Copyright
by
Utku Eren
2015

The Report Committee for Utku Eren

Certifies that this is the approved version of the following report:

A Mixed Integer Convex Programming Approach to Constrained Attitude Guidance

APPROVED BY

SUPERVISING COMMITTEE:

Behçet Açıkmeşe, Supervisor

Maruthi R. Akella

A Mixed Integer Convex Programming Approach to Constrained Attitude Guidance

by

Utku Eren, B.S., B.S., M.S.

REPORT

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN ENGINEERING

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2015

Dedicated to my family.

Acknowledgments

I would like to begin with expressing my gratitude to my advisor Dr. Behçet Açıkmeşe for his support and guidance throughout this study. I would also like to thank Dr. Daniel P. Scharf for his continuous support and motivation throughout this research. I want to acknowledge Dr. Maruthi R. Akella for taking time to review this report and for his valuable suggestions.

I would like thank Can Pehlivantürk who helped me in various topics with extremely fruitful discussions; Nazli Demirer with whom I endured stressful periods of my graduate studies and Daniel Dueri whose support made me aware of different perspectives. I also want to acknowledge Tim Lowery and Micheal Szmuk for their efforts on managing the lab and experimental setups. Special thanks go to all my friends in Turkey for their continuous support.

Finally and most importantly, I can not thank enough to my parents, Aliye & Sinan Eren, for being endless source of my motivation. I am deeply thankful for their love and support throughout my life. The respect and gratitude I have for them is simply beyond words. Also, a special thank goes to my girlfriend Güliz Tokadlı for her endless support throughout my graduate studies.

This research was supported in part by Jet Propulsion Laboratory, California Institute of Technology, Contract No. 1492781. Part of this research was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Also this work appeared in 14$^{\text{th}}$ annual *European Control Conference*, Austria, 2015.

A Mixed Integer Convex Programming Approach to Constrained Attitude Guidance

Utku Eren, M.S.E.
The University of Texas at Austin, 2015

Supervisor: Behçet Açıkmeşe

This brief report introduces a new algorithm for attitude motion planning, Constrained Attitude Guidance (CAG) problem, in the presence of angular rate constraints and conic exclusion regions (pointing constraints). The CAG problem is solved by considering only the quaternion kinematics in the formulation and using constraints on quaternions and its time derivatives to indirectly apply bounds on the angular rates and accelerations. The CAG formulation makes use of Mixed Integer Convex Programming (MICP) in order to impose, approximately, the unity constraint on the quaternion magnitude, where the approximation accuracy can be set to a desired accuracy. The solution complexity of the MICP formulation increases exponentially with the number of binary variables that are used to impose the unit norm constraint on the quaternion. Since this number is independent of the number of exclusion pointing constraints, the solution approach has favorable complexity in terms of the number of pointing constraints. The report also provides a numerical example that incorporates both angular rate and pointing constraints.

Table of Contents

Acknowledgments **v**

Abstract **vii**

List of Figures **x**

Chapter 1. Introduction **1**

1.1 Motivation 1

1.2 Challenges & Existing Work 2

1.3 Notation 4

Chapter 2. Problem Statement **5**

2.1 Rotational Kinematics & Dynamics 5

2.2 Constraints 6

Chapter 3. Proposed Formulation for CAG Problem **8**

3.1 Convexification of Pointing Constraints 8

3.2 Handling of the Nonlinear Dynamics 9

3.3 Derivation of Angular Velocity and Angular Acceleration constraints 11

3.4 Approximation of Quaternion Space to Impose Unity Quaternion Constraint 14

Chapter 4. Utilization of Mixed Integer Convex Programming **17**

4.1 Time Discretization 17

4.2 Use of Binary Variables 18

4.3 Cost Function 19

Chapter 5. A Numerical Example **21**

Chapter 6. Conclusions **27**

Bibliography **28**

List of Figures

3.1 Diagram of the Proposed Algorithm: (i) For given initial and final conditions Constraint Attitude Guidance Problem is solved with pure quaternion formulation. (ii) Successfully obtained sequence of quaternions are fed into angular velocity/acceleration calculator and (iii) results are used to construct control signal sequence that generates the whole trajectory. 10

3.2 3-Dimensional Illustration of Polyhedra Approximation for Hyperspherical Surface: Half-spherical surface that defines the original solution domain (left), depiction of the hyperplanes that surrounds the half-sphere (middle), the new approximate 3-dimensional surface that hosts our solutions which is generated with 17 hyperplanes (right). 15

5.1 Path that directions of sensitive instruments in body frame follow: Two avoidance regions are depicted as blue ($\theta_1 = 40^\circ$) and green ($\theta_2 = 30^\circ$) cones. Blue dots with transparent lines represent the motion of the body vector $x_1 = [0, 0, 1]^{\mathrm{T}}$ in inertial frame same as green dots gives the motion of the body vector $x_2 = [1, 0, 0]^{\mathrm{T}}$. Black dots in the figure illustrates the motion of the body vector $e_2 = [0, 1, 0]^{\mathrm{T}}$ in inertial frame in order to provide a complete set of axes. The transparency in vectors emphasizes the direction of the motion i.e. towards final orientation vectors become more opaque. 23

5.2 Results of Mixed-Integer Convex Optimization for proposed CAG problem formulation: Resulting quaternion vectors with magnitude (top), angular velocity vector with its magnitude along with upper bound (middle), angular acceleration vector with its magnitude along with upper bound of $\|\ddot{q}\|$ (bottom). . 25

5.3 Results of Mixed-Integer Convex Optimization for proposed CAG problem formulation: Corresponding Euler angles (top), angle between the body vector that represents sensitive instrument on-board and the initial vector that centers the avoidance region (middle), control signal vector with its magnitude (bottom). 26

Chapter 1

Introduction

Attitude motion planning, or guidance is an important control problem in many aerospace applications. Spacecraft, airplanes, and helicopters all rely on some form of attitude guidance in order to achieve their mission objectives.

1.1 Motivation

This work is specifically motivated by the spacecraft attitude guidance problem, where, for example, a spacecraft must rotate from a given initial orientation to a final one while avoiding some prescribed pointing directions. Additionally, strict pointing constraints are frequently used to keep sensitive instruments from being exposed to direct sunlight (such as star-sensors). In most cases, there can be additional constraints that limit angular velocities and accelerations for mission related reasons or to prevent the excitation of structural modes. To this end, this report aims to to introduce a Constrained Attitude Guidance (CAG) methodology that can be used to obtain quaternion trajectories that meet these needs with a bounded control authority.

