Autonomous Guidance for Asteroid Mapping and Touch and Go Descent

Dual Quaternion Approach

S. Hazra

Technische Universiteit Delft

Autonomous Guidance for Asteroid Mapping and Touch and Go Descent

Dual Quaternion Approach

by

S. Hazra

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on –

Student number:	4598342	
Project duration:	November 18, 2017 – June 18, 2018	
Thesis committee:	Prof. dr. ir. E. Mooij,	TU Delft
	Dr. ir. M. Sagliano,	DLR

This thesis is confidential and cannot be made public until –

Abstract

With the onset of the age of space travel, asteroid missions have been steadily gaining interest. The pristine nature of asteroids due to their preserved state since the formation of the solar system is an opportunity to unravel many mysteries about the solar system. Also with the ever growing need for resources, asteroids prove to be a plentiful source. With the discovery of asteroids in the close vicinity of our planet and a probable threat to the preservation of life, a need for defence missions has also arisen. With many successful missions like NEAR to Eros, Rosetta to Comet 67P, Hayabusa to Itokawa etcetera, the requirement of precise navigation with autonomous guidance and control for future missions has been established.

The ever-growing need for better computational speed and accuracy has led to the development of new representations for attitude and position of the spacecraft in the past. The usual methods of representations of position and attitude (pose) are the Cartesian coordinates and quaternions. A recent development is the simultaneous representation of the pose of the spacecraft using dual quaternions which are eight-dimensional vectors. As of December 2017, a variety of missions using dual quaternions for relative navigation, rendezvous and docking, entry, descent and landing have been conceptualized.

In this thesis, we try to combine two recently developed guidance methods: *Sampling Based Motion Planning Optimisation* and *Successive Convexification* with a *Dual Quaternion Approach* for mapping the asteroid and descending on it respectively. To achieve this we browse through past missions to establish the requirements of a mission to asteroids, explore the augmented algebra of dual numbers and their application in the concept of dual quaternions, establish how the environment around the asteroid needs to be modelled for a realistic validation of the GN&C algorithm and also study the kinematics and dynamics of the spacecraft using dual quaternions in relative frame.

The relative navigation for an asteroid mission using dual quaternions (Razgus 2017) and (Mao, Szmuk, and Acıkmeşe 2016) (Surovik and Scheeres 2014) form the base of this thesis and we build the guidance algorithms to incorporate them.

Preface

qwertzuiop

S. Hazra
Delft, 2018

Nomenclature/Abbreviations

Abbreviations

AOCS	Attitude and Orbit Control System
ADCS	Attitude Determination and Control System
AU	Astronomical Unit
COM	Centre of Mass
DCM	Direction Cosine Matrix
DLR	Deutsches Zentrum für Luft- und Raumfahrt e.V.
DOF	Degree of Freedom
DQ	Dual Quaternion
DQEKF	Dual Quaternion Extended Kalman Filter
DQMEKF	Dual Quaternion Multiplicative Extended Kalman Filter
EKF	Extended Kalman Filter
ESA	European Space Agency
FDIR	Fault Idnetification, Detection and Recovery
FOV	Field of View
GCP	Global Control Point
GG	Gravity Gradient
GNC	Guidance, Navigation and Control
GPS	Global Positioning System
HDA	Hazard Detection and Avoidance
IMU	Inertial Measurement Unit
IRU	Inertial Reference Units
IVP	Initial Value Problem
JAXA	Japan Aerospace Exploration Agency
JPL	Jet Propulsion Laboratory
LIDAR	Light Imaging, Detection, And Ranging
LOS	Line of Sight
LR(F)	Laser Ranger (Finder)
LVLH	Local Vertical Local Horizontal
MBA	Main Belt Asteroids
MC	Monte Carlo
MEKF	Multiplicative Extended Kalman Filter
MEMS	Micro-Electro-Mechanical System
MVM	Mission Vehicle Management
NAIF	Navigation and Ancillary Information Facility
NASA	National Aeronautics and Space Administration
NAVCAM	Navigation Camera
NEA	Near-Earth Asteroid
NEAR	Near Earth Asteroid Rendezvous
OCP	Optimal Control Problems
OD	Orbit Determination
ONC	Optical Navigation Camera
QVEKF	Quaternion-Vector Extended Kalman Filter
RADAR	RAdio Detection And Ranging
RK45	Runge-Kutta 45
SBMPO	Sampling Based Motion Planning Optimisation
SC	Spacecraft
SCvx	Successive Convexification

SOCP	Second Order Cone Programming
SPICE	Spacecraft, Planet, Instrument, C-matrix, Events
SRP	Solar-Radiation Pressure
STT	Star Tracker
TAG	Touch and Go
TNO	Trans-Neptunian Objects
TPBVP	Two-Point Boundary Value Problem
YORP	Yarkovsky–O'Keefe–Radzievskii–Paddack effect

Nomenclature

Contents

Preface v

Nomenclature/Abbreviations vii

1 Successive Convex Optimisation 1

1.1 Basic Concepts of Convex optimisation . . . 2

1.1.1 Convex Sets and Functions. . . . 2

1.1.2 Convex optimisation Problems: Second-Order Cone Programming . . . 4

1.2 Lossless Convexification . . . 5

1.2.1 Introduction of Slack Variables. . . . 6

1.2.2 Change of Variables . . . 7

1.3 Dual Quaternion Problem Formulation . . . 9

1.4 Successive Convexification . . . 10

1.4.1 Linearisation. . . . 10

1.4.2 Virtual Control . . . 14

1.4.3 Trust Regions . . . 15

1.5 Discretisation . . . 15

1.5.1 Control Parametrisation . . . 16

1.6 Extended Convex Guidance . . . 16

1.6.1 Need for Extended Constraints. . . . 16

1.6.2 Glide-Slope Constraint. . . . 17

1.6.3 Attitude Constraints . . . 17

1.6.4 Thrust Direction Constraint . . . 17

1.6.5 Stacking Equations. . . . 18

1.7 Algorithm . . . 18

A Mathematical Properties 19

B Verification using the Comet 67P Mapping 21

C Verification using a 6-DOF Mars Descent 23

1

Successive Convex Optimisation

Mathematical optimisation finds the best solution (maxima or minima) out of feasible solutions for a given problem. The standard formulation of an optimisation problem is as follows (Boyd and Vandenberghe 2010)

$$\begin{aligned} \text{minimise} \quad & f_0(x) \\ \text{subject to} \quad & g_i(x) \leq b_i, \quad i = 1, \ldots, m. \\ & h_j(x) = c_j, \quad j = 1, \ldots, p. \end{aligned} \tag{1.1}$$

where, $x = (x_1, \ldots, x_n)$ is the *optimisation variable* of the problem, $f_0 : \mathbb{R}^n \rightarrow \mathbb{R}$ is the *objective* function of the variable which needs to be optimised (minimised in the case above). The objective function is subject to the functions $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ and $h_i : \mathbb{R}^n \rightarrow \mathbb{R}$ are the *inequality* and *equality* constraints and the constants $b_1, \ldots, b_m$ and $c_j, \ldots, c_p$ are the bounds on them. The solution of this problem is given by a vector $\boldsymbol{x}^*$ which gives the minimal value of the objective function whilst satisfying all the constraint equations such that,

$$f_0(x^*) \leq f_0(z) \quad \forall\ z \in \mathbb{R}^n \quad \text{with} \quad g_i(z) \leq b_i \quad \& \quad h_j(z) \leq c_j \tag{1.2}$$

The optimisation problem is classified as *linear* or *nonlinear* depending on the form of the objective and constraint functions. A linear program has objective and constraint functions that satisfy:

$$g_i(\alpha x + \beta y) = \alpha g_i(x) + \beta g_i(y) \quad \forall\ x, y \in \mathbb{R}^n \quad \alpha, \beta \in \mathbb{R} \tag{1.3}$$

The small solar system body descent problem is plagued with non-convex constraints as well as nonlinearities due to time varying gravity fields and other perturbing forces. We can therefore no longer apply methods to solve linear problems directly. This chapter walks us through the process of convex optimisation and how it can be utilized by successively convexifying non-linear or non-convex problems. The Figure 1.1 provides with the logical flow of this chapter.

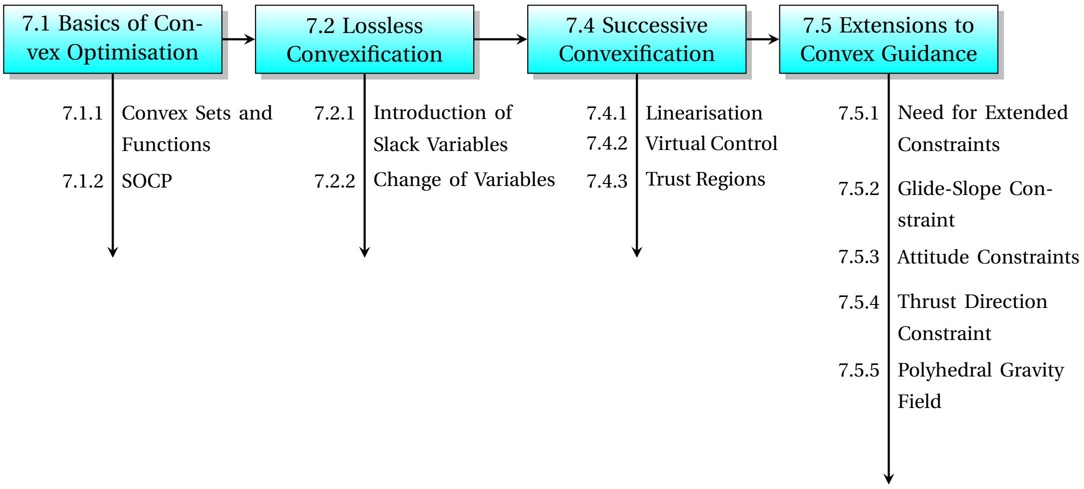

7.1 Basics of Convex Optimisation
7.2 Lossless Convexification
7.4 Successive Convexification
7.5 Extensions to Convex Guidance
7.1.1 Convex Sets and Functions
7.1.2 SOCP
7.2.1 Introduction of Slack Variables
7.2.2 Change of Variables
7.4.1 Linearisation
7.4.2 Virtual Control
7.4.3 Trust Regions
7.5.1 Need for Extended Constraints
7.5.2 Glide-Slope Constraint
7.5.3 Attitude Constraints
7.5.4 Thrust Direction Constraint
7.5.5 Polyhedral Gravity Field

