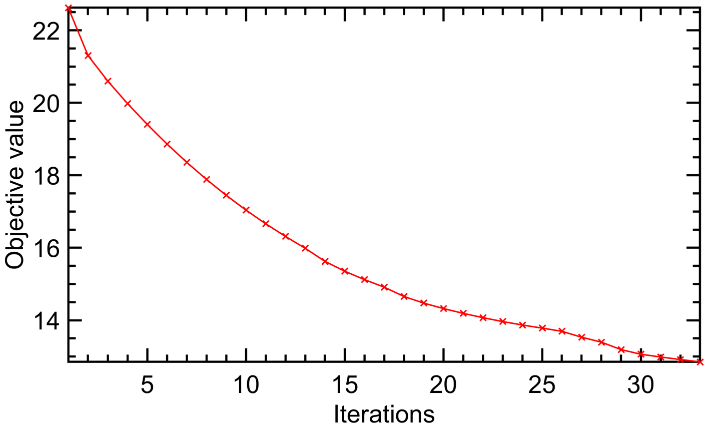


Figure 7.6: Objective value

Chapter 8
Conclusions and future work

8.1 Summary and conclusions

Motivated by recent advances in computational guidance and control of space systems, this dissertation focuses on advances in robust, real-time, and optimal planning and control of space-robotic systems using convex programming.

The classical problem of planning the motion of a free-floating space manipulator is reformulated and solved as a quadratic program. The method is general and accommodates different types of constraints including minimizing base rotational disturbance, obstacle avoidance, and end-effector task constraints. Compared with current approaches using non-convex and sampling based techniques, the proposed quadratic programming method is several orders of magnitude faster. This enables rapid planning and re-planning of space robot trajectories which is suitable in a dynamically changing environment.

For a spacecraft in the circular restricted three-body dynamical regime, the station-keeping problem is studied for Halo orbits in the Sun-Earth system and Lissajous orbits in the Earth-Moon system. Given a desired reference trajectory to follow, the equations of motion for the spacecraft are expanded up-to third degree using Taylor expansions. The resulting polynomial system serves as the dynamics model which is used in a receding horizon setting. Contrary to local nonlinear approaches, the receding horizon problem is converted to a polynomial optimization problem and solved to global optimality using sum-of-squares relaxations. The globally optimal model predictive control approach provides lower

ΔV budget as compared to other competing approaches such as linear quadratic regulator and linear and locally optimal nonlinear Model predictive control.

Robust control of nonlinear polynomial systems is addressed in the framework of nonlinear disturbance observers. The synthesis of these observers is extremely challenging for general nonlinear systems. By restricting the focus on systems with polynomial vector or matrix fields, the observer synthesis problem is posed and solved as a polynomial optimization using the sum-of-squares methodology. In addition, the design of a compensation matrix for systems with mismatched uncertainties is numerically validated using computational Lyapunov analysis. To further illustrate the efficacy of this framework, the robust rigid-body relative attitude tracking problem is studied where disturbance torques are assumed to act on both the chaser and target spacecraft. The robustness of the disturbance observer is verified using numerical simulations with a simple proportional-derivative feedback control scheme.

The sequential convex approach for solving nonlinear optimal control problems is also addressed. To meet the demands of rapid generation of optimal trajectories for problems with non-convex constraints and linear dynamics, an iteratively feasible technique is investigated. Contrary to existing methods such as the penalty based trust region approach, the technique shown in this dissertation maintains *any-time* feasibility and sub-optimality of the iterates. This is made possible by leveraging difference of convex function programming for polynomial non-convex constraints. As an application, the optimal spacecraft guidance problem in the framework of Clohessy-Hill-Wiltshire dynamics is solved. Although the underlying dynamics are linear, the problem is complicated due to the presence of non-convex obstacle avoidance constraints. By optimally decomposing these non-convex constraints as the difference of two convex functions, the recursive feasibility of the iterates is preserved.

8.2 Recommendations for future work

Based on the convex programming tools developed to solve a diverse range of space-robotic planning and control problems, this thesis opens several avenues for improvement.

- In Chapter-4, the trajectory planning of free-floating space robots only focuses on kinematically feasible trajectories. While theoretically the approach is suitable for kino-dynamic trajectories as well, it has not been numerically validated. Experimental validation of such trajectories in a free-floating setting is also of interest. Lastly, robustness of the planning technique to external disturbances can be incorporated in the framework of robust optimization.

- A significant challenge in implementing the polynomial optimization based globally optimal Model Predictive Control approach is the computational complexity since it requires solving large dimensional semi-definite programs. The reduction of problem structure and incorporating sparsity can be explored to make this approach suitable for real-time applications.

- The disturbance observer framework proposed generally works well and does not require any assumptions on the disturbance structure. However, it would be interesting to study outlier cases of external disturbances where the proposed approach may not work. Improvements in theory need to be made to solve problems where the disturbance is fast-varying with time. Secondly, in the framework proposed, the observer design is considered separate and decoupled and as such the designed observer can be used as a patch with any asymptotically feasible control law. However, this may result in actuator saturation especially if the disturbance estimate is large and converges slowly to the actual disturbance. In these settings, the investigation of saturated composite observer-control laws will be of interest.

- The sequential convex approach to solving nonlinear optimal control problems only consider non-convex, polynomial constraints to ensure iterate feasibility. Future work is needed to make the method proposed in this dissertation amenable to non-linear dynamics. In this setting, the investigation of inner convex approximations of nonlinear dynamics either through feedback/partial-feedback linearization or coordinate transformations will be useful.