1.2 Challenges & Existing Work

The CAG problem is inherently challenging due to the sources of non-convexity involved. There are three major sources of non-convexity in a CAG problem formulation: the non-convexity introduced by having exclusion cones in the quaternion domain, the non-convexity associated with nonlinear attitude dynamics, and the non-convexity associated with enforcing that the attitude quaternions to be unit vectors (the surface of a hypersphere in $\mathbb{R}^4$ is a non-convex domain). There have been several innovative methods proposed to deal with these challenges. One such method uses gridding to map the constrained attitude domain into a (potentially non-convex) three dimensional set that represents the set of feasible attitudes [17]. This set can then be searched very efficiently to find attitude trajectories in real-time. However, this method does not lend itself well to unexpected changes in the attitude constraints, since the grid must be resampled accordingly. In [5, 18], an LMI based approach is proposed that directly utilizes the direction cosine matrix formulation of the attitude dynamics and kinematics. Though it is a different technical approach than the one taken in this report, it has a similar insight. It hierarchically decomposes the problem into two subproblems, and applies a graph search to solve the higher level subproblem to determine a set-point sequence, which effectively makes the resulting approach tractable via a convex optimization. In [9, 10, 12], a novel lossless convexification of the non-convex pointing constraints for exclusion cones is introduced, which is also leveraged in this report to handle the first source of non-convexity. The non-linearity

in the dynamics are circumvented by not considering the dynamics explicitly and by imposing the control constraints (bounds on available angular accelerations) as constraints on the first and second time derivatives of the quaternion, which are convex constraints. Lastly binary variables are utilized to impose the unity constraint on the quaternion. The resulting problem can then be converted into a Mixed Integer Convex Programming (MICP).

MICPs are frequently proposed to handle non-convexities in motion planning [15]. For the formulation proposed in this work, we make use of a polytope in $\mathbb{R}^4$ to approximate the hyperspherical surface as a series of outer-bounding planar constraints that can be enforced as a series of binary equality constraints. A number of efficient solution methods exist to solve MICP problems, by using a branch and bound algorithm [7]. For this particular formulation, each node in the binary search tree is a specific type of convex optimization problem known as a Second Order Cone Programming (SOCP) problem. Thus, each node in the tree can be solved to global optimality in polynomial time by making use of Interior Point Method (IPM) solvers [6, 13]. Moreover, IPMs can certify the feasibility of each node and find solutions fast - especially with the use of customized solvers [4, 14]. For the purposes of this report, we make use of CVX [6] and Gurobi [7] to obtain numerical solutions to the MICP formulation.

As far as the controller level of the Constrained Attitude Guidance problem is concerned, the guidance method presented in this paper can be extended to Model Predictive Control framework [1, 3] which is a potential

subject for a future research.

The report is organized as follows. Chapter 2 gives the general problem statement. Chapter 3 introduces the Constrained Attitude Guidance problem by purely using quaternions and introduces the conic pointing constraints that we make use of. Chapter 4 formulates the CAG problem as a MICP problem. Chapter 5 provides numerical examples to illustrate the theoretical results of the report. Finally, conclusions are stated in Chapter 6.

1.3 Notation

The following is a partial list of the notation that is used throughout this report. $\mathbb{R}^n$ is the n dimensional real vector space; $\mathbb{R}^n_+$ is the set of non-negative real numbers in an n dimensional real vector space; $\|\cdot\|$ is the vector 2-norm; I is the identity matrix and I_m is the identity matrix in $\mathbb{R}^{m\times m}$; $Q = Q^{\mathrm{T}} \succ (\succeq)\, 0$ implies Q is a symmetric positive (semi-)definite matrix.

Chapter 2

Problem Statement

This chapter, introduces the specific formulation of Constrained Attitude Guidance (CAG) problem adapted in this report. The objective of the problem is to find a quaternion time history, $q(t)$, over a time interval $[t_0, t_f]$ that is subject to initial/final state conditions, pointing constraints, and angular rate constraints.

2.1 Rotational Kinematics & Dynamics

The quaternion and angular velocity vectors are partitioned into components as follows,

$$q(t) = [\mu(t)^{\mathrm{T}}, \epsilon(t)]^{\mathrm{T}},$$

$$\omega(t) = [\omega_1(t), \omega_2(t), \omega_3(t)]^{\mathrm{T}}$$

Here, $\mu(t) \in \mathbb{R}^3$ represents the vector part of the quaternion while $\epsilon(t)$ stands for the scalar component. When the body coordinate system for the vehicle is aligned with the principal axes of inertia, the attitude dynamics can be expressed as

$$\begin{aligned} J_1\dot{\omega}_1(t) - (J_2 - J_3)\omega_2(t)\omega_3(t) &= u_1(t), \\ J_2\dot{\omega}_2(t) - (J_3 - J_1)\omega_3(t)\omega_1(t) &= u_2(t), \\ J_3\dot{\omega}_3(t) - (J_1 - J_2)\omega_1(t)\omega_2(t) &= u_3(t), \end{aligned} \tag{2.1}$$

where J_i ($i \in \{1,2,3\}$) are the principle moments of inertia of the vehicle, ω_i are the components of the angular velocity vector about the principal axes and u_i are the control torques. The following is the expression for the quaternion rate, where we note that its associated dynamics are norm preserving, i.e., $\|q(t)\|=1$:

$$\dot{q}(t) = \frac{1}{2}\Omega(\omega(t))q(t) \tag{2.2}$$

Here, $\Omega(\omega(t))$ is a skew-symmetric matrix that is a function of the angular velocity vector:

$$\Omega(\omega(t)) = \begin{bmatrix} 0 & \omega_3(t) & -\omega_2(t) & \omega_1(t) \\ -\omega_3(t) & 0 & \omega_1(t) & \omega_2(t) \\ \omega_2(t) & -\omega_1(t) & 0 & \omega_3(t) \\ -\omega_1(t) & -\omega_2(t) & -\omega_3(t) & 0 \end{bmatrix}$$

2.2 Constraints

Along with the kinematics & dynamics, it must be ensured that the angular velocity and acceleration vectors are sufficiently bounded in order to impose the control authority constraints, as well as the state constraints due to mission considerations:

$$\left.\begin{array}{l} \|\omega(t)\| \leq \delta_1 \\ \|\dot{\omega}(t)\| \leq \delta_2 \end{array}\right\} \quad \forall\, t \in [t_0, t_f]. \tag{2.3}$$