Figure 1.1: Chapter Overview

1.1. Basic Concepts of Convex optimisation

Convex optimisation is a rather young development as compared to well established methods like least squares or linear programming and was explored to the current status mostly in the 20^{th} century. In this short period it has proved its worth with numerous beneficial properties. Açıkmeşe and Ploen 2007 presented their work to find a soft-landing guidance algorithm for a powered descent on Mars implementing convex optimisation. Since then there have been quite a few developments in the treatment of the powered descent problem as discussed in the Chapter 2. We will discuss some basic concepts in this section and work our way towards understanding the presently developed algorithms and finally formulate the problem using successive convexification using dual quaternions.

1.1.1. Convex Sets and Functions

A convex optimisation problem calls for convex objective and constraint functions which means they should be of the form:

$$g_i(\alpha x+\beta y) \leq \alpha g_i(x)+\beta g_i(y) \quad \forall\ x, y \in \mathbb{R}^n \quad \alpha, \beta \in \mathbb{R} \quad \alpha+\beta=1, \quad \alpha \geq 0, \beta \geq 0 \tag{1.4}$$

A function is therefore convex if the function of the weighted average is less than or equal to the weighted average of the function. As can be seen from the form of convex objective or constraints, convex problems are nothing but a generalization of linear problems by replacing the equality by inequality and with a restriction on α and β.

Convex sets are fundamental to understanding the theory of convex optimisation and hence will be discussed briefly. First we define a line segment between two points $x_1,\ x_2 \in \mathbb{R}^n$ and $x_1 \neq x_2$ as:

$$y = \theta x_1 + (1-\theta)x_2, \quad \theta \in \mathbb{R}, \quad 0 \leq \theta \leq 1 \tag{1.5}$$

An *affine set* is such that a line through any two points in the set lies in the set itself, meaning that a set, $\mathscr{C} \subseteq \mathbb{R}^n$ is an *affine set* if for any $x_1, x_2 \in \mathscr{C}$ and $\theta \in \mathbb{R}$, its true that $\theta x_1 + (1-\theta)x_2 \in \mathscr{C}$, An affine combination, $\theta_1 x_1 + \ldots + \theta_k x_k$, with $\theta_1 + \ldots + \theta_k$ lies in $\mathscr{C}$ too. *Convex sets* are in a way extension of affine sets where θ is restricted between 0 and 1 along with $\sum \theta_i = 1$. The set $\mathscr{C} \subseteq \mathbb{R}^n$ is convex for any $x_1,\ x_2 \in \mathscr{C}$ and $0 \leq \theta \leq 1$ such that $\theta x_1 + (1-\theta)x_2 \in \mathscr{C}$. "If every point in the set can be seen by every other point in the set, along an unobstructed straight path, its a convex set" (Boyd and Vandenberghe 2010). Figure 1.2 represents geometric examples of convex and non-convex sets. A convex combination can therefore be represented as

$$\theta_1 x_1 + \ldots + \theta_k x_k \in \mathscr{C}, \quad x_i \in \mathscr{C}, \quad \theta_i \geq 0, \quad \sum \theta_i = 1,\ i = 1, \ldots, k \tag{1.6}$$

A special kind of convex set is the convex cone, which is used in the formulation of *second order conic programming* which is a method to treat convex optimisation problems. The set $\mathscr{C}$ is a convex cone if for any $x_1, x_2 \in \mathscr{C}$ and $\theta_1, \theta_2 \geq 0$, we have

$$\theta_1 x_1 + \theta_2 x_2 \in \mathscr{C} \tag{1.7}$$

Geometrically the above equation represents a 2D slice of pie, with its apex at 0 and its edges passing through x_1 and x_2. Figure 1.3 represents this convex cone on the left and also shows an example of a cone that is non-convex on the right since the line passing through x_1 and x_2 does not lie within the set. With this the second-order cone or the *Euclidean*[1] norm cone in can be represented as

$$\begin{aligned} \mathscr{C} &= \{(x,t) \mid \|x\|_2 \leq t,\} \subseteq \mathbb{R}^{n+1} \\ &= \left\{ \begin{pmatrix} x \\ t \end{pmatrix} \middle| \begin{pmatrix} x \\ t \end{pmatrix}^T \begin{bmatrix} \boldsymbol{I}_{n\times n} & 0 \\ 0 & -1 \end{bmatrix} \begin{pmatrix} x \\ t \end{pmatrix} \leq 0,\ t \geq 0 \right\} \end{aligned} \tag{1.8}$$

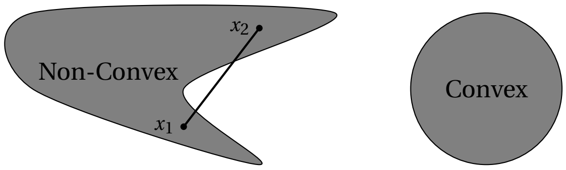
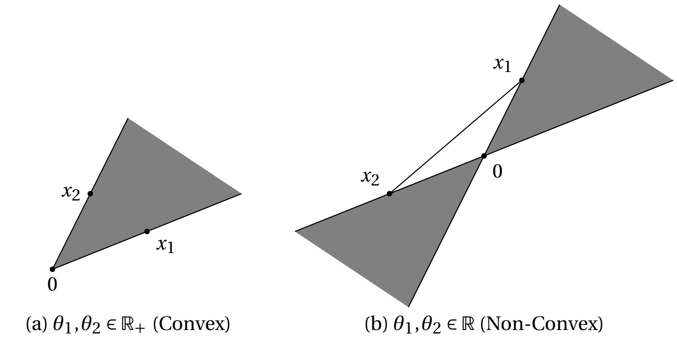


Figure 1.2: Convex and non-convex set examples.

[1] Euclidean norm is the second norm given by $\|x_1 - x_2\|_2 = \sqrt{(x_1 - x_2)^T (x_1 - x_2)}$



(a) $\theta_1, \theta_2 \in \mathbb{R}_+$ (Convex)

(b) $\theta_1, \theta_2 \in \mathbb{R}$ (Non-Convex)

Figure 1.3: Convex and non-convex cone examples.

The second-order cone in $\mathbb{R}^3$ is also known as *quadratic cone* or *Lorentz cone* or *ice-cream cone* and can be represented as $\mathscr{C} = \{(x_1, x_2, t) \mid \|x_i\|_2 \leq t,\ i = 1,2\} \subseteq \mathbb{R}^3$ and is shown in the Figure 1.4.

To formulate an optimisation problem as a convex one, all the functions in the problem must be convex. Problems like the descent guidance do not have convex constraints due to the lower bound on the thrust and also the dynamics governing the motion of the SC is non-linear. Hence it is necessary to be able to convert these to convex functions to be able to solve the problem as a convex optimisation problem. A function, $f : \mathbb{R}^n \to \mathbb{R}$ is said to be convex if it satisfies the following three conditions:

(i) dom f is a convex set

(ii) All $x,\ y \in \mathrm{dom} f$

(iii) $0 \leq \theta \leq 1$

This leads to the inequality $f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$ to hold true. Figure 1.5a shows the geometric interpretation of this inequality. As can be seen that for a function to be convex, the line joining any two points of the function in the set must lie above the graph of the function of all the points. Since it cannot be intuitively said if a function is convex or not by simply looking at the inequality function, the way to recognize the function's convexity is that it must fulfill the first and second order necessary conditions.

First Order: If f is differentiable, then f is convex iff dom f is convex and

$$f(x) \geq f(x_0) + \nabla f(x_0)^T (x - x_0), \quad \forall x, y \in \mathrm{dom} f \tag{1.9}$$

Second Order: If f is twice differentiable, then f is convex iff dom f is convex and $\nabla^2 f$ is positive semidefinite for all $x \in f$

$$\nabla^2 f \succeq 0 \tag{1.10}$$

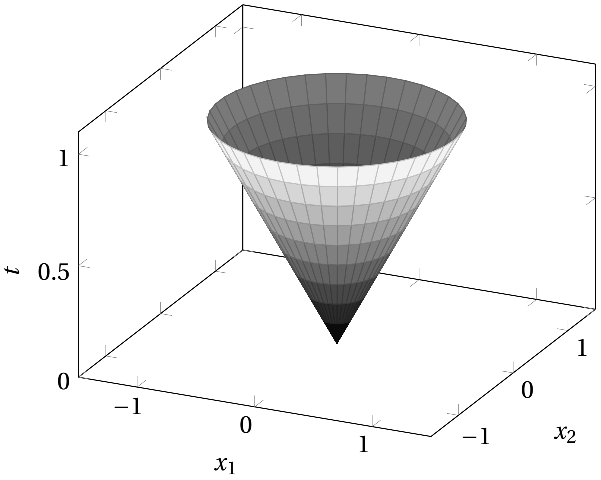
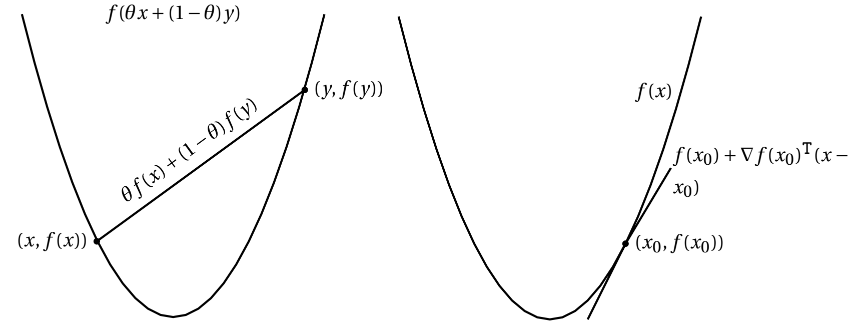


Figure 1.4: Second order cone boundary. It represents a filled cone.