References

[1] ARIADNA call for ideas: Active removal of space debris. Technical report, European Space Agency, 2010.

[2] Behcet Acikmese and Scott R Ploen. Convex programming approach to powered descent guidance for mars landing. *Journal of Guidance, Control, and Dynamics*, 30(5):1353–1366, 2007.

[3] Behcet Acikmese, Daniel Scharf, Lars Blackmore, and Aron Wolf. Enhancements on the convex programming based powered descent guidance algorithm for mars landing. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 6426, 2008.

[4] Farhad Aghili. Pre-and post-grasping robot motion planning to capture and stabilize a tumbling/drifting free-floater with uncertain dynamics. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 5461–5468. IEEE, 2013.

[5] Om P Agrawal and Yangsheng Xu. On the global optimum path planning for redundant space manipulators. *Systems, Man and Cybernetics, IEEE Transactions on*, 24(9):1306–1316, 1994.

[6] Amir Ali Ahmadi and Georgina Hall. DC decomposition of nonconvex polynomials with algebraic techniques. *Mathematical Programming*, 169(1):69–94, May 2018.

[7] Jasim Ahmed, Vincent T Coppola, and Dennis S Bernstein. Adaptive asymptotic tracking of spacecraft attitude motion with inertia matrix identification. *Journal of Guidance, Control, and Dynamics*, 21(5):684–691, 1998.

[8] Buzz Aldrin. *Line-of-sight guidance techniques for manned orbital rendezvous*. PhD thesis, Massachusetts Institute of Technology, 1963.

[9] Claudio Altafini. The de casteljau algorithm on SE (3). In *Nonlinear control in the year 2000*, pages 23–34. Springer, 2001.

[10] Claudio Altafini. Inverse kinematics along a geometric spline for a holonomic mobile manipulator. In *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 2, pages 1265–1270. IEEE, 2001.

[11] Federico Augugliaro, Angela P Schoellig, and Raffaello D'Andrea. Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1917–1922. IEEE, 2012.

[12] Xiaoli Bai and John L Junkins. Modified chebyshev-picard iteration methods for station-keeping of translunar halo orbits. *Mathematical Problems in Engineering*, 2012, 2012.

[13] Calin Belta and Vijay Kumar. An svd-based projection method for interpolation on se (3). *IEEE transactions on Robotics and Automation*, 18(3):334–345, 2002.

[14] Gino van den Bergen. A fast and robust gjk implementation for collision detection of convex objects. *Journal of graphics tools*, 4(2):7–25, 1999.

[15] Lasserre Jean Bernard. *Moments, positive polynomials and their applications*, volume 1. World Scientific, 2009.

[16] John T Betts. *Practical methods for optimal control and estimation using nonlinear programming*, volume 19. Siam, 2010.

[17] Dankmar Bhning and Bruce G. Lindsay. Monotonicity of quadratic-approximation algorithms. *Annals of the Institute of Statistical Mathematics*, 40(4):641–663, December 1988.

[18] William N Bittle. Collision detection using the gjk algorithm. *Algorithmic and Architectural Gaming Design: Implementation and Development: Implementation and Development*, page 253, 2012.

[19] Lars Blackmore, Behçet Açıkmeşe, and John M Carson. Lossless convexification of control constraints for a class of nonlinear optimal control problems. *Systems & Control Letters*, 61(8):863–870, 2012.

[20] Lars Blackmore, Behcet Acikmese, and Daniel P Scharf. Minimum-landing-error powered-descent guidance for mars landing using convex optimization. *Journal of guidance, control, and dynamics*, 33(4):1161–1171, 2010.

[21] Lars Blackmore and Brian Williams. Optimal manipulator path planning with obstacles using disjunctive programming. In *American Control Conference, 2006*, pages 3–pp. IEEE, 2006.

[22] DART Mishap Investigation Board. Overview of the dart mishap investigation results. Technical report, Tech. rep., NASA, 2006. Available at http://www. nasa. gov/pdf , 2006.

[23] Immanuel M Bomze and Marco Locatelli. Undominated DC decompositions of quadratic functions and applications to branch-and-bound approaches. *Computational Optimization and Applications*, 28(2):227–245, 2004.

[24] Stephen Boyd, Martin Hast, and Karl Johan Åström. Mimo pid tuning via iterated lmi restriction. *International Journal of Robust and Nonlinear Control*, 26(8):1718–1731, 2016.

[25] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[26] Richard H Byrd, Jorge Nocedal, and Richard A Waltz. Knitro: An integrated package for nonlinear optimization. In *Large-scale nonlinear optimization*, pages 35–59. Springer, 2006.

[27] James A Chamberlin and James T Rose. Gemini rendezvous program. *Journal of Spacecraft and Rockets*, 1(1):13–18, 1964.

[28] Dian-Jing Chen and Mehran Mesbahi. Constrained attitude control, psd lifts, and semidefinite programming. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2134–2139. IEEE, 2017.

[29] W. Chen, J. Yang, L. Guo, and S. Li. Disturbance-Observer-Based Control and Related Methods-An Overview. *IEEE Transactions on Industrial Electronics*, 63(2):1083–1095, February 2016.

[30] W.-H. Chen, S. Li, and J. Yang. Non-linear disturbance observer-based robust control for systems with mismatched disturbances/uncertainties. *IET Control Theory & Applications*, 5(18):2053–2062, December 2011.

[31] Stefano Chiaverini and Olay Egeland. A solution to the singularity problem for six-joint manipulators. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pages 644–649. IEEE, 1990.