Finally, pointing constraints are enforced to protect sensitive on-board instruments by avoiding predefined conic regions, see [9] for the details of the formulation. This type of conic exclusion zone can be defined with the following

non-convex quadratic constraint:

$$q(t)^{\mathrm{T}} P_i(x_i, y_i, \theta_i) q(t) \leq 0, \quad i = 1, ..., m, \tag{2.4}$$

where the inequality given in (2.4) defines m number of conic exclusion zones by expressing them as m number of non-convex quadratic constraints, and,

$$\begin{aligned} P_i(x_i, y_i, \theta_i) &= \begin{bmatrix} A_i & b_i \\ b_i^{\mathrm{T}} & d_i \end{bmatrix} \in \mathbb{R}^{4\times 4} \\ A_i &:= x_i y_i^{\mathrm{T}} + y_i x_i^{\mathrm{T}} - \left(x_i^{\mathrm{T}} y_i + \cos\theta_i\right) I_3 \\ b_i &:= -x_i \times y_i, \qquad d_i := x_i^{\mathrm{T}} y_i - \cos\theta_i, \end{aligned} \tag{2.5}$$

where x_i defines the direction of the i^{th} sensitive instrument in the body frame, y_i defines the center of the i^{th} conic exclusion zone, and θ_i represents the half cone angle about y_i. Now the CAG problem can be written as:

Compute $q(t)$, $\omega(t)$, and $u(t)$ over $t \in [t_0, t_f]$ such that

$$q(t_0) = q_0, \quad \omega(t_0) = \omega_0, \quad q(t_f) = q_f, \quad \omega(t_f) = \omega_f, \tag{2.6}$$

and the conditions (2.1, 2.2, 2.3, 2.4) hold for all $[t_0, t_f]$.

Chapter 3

Proposed Formulation for CAG Problem

The problem statement given in the previous chapter has challenges from a convex optimization perspective. The most pressing challenge is that the rotational equations of motion for the rigid body dynamics are non-linear function of the angular velocities. The same situation emerges in the quaternion kinematics. Furthermore, non-convex nature of the quadratic pointing constraints prevents its direct use in convex programming. This chapter intends to present elimination of each challenge.

3.1 Convexification of Pointing Constraints

The pointing constraints are conic exclusion zones defined with quadratic inequality constraints as given in (2.4). The P_i matrices in these inequalities are sign-indefinite matrices, hence they are not suitable for convex programming. However, since attitude quaternions are unit vectors (that is, $\|q(t)\| = 1 \ \forall t \in [t_0, t_f]$), the inequalities in (2.4) can be converted into the form given below:

$$\left.\begin{array}{c} q(t)^{\mathrm{T}} \hat{P}_i(x, y, \theta) q(t) \leq l_i, \\ \hat{P}_i = P_i + l_i I_4 \end{array}\right\} \quad i = 1, ..., m \tag{3.1}$$

where l_i is any positive real number which is greater than the smallest (most negative) eigenvalue of P_i. Essentially, the pointing constraints are converted into semi-definite constraints by shifting the spectrum of the P_i matrices. Again note that this inequality is only valid if the quaternions have unit norms. For the derivation of this convexification of the pointing constraints and their associated proofs, the authors refer interested readers to [9, 11].

3.2 Handling of the Nonlinear Dynamics

The quaternion histories are computed without imposing the dynamics and kinematics equations given by (2.1) and (2.2). Instead, constraints are imposed on first and second time derivatives of the quaternions. Bounding these rates implies bounds on the control authority, and hence the resulting trajectories are still dynamically feasible. To that end, we first discretize the quaternion time derivatives and impose all the constraints on the discretized variables. Then we solve a convex optimization problem to obtain a feasible quaternion history sampled at finite number of time instances. The quaternions between consecutive samples are used to compute the actual angular velocity vector by assuming constant acceleration between the time samples, see Figure 3.1. The formulation for this phase of the algorithm is given as:

$$\begin{bmatrix} \dfrac{2\alpha_k \cos\alpha_k}{\Delta t \sin\alpha_k} \\ \omega_{3(k)} \\ \omega_{2(k)} \\ \omega_{1(k)} \end{bmatrix} = \left(\frac{2\alpha_k}{\Delta t \sin\alpha_k} \right) \Psi^{-1}(q_k) q_{k+1} \tag{3.2}$$

where $q_{k+1} = [\mu_{(k+1)}^{\mathrm{T}},\ \epsilon_{(k+1)}]^{\mathrm{T}}$, the matrix $\Psi(q_k)$ is

$$\Psi(q_k) = \begin{bmatrix} \mu_{1(k)} & \mu_{2(k)} & -\mu_{3(k)} & \epsilon_{(k)} \\ \mu_{2(k)} & -\mu_{1(k)} & \epsilon_{(k)} & \mu_{3(k)} \\ \mu_{3(k)} & \epsilon_{(k)} & \mu_{1(k)} & -\mu_{2(k)} \\ \epsilon_{(k)} & -\mu_{3(k)} & -\mu_{2(k)} & -\mu_{1(k)} \end{bmatrix},$$

and α_k is the angle between two unit quaternions:

$$\alpha_k = \arccos\left(q_k^{\mathrm{T}}\, q_{k+1}\right),$$

where $\|q_k\|,\ \|q_{k+1}\| = 1$. Finally, the required torques i.e. control inputs to achieve the optimal trajectory are calculated directly from rotational rigid body dynamics given as (2.1).

It is important to indicate that, our CAG problem formulation is capable of imposing the initial and final conditions on the angular velocity vector.

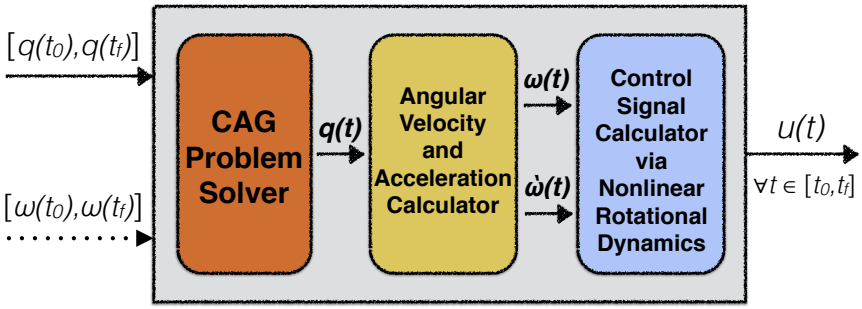

Figure 3.1: Diagram of the Proposed Algorithm: (i) For given initial and final conditions Constraint Attitude Guidance Problem is solved with pure quaternion formulation. (ii) Successfully obtained sequence of quaternions are fed into angular velocity/acceleration calculator and (iii) results are used to construct control signal sequence that generates the whole trajectory.

For given initial quaternion, angular velocity duo $(q(t_0), \omega(t_0))$, the quaternion in the next time instance $q(t_0 + \Delta t)$ can be obtained by zeroth-order forward integration of (2.2) under the assumption that initial angular velocity is constant during the time interval Δt. Similarly, for given final quaternion, angular velocity duo $(q(t_f), \omega(t_f))$, the previous quaternion $q(t_f - \Delta t)$ can be obtained by zeroth-order backward integration. The results of these zeroth order integration processes are transformed into changes in quaternion vectors and these changes are enforced as equality constraints in the CAG problem formulation ensuring that the first and last motions take place accordingly.