$f(\theta x+(1-\theta)y)$
$(y,f(y))$
$\theta f(x)+(1-\theta)f(y)$
$(x,f(x))$
$f(x)$
$f(x_0)+\nabla f(x_0)^{\mathrm{T}}(x-x_0)$
$(x_0,f(x_0))$

(a) The line joining the two points will always lie above the graph

(b) First order condition for convex functions

Figure 1.5: Graphical representation and properties of a convex function.

As can be seen the right hand side of the first order condition is a Taylor approximation of the function $f(x)$ and the inequality makes it a global underestimator of the function if its convex i.e. the line shown in Figure 1.5b always lies below the graph if the function is convex. This enables deriving global information from local information making it the most important property of convex functions and resulting in beneficial properties in convex optimisation problems Boyd and Vandenberghe 2010. The second order condition is nothing but that the Hessian must be positive semidefinite i.e. the derivative must be non-decreasing and the function of x should have an upward curvature.

These conditions hold for differentiable functions but non differentiable convex functions exist, eg. $f(x) = |x|$. As can be interpreted at $x = 0$, there is a kink in the graph and is a non differential at the point, but it is a convex function. hence the above conditions are not a general requirement for a function to be convex but are a check for differentiable functions.

1.1.2. Convex optimisation Problems: Second-Order Cone Programming

Convex optimisation problems possess certain properties that enable efficient solving of the problem with robust solutions. The major difficulty is the recognition and formulation of the convex problem. Once the two are achieved, solving the problem is almost as similar as least squares or linear programming. Here we will discuss SOCP in detail, since it is the form used for the descent guidance. We put forth the form of *Convex optimisation problems* (Boyd and Vandenberghe 2010)

$$\begin{aligned} \text{minimise} \quad & f_0(x) \\ \text{subject to} \quad & g_i(x) \le 0, \quad i = 1,\dots,m. \\ & a_j^T x = b_j, \quad j = 1,\dots,p. \end{aligned} \tag{1.11}$$

In comparison with the general optimisation problem, the convex optimisation problem has the following three requirements,

(i) The objective function f_0 has to be a *convex* function; and

(ii) The inequality constraints $g_1,\dots,g_m$ have to be *convex*; and

(iii) The equality constraints $a_j^T x - b_j$ or $h_1,\dots,h_p$ must be *affine*.

The advantage of the constraints being convex is the feasible set of the problem from the intersection of all the constraints is also convex. This is because of the property that intersection of convex sets is convex (Boyd and Vandenberghe 2010).

$$\mathcal{D} = \bigcap_{i=0}^{m} \operatorname{dom} g_i \tag{1.12}$$

For the problem to be categorised as a SOCP, the following requirements must be met,

(i) The objective function f_0 has to be a *affine* function; and

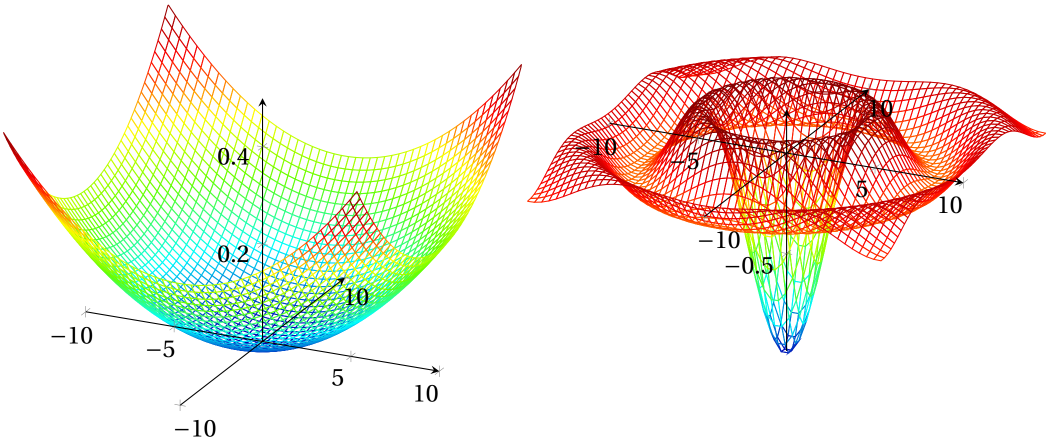


Figure 1.6: Optimal solutions of convex and non-convex problems

(ii) The inequality constraints $g_1, \dots, g_m$ have to be *convex cones*; and

(iii) The equality constraints $a_j^T x - b_j$ or $h_1, \dots, h_p$ must be *affine*.

The form of the convex problem is updated to SOCP as:

$$\begin{aligned} \text{minimise} \quad & f^T x \\ \text{subject to} \quad & \|A_i(x) + b_i\| \leq c_i^T x + d_i, \quad i = 1, \dots, m. \\ & Fx = g \end{aligned} \tag{1.13}$$

where $A \in \mathbb{R}^{k \times n}$ and $F \in \mathbb{R}^{n+1}$. Comparing with Eq. 8.8 it can be seen that the inequality constraints represent a second-order cone.

$$\mathscr{C} = \{(A_i x + b_i, c_i^T x + d_i) \mid \|A_i x = b_i\| \leq c_i^T x + d_i,\ i + 1, \dots, m\} \subseteq \mathbb{R}^{k+1} \tag{1.14}$$

The obvious reason for using convex optimisation is the necessary existence of *only one global minimum* for convex function over its domain. The probability of getting stuck at a local minimum is therefore nil. Also the negative gradient of the function will always lead to the optimum. A parabola is used to represent a convex function in Figure 1.6a since it is easy to visualize but it does not generalise convexity. In reality convex sets are not intuitive or easy to visualize.

As mentioned earlier the difficult part is the problem formulation in the SOCP form. SOCP possesses remarkable properties which enable speedy solving of the problem. These solvers are on the verge of being a mature technology. At the end of this chapter the descent guidance will be formulated as an SOCP.

The next sections are towards understanding the steps how a nonconvex, nonlinear problem can be reformulated to a convex one without compromising on the constraints or objective function. The convex guidance developed by Açıkmeşe and Ploen 2007 called *lossless convexification* will be briefly discussed to give an overview of the existing methods. Then we will move on to *successive convexification* which has been developed and used in a few recent works (Mao, Szmuk, and Acıkmeşe 2016), (Szmuk, Utku, and Açıkmeşe 2017), (Szmuk and Açıkmeşe 2018), (Açıkmeşe, Carson, and Blackmore 2013) and (Xinfu and Ping 2014).

1.2. Lossless Convexification

Referring back to Problem 2b on page—, it can be seen that the source of nonlinearity comes from the dynamics due to nonlinear gravitational acceleration of the target body and nonconvexity from the control constraint. The method of lossless convexification by Açıkmeşe and Ploen 2007 provides a way to mathematically manipulate nonconvex functions to a second order cone form, thereby making it convex and enabling the computation of the global minimum to a given accuracy with a deterministic upper bound on the number of iterations for convergence. We discuss each step of the procedure in the section and reformulate the nonconvex constraints to a convex form.

1.2.1. Introduction of Slack Variables

The control constraint dealing with the thrust magnitude given in the optimisation problem is the primary nonconvex constraint in the problem. Once the thrusters have been initiated, they cannot be turned off during the maneuver and also there is a minimum level of thrust at which the propulsion system can operate. Below this level the operation is unreliable and hence the thrust magnitude needs to be bounded at the lower end too.

$$0 < T_{min} \leq \|\boldsymbol{T}(t)\| \leq T_{max} \tag{1.15}$$

The lower bound defines the nonconvex feasible control space.[2] This can be proved geometrically by considering a two dimensional thrust as follows.[3]

$$sup\{\|\boldsymbol{T}\| \mid T_z = 0,\ \|\boldsymbol{T}(t)\| \leq T_{max}\} = \{\|\boldsymbol{T}\| \mid \sqrt{T_x^2 + T_y^2} = T_{max}\} \tag{1.16}$$

$$inf\{\|\boldsymbol{T}\| \mid T_z = 0,\ 0 < T_{min} \leq \|\boldsymbol{T}(t)\|\} = \{\|\boldsymbol{T}\| \mid \sqrt{T_x^2 + T_y^2} = T_{min}\} \tag{1.17}$$

$$\tag{1.18}$$

Considering T_{max} and T_{min}, the constraint is geometrically shown in Figure 1.7 as an annular diagram. Repeating the definition of a convex function, it states that a line drawn between two points in the convex set must also lie in the set. As can be seen that the lines passing through the central region of the annulus to connect two points do not lie inside the convex set themselves hence proving the nonconvexity of the function.