[32] WH Clohessy. Terminal guidance system for satellite rendezvous. *Journal of the Aerospace Sciences*, 27(9):653–658, 1960.

[33] Silvio Cocuzza, Isacco Pretto, and Stefano Debei. Least-squares-based reaction control of space manipulators. *Journal of Guidance, Control, and Dynamics*, 35(3):976–986, 2012.

[34] Peter Crouch and F Silva Leite. The dynamic interpolation problem: on riemannian manifolds, lie groups, and symmetric spaces. *Journal of Dynamical and control systems*, 1(2):177–202, 1995.

[35] CW De Silva. Trajectory design for robotic manipulators in space applications. *Journal of guidance, control, and dynamics*, 14(3):670–674, 1991.

[36] S. DiCairano, H. Park, and I. Kolmanovsky. Model Predictive Control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control*, 22(12):1398–1427, August 2012.

[37] Hao Ding, Mingxiang Zhou, and Olaf Stursberg. Optimal path planning in the workspace for articulated robots using mixed integer programming. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 5770–5775. IEEE, 2009.

[38] Haibo Du, Shihua Li, and Chunjiang Qian. Finite-time attitude tracking control of spacecraft with application to attitude synchronization. *IEEE Transactions on Automatic Control*, 56(11):2711–2717, 2011.

[39] Steven Dubowsky and Miguel A Torres. Path planning for space manipulators to minimize spacecraft attitude disturbances. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pages 2522–2528. IEEE, 1991.

[40] Daniel Dueri, Behçet Açıkmeşe, Daniel P Scharf, and Matthew W Harris. Customized real-time interior-point methods for onboard powered-descent guidance. *Journal of Guidance, Control, and Dynamics*, 40(2):197–212, 2017.

[41] Gamal Elnagar, Mohammad A Kazemi, and Mohsen Razzaghi. The pseudospectral legendre method for discretizing optimal control problems. *IEEE transactions on Automatic Control*, 40(10):1793–1796, 1995.

[42] Samira S Farahani, Ivan Papusha, Catharine McGhan, and Richard M Murray. Constrained autonomous satellite docking via differential flatness and model predictive control. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 3306–3311. IEEE, 2016.

[43] Robert W Farquhar, David W Dunham, Yanping Guo, and James V McAdams. Utilization of libration points for human exploration in the sun–earth–moon system and beyond. *Acta Astronautica*, 55(3):687–700, 2004.

[44] Angel Flores-Abad, Ou Ma, Khanh Pham, and Steve Ulrich. A review of space robotics technologies for on-orbit servicing. *Progress in Aerospace Sciences*, 68:1–26, 2014.

[45] Ioannis A Fotiou, Philipp Rostalski, Pablo A Parrilo, and Manfred Morari. Parametric optimization and optimal control using algebraic geometry methods. *International Journal of Control*, 79(11):1340–1358, 2006.

[46] Jean Gallier. *Geometric methods and applications: for computer science and engineering*, volume 38. Springer Science & Business Media, 2011.

[47] G Gerard et al. *Dynamics and mission design near libration points*, volume 1. World Scientific, 2001.

[48] Mehrdad Ghorbani and Nima Assadian. Optimal Station-Keeping near EarthMoon Collinear Libration Points Using Continuous and Impulsive Maneuvers. *Advances in Space Research*, 52(12):2067–2079, December 2013.

[49] Elmer G Gilbert, Daniel W Johnson, and S Sathiya Keerthi. A fast procedure for computing the distance between complex objects in three-dimensional space. *IEEE Journal on Robotics and Automation*, 4(2):193–203, 1988.

[50] Donald Goldfarb and Shucheng Liu. An $\mathcal{O}(n^3 l)$ primal interior point algorithm for convex quadratic programming. *Mathematical Programming*, 49(1):325–340, 1990.

[51] Gurobi Optimization, Inc. Gurobi optimizer reference manual, 2015.

[52] Wassim M Haddad and VijaySekhar Chellaboina. *Nonlinear dynamical systems and control: a Lyapunov-based approach*. Princeton University Press, 2011.

[53] Eranda Harinath, Lucas C Foguth, Joel A Paulson, and Richard D Braatz. Nonlinear model predictive control using polynomial optimization methods. In *American Control Conference (ACC), 2016*, pages 1–6. IEEE, 2016.

[54] Edward N. Hartley, Marco Gallieri, and Jan M. Maciejowski. Terminal spacecraft rendezvous and capture with LASSO model predictive control. *International Journal of Control*, 86(11):2104–2113, November 2013.

[55] Edward N Hartley, Paul A Trodden, Arthur G Richards, and Jan M Maciejowski. Model predictive control system design and implementation for spacecraft rendezvous. *Control Engineering Practice*, 20(7):695–713, 2012.

[56] Philip Hartman et al. On functions representable as a difference of convex functions. *Pacific Journal of Mathematics*, 9(3):707–713, 1959.

[57] Didier Henrion and J-B Lasserre. Convergent relaxations of polynomial matrix inequalities and static output feedback. *IEEE Transactions on Automatic Control*, 51(2):192–202, 2006.

[58] Didier Henrion, Jean-Bernard Lasserre, and Johan Löfberg. Gloptipoly 3: moments, optimization and semidefinite programming. *Optimization Methods & Software*, 24(4-5):761–779, 2009.

[59] Carl Glen Henshaw. The DARPA Phoenix spacecraft servicing program: Overview and plans for risk reduction. In *Proceedings of 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2014),(Montreal, Canada)*, 2014.