3.3 Derivation of Angular Velocity and Angular Acceleration constraints

This section derives expressions for angular velocity and acceleration constraints in terms of quaternions under the assumption that the quaternion norm stays unity throughout the whole trajectory. All of the derivations in this section are the results obtained from manipulation of (2.2).

Lemma 3.3.1. *The magnitude of angular velocity vector $\|\omega(t)\|$ can be bounded by bounding the magnitude of the quaternion rates $\|\dot{q}(t)\|$ as:*

$$\|\dot{q}(t)\| \leq \bar{\omega} \;\Rightarrow\; \|\omega(t)\| \leq 2\bar{\omega}. \tag{3.3}$$

Proof. Quaternions evolve according to the following differential equation,

$$\dot{q}(t) = \frac{1}{2}\Omega(t)q(t)$$

where skew symmetric matrix $\Omega(t)$ satisfies the following equation

$$\Omega(t)^{\mathrm{T}} = -\Omega(t).$$

Manipulating the kinematic constraint and employing the property of skew symmetric matrix implies

$$\begin{aligned}
\|\dot{q}(t)\|^2 = \dot{q}(t)^{\mathrm{T}}\dot{q}(t) &= \frac{1}{4}q(t)^{\mathrm{T}}\Omega(t)^{\mathrm{T}}\Omega(t)q(t) \\
&= -\frac{1}{4}q(t)^{\mathrm{T}}\Omega^2(t)q(t) \\
&= -\frac{1}{4}q(t)^{\mathrm{T}}(-\|\omega(t)\|^2 I_4)q(t) \\
&= \left\|\frac{\omega(t)}{2}\right\|^2 q(t)^{\mathrm{T}}q(t)
\end{aligned}$$

given that $q(t)^{\mathrm{T}}q(t) = 1$, the inequality given as (3.3) implies the following inequality:

$$\|\omega(t)\| \leq 2\bar{\omega} \tag{3.4}$$

□

Lemma 3.3.2. *The magnitude of angular acceleration vector $\|\dot{\omega}(t)\|$ can be bounded by bounding the magnitude of second derivative of quaternions $\|\ddot{q}(t)\|$ as:*

$$\|\ddot{q}(t)\| \leq \bar{\alpha} \;\Rightarrow\; \sqrt{\|\dot{\omega}(t)\|^2 + \frac{\|\omega(t)\|^4}{4}} \leq 2\bar{\alpha}. \tag{3.5}$$

Proof. From derivative of quaternion-based kinematic constraint:

$$\begin{aligned}
\frac{\partial \dot{q}(t)}{\partial t} &= \frac{1}{2}\frac{\partial(\Omega(t)q(t))}{\partial t} \\
\ddot{q}(t) &= \frac{1}{2}\Big\{\dot{\Omega}(t)q(t) + \Omega(t)\dot{q}(t)\Big\} \\
&= \frac{1}{2}\Big\{\dot{\Omega}(t) + \frac{1}{2}\Omega^2(t)\Big\}q(t) \\
&= \frac{1}{2}\Big\{\dot{\Omega}(t) - \frac{1}{2}\|\omega(t)\|^2 I_4\Big\}q(t)
\end{aligned}$$

which implies

$$\begin{aligned}
\|\ddot{q}(t)\|^2 &= \ddot{q}(t)^{\mathrm{T}}\ddot{q}(t) \\
&= \frac{1}{4}q^{\mathrm{T}}\Big\{\dot{\Omega}^{\mathrm{T}}\dot{\Omega} - \frac{1}{2}\|\omega\|^2\Big(\dot{\Omega}^{\mathrm{T}} + \dot{\Omega}\Big) + \frac{1}{4}\|\omega\|^4 I_4\Big\}q \\
&\text{using the property of skew-symmetric matrices } \dot{\Omega}^{\mathrm{T}} = -\dot{\Omega} \\
&= \frac{1}{4}q(t)^{\mathrm{T}}\Big\{ -\dot{\Omega}^2(t) + \frac{1}{4}\|\omega(t)\|^4 I_4\Big\}q(t) \\
&= \frac{1}{4}q(t)^{\mathrm{T}}\Big\{\|\dot{\omega}(t)\|^2 I_4 + \frac{1}{4}\|\omega(t)\|^4 I_4\Big\}q(t) \\
&= \Big\{\Big\|\frac{\dot{\omega}(t)}{2}\Big\|^2 + \Big\|\frac{\omega(t)}{2}\Big\|^4\Big\}q(t)^{\mathrm{T}}q(t)
\end{aligned}$$

given that $q(t)^{\mathrm{T}}q(t) = 1$, the inequality given as (3.5) implies the following inequality:

$$\sqrt{\|\dot{\omega}(t)\|^2 + \frac{\|\omega(t)\|^4}{4}} \leq 2\bar{\alpha}. \tag{3.6}$$

from Lemma 3.3.1, $\|\omega(t)\|$ is bounded, hence, via inequality (3.6) $\|\dot{\omega}(t)\|$ is also bounded. □

By Lemma 3.3.1 and 3.3.2, the magnitudes of angular velocity and angular acceleration vectors can be bounded via pure quaternion constraints.

The second bound presented in Lemma 3.3.2 is not the tightest upper bound for angular acceleration since the inequality contains both angular acceleration and angular velocity. Hence it can be used by introducing some conservatism in terms of imposing a constraint on $\|\dot{\omega}\|$. Introducing this conservatism is a price that is paid in order to keep the formulation convex. Note that, the conservatism decreases as the bound on $\|\omega\|$ decreases.

3.4 Approximation of Quaternion Space to Impose Unity Quaternion Constraint

Every quaternion represents the same rotation with its antipodal, i.e., $q = [\mu^{\mathrm{T}}, \epsilon]^{\mathrm{T}}$ and $-q = [-\mu^{\mathrm{T}}, -\epsilon]^{\mathrm{T}}$ define the same rotation [8]. Therefore the quaternion solution domain $\mathcal{V}$ of the CAG problem is a surface of 4-dimensional hemisphere:

$$\mathcal{V} = \{q = [\mu^{\mathrm{T}}, \epsilon]^{\mathrm{T}} : \mu \in \mathbb{R}^3, \;\; \epsilon \geq 0, \;\; \|q\| = 1\} \tag{3.7}$$

where $\mathcal{V}$ is a subset of whole quaternion space $\mathbb{H}$. The set $\mathcal{V}$ is clearly not a convex set, hence in order to tackle this challenge, we approximate the domain given by (3.7) with convex subsets i.e. hyperplanes that surround the domain. An illustrative example of this procedure is given in Figure 3.2. Gathering these arbitrary number of hyperplanes defines a new surface which is an approximation of the original solution domain of the CAG problem and each hyperplane provides a convex subset, thus, searching solutions in these subsets can be accomplished via a mixed integer convex optimization (MICO) formulation.