Lossless convexification is a simple but effective way of manipulating the nonconvex function to make it convex. A scalar slack variable is introduced to relax the control constraint such that the control space becomes *affine* which makes it inherently convex (Açıkmeşe, Carson, and Blackmore 2013). For the thrust control constraint, its magnitude is replaced by the scalar variable, Γ and the variable itself is constrained by the thrust magnitude.

$$0 < T_{min} \leq \Gamma \leq T_{max} \tag{1.19}$$

$$\|\boldsymbol{T}(t)\| \leq \Gamma \tag{1.20}$$

The thrust magnitude has been relaxed by Γ without a lower bound on it, which means that $\|\boldsymbol{T}(t)\| \leq T_{min}$ is feasible in the new problem, but not in the original problem, with the bounded thrust magnitude. The original control space is therefore a subset of the relaxed control space, represented as below

$$\{\boldsymbol{T} \mid 0 \leq T_{min} \leq \|\boldsymbol{T}(t)\| \leq T_{max}\} \subset \{\boldsymbol{T},\ \Gamma \mid 0 < T_{min} \leq \Gamma \leq T_{max}\} \tag{1.21}$$

However, Açıkmeşe and Ploen 2007 have proved that an optimal solution to the new problem with relaxed constraints will satisfy the original thrust constraints as well. The proof is quite long and hence will not be presented in this thesis and the interested reader can refer to Açıkmeşe and Ploen 2007 for the same. The central conclusion from the proof is that the optimal control history for both Γ and $\|\boldsymbol{T}\|$ is same as shown below which also requires $\|\boldsymbol{T}^*\|$ to satisfy the Eq. 1.19.

$$\Gamma^*(t) = \|\boldsymbol{T}^*(t)\| \tag{1.22}$$

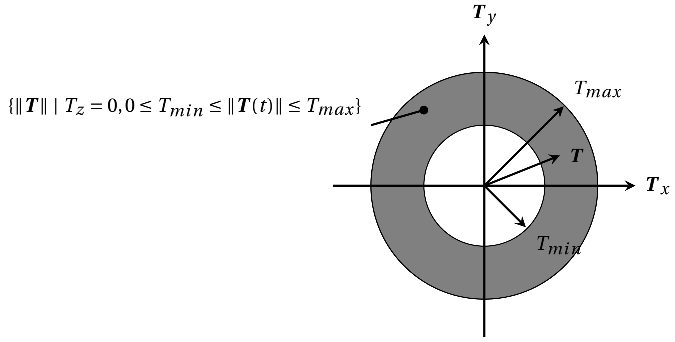

$\boldsymbol{T}_y$
$\{\|\boldsymbol{T}\| \mid T_z = 0, 0 \leq T_{min} \leq \|\boldsymbol{T}(t)\| \leq T_{max}\}$
T_{max}
$\boldsymbol{T}$
$\boldsymbol{T}_x$
T_{min}

Figure 1.7: Optimal solutions of convex and non-convex problems

[2] An upper bounded constraint, $0 \leq \|\boldsymbol{T}_{\mathrm{B}}(t)\| \leq T_{max}$ would be convex. Visualizing Figure 1.7, with just an upper bound, the set would be a closed circle which is convex.

[3] For $T_z \neq 0$, the region could be imagined as a partially hollow sphere.

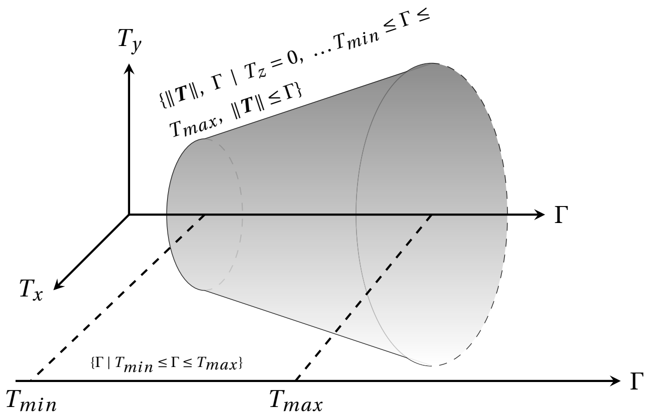

T_y
$\{\|\mathbf{T}\|,\ \Gamma\ |\ T_z = 0,\ \ldots T_{min} \le \Gamma \le$
$T_{max},\ \|\mathbf{T}\| \le \Gamma\}$
Γ
T_x
$\{\Gamma \mid T_{min} \le \Gamma \le T_{max}\}$
T_{min}
T_{max}
Γ

Figure 1.8: Second order cone boundary. It represents a filled cone.

Figure 1.8 represents the visualisation of the relaxed control space for understanding the convexity of the new constraints. The combination of both the relaxed constraints leads to a truncated cone as the control space, which is convex by nature. The affine control space representation shown on the axis Γ is represented similarly in the combined figure. The binding values T_{min} and T_{max} form the end caps of the truncated cone at their locations on the Γ axis parallel to the plane of the torque axes.

In Subsection 1.1.2 the require form for a constraint has been put forth in Eq. 1.13. To take advantage of the beneficial properties of SOCPs, the new constraints need to be in the same form. Therefore any other constraints that might be needed for the powered descent guidance will be added to the problem in later stages in this form. For now, the newly formulated convex problem is as shown in Problem 2c.

➽ PROBLEM 2c: Convexified Control Constraints

minimise $-m(t_f)$

subject to:

Kinematics:

$$\dot{\boldsymbol{r}}^{\mathtt{I}}(t) = \boldsymbol{v}^{\mathtt{I}}_{\mathtt{B/I}}(t), \qquad \dot{\boldsymbol{q}}_{\mathtt{B/I}}(t) = \frac{1}{2}\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t) \otimes \boldsymbol{q}_{\mathtt{B/I}}(t)$$

Dynamics:

$$\dot{m}(t) = -\alpha\Gamma(t), \qquad \dot{\boldsymbol{v}}^{\mathtt{I}}_{\mathtt{B/I}}(t) = \frac{\boldsymbol{T}^{\mathtt{I}}(t)}{m(t)} + \boldsymbol{g}_{\mathtt{I}}$$

$$\dot{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}}(t) = J^{-1}(\boldsymbol{r}_{\mathtt{B}}(t) \times \boldsymbol{T}^{\mathtt{B}}(t) - \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t) \times J\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t))$$

Control Constraints:

$$0 < T_{min} \le \Gamma \le T_{max}$$

$$\|\boldsymbol{T}^{\mathtt{I}}(t)\| \le \Gamma$$

Boundary Conditions:

$$m(0) = m_{wet}$$

$$\boldsymbol{r}^{\mathtt{I}}(0) = \boldsymbol{r}^{\mathtt{I}}_0, \quad \boldsymbol{r}^{\mathtt{I}}(t_f) = \boldsymbol{r}^{\mathtt{I}}_f, \quad \boldsymbol{v}^{\mathtt{I}}_{\mathtt{B/I}}(0) = \boldsymbol{v}^{\mathtt{I}}_{\mathtt{B/I}_0}, \quad \boldsymbol{v}^{\mathtt{I}}_{\mathtt{B/I}}(t_f) = \boldsymbol{v}^{\mathtt{I}}_{\mathtt{B/I}_f}$$

$$\boldsymbol{q}_{\mathtt{B/I}}(0) = \boldsymbol{q}_{\mathtt{B/I}_0}, \quad \boldsymbol{q}_{\mathtt{B/I}}(t_f) = \boldsymbol{q}_{\mathtt{B/I}_f}, \quad \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(0) = \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}_0}, \quad \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t_f) = \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}_f}$$

Additional Constraints:

$$\|A_i(\boldsymbol{x}) + b_i\| \le c_i^T\boldsymbol{x} + d_i, \quad i = 1, \ldots, m$$

1.2.2. Change of Variables

The Problem 2c has convexified control and additional constraints but is still not fit for convex solvers due to the nonlinearity in the dynamics. The nonlinearity in the mass flow dynamics will be dealt by the change of variables. The control variables $\boldsymbol{T}(t)$ and $\Gamma(t)$ will be changed to establish linear second order differential equation that can ultimately represent a linear, time-invariant, state-space system (LTI). The control variables are transformed to accelerations from thrust forces as below,

$$\sigma(t) \triangleq \frac{\Gamma(t)}{m(t)}, \quad \boldsymbol{\tau}(t) \triangleq \frac{\boldsymbol{T}(t)}{m(t)} \tag{1.23}$$

It is important to mention here, that this change of variables maintains the equivalence in the optimal controls as it was for $\boldsymbol{T}(t)$ and $\Gamma(t)$ i.e. $\|\boldsymbol{\tau}^*(t)\| = \sigma^*(t)$. Using the new variables $\sigma(t)$ and $\boldsymbol{\tau}(t)$ the EOMs for translational motion and mass flow can be rephrased as:

$$\boldsymbol{v}(t) = \boldsymbol{\tau}(t) + \boldsymbol{g} \tag{1.24}$$

$$\frac{\dot{m}(t)}{m(t)} = -\alpha\sigma(t) \tag{1.25}$$

As can be seen the EOMs are still not linearised due to the nonlinear gravity field in Eq. 1.24 and the time dependent mass term in Eq. 1.25. Further steps need to be taken in order to linearise them as well as the other nonlinear and dynamics constraints. The cost function and the control constraints need to be changed with these variables. Eq. 1.25 is a first order, ordinary differential equation (ODE). This ODE can be solved as (Boyce and DiPrima 2008):

$$m(t) = m_0 \exp\left(-\alpha \int_0^{t_f} \sigma(t)dt\right) \tag{1.26}$$

From the above equation, it can be seen that minimizing the exponential function would maximise $m(t)$. Hence the cost function is reformulated to a new but valid cost function as shown below.

$$\text{minimise} - m(t_f) = \text{minimise} \int_0^{t_f} \sigma(t)dt \tag{1.27}$$