[60] Reiner Horst and Nguyen V Thoai. Dc programming: overview. *Journal of Optimization Theory and Applications*, 103(1):1–43, 1999.

[61] Qinglei Hu, Bo Li, and Juntong Qi. Disturbance observer based finite-time attitude control for rigid spacecraft under input saturation. *Aerospace Science and Technology*, 39:13–21, December 2014.

[62] H Ichihara. Computational approach to input-to-state stability analysis of a class of nonlinear systems. In *Proceedings of ICROS-SICE International Joint Conference 2009*, pages 3914–3918, 2009.

[63] Hiroyuki Ichihara. Sum of squares based input-to-state stability analysis of polynomial nonlinear systems. *SICE Journal of Control, Measurement, and System Integration*, 5(4):218–225, 2012.

[64] Francis James, Suril V Shah, Arun K Singh, K Madhava Krishna, and Arun K Misra. Reactionless maneuvering of a space robot in precapture phase. *Journal of Guidance, Control, and Dynamics*, pages 1–7, 2016.

[65] John L Junkins and Hanspeter Schaub. *Analytical mechanics of space systems*. American Institute of Aeronautics and Astronautics, 2009.

[66] U Kalabic, A Weiss, S Di Cairano, and I Kolmanovsky. Station-keeping and momentum-management on halo orbits around l2: Linear-quadratic feedback and model predictive control approaches. In *Proc. AAS Space Flight Mechanics Meeting*, pages 15–307, 2015.

[67] Reza Kamyar and Matthew Peet. Polynomial optimization with applications to stability analysis and control-alternatives to sum of squares. *arXiv preprint arXiv:1408.5119*, 2014.

[68] Fumio Kanehiro, Florent Lamiraux, Oussama Kanoun, Eiichi Yoshida, and Jean-Paul Laumond. A local collision avoidance method for non-strictly convex polyhedra. *Proceedings of robotics: science and systems IV*, 2008.

[69] M Karpenko, S Bhatt, N Bedrossian, A Fleming, and IM Ross. First flight results on time-optimal spacecraft slews. *Journal of Guidance, Control, and Dynamics*, 35(2):367–376, 2012.

[70] Hassan K Khalil and Jessy W Grizzle. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ, 2002.

[71] Oussama Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research*, 5(1):90–98, 1986.

[72] Wang Sang Koon, Martin W Lo, Jerrold E Marsden, and Shane D Ross. Dynamical systems, the three-body problem and space mission design. 2000.

[73] Basil Kouvaritakis and Mark Cannon. *Model Predictive Control: Classical, Robust and Stochastic*. Springer, 2015.

[74] Jayant E Kulkarni, Mark E Campbell, and Geir E Dullerud. Stabilization of spacecraft flight in halo orbits: An $\mathcal{H}_\infty$ approach. *IEEE transactions on control systems technology*, 14(3):572–578, 2006.

[75] R Lampariello. Motion planning for the on-orbit grasping of a non-cooperative target satellite with collision avoidance. In *Int. Symp. Artif. Intell. Robot. Automat. Space*, volume 1, pages 636–643, 2010.

[76] Roberto Lampariello and Gerd Hirzinger. Generating feasible trajectories for autonomous on-orbit grasping of spinning debris in a useful time. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 5652–5659. IEEE, 2013.

[77] JB Lasserre. A unified criterion for positive definiteness and semidefiniteness. *LAAS-CNRS, Toulouse, France, Res. Rep. 05*, 283:219, 2005.

[78] Jean B Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001.

[79] Jean B Lasserre. Robust global optimization with polynomials. *Mathematical programming*, 107(1-2):275–293, 2006.

[80] Jean B Lasserre. Representation of nonnegative convex polynomials. *Archiv der Mathematik*, 91(2):126–130, 2008.

[81] Jean B Lasserre, Didier Henrion, Christophe Prieur, and Emmanuel Trélat. Nonlinear optimal control via occupation measures and lmi-relaxations. *SIAM journal on control and optimization*, 47(4):1643–1666, 2008.

[82] Jean Bernard Lasserre. *Moments, positive polynomials and their applications*. World Scientific, 2009.

[83] Jean Bernard Lasserre. *An introduction to polynomial and semi-algebraic optimization*, volume 52. Cambridge University Press, 2015.

[84] Steven M LaValle and James J Kuffner Jr. Rapidly-exploring random trees: Progress and prospects. 2000.

[85] Eugene L Lawler and David E Wood. Branch-and-bound methods: A survey. *Operations research*, 14(4):699–719, 1966.

[86] Hoai An Le Thi, Hoai Minh Le, Tao Pham Dinh, et al. A dc programming approach for feature selection in support vector machines learning. *Advances in Data Analysis and Classification*, 2(3):259–278, 2008.

[87] Daero Lee. Nonlinear disturbance observer-based robust control of attitude tracking of rigid spacecraft. *Nonlinear Dynamics*, 88(2):1317–1328, 2017.

[88] Kwang-Kyu Lee and Martin Buss. Obstacle avoidance for redundant robots using jacobian transpose method. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 3509–3514. IEEE, 2007.

[89] Unsik Lee and Mehran Mesbahi. Spacecraft reorientation in presence of attitude constraints via logarithmic barrier potentials. In *Proceedings of the 2011 American Control Conference*, pages 450–455. IEEE, 2011.

[90] Shihua Li, Jun Yang, Wen-Hua Chen, and Xisong Chen. *Disturbance observer-based control: methods and applications*. CRC press, 2016.