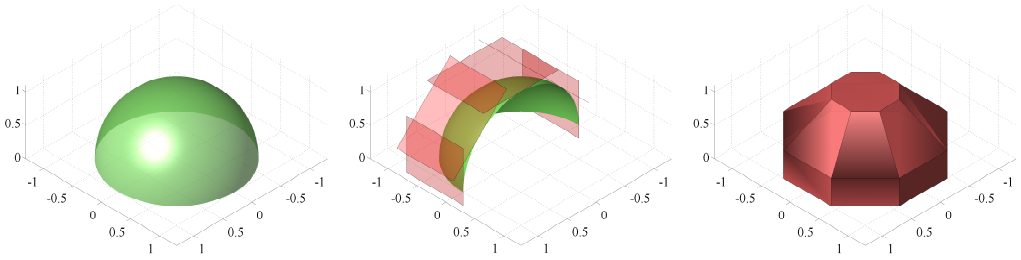

Figure 3.2: 3-Dimensional Illustration of Polyhedra Approximation for Hyperspherical Surface: Half-spherical surface that defines the original solution domain (left), depiction of the hyperplanes that surrounds the half-sphere (middle), the new approximate 3-dimensional surface that hosts our solutions which is generated with 17 hyperplanes (right).

As it is presented earlier, both convexified pointing constraint given as (3.1) and angular velocity/acceleration constraints given by Lemmas 3.3.1-3.3.2 require $\|q\| = 1$. However, the polyhedral approximation introduced for quaternion surface leads to a solution domain in which all quaternions satisfy the following inequality:

$$1 \leq \|q\| \leq \beta$$

where β is an upper bound on the norm of solutions naturally enforced by the approximation and its value is dependent on the number of hyperplanes. In this approximation, having $1 \leq \|q\|$ brings conservatism to the CAG problem. For instance, if the pointing constraint has a 30-degrees keep-out angle, the trajectory solution may come no closer than 33 degrees due to the number of hyperplanes used to approximate the unit hemisphere. However, the β value is also an upper bound on conservatism, which means that it can be reduced

arbitrarily at the cost of computation time by adding more hyperplanes. We next describe how this polyhedral approximation is incorporated into the attitude guidance formulation via using binary solution variables.

Chapter 4

Utilization of Mixed Integer Convex Programming

First, time discretization of the Constrained Attitude Guidance problem is introduced, which will be followed by the introduction of binary variables to impose the polyhedral approximation of the unit hemisphere for quaternions.

4.1 Time Discretization

Infinite dimensional CAG problem is transformed into a finite dimensional problem by choosing the number of time steps as N, and a time step Δt such that $t_f = N\Delta t$. The discrete-time approximation of continues-time quaternions are given as:

$$q(t) = q_k \quad \forall t \in [k\Delta t, (k+1)\Delta t),$$

and the first and second derivative of quaternions are obtained with central Euler differentiation:

$$\dot{q}_k = \{q_{k+1} - q_{k-1}\}/2\Delta t$$
$$\ddot{q}_k = \{q_{k+1} - 2q_k + q_{k-1}\}/\Delta t^2.$$

For the initial and final conditions for the second derivatives of quaternions, the following formulation is utilized along with the initial and final conditions on q and $\dot{q}$:

$$\ddot{q}_0 = \{q_1 - q_0\}/(0.5\Delta t)^2$$
$$\ddot{q}_N = \{q_N - q_{N-1}\}/(0.5\Delta t)^2.$$

4.2 Use of Binary Variables

After defining a set of hyperplanes to approximate the unit 4-dimensional hemisphere, the problem can be transformed into a mixed integer form by introducing binary variables [15]. For each hyperplane at each time step, a binary variable $z_{ik} \in \{0, 1\}$ is introduced to construct the following form:

$$\left.\begin{array}{l} S_i q_k \leq 1 + M z_{ik} \\ S_i q_k \geq 1 - M z_{ik} \end{array}\right\} \; i = 1, \ldots, n_p, \; k = 0, \ldots, N \tag{4.1}$$

where S_i is the normal of i^{th} hyperplane and M is a big positive number that is larger than any distance in unitary quaternion space. This "big-M" formulation [16] adds a pair of inequalities for n_p number of hyperplanes at $N + 1$ number of time steps. Here, having $z_{ik} = 0$ means solution q_k lies in hyperplane S_i at time $k\Delta t$. The following constraint is also employed in order to ensure that at least one of the z_{ik} values is zero i.e. solution has to be on one of those hyperplanes at all time: $\sum_{i=1}^{n_p} z_{ik} \leq n_p - 1$

4.3 Cost Function

The final element for the optimization framework is the cost function of the MICP problem. The primary goal is to find a feasible trajectory, and since the CAG problem is formulated with pure quaternions, the cost is also chosen to be a function of quaternions: $\sum \lambda_1 \|\ddot{q}_k\| + \lambda_2 \|\dot{q}_k\|$. As Lemma 3.3.1 and 3.3.2 shows, minimizing this cost function means minimizing a weighted some of angular velocity and angular accelerations related expressions. For instance, by minimizing $\sum \|\ddot{q}_k\|$, we minimize the following expression:

$$\sum \|\ddot{q}_k\| = \sum \left(\sqrt{\|\dot{\omega}_k\|^2 + \frac{\|\omega_k\|^4}{4}} \right).$$

Combining the results up to this point, we obtain the following MICP formulation in discrete time. This MICP formulation can be solved to global optimality by using commercial solvers [7]. Furthermore, in this formulation Δt can also be treated as a solution variable. Since it is a scalar quantity, a line search [2] on Δt can be performed to find the global optimum by solving a MICP for each cost evaluation of the line search. Moreover, for a fixed Δt, the minimum time trajectory can also be obtained by reducing the final time t_f and repeating the optimization until the MICO solver fails to find a solution. Hence the following formulation can be adapted to handle many cost metrics.