The control constraints with the new variables are given as

$$\|\boldsymbol{\tau}(t)\| \le \sigma(t) \tag{1.28}$$

$$\frac{T_{min}}{m(t)} \le \sigma(t) \le \frac{T_{max}}{m(t)} \quad \forall\ t \in [0, t_f] \tag{1.29}$$

But as can be seen the second constraint is again a source of nonlinearity due to the time dependent mass. —-

This nonconvexity can be removed by another change of variables, given as

$$z(t) \triangleq \ln m(t) \tag{1.30}$$

The ODE for mass flow in Eq. 1.26 is changed to

$$\dot{z}(t) = -\alpha\sigma(t) \tag{1.31}$$

The inequality control constraint is therefore now rewritten as

$$T_{min}\, e^{-z(t)} \le \sigma(t) \le T_{max}\, e^{-z(t)} \tag{1.32}$$

This constraint is still not convexified as can be understood from the similar graph of the function $1/z$ and also the exponential cannot be a part of the SOCP constraint. In order to take care of this a Taylor series approximation is used for the the exponential function as below (Boyce).

$$f(z) = \sum_{n=0}^{\infty} \frac{f^{(n)} z_0}{n!} (z - z_0)^n \tag{1.33}$$

where, z_0 and n denote the nominal around which the series is to be extended and the n^{th} order derivative respectively. Considering upto the 2nd order derivative would suffice since it provides enough precision and also it would keep the lower bound thrust convex, which is the purpose of the variable changes. The lower bound on the thrust can be represented as

$$T_{min}\, e^{-z_1(t)} [1 - (z(t) - z_1(t) + \frac{1}{2} z(t) - z_1(t))^2] \le \sigma(t) \tag{1.34}$$

where, z_1 is the nominal point. For the upper bound on the thrust as discussed above, anything above the first order would maintain nonconvexity and therefore we just the first order Taylor expansion will be used

$$\sigma(t) \leq T_{max}\, e^{-z_2(t)}[1-(z(t)-z_2(t))] \tag{1.35}$$

where, z_2 is the nominal point. The pre-factors multiplied to the expansion series can be abbreviated as

$$\mu_1(t) = T_{min}\, e^{-z_1(t)}, \quad \mu_2(t) = T_{max}\, e^{-z_2(t)} \tag{1.36}$$

It is important that the mass equivalent variables z_1 and z_2 or the nominal points are logical to ensure sufficient accuracy of the newly defined constraints. This can be done by reiterating the definition of z and setting the slack variable Γ to the lower and upper thrust values for the lower and upper bounds respectively.

$$z_1(t) = \ln m(t) = \ln(m_0 - \alpha T_{min} t) \tag{1.37}$$

$$z_2(t) = \ln m(t) = \ln(m_0 - \alpha T_{max} t) \tag{1.38}$$

An additional constraint needs to be put on z such that the physical bounds on it remain intact.

$$z_1(t) \leq z(t) \leq z_2(t) \tag{1.39}$$

Using all the newly defined constraints the SOCP can be set up for the next phase where we need to convexify the remaining nonlinear dynamics. Also using the equations for relative frame kinematics and dynamics we can update the convexified problem in 2c.

➽ PROBLEM 2d: Problem Formulation in the Relative Frame

minimise $\int_0^{t_f} \sigma(t)dt$

subject to:

Kinematics:

$\dot{\boldsymbol{r}}^{\mathtt{A}}(t) = \boldsymbol{v}^{\mathtt{A}}_{\mathtt{B/A}}(t), \qquad \dot{\boldsymbol{q}}_{\mathtt{B/A}}(t) = \frac{1}{2}\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}}(t) \otimes \boldsymbol{q}_{\mathtt{B/A}}(t)$

Dynamics:

$\dot{z}(t) = -\alpha\sigma(t)$

$\dot{\boldsymbol{v}}^{\mathtt{A}}_{\mathtt{B/A}}(t) = \boldsymbol{\tau}^{\mathtt{A}}(t) + \boldsymbol{g}^{A} - 2\boldsymbol{\omega}^{\mathtt{A}}_{\mathtt{A/I}(t)} \times \boldsymbol{v}^{\mathtt{A}}_{\mathtt{B/A}}(t) - \boldsymbol{\omega}^{\mathtt{A}}_{\mathtt{A/I}}(t) \times \boldsymbol{\omega}^{\mathtt{A}}_{\mathtt{A/I}}(t) \times \boldsymbol{r}^{\mathtt{A}}(t)$

$\dot{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}}(t) = \boldsymbol{J}^{-1}(\boldsymbol{T}^{\mathtt{B}}(t) - \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}}(t) \times \boldsymbol{J}\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}}(t) - \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{A/I}}(t) \times \boldsymbol{J}\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{A/I}})(t) - \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{A/I}}(t) \times \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}}(t)$

Control Constraints:

$\|\boldsymbol{\tau}^{\mathtt{A}}(t)\| \leq \sigma(t)$

$\mu_1(t)\{1-[z(t)-z_1(t)] + \frac{1}{2}[z_1(t)-z_2(t)]^2\} \leq \sigma(t) \leq \mu_2(t)\{1-[z(t)-z_2(t)]\}$

$z_1(t) \leq z(t) \leq z_2(t)$

Boundary Conditions:

$m(0) = m_{wet}, \quad \boldsymbol{q}_{\mathtt{A/I}}(0) = \boldsymbol{q}_{\mathtt{A/I}_0}$

$\boldsymbol{r}^{\mathtt{A}}(0) = \boldsymbol{r}^{\mathtt{A}}_0, \quad \boldsymbol{r}^{\mathtt{A}}(t_f) = \boldsymbol{r}^{\mathtt{A}}_f, \quad \boldsymbol{v}^{\mathtt{A}}_{\mathtt{B/A}}(0) = \boldsymbol{v}^{\mathtt{A}}_{\mathtt{B/A}_0}, \quad \boldsymbol{v}^{\mathtt{A}}_{\mathtt{B/A}}(t_f) = \boldsymbol{v}^{\mathtt{A}}_{\mathtt{B/A}_f}$

$\boldsymbol{q}_{\mathtt{B/A}}(0) = \boldsymbol{q}_{\mathtt{B/A}_0}, \quad \boldsymbol{q}_{\mathtt{B/A}}(t_f) = \boldsymbol{q}_{\mathtt{B/A}_f}, \quad \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}}(0) = \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}_0}, \quad \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}}(t_f) = \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/A}_f}$

Additional Constraints:

$\|A_i(\boldsymbol{x}) + b_i\| \leq c_i^T \boldsymbol{x} + d_i, \quad i = 1, \ldots, m$

1.3. Dual Quaternion Problem Formulation

Before we can move to the next section of dealing with non linearities in the dynamics, we need to reformulate the problem in the format of DQs. We will deal with the kinematics and dynamics step by step along with the control constraints and boundary conditions to establish the problem in DQs. In the next section of successive convexification we will take care of the non linear kinematics and dynamics in the DQ form.

As can be seen the problem in 2d is in the inertial frame and this is appealing due to the simpler EOMs. However this leads to the final time boundary conditions being dependent on time due to the rotation of the target body about itself. This could render the problem intractable. Also additional constraints of the extended guidance problem (to be discussed in Section 8.6) are defined with respect to the landing site and are convex by nature. If they are to be redefined in the inertial frame, non-convexities will occur in these constraints. To avoid these problems we define the whole problem in the asteroid rotating reference frame. This retains the convexifications done so far and maintains the convex nature of the additional constraints to be added later on. It also has constant initial and final boundary conditions retaining the tractability of the problem. The problem in asteroid rotating frame, $\mathtt{A}$ is given as,

➽ PROBLEM 2e: Problem Formulation using Dual Quaternions

minimise $\int_0^{t_f} \sigma(t) dt$

subject to:

Kinematics:
$$\dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}} = \frac{1}{2}\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}} \check{\otimes} \check{\boldsymbol{q}}_{\mathtt{B/A}}$$

Dynamics:
$$\dot{z}(t) = -\alpha\sigma(t)$$
$$\dot{\check{\boldsymbol{\omega}}}^{\mathtt{B}}_{\mathtt{B/A}} = \check{\boldsymbol{J}}^{-1}(\check{\boldsymbol{F}}^{\mathtt{B}} + \check{\boldsymbol{F}}^{\mathtt{B}}_{D} - (\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}} + \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}) \check{\times} \check{\boldsymbol{J}}(\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}} + \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}) - \check{\boldsymbol{J}}\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}} \check{\times} \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}} - \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}} \check{\times} \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}} \check{\times} \boldsymbol{A})$$

Control Constraints:
$$\|\boldsymbol{\tau}^{\mathtt{A}}(t)\| \le \sigma(t)$$
$$\mu_1(t)\{1 - [z(t) - z_1(t)] + \frac{1}{2}[z_1(t) - z_2(t)]^2\} \le \sigma(t) \le \mu_2(t)\{1 - [z(t) - z_2(t)]\}$$
$$z_1(t) \le z(t) \le z_2(t)$$

Boundary Conditions:
$$m(0) = m_{wet}, \quad \boldsymbol{\omega}^{\mathtt{A}}_{\mathtt{A/I}} = \text{constant}$$
$$\dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}}(0) = \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A_0}}, \quad \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}}(t_f) = \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A_0}}, \quad \dot{\check{\boldsymbol{\omega}}}^{\mathtt{B}}_{\mathtt{B/A}}(0) = \dot{\check{\boldsymbol{\omega}}}^{\mathtt{B}}_{\mathtt{B/A_0}}, \quad \dot{\check{\boldsymbol{\omega}}}^{\mathtt{B}}_{\mathtt{B/A}}(t_f) = \dot{\check{\boldsymbol{\omega}}}^{\mathtt{B}}_{\mathtt{B/A}_f}$$

Additional Constraints:
$$\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{A/I}} = \boldsymbol{C}_{\mathtt{B/A}}\boldsymbol{\omega}^{\mathtt{A}}_{\mathtt{A/I}}$$
$$\|A_i(\boldsymbol{x}) + b_i\| \le c_i^T\boldsymbol{x} + d_i, \quad i = 1, \dots, m$$

1.4. Successive Convexification

Until here the method of lossless convexification proves useful for convexifying the control constraint but it fails to convexify the remaining nonlinear kinematics and dynamics. Mao, Szmuk, and Acıkmeşe 2016 presented a novel method to deal with such nonlinear dynamics, *successive convexification.* The underlying logic is to linearise the non convexities in the dynamics and control constraints about the solution for that iteration, say k^{th} iteration. This leads to a convex sub-problem preferably in the SOCP format, which can be solved to full optimality resulting in a new solution for the $(k+1)^{th}$ iterate. This process is continued successively till convergence. A continuous time convergence analysis done by Mao, Szmuk, and Acıkmeşe 2016 guarantees convergence to the optimal solution of the original problem. The proof is too extensive to be explained in this thesis and can be referred to in Mao, Szmuk, and Acıkmeşe 2016. The procedure for successive convexification will be explained step by step through the sub sections below.