[91] Yijun Lian, Gerard Gmez, Josep J. Masdemont, and Guojian Tang. Station-Keeping of Real EarthMoon Libration Point Orbits Using Discrete-Time Sliding Mode Control. *Communications in Nonlinear Science and Numerical Simulation*, 19(10):3792–3807, October 2014.

[92] JC Liou. An update on LEO environment remediation with active debris removal. *Orbital Debris Quarterly News*, 15(2):4–6, 2011.

[93] Thomas Lipp and Stephen Boyd. Variations and extension of the convex–concave procedure. *Optimization and Engineering*, 17(2):263–287, 2016.

[94] Xiaodong Liu, Hexi Baoyin, and Xingrui Ma. Optimal path planning of redundant free-floating revolute-jointed space manipulators with seven links. *Multibody System Dynamics*, 29(1):41–56, 2013.

[95] Xinfu Liu. Fuel-optimal rocket landing with aerodynamic controls. *Journal of Guidance, Control, and Dynamics*, 42(1):65–77, 2018.

[96] Xinfu Liu and Ping Lu. Robust trajectory optimization for highly constrained rendezvous and proximity operations. In *AIAA Guidance, Navigation, and Control (GNC) Conference*, page 4720, 2013.

[97] Xinfu Liu, Zuojun Shen, and Ping Lu. Entry trajectory optimization by second-order cone programming. *Journal of Guidance, Control, and Dynamics*, 39(2):227–241, 2015.

[98] Xinfu Liu, Zuojun Shen, and Ping Lu. Exact convex relaxation for optimal flight of aerodynamically controlled missiles. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1881–1892, 2016.

[99] Zhijie Liu, Jinkun Liu, and Lijun Wang. Disturbance observer based attitude control for flexible spacecraft with input magnitude and rate constraints. *Aerospace Science and Technology*, 72:486–492, January 2018.

[100] Johan Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289. IEEE, 2004.

[101] Ping Lu and Xinfu Liu. Autonomous trajectory planning for rendezvous and proximity operations by conic optimization. *Journal of Guidance, Control, and Dynamics*, 36(2):375–389, 2013.

[102] Ping Lu and Xinfu Liu. Autonomous trajectory planning for rendezvous and proximity operations by conic optimization. *Journal of Guidance, Control, and Dynamics*, 36(2):375–389, 2013.

[103] Anthony A Maciejewski and Charles A Klein. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *The international journal of robotics research*, 4(3):109–117, 1985.

[104] Yuanqi Mao, Daniel Dueri, Michael Szmuk, and Behçet Açıkmeşe. Convexification and real-time optimization for mpc with aerospace applications. In *Handbook of Model Predictive Control*, pages 335–358. Springer, 2019.

[105] Yuanqi Mao, Michael Szmuk, and Behcet Acikmese. Successive convexification of non-convex optimal control problems and its convergence properties. *arXiv preprint arXiv:1608.05133*, 2016.

[106] Darren Mcknight. US active debris removal (ADR) efforts. *United Nations Scientific and Technical Subcommittee of the UNCOPUOS*, 2013.

[107] A Megretski. Systems polynomial optimization tools (spot). 2010.

[108] Gaurav Misra and Xiaoli Bai. Optimal path planning for free-flying space manipulators via sequential convex programming. *Journal of Guidance, Control, and Dynamics*, 40(11):3019–3026, 2017.

[109] Gaurav Misra and Xiaoli Bai. Task-constrained trajectory planning of free-floating space-robotic systems using convex optimization. *Journal of Guidance, Control, and Dynamics*, 40(11):2857–2870, 2017.

[110] Gaurav Misra and Xiaoli Bai. Nonlinear disturbance observer based control for polynomial systems with mismatched uncertainties using sum-of-squares programming. In *2019 Annual American Control Conference (ACC)*. IEEE, 2019.

[111] Gaurav Misra, Maziar Izadi, Amit Sanyal, and Daniel Scheeres. Coupled orbit-attitude dynamics and relative state estimation of spacecraft near small solar system bodies. *Advances in Space Research*, 2015.

[112] Gaurav Misra, Hao Peng, and Xiaoli Bai. Halo orbit station-keeping using nonlinear MPC and polynomial optimization. In *2018 Space Flight Mechanics Meeting*, page 1454, 2018.

[113] Gaurav Misra, Amit Sanyal, and Ehsan Samiei. Asteroid landing guidance design in the framework of coupled orbit-attitude spacecraft dynamics. In _ *25th AAS/AIAA Spaceflight Mechanics Meeting, Advances in the Astronautical Sciences* _, volume 155, pages 1969–1980, 2015.

[114] Gaurav Misra and Amit K Sanyal. Analysis of orbit-attitude coupling of spacecraft near small solar system bodies. In *AIAA Guidance, Navigation, and Control Conference*, page 1777, 2015.

[115] A. Mohammadi, H. J. Marquez, and M. Tavakoli. Nonlinear Disturbance Observers: Design and Applications to Euler-Lagrange Systems. *IEEE Control Systems*, 37(4):50–72, August 2017.

[116] Daniel Morgan, Soon-Jo Chung, and Fred Y Hadaegh. Model predictive control of swarms of spacecraft using sequential convex programming. *Journal of Guidance, Control, and Dynamics*, 37(6):1725–1740, 2014.

[117] APS Mosek. The mosek optimization software. *Online at http://www. mosek. com*, 2010.

[118] Richard M Murray, Zexiang Li, S Shankar Sastry, and S Shankara Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[119] Yoshihiko Nakamura and Ranjan Mukherjee. Nonholonomic path planning of space robots via a bidirectional approach. *Robotics and Automation, IEEE Transactions on*, 7(4):500–514, 1991.