Discretized Constrained Attitude Guidance Problem

$$\min_{q,z_{ik}} \sum \|\ddot{q}_k\| \qquad \text{s.t}$$

$$\epsilon_k \geq 0, \qquad\qquad\qquad\qquad\qquad k = 0,\ldots,N$$

$$\|\dot{q}_k\| \leq \bar{\omega}, \qquad \|\ddot{q}_k\| \leq \bar{\alpha}, \qquad k = 0,\ldots,N$$

$$q_k^{\mathrm{T}} \hat{P}_i(x,y,\theta) q_k \leq b, \quad i = 1,\ldots,n_c, \quad k = 0,\ldots,N$$

$$q_k = [\mu_{1k}, \mu_{2k}, \mu_{3k}, \epsilon_k]^{\mathrm{T}} \qquad k = 0,\ldots,N$$

$$\dot{q}_k = \{q_{k+1} - q_{k-1}\}/2\Delta t, \qquad k = 1,\ldots,N-1$$

$$\ddot{q}_k = \{q_{k+1} - 2q_k + q_{k-1}\}/\Delta t^2, \qquad k = 1,\ldots,N-1$$

$$q_0 = \nu_0, \qquad q_N = \nu_N$$

$$\dot{q}_0 = \dot{\nu}_0, \qquad \dot{q}_N = \dot{\nu}_N$$

$$\ddot{q}_0 = \{q_1 - q_0\}/(0.5\Delta t)^2,$$

$$\ddot{q}_N = \{q_N - q_{N-1}\}/(0.5\Delta t)^2,$$

$$S_i q_k \leq 1 + M z_{ik}, \quad i = 1,\ldots,n_p, \qquad k = 0,\ldots,N$$

$$S_i q_k \geq 1 - M z_{ik}, \quad i = 1,\ldots,n_p, \qquad k = 0,\ldots,N$$

$$\sum_{i=1}^{n_p} z_{ik} \leq n_p - 1$$

$$z_{ik} \in \{0,1\}$$

Chapter 5

A Numerical Example

This chapter, presents an example that illustrates the proposed algorithm for the CAG problem, with angular rate constraints along with two exclusion cones. We consider a spacecraft with the inertia matrix along the principle axes given as:

$$J = \text{diag}([J_1, J_2, J_3]) = \text{diag}([\ 3,\ 1,\ 2\ ])\ \text{kg m}^2.$$

The initial and final quaternions are chosen as

$$q(t_0) = [-0.6744, 0.2126, -0.1530, 0.6903]^{\mathrm{T}}$$
$$q(t_f) = [0, 0, 0, 1]^{\mathrm{T}}$$

and the initial and final angular velocities are assumed to be

$$\omega(t_0) = [0, 0, 0]^{\mathrm{T}}\ \text{rad/s}$$
$$\omega(t_f) = [0, 0, 0]^{\mathrm{T}}\ \text{rad/s}.$$

The constraints on angular velocity and angular acceleration that spacecraft is subjected to are enforced through the following constraints:

$$\|\dot{q}(t)\| \leq 0.22, \quad \|\ddot{q}(t)\| \leq 0.19,$$

which corresponds to constraints given below by Lemma 3.3.1 and Lemma 3.3.2

$$\|\omega(t)\| \leq 0.44 \text{ rad/s}$$
$$\sqrt{\|\dot{\omega}(t)\|^2 + \frac{\|\omega(t)\|^4}{4}} \leq 0.38 \text{ rad/s}^2.$$

As pointing constraints, two cones as avoidance regions are determined for two different sensitive instruments where each instrument has its own avoidance cone. Each sensitive instrument is deemed as a direction in body frame of the spacecraft. These directions along with cone directions in inertial frame and half cone angles associated with the avoidance regions are given as:

$$x_1 = [0, 0, 1]^{\mathrm{T}}, \quad y_1 = [0, 1, 1]^{\mathrm{T}}, \quad \theta_1 = 40^\circ$$
$$x_2 = [1, 0, 0]^{\mathrm{T}}, \quad y_2 = [1, 0, 1]^{\mathrm{T}}, \quad \theta_2 = 30^\circ.$$

For this example total maneuver time is taken as 10 seconds and time step Δt is set to be 0.2 seconds. Also, non-convex 4-dimensional hemispherical surface i.e. the unit norm quaternion domain is approximated with a boundary of a polyhedra that contains 13 hyperplanes.

In Figure 5.1, two avoidance regions are illustrated with blue ($\theta_1 = 40^\circ$) and green ($\theta_2 = 30^\circ$) cones. The blue dots with transparent lines represent the motion of body direction $x_1 = [0, 0, 1]^{\mathrm{T}}$ in inertial frame while green ones stand for $x_2 = [1, 0, 0]^{\mathrm{T}}$. The black dots are given to provide a complete set of body vectors $e_2 = [0, 1, 0]^{\mathrm{T}}$. The resulting paths reveal that rotational maneuver has been accomplished without any violation in terms of avoided regions. The

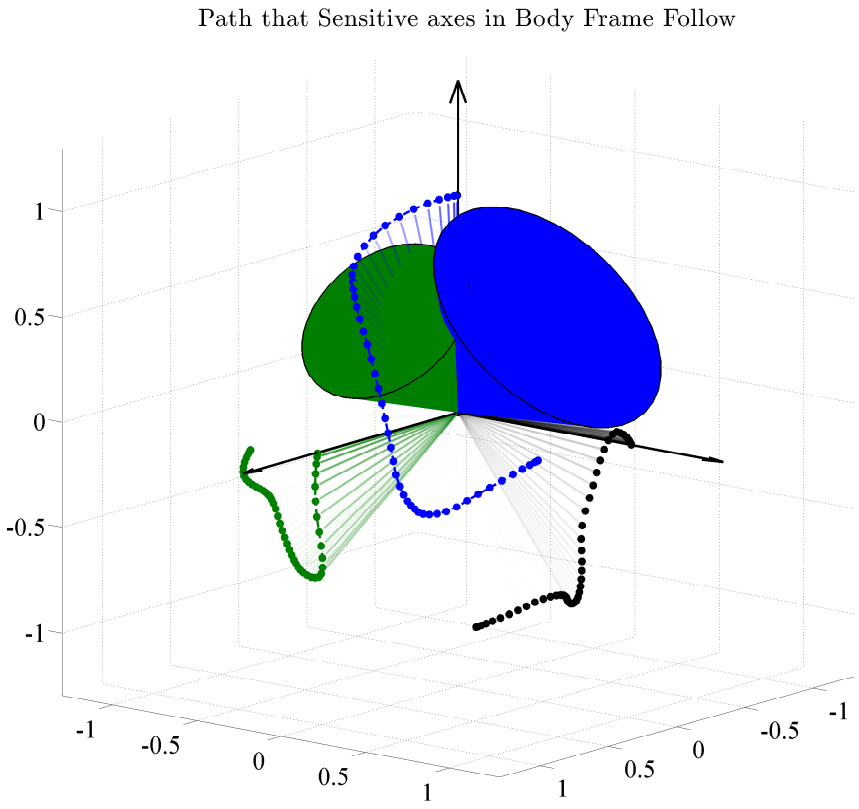

Figure 5.1: Path that directions of sensitive instruments in body frame follow: Two avoidance regions are depicted as blue ($\theta_1 = 40°$) and green ($\theta_2 = 30°$) cones. Blue dots with transparent lines represent the motion of the body vector $x_1 = [0, 0, 1]^{\mathrm{T}}$ in inertial frame same as green dots gives the motion of the body vector $x_2 = [1, 0, 0]^{\mathrm{T}}$. Black dots in the figure illustrates the motion of the body vector $e_2 = [0, 1, 0]^{\mathrm{T}}$ in inertial frame in order to provide a complete set of axes. The transparency in vectors emphasizes the direction of the motion i.e. towards final orientation vectors become more opaque.

blue path is allowed to pass across green cone and it goes around the blue cone without any violations as the green path does not intersect with green cone.