1.4.1. Linearisation

Problem 2b in Chapter 5 shows the presence of nonlinearities in the attitude kinematics, dynamics, translational dynamics, mass dynamics and thrust constraints. We deal with a non-linear polyhedron gravity field for asteroids which adds non linearity in the gravitational acceleration present in the translational dynamics and also a non linearity due to the perturbing acceleration. All of these need to be convexified so to be suitable for the SOCP problem formulation. For this we first redefine the state vectors discussed in Chapter 4 as below and also define the control vector.

$$\check{\boldsymbol{x}}_{\mathtt{A}}(t) = \begin{pmatrix} m(t) \\ \check{\boldsymbol{q}}_{\mathtt{B/A}}(t) \\ \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}}(t) \\ \check{\boldsymbol{F}}^{\mathtt{B}}(t) \\ \dot{\check{\boldsymbol{F}}}^{\mathtt{B}}(t) \end{pmatrix}, \quad \check{\boldsymbol{u}}_{\mathtt{A}}(t) = \left(\ddot{\check{\boldsymbol{F}}}^{\mathtt{B}}(t)\right) \tag{1.40}$$

The inclusion of the dual force and its first derivative in the state vector whilst the second derivative of the dual force in the control vector separates the state and control vector. This separation is beneficial since it prevents high frequency jitters in control due to the nonlinear dependency of the state on the control. The presence of these high frequency jitters have been found in **paper** and their absence by separation has also been proved. Hence we will go ahead with the separated vectors.

The convexification of the equality constraint is done by linearising them using first order Taylor approximation. But this does not ensure the optimal solution of the problem with linearised constraints to be the one for the original problem as well. To recover this optimality they are successively linearised till convergence. The continuous time nonlinear dynamics is of the form,

$$\dot{\boldsymbol{x}}(t) = f(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \quad a.e. \quad 0 \leq t \leq t_f \tag{1.41}$$

where, $\boldsymbol{x} : [0, t_f] \to \mathbb{R}^n$ represents the state trajectory, $\boldsymbol{u} : [0, t_f] \to \mathbb{R}^m$, the control inputs and $f : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$, the control-state mapping function. The control-state function will be assumed to be Fréchet differentiable [4] with respect to all arguments and the control input to be Lebesgue integrable i.e. $\boldsymbol{u}$ is a measurable non negative decreasing function with a finite Lebesgue integral.[5] The first order Taylor expansion of the dynamics can be given as

$$\begin{aligned} \dot{\boldsymbol{x}}(t) = f(\bar{\boldsymbol{x}}_{k-1}(t), \bar{\boldsymbol{u}}_{k-1}(t), t) + \frac{\partial}{\partial x} f(\boldsymbol{x}_{k-1}(t), \boldsymbol{u}_{k-1}(t), t)\Big(\boldsymbol{x}_{k-1}(t) - \bar{\boldsymbol{x}}_{k-1}(t)\Big) + \ldots \\ + \frac{\partial}{\partial u} f(\boldsymbol{x}_{k-1}(t), \boldsymbol{u}_{k-1}(t), t)\Big(\boldsymbol{u}_{k-1}(t) - \bar{\boldsymbol{u}}_{k-1}(t)\Big) \end{aligned} \tag{1.42}$$

The above equation can be abbreviated with the following substitutions,

$$\boldsymbol{A}(t) = \frac{\partial}{\partial x} f(\boldsymbol{x}_{k-1}(t), \boldsymbol{u}_{k-1}(t), t)\bigg|_{\bar{x}_{k-1}, \bar{u}_{k-1}} \tag{1.43}$$

$$\boldsymbol{B}(t) = \frac{\partial}{\partial u} f(\boldsymbol{x}_{k-1}(t), \boldsymbol{u}_{k-1}(t), t)\bigg|_{\bar{x}_{k-1}, \bar{u}_{k-1}} \tag{1.44}$$

$$\boldsymbol{z}(t) = f(\bar{\boldsymbol{x}}_{k-1}(t), \bar{\boldsymbol{u}}_{k-1}(t), t) - A(t)\bar{\boldsymbol{x}}_{k-1}(t) - B(t)\bar{\boldsymbol{u}}_{k-1}(t) \tag{1.45}$$

The linearised dynamics would therefore look like,

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_{k-1}(t)\boldsymbol{x}_{k-1}(t) + \boldsymbol{B}_{k-1}(t)\boldsymbol{u}_{k-1}(t) + \boldsymbol{z}_{k-1}(t) \tag{1.46}$$

The problem formulation as shown in Problem 2e, is as per lossless convexification and includes slack variables to convexify the nonconvex controls. However the nonconvexity in the dynamics still exists. We can linearise all the non convex constraints and dynamics in the successive convexification algorithm or we can include the slack variables for the convexified constraints. From the state vector and control vector in Eq. 7.40 we can state the dynamic state as shown in the next page.

The approach is to directly linearise the nonlinearities in the dynamic state at the initial and final time about $\bar{\boldsymbol{x}}(t)$. We reiterate the state differential from the relative state kinematics and dynamics as discussed in Chapter 4 and derive the partial differentials with respect to the state vector to linearise the non-linear dynamics using Taylor series expansion of the first order.

$$\dot{\check{\boldsymbol{x}}}_{\mathtt{A}}(t) = \begin{pmatrix} -\alpha\|\boldsymbol{F}^{\mathtt{B}}(t)\| \\ \frac{1}{2}\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}}(t)\check{\otimes}\check{\boldsymbol{q}}_{\mathtt{B/A}}(t) \\ \check{\boldsymbol{J}}^{-1}(\check{\boldsymbol{F}}^{\mathtt{B}}(t) + \check{\boldsymbol{F}}^{\mathtt{B}}_{D}(t) - (\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}}(t) + \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}(t))\check{\times}\check{\boldsymbol{J}}(\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}}(t) + \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}(t)) - \check{\boldsymbol{J}}\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}(t)\check{\times}\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/A}}(t) - \check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}(t)\check{\times}\check{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{A/I}}(t)\check{\times}\boldsymbol{R}(t)) \\ \dot{\check{\boldsymbol{F}}}^{\mathtt{B}}(t) \\ \check{\boldsymbol{u}}_{\mathtt{B}}(t) \end{pmatrix} \tag{1.47}$$

[4] A function f is Fréchet differentiable at a if $\lim_{x \to a} = \frac{f(x) - f(a)}{x - a}$ exists.

[5] Lebesgue integral is by definition $\int_0^\infty u(t)dt$.

$$f_{\boldsymbol{F}^{\mathrm{B}},t} = \left(\|\bar{\boldsymbol{F}}^{\mathrm{B}}(t)\| + \frac{\bar{\boldsymbol{F}}^{\mathrm{B}}(t)}{\|\bar{\boldsymbol{F}}^{\mathrm{B}}(t)\|} (\boldsymbol{F}^{\mathrm{B}}(t) - \bar{\boldsymbol{F}}^{\mathrm{B}}(t) \right) \tag{1.48}$$

$$f_{\dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}},t} = \dot{\check{\bar{\boldsymbol{q}}}}_{\mathtt{B/A}}(t) + \frac{\partial \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}}}{\partial \boldsymbol{x}} \bigg|_{\bar{\boldsymbol{x}}_{\mathtt{A}}(t)} (\boldsymbol{x}_{\mathtt{A}}(t) - \bar{\boldsymbol{x}}_{\mathtt{A}}(t)) \tag{1.49}$$

$$f_{\dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}},t} = \dot{\check{\bar{\boldsymbol{\omega}}}}^{\mathrm{B}}_{\mathtt{B/A}}(t) + \frac{\partial \dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}}{\partial \boldsymbol{x}} \bigg|_{\bar{\boldsymbol{x}}_{\mathtt{A}}(t)} (\boldsymbol{x}_{\mathtt{A}}(t) - \bar{\boldsymbol{x}}_{\mathtt{A}}(t)) \tag{1.50}$$

Now we will derive the linearisation matrices for continuous time, for which we first derive the partial differentials with respect to state and controls.