[120] Kostas Nanos and Evangelos Papadopoulos. On the use of free-floating space robots in the presence of angular momentum. *Intelligent Service Robotics*, 4(1):3–15, 2011.

[121] Kostas Nanos and Evangelos Papadopoulos. On cartesian motions with singularities avoidance for free-floating space robots. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 5398–5403. IEEE, 2012.

[122] Morad Nazari, Eric A Butcher, and William Anthony. Earth–moon l1 libration point orbit continuous stationkeeping control using time-varying lqr and backstepping. *International Journal of Dynamics and Control*, pages 1–14, 2016.

[123] Dragomir Nenchev, Yoji Umetani, and Kazuya Yoshida. Analysis of a redundant free-flying spacecraft/manipulator system. *IEEE Transactions on Robotics and Automation*, 8(1):1–6, 1992.

[124] Dragomir N Nenchev, Kazuya Yoshida, Prasart Vichitkulsawat, and Masaru Uchiyama. Reaction null-space control of flexible structure mounted manipulator systems. *IEEE Transactions on Robotics and Automation*, 15(6):1011–1023, 1999.

[125] Yi-Shuai Niu, Joaquim Júdice, Hoai An Le Thi, and Dinh Tao Pham. Improved dc programming approaches for solving the quadratic eigenvalue complementarity problem. *Applied Mathematics and Computation*, 353:95–113, 2019.

[126] Dennis Normile. Spunky hayabusa heads home with possible payload, 2010.

[127] Mitsushige Oda. Space robot experiments on nasda's ETS-VII satellite-preliminary overview of the experiment results. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 2, pages 1390–1395. IEEE, 1999.

[128] Mitsushige Oda et al. ETS-VII: achievements, troubles and future. In *Proceedings of the 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: ISAIRAS 2001*, 2001.

[129] Saurabh Pandey and Sunil K Agrawal. Path planning of free floating prismatic-jointed manipulators. *Multibody System Dynamics*, 1(1):127–140, 1997.

[130] Evangelos G Papadopoulos. Path planning for space manipulators exhibiting nonholonomic behavior. In *Proceedings of the 1992 lEEE/RSJ International Conference on*, volume 1, pages 669–675. IEEE, 1992.

[131] Pablo A Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization.* PhD thesis, California Institute of Technology, 2000.

[132] Pablo A Parrilo and Bernd Sturmfels. Minimizing polynomial functions. *Algorithmic and quantitative real algebraic geometry, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 60:83–99, 2003.

[133] Michael A Patterson and Anil V Rao. Gpops-ii: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming. *ACM Transactions on Mathematical Software (TOMS)*, 41(1):1, 2014.

[134] Hao Peng, Yuxin Liao, Xiaoli Bai, and Shijie Xu. Maintenance of Libration Point Orbit in Elliptic Sun-Mercury Model. *IEEE Transactions on Aerospace and Electronic Systems*, PP(99):1–15, 2017.

[135] Sven Mikael Persson and Inna Sharf. Sampling-based A* algorithm for robot path-planning. *The International Journal of Robotics Research*, 33(13):1683–1708, 2014.

[136] Robin Pinson and Ping Lu. Trajectory design employing convex optimization for landing on irregularly shaped asteroids. In *AIAA/AAS Astrodynamics Specialist Conference*, page 5378, 2016.

[137] Stephen Prajna, Antonis Papachristodoulou, and Pablo A Parrilo. Introducing sostools: A general purpose sum of squares programming solver. In *Decision and Control, 2002, Proceedings of the 41st IEEE Conference on*, volume 1, pages 741–746. IEEE, 2002.

[138] D. Pylorof and E. Bakolas. Robust control of input constrained nonlinear systems subject to unknown bounded disturbances based on convex optimization. In *2017 American Control Conference (ACC)*, pages 3700–3705, May 2017.

[139] T Raff, C Ebenbauer, R Findeisen, and F Allgöwer. Nonlinear model predictive control and sum of squares techniques. In *Fast Motions in Biomechanics and Robotics*, pages 325–344. Springer, 2006.

[140] Tobias Raff, Rolf Findeisen, Christian Ebenbauer, and Frank Allgwer. Model Predictive Control for Discrete Time Polynomial Control Systems: A Convex Approach. *IFAC Proceedings Volumes*, 37(21):123–128, December 2004.

[141] Anil V Rao. A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, 135(1):497–528, 2009.

[142] Arthur Richards, Tom Schouwenaars, Jonathan P How, and Eric Feron. Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming. *Journal of Guidance, Control, and Dynamics*, 25(4):755–764, 2002.

[143] David L Richardson. Analytic construction of periodic orbits about the collinear points. *Celestial Mechanics and Dynamical Astronomy*, 22(3):241–253, 1980.

[144] I Michael Ross. Users manual for dido: A matlab application package for solving optimal control problems. *Tomlab Optimization, Sweden*, page 65, 2004.

[145] Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1976.

[146] Timothy E Rumford. Demonstration of autonomous rendezvous technology (dart) project summary. In *Space Systems Technology and Operations*, volume 5088, pages 10–20. International Society for Optics and Photonics, 2003.

[147] Amit K Sanyal, Maziar Izadi, Daniel J Scheeres, Gaurav Misra, and Ehsan Samiei. Estimation of dynamics of space objects from visual feedback during proximity operations. In *AIAA/AAS Astrodynamics Specialist Conference*, page 4419, 2014.