The effects of polyhedra approximation for unit hemisphere can be observed from magnitudes of quaternions given in Figure 5.2. Outer-bounding polyhedra ensures that the quaternion norms are at least one. As it is enforced, the angular velocity vector starts with vector of zeros and ends with zeros as well. The angular acceleration, on the other hand, starts with near maximum norm and decreases with increasing angular velocity. Consequently, as the magnitude of angular velocity gets close to zero, the magnitude of angular acceleration reaches its maximum feasible value. In either case, both angular velocity and angular acceleration vectors are successfully bounded. Finally, the actual control input needed for the trajectory is presented in Figure 5.3.

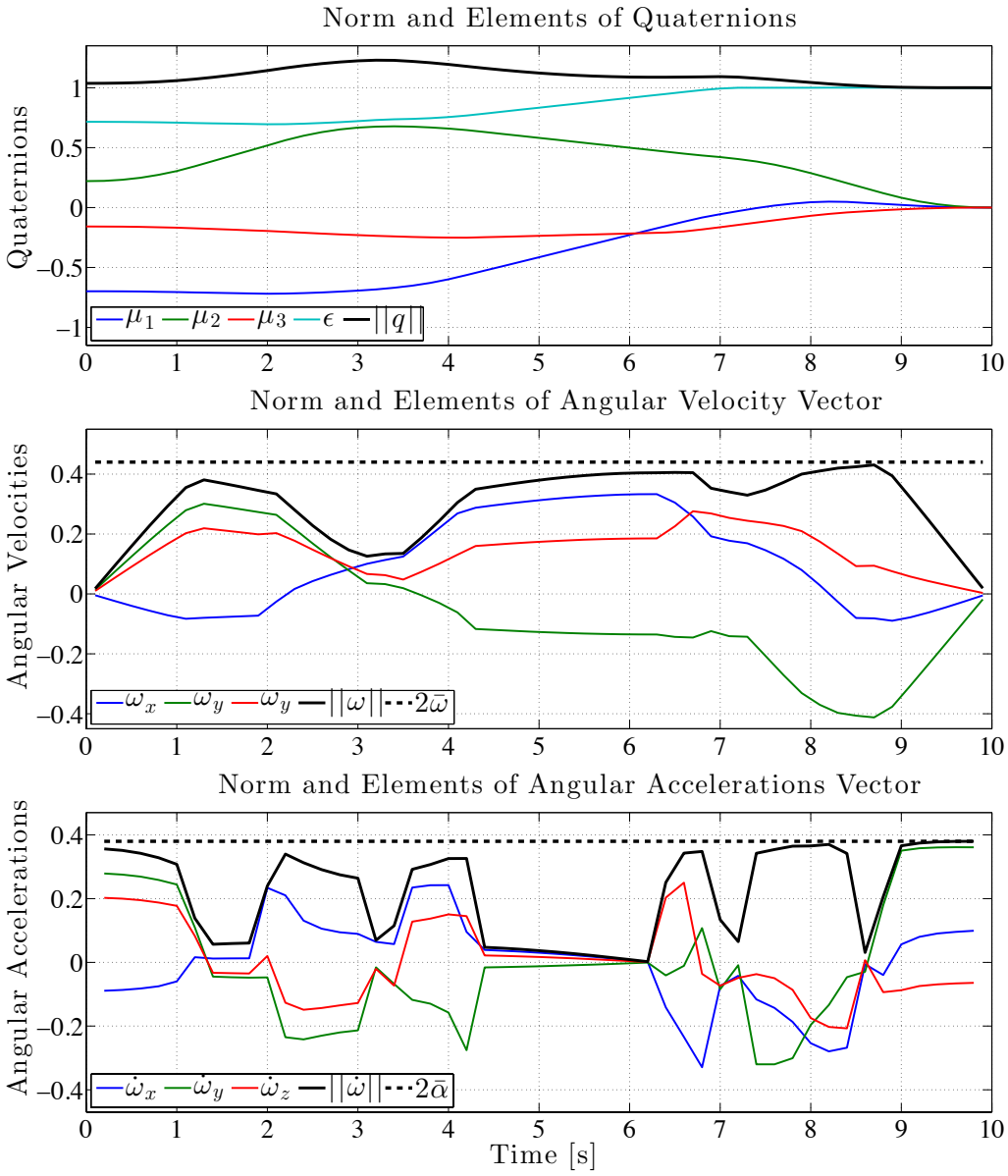

Figure 5.2: Results of Mixed-Integer Convex Optimization for proposed CAG problem formulation: Resulting quaternion vectors with magnitude (top), angular velocity vector with its magnitude along with upper bound (middle), angular acceleration vector with its magnitude along with upper bound of $\|\ddot{q}\|$ (bottom).