Kinematics:

$$\begin{aligned} \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}}(t) &= \frac{1}{2}\left[\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}(t)\check{\otimes}\right]\check{\boldsymbol{q}}_{\mathtt{B/A}}(t) \\ &= \frac{1}{2}\begin{bmatrix} [\boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{B/A}}\otimes] & 0_{4\times4} \\ [\boldsymbol{v}^{\mathrm{B}}_{\mathtt{B/A}}\otimes] & [\boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{B/A}}\otimes] \end{bmatrix} \begin{pmatrix} \boldsymbol{q}_{\mathtt{B/A}} \\ \boldsymbol{q}_{d_{\mathtt{B/A}}} \end{pmatrix} \\ &= \frac{1}{2}\begin{bmatrix} [\boldsymbol{q}_{\mathtt{B/A}}\odot] & 0_{4\times4} \\ [\boldsymbol{q}_{d_{\mathtt{B/A}}}\odot] & [\boldsymbol{q}_{\mathtt{B/A}}\odot] \end{bmatrix} \begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{B/A}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{B/A}} \end{pmatrix} \end{aligned} \tag{1.51}$$

We find the partial differentials below:

$$\Phi_{\check{q}} = \frac{\partial \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}}}{\partial \check{\boldsymbol{q}}_{\mathtt{B/A}}}\bigg|_{\bar{\boldsymbol{x}}} = \begin{bmatrix} [\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}\otimes] & 0_{4\times4} \\ [\bar{\boldsymbol{v}}^{\mathrm{B}}_{\mathtt{B/A}}\otimes] & [\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}\otimes] \end{bmatrix} \tag{1.52}$$

$$\Phi_{\check{\omega}} = \frac{\partial \dot{\check{\boldsymbol{q}}}_{\mathtt{B/A}}}{\partial \check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}}\bigg|_{\bar{\boldsymbol{x}}} = \begin{bmatrix} [\bar{\boldsymbol{q}}_{\mathtt{B/A}}\odot] & 0_{4\times4} \\ [\bar{\boldsymbol{q}}_{d_{\mathtt{B/A}}}\odot] & [\bar{\boldsymbol{q}}_{\mathtt{B/A}}\odot] \end{bmatrix} \tag{1.53}$$

Dynamics:

$$\begin{aligned} \dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}(t) &= \check{\boldsymbol{J}}^{-1}\big(\check{\boldsymbol{F}}^{\mathrm{B}}(t) + \check{\boldsymbol{F}}^{\mathrm{B}}_{D}(t) - (\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}(t) + \check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}(t))\check{\times}\check{\boldsymbol{J}}(\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}(t) + \check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}(t)) - \check{\boldsymbol{J}}\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}(t)\check{\times}\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}(t) - \check{\boldsymbol{J}}(\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}(t)\check{\times}\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}(t)\check{\times}\boldsymbol{R}(t))\big) \\ &= \check{\boldsymbol{J}}^{-1}\left(\begin{pmatrix} \boldsymbol{F}^{\mathrm{B}} \\ 0 \\ \boldsymbol{r}_{ex}\times\boldsymbol{F}^{\mathrm{B}} \\ 0 \end{pmatrix} + \begin{pmatrix} m\boldsymbol{g}^{\mathrm{B}} \\ 0 \\ \boldsymbol{T}^{\mathrm{B}}_{GG} \\ 0 \end{pmatrix} - \left(\begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{B/A}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{B/A}} \end{pmatrix} + \begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{A/I}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{A/I}} \end{pmatrix}\right)\check{\times}\check{\boldsymbol{J}}\left(\begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{B/A}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{B/A}} \end{pmatrix} + \begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{A/I}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{A/I}} \end{pmatrix}\right) - \check{\boldsymbol{J}}\begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{A/I}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{A/I}} \end{pmatrix}\check{\times}\begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{B/A}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{B/A}} \end{pmatrix} - \check{\boldsymbol{J}}\begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{A/I}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{A/I}} \end{pmatrix}\check{\times}\begin{pmatrix} \boldsymbol{\omega}^{\mathrm{B}}_{\mathtt{A/I}} \\ \boldsymbol{v}^{\mathrm{B}}_{\mathtt{A/I}} \end{pmatrix}\check{\times}\boldsymbol{R}\right) \\ &= \check{\boldsymbol{J}}^{-1}\boldsymbol{W} \end{aligned} \tag{1.54}$$

We find the partial differentials below:

$$\check{\boldsymbol{J}}' = \frac{\partial \check{\boldsymbol{J}}}{\partial m}\bigg|_{\bar{\boldsymbol{x}}} = \left[\begin{array}{cc|cc} 0_{3\times3} & 0 & I_3 & 0 \\ 0 & 0 & 0 & 1 \\ \hline 0_{3\times3} & 0 & 0_{3\times3} & 0 \\ 0 & 1 & 0 & 0 \end{array}\right], \quad \check{\boldsymbol{J}}'' = \frac{\partial \check{\boldsymbol{J}}^{-1}}{\partial m}\bigg|_{\bar{\boldsymbol{x}}} = \tag{1.55}$$

$$\Theta_m = \frac{\partial \dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}}{\partial m}\bigg|_{\bar{\boldsymbol{x}}} = \check{\boldsymbol{J}}^{-1}\left(\begin{pmatrix} \nabla U^{\mathrm{B}} \\ 0 \\ \boldsymbol{T}^{\mathrm{B}}_{GG} \\ 0 \end{pmatrix} - (\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}} + \check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}})\check{\times}\check{\boldsymbol{J}}'(\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}} + \check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}) - \check{\boldsymbol{J}}'\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}\check{\times}\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}} - \check{\boldsymbol{J}}'(\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}\check{\times}\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}\check{\times}\boldsymbol{R})\right) + \check{\boldsymbol{J}}''\boldsymbol{W} \tag{1.56}$$

$$\Theta_{\check{q}} = \frac{\partial \dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}}{\partial \check{\boldsymbol{q}}_{\mathtt{B/A}}}\bigg|_{\bar{\boldsymbol{x}}} = \check{\boldsymbol{J}}^{-1} \tag{1.57}$$

$$\Theta_{\check{\omega}} = \frac{\partial \dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}}{\partial \check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}}\bigg|_{\bar{\boldsymbol{x}}} = \check{\boldsymbol{J}}^{-1}\left(\left[\check{\boldsymbol{J}}(\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}\check{\times}\right] - \left[\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}\check{\times}\right]\check{\boldsymbol{J}} - \left[\check{\boldsymbol{J}}(\check{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{A/I}}\check{\times}\right]\right) \tag{1.58}$$

$$\Theta_{\check{F}} = \frac{\partial \dot{\check{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}}{\partial \check{\boldsymbol{F}}^{\mathrm{B}}}\bigg|_{\bar{\boldsymbol{x}}} = \check{\boldsymbol{J}}^{-1}\left[\begin{array}{cc|cc} I_3 & 0 & [\boldsymbol{r}_e\times]^{-1} & 0 \\ 0 & 0 & 0 & 0 \\ \hline [\boldsymbol{r}_e\times] & 0 & I_3 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right] \tag{1.59}$$