[148] Daniel P Scharf, Behçet Açıkmeşe, Daniel Dueri, Joel Benito, and Jordi Casoliva. Implementation and experimental demonstration of onboard powered-descent guidance. *Journal of Guidance, Control, and Dynamics*, 40(2):213–229, 2017.

[149] Carsten W Scherer. Lmi relaxations in robust control. *European Journal of Control*, 12(1):3–29, 2006.

[150] Rune Schlanbusch, Esten Ingar Grøtli, Antonio Loria, and Per Johan Nicklasson. Hybrid attitude tracking of rigid bodies without angular velocity measurement. *Systems & Control Letters*, 61(4):595–601, 2012.

[151] Tom Schouwenaars, Arthur Richards, Eric Feron, and Jonathan How. Plume avoidance maneuver planning using mixed integer linear programming. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, page 4091, 2001.

[152] John Schulman, Jonathan Ho, Alex X Lee, Ibrahim Awwal, Henry Bradlow, and Pieter Abbeel. Finding locally optimal, collision-free trajectories with sequential convex optimization. In *Robotics: science and systems*, volume 9, pages 1–10. Citeseer, 2013.

[153] Volker Schulz. *Reduced SQP methods for large scale optimal control problems in DAE with application to path planning problems for satellite mounted robots.* PhD thesis, University of Heildelberg, 1995.

[154] Shay Segal and Pini Gurfil. Effect of kinematic rotation-translation coupling on relative spacecraft translational dynamics. *Journal of Guidance, Control, and Dynamics*, 32(3):1045–1050, 2009.

[155] Krishna Shankar, Joel W Burdick, and Nicolas H Hudson. A quadratic programming approach to quasi-static whole-body manipulation. In *Algorithmic Foundations of Robotics XI*, pages 553–570. Springer, 2015.

[156] Maksim Shirobokov, Sergey Trofimov, and Mikhail Ovchinnikov. Survey of Station-Keeping Techniques for Libration Point Orbits. *Journal of Guidance, Control, and Dynamics*, 40(5):1085–1105, 2017.

[157] Bruno Siciliano and Oussama Khatib. *Springer handbook of robotics.* Springer Science & Business Media, 2008.

[158] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Oriolo. *Robotics: modelling, planning and control.* Springer Science & Business Media, 2010.

[159] Eduardo D Sontag and Yuan Wang. On characterizations of the input-to-state stability property. *Systems & Control Letters*, 24(5):351–359, 1995.

[160] Joseph A Starek, Behçet Açıkmeşe, Issa A Nesnas, and Marco Pavone. Spacecraft autonomy challenges for next-generation space missions. In *Advances in Control System Technology for Aerospace Applications*, pages 1–48. Springer, 2016.

[161] Jos F Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653, 1999.

[162] Joshua Sullivan, Sebastian Grimberg, and Simone DAmico. Comprehensive survey and assessment of spacecraft relative motion dynamics models. *Journal of Guidance, Control, and Dynamics*, 40(8):1837–1859, 2017.

[163] L. Sun and Z. Zheng. Disturbance-Observer-Based Robust Backstepping Attitude Stabilization of Spacecraft Under Input Saturation and Measurement Uncertainty. *IEEE Transactions on Industrial Electronics*, 64(10):7994–8002, October 2017.

[164] L. Sun and Z. Zheng. Disturbance Observer-Based Robust Saturated Control for Spacecraft Proximity Maneuvers. *IEEE Transactions on Control Systems Technology*, 26(2):684–692, March 2018.

[165] Liang Sun, Wei Huo, and Zongxia Jiao. Disturbance observer-based robust relative pose control for spacecraft rendezvous and proximity operations under input saturation. *IEEE Transactions on Aerospace and Electronic Systems*, 2018.

[166] Victory Szebehely. *Theory of orbit: The restricted problem of three Bodies.* Elsevier, 2012.

[167] Michael Szmuk, Utku Eren, and Behcet Acikmese. Successive convexification for mars 6-dof powered descent landing guidance. In *AIAA Guidance, Navigation, and Control Conference*, page 1500, 2017.

[168] Michael Szmuk, Carlo Alberto Pascucci, Daniel Dueri, and Behcet Açikmeşe. Convexification and real-time on-board optimization for agile quad-rotor maneuvering and obstacle avoidance. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4862–4868. IEEE, 2017.

[169] Margaret Tam and E Glenn Lightsey. Constrained spacecraft reorientation using mixed integer convex programming. *Acta Astronautica*, 127:31–40, 2016.

[170] Jérôme Thai, Timothy Hunter, Anayo K Akametalu, Claire J Tomlin, and Alexandre M Bayen. Inverse covariance estimation from data with missing values using the concave-convex procedure. In *53rd IEEE Conference on Decision and Control*, pages 5736–5742. IEEE, 2014.

[171] J Tschauner and P Hempel. Optimale beschleunigungsprogramme fur das rendezvous-manover. *Astronautica Acta*, 10(5-6):296–+, 1964.

[172] Yuri Ulybyshev. Long-Term Station Keeping of Space Station in Lunar Halo Orbits. *Journal of Guidance, Control, and Dynamics*, 38(6):1063–1070, June 2015.

[173] Masashi Uo, Kenichi Shirakawa, Tatsuaki Hashimoto, Takashi Kubota, and Jun'ichiro Kawaguchi. Attitude control challenges and solutions for hayabusa spacecraft. In *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, page 6534, 2006.

[174] Diederik Verscheure, Bram Demeulenaere, Jan Swevers, Joris De Schutter, and Moritz Diehl. Time-optimal path tracking for robots: A convex optimization approach. *Automatic Control, IEEE Transactions on*, 54(10):2318–2327, 2009.