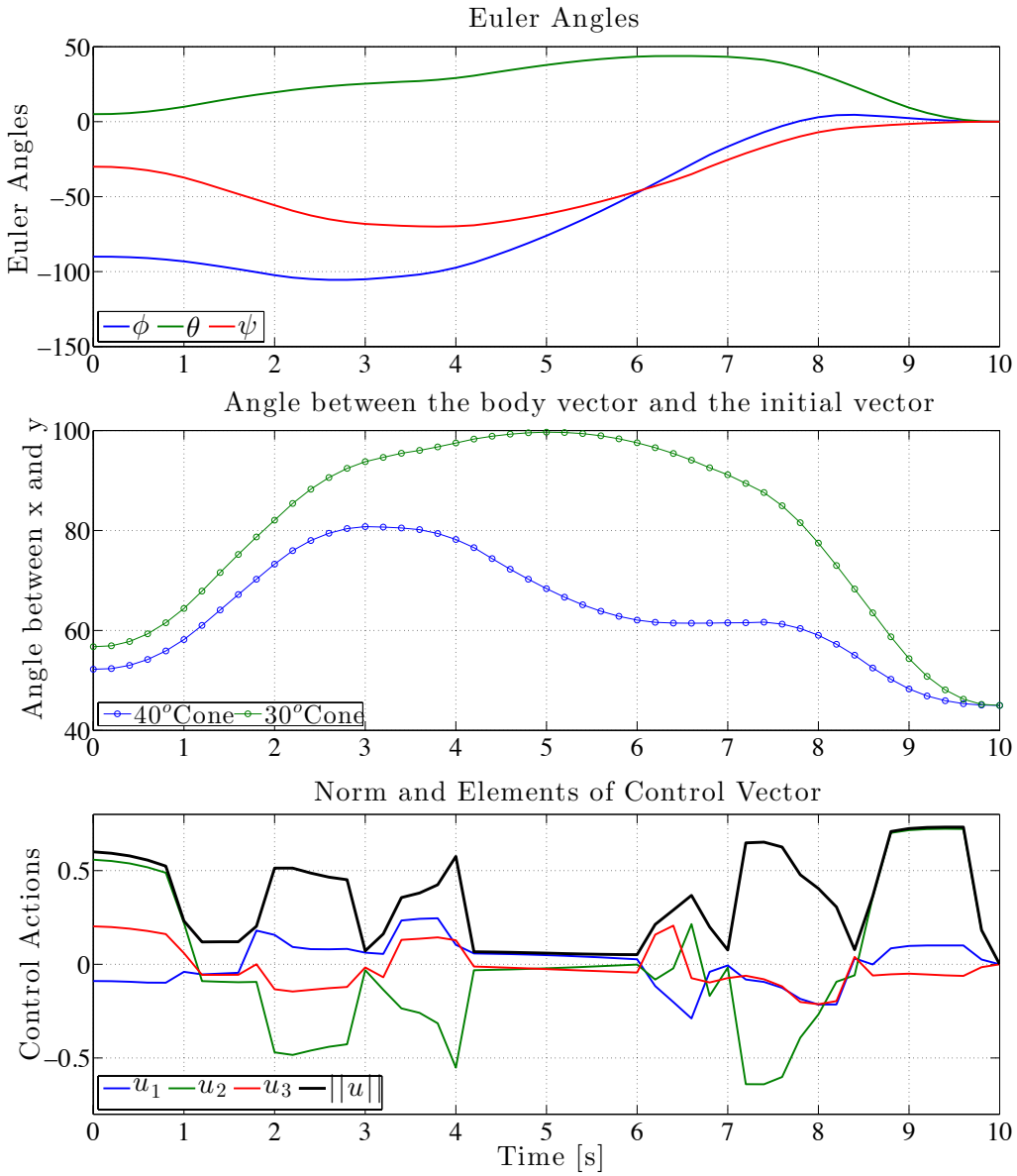

Figure 5.3: Results of Mixed-Integer Convex Optimization for proposed CAG problem formulation: Corresponding Euler angles (top), angle between the body vector that represents sensitive instrument on-board and the initial vector that centers the avoidance region (middle), control signal vector with its magnitude (bottom).

Chapter 6

Conclusions

This brief report presents a new mathematical formulation and a Mixed Integer Convex Programming (MICP) based solution method that provides globally, essentially, optimal solutions to the Constraint Attitude Guidance problem. The word "essentially" emphasizes the fact that the true optimal solution can be found to arbitrary precision by introducing finer polyhedral approximations to the unit hemisphere to impose the unity quaternion constraint, in exchange for more computation. The advantage of using MICP formulation is the guarantee of finding its optimal solution. The disadvantage is due the exponential increase in complexity with the number of binary variables. However, since the binary variables are solely used to impose the unity quaternion constraint, adding more attitude exclusion cone constraints does not introduce any new binary variables, hence the solution complexity does not increase exponentially with the number of exclusion constraints. Future work will focus on reducing the conservatism in the formulation and the real-time implementation of the solution method.

Bibliography

[1] Behçet Açıkmeşe, John M Carson, and David S Bayard. A robust model predictive control algorithm for incrementally conic uncertain/nonlinear systems. *International Journal of Robust and Nonlinear Control*, 21(5):563–590, 2011.

[2] D.P. Bertsekas. *Nonlinear Programming Second Edition*. Athena Scientific, 2000.

[3] John M Carson, Behçet Açıkmeşe, Richard M Murray, and Douglas G MacMartin. A robust model predictive control algorithm augmented with a reactive safety mode. *Automatica*, 49(5):1251–1260, 2013.

[4] D. Dueri, J. Zhang, and B. Açıkmese. Automated custom code generation for embedded real-time second order cone programming. In *IFAC World Congress*. IFAC, 2014.

[5] J. R. Forbes and A. de Ruiter. An LMI-based solution to wahba's problem. *AIAA Journal of Guidance, Control, and Dynamics*, 2013.

[6] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, March 2014.

[7] Inc. Gurobi Optimization. Gurobi optimizer version 5.6. `http://www.gurobi.com`, 2014.

[8] Berthold K.P. Horn. Some notes on unit quaternions and rotation. `http://people.csail.mit.edu/bkph/articles/Quaternions.pdf`, 2001.

[9] Y. Kim and M. Mesbahi. Quadratically constrained attitude control via semidefinite programming. *IEEE Transactions on Automatic Control*, 49(5), May 2004.

[10] Y. Kim, M. Mesbahi, G. Singh, and F. Y. Hadaegh. On the constrained attitude control problem. *IEEE Transactions on Aerospace and Electronic Systems*, May 2010.

[11] U. Lee and M. Mesbahi. Spacecraft attitude synchronization in presence of constrained zones. In *American Control Conference*, 2012.

[12] U. Lee and M. Mesbahi. Quaternion-based optimal spacecraft reorientation under complex attitude constrained zones. In *AAS/AIAA Astrodynamics Specialist Conference*. AAS/AIAA, 2013.

[13] J Löfberg. YALMIP 3. *http://control.ee.ethz.ch/ joloef/yalmip.msql*, 2004.

[14] J. Mattingley and S. Boyd. CVXGEN: a code generator for embedded convex optimization. *Optimization and Engineering*, 13(1):1–27, 2012.

[15] A. Richards, T. Schouwenaars, J. How, and E. Feron. Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming. *AIAA Journal of Guidance, Control, and Dynamics*, 25(4), July-August 2002.

[16] T. Schouwenaars, B. De Moor, E. Feron, and J. How. Mixed integer programming for multi-vehicle path planning. In *Proceedings European Control Conference*, 2001.

[17] Sergei Tanygin. Three-axis constrained attitude pathfinding and visualization: Charting a course on higher-dimensional maps. In *AIAA/AAS Astrodynamics Specialist Conference*, August 2014.

[18] A. Weiss, F. Leve, M. Baldwin, J. R. Forbes, and I. Kolmovsky. Spacecraft constrained attitude control using positively invariant constraint admissible sets on SO(3)xR^3. In *American Control Conference*, pages 4955–4960, 2014.