[175] Josep Virgili-Llop and Marcello Romano. A recursively feasible and convergent sequential convex programming procedure to solve non-convex problems with linear equality constraints. *arXiv preprint arXiv:1810.10439*, 2018.

[176] Josep Virgili-Llop, Costantinos Zagaris, Richard Zappulla, Andrew Bradstreet, and Marcello Romano. A convex-programming-based guidance algorithm to capture a tumbling object on orbit using a spacecraft equipped with a robotic manipulator. *The International Journal of Robotics Research*, 38(1):40–72, 2019.

[177] Jinbo Wang, Naigang Cui, and Changzhu Wei. Optimal rocket landing guidance using convex optimization and model predictive control. *Journal of Guidance, Control, and Dynamics*, pages 1–15, 2019.

[178] Lu Wang and Jianbo Su. Robust disturbance rejection control for attitude tracking of an aircraft. *IEEE Transactions on Control Systems Technology*, 23(6):2361–2368, 2015.

[179] Mingming Wang, Jianjun Luo, and Ulrich Walter. Novel synthesis method for minimizing attitude disturbance of the free-floating space robots. *Journal of Guidance, Control, and Dynamics*, 38(11):695–704, 2015.

[180] Zhenbo Wang and Michael J Grant. Constrained trajectory optimization for planetary entry via sequential convex programming. In *AIAA Atmospheric Flight Mechanics Conference*, page 3241, 2016.

[181] Zhenbo Wang and Michael J Grant. Minimum-fuel low-thrust transfers for spacecraft: A convex approach. *IEEE Transactions on Aerospace and Electronic Systems*, 54(5):2274–2290, 2018.

[182] Zhenbo Wang and Michael J Grant. Improved sequential convex programming algorithms for entry trajectory optimization. In *AIAA Scitech 2019 Forum*, page 0667, 2019.

[183] A. Weiss, M. Baldwin, R. S. Erwin, and I. Kolmanovsky. Model Predictive Control for Spacecraft Rendezvous and Docking: Strategies for Handling Constraints and Case Studies. *IEEE Transactions on Control Systems Technology*, 23(4):1638–1647, July 2015.

[184] A. Weiss, I. Kolmanovsky, M. Baldwin, and R. S. Erwin. Model Predictive Control of three dimensional spacecraft relative motion. In *2012 American Control Conference (ACC)*, pages 173–178, June 2012.

[185] Guang Q Xing and Shabbir A Parvez. Nonlinear attitude state tracking control for spacecraft. *Journal of Guidance, Control, and Dynamics*, 24(3):624–626, 2001.

[186] Yangsheng Xu and Takeo Kanade. *Space robotics: dynamics and control*, volume 188. Springer Science & Business Media, 1992.

[187] Yangsheng Xu and Heung-Yeung Shum. Dynamic control and coupling of a free-flying space robot system. *Journal of Robotic Systems*, 11(7):573–589, 1994.

[188] Hongwei Yang, Xiaoli Bai, and Hexi Baoyin. Rapid generation of time-optimal trajectories for asteroid landing via convex optimization. *Journal of Guidance, Control, and Dynamics*, 40(3):628–641, 2017.

[189] Jun Yang, Shihua Li, and Wen-Hua Chen. Nonlinear disturbance observer-based control for multi-input multi-output nonlinear systems subject to mismatching condition. *International Journal of Control*, 85(8):1071–1082, 2012.

[190] Kazuya Yoshida, Dimitar Dimitrov, and Hiroki Nakanishi. On the capture of tumbling satellite by a space robot. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 4127–4132. IEEE, 2006.

[191] Makoto Yoshikawa, Junichiro Kawaguchi, Akira Fujiwara, and Akira Tsuchiyama. Hayabusa sample return mission. *Asteroids IV, Univ. Arizona Press, Tucson*, pages 397–418, 2015.

[192] Alan L Yuille and Anand Rangarajan. The concave-convex procedure. *Neural computation*, 15(4):915–936, 2003.

[193] MC Zanardi and FF Real. Environmental torques acting on a low earth orbiter cylindrical spacecraft. *Advances in Space Research*, 31(8):1981–1986, 2003.

[194] Milos Zefran and Vijay Kumar. Two methods for interpolating rigid body motions. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 4, pages 2922–2927. IEEE, 1998.

[195] Fuzhen Zhang. Positive semidefinite matrices. In *Matrix Theory*, pages 199–252. Springer, 2011.

[196] Qiang Zhang, Shu-Rong Li, and Xiao-Shan Gao. Practical smooth minimum time trajectory planning for path following robotic manipulators. In *American Control Conference (ACC), 2013*, pages 2778–2783. IEEE, 2013.

[197] Yunong Zhang, Shuzhi Sam Ge, and Tong Heng Lee. A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(5):2126–2132, 2004.

[198] Zhe Zhang, Jianxun Li, and Jun Wang. Sequential convex programming for nonlinear optimal control problems in uav path planning. *Aerospace Science and Technology*, 76:280–290, 2018.

[199] Zhijie Zhu, Edward Schmerling, and Marco Pavone. A convex optimization approach to smooth trajectories for motion planning with car-like robots. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 835–842. IEEE, 2015.

[200] A. Zou, A. H. J. de Ruiter, and K. D. Kumar. Disturbance Observer-Based Attitude Control for Spacecraft with Input MRS. *IEEE Transactions on Aerospace and Electronic Systems*, pages 1–1, 2018.