$$
\boldsymbol{A}(t) = \left[\begin{array}{ccccccccccccccccccccccccccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\alpha(\frac{\bar{\boldsymbol{F}}^{\mathrm{B}}(t)_1}{\|\bar{\boldsymbol{F}}^{\mathrm{B}}(t)\|}) & -\alpha(\frac{\bar{\boldsymbol{F}}^{\mathrm{B}}(t)_2}{\|\bar{\boldsymbol{F}}^{\mathrm{B}}(t)\|}) & -\alpha(\frac{\bar{\boldsymbol{F}}^{\mathrm{B}}(t)_3}{\|\bar{\boldsymbol{F}}^{\mathrm{B}}(t)\|}) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{11}} & \Phi_{\check{q}_{12}} & \Phi_{\check{q}_{13}} & \Phi_{\check{q}_{14}} & \Phi_{\check{q}_{15}} & \Phi_{\check{q}_{16}} & \Phi_{\check{q}_{17}} & \Phi_{\check{q}_{18}} & \Phi_{\check{\omega}_{11}} & \Phi_{\check{\omega}_{12}} & \Phi_{\check{\omega}_{13}} & \Phi_{\check{\omega}_{14}} & \Phi_{\check{\omega}_{15}} & \Phi_{\check{\omega}_{16}} & \Phi_{\check{\omega}_{17}} & \Phi_{\check{\omega}_{18}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{21}} & \Phi_{\check{q}_{22}} & \Phi_{\check{q}_{23}} & \Phi_{\check{q}_{24}} & \Phi_{\check{q}_{25}} & \Phi_{\check{q}_{26}} & \Phi_{\check{q}_{27}} & \Phi_{\check{q}_{28}} & \Phi_{\check{\omega}_{21}} & \Phi_{\check{\omega}_{22}} & \Phi_{\check{\omega}_{23}} & \Phi_{\check{\omega}_{24}} & \Phi_{\check{\omega}_{25}} & \Phi_{\check{\omega}_{26}} & \Phi_{\check{\omega}_{27}} & \Phi_{\check{\omega}_{28}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{31}} & \Phi_{\check{q}_{32}} & \Phi_{\check{q}_{33}} & \Phi_{\check{q}_{34}} & \Phi_{\check{q}_{35}} & \Phi_{\check{q}_{36}} & \Phi_{\check{q}_{37}} & \Phi_{\check{q}_{38}} & \Phi_{\check{\omega}_{31}} & \Phi_{\check{\omega}_{32}} & \Phi_{\check{\omega}_{33}} & \Phi_{\check{\omega}_{34}} & \Phi_{\check{\omega}_{35}} & \Phi_{\check{\omega}_{36}} & \Phi_{\check{\omega}_{37}} & \Phi_{\check{\omega}_{38}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{41}} & \Phi_{\check{q}_{42}} & \Phi_{\check{q}_{43}} & \Phi_{\check{q}_{44}} & \Phi_{\check{q}_{45}} & \Phi_{\check{q}_{46}} & \Phi_{\check{q}_{47}} & \Phi_{\check{q}_{48}} & \Phi_{\check{\omega}_{41}} & \Phi_{\check{\omega}_{42}} & \Phi_{\check{\omega}_{43}} & \Phi_{\check{\omega}_{44}} & \Phi_{\check{\omega}_{45}} & \Phi_{\check{\omega}_{46}} & \Phi_{\check{\omega}_{47}} & \Phi_{\check{\omega}_{48}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{51}} & \Phi_{\check{q}_{52}} & \Phi_{\check{q}_{53}} & \Phi_{\check{q}_{54}} & \Phi_{\check{q}_{55}} & \Phi_{\check{q}_{56}} & \Phi_{\check{q}_{57}} & \Phi_{\check{q}_{58}} & \Phi_{\check{\omega}_{51}} & \Phi_{\check{\omega}_{52}} & \Phi_{\check{\omega}_{53}} & \Phi_{\check{\omega}_{54}} & \Phi_{\check{\omega}_{55}} & \Phi_{\check{\omega}_{56}} & \Phi_{\check{\omega}_{57}} & \Phi_{\check{\omega}_{58}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{61}} & \Phi_{\check{q}_{62}} & \Phi_{\check{q}_{63}} & \Phi_{\check{q}_{64}} & \Phi_{\check{q}_{65}} & \Phi_{\check{q}_{66}} & \Phi_{\check{q}_{67}} & \Phi_{\check{q}_{68}} & \Phi_{\check{\omega}_{61}} & \Phi_{\check{\omega}_{62}} & \Phi_{\check{\omega}_{63}} & \Phi_{\check{\omega}_{64}} & \Phi_{\check{\omega}_{65}} & \Phi_{\check{\omega}_{66}} & \Phi_{\check{\omega}_{67}} & \Phi_{\check{\omega}_{68}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{71}} & \Phi_{\check{q}_{72}} & \Phi_{\check{q}_{73}} & \Phi_{\check{q}_{74}} & \Phi_{\check{q}_{75}} & \Phi_{\check{q}_{76}} & \Phi_{\check{q}_{77}} & \Phi_{\check{q}_{78}} & \Phi_{\check{\omega}_{71}} & \Phi_{\check{\omega}_{72}} & \Phi_{\check{\omega}_{73}} & \Phi_{\check{\omega}_{74}} & \Phi_{\check{\omega}_{75}} & \Phi_{\check{\omega}_{76}} & \Phi_{\check{\omega}_{77}} & \Phi_{\check{\omega}_{78}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_{\check{q}_{81}} & \Phi_{\check{q}_{82}} & \Phi_{\check{q}_{83}} & \Phi_{\check{q}_{84}} & \Phi_{\check{q}_{85}} & \Phi_{\check{q}_{86}} & \Phi_{\check{q}_{87}} & \Phi_{\check{q}_{88}} & \Phi_{\check{\omega}_{81}} & \Phi_{\check{\omega}_{82}} & \Phi_{\check{\omega}_{83}} & \Phi_{\check{\omega}_{84}} & \Phi_{\check{\omega}_{85}} & \Phi_{\check{\omega}_{86}} & \Phi_{\check{\omega}_{87}} & \Phi_{\check{\omega}_{88}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_1} & \Theta_{\check{q}_{11}} & \Theta_{\check{q}_{12}} & \Theta_{\check{q}_{13}} & \Theta_{\check{q}_{14}} & \Theta_{\check{q}_{15}} & \Theta_{\check{q}_{16}} & \Theta_{\check{q}_{17}} & \Theta_{\check{q}_{18}} & \Theta_{\check{\omega}_{11}} & \Theta_{\check{\omega}_{12}} & \Theta_{\check{\omega}_{13}} & \Theta_{\check{\omega}_{14}} & \Theta_{\check{\omega}_{15}} & \Theta_{\check{\omega}_{16}} & \Theta_{\check{\omega}_{17}} & \Theta_{\check{\omega}_{18}} & \Theta_{\check{F}_{11}} & \Theta_{\check{F}_{12}} & \Theta_{\check{F}_{13}} & \Theta_{\check{F}_{14}} & \Theta_{\check{F}_{15}} & \Theta_{\check{F}_{16}} & \Theta_{\check{F}_{17}} & \Theta_{\check{F}_{18}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_2} & \Theta_{\check{q}_{21}} & \Theta_{\check{q}_{22}} & \Theta_{\check{q}_{23}} & \Theta_{\check{q}_{24}} & \Theta_{\check{q}_{25}} & \Theta_{\check{q}_{26}} & \Theta_{\check{q}_{27}} & \Theta_{\check{q}_{28}} & \Theta_{\check{\omega}_{21}} & \Theta_{\check{\omega}_{22}} & \Theta_{\check{\omega}_{23}} & \Theta_{\check{\omega}_{24}} & \Theta_{\check{\omega}_{25}} & \Theta_{\check{\omega}_{26}} & \Theta_{\check{\omega}_{27}} & \Theta_{\check{\omega}_{28}} & \Theta_{\check{F}_{21}} & \Theta_{\check{F}_{22}} & \Theta_{\check{F}_{23}} & \Theta_{\check{F}_{24}} & \Theta_{\check{F}_{25}} & \Theta_{\check{F}_{26}} & \Theta_{\check{F}_{27}} & \Theta_{\check{F}_{28}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_3} & \Theta_{\check{q}_{31}} & \Theta_{\check{q}_{32}} & \Theta_{\check{q}_{33}} & \Theta_{\check{q}_{34}} & \Theta_{\check{q}_{35}} & \Theta_{\check{q}_{36}} & \Theta_{\check{q}_{37}} & \Theta_{\check{q}_{38}} & \Theta_{\check{\omega}_{31}} & \Theta_{\check{\omega}_{32}} & \Theta_{\check{\omega}_{33}} & \Theta_{\check{\omega}_{34}} & \Theta_{\check{\omega}_{35}} & \Theta_{\check{\omega}_{36}} & \Theta_{\check{\omega}_{37}} & \Theta_{\check{\omega}_{38}} & \Theta_{\check{F}_{31}} & \Theta_{\check{F}_{32}} & \Theta_{\check{F}_{33}} & \Theta_{\check{F}_{34}} & \Theta_{\check{F}_{35}} & \Theta_{\check{F}_{36}} & \Theta_{\check{F}_{37}} & \Theta_{\check{F}_{38}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_4} & \Theta_{\check{q}_{41}} & \Theta_{\check{q}_{42}} & \Theta_{\check{q}_{43}} & \Theta_{\check{q}_{44}} & \Theta_{\check{q}_{45}} & \Theta_{\check{q}_{46}} & \Theta_{\check{q}_{47}} & \Theta_{\check{q}_{48}} & \Theta_{\check{\omega}_{41}} & \Theta_{\check{\omega}_{42}} & \Theta_{\check{\omega}_{43}} & \Theta_{\check{\omega}_{44}} & \Theta_{\check{\omega}_{45}} & \Theta_{\check{\omega}_{46}} & \Theta_{\check{\omega}_{47}} & \Theta_{\check{\omega}_{48}} & \Theta_{\check{F}_{41}} & \Theta_{\check{F}_{42}} & \Theta_{\check{F}_{43}} & \Theta_{\check{F}_{44}} & \Theta_{\check{F}_{45}} & \Theta_{\check{F}_{46}} & \Theta_{\check{F}_{47}} & \Theta_{\check{F}_{48}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_5} & \Theta_{\check{q}_{51}} & \Theta_{\check{q}_{52}} & \Theta_{\check{q}_{53}} & \Theta_{\check{q}_{54}} & \Theta_{\check{q}_{55}} & \Theta_{\check{q}_{56}} & \Theta_{\check{q}_{57}} & \Theta_{\check{q}_{58}} & \Theta_{\check{\omega}_{51}} & \Theta_{\check{\omega}_{52}} & \Theta_{\check{\omega}_{53}} & \Theta_{\check{\omega}_{54}} & \Theta_{\check{\omega}_{55}} & \Theta_{\check{\omega}_{56}} & \Theta_{\check{\omega}_{57}} & \Theta_{\check{\omega}_{58}} & \Theta_{\check{F}_{51}} & \Theta_{\check{F}_{52}} & \Theta_{\check{F}_{53}} & \Theta_{\check{F}_{54}} & \Theta_{\check{F}_{55}} & \Theta_{\check{F}_{56}} & \Theta_{\check{F}_{57}} & \Theta_{\check{F}_{58}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_6} & \Theta_{\check{q}_{61}} & \Theta_{\check{q}_{62}} & \Theta_{\check{q}_{63}} & \Theta_{\check{q}_{64}} & \Theta_{\check{q}_{65}} & \Theta_{\check{q}_{66}} & \Theta_{\check{q}_{67}} & \Theta_{\check{q}_{68}} & \Theta_{\check{\omega}_{61}} & \Theta_{\check{\omega}_{62}} & \Theta_{\check{\omega}_{63}} & \Theta_{\check{\omega}_{64}} & \Theta_{\check{\omega}_{65}} & \Theta_{\check{\omega}_{66}} & \Theta_{\check{\omega}_{67}} & \Theta_{\check{\omega}_{68}} & \Theta_{\check{F}_{61}} & \Theta_{\check{F}_{62}} & \Theta_{\check{F}_{63}} & \Theta_{\check{F}_{64}} & \Theta_{\check{F}_{65}} & \Theta_{\check{F}_{66}} & \Theta_{\check{F}_{67}} & \Theta_{\check{F}_{68}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_7} & \Theta_{\check{q}_{71}} & \Theta_{\check{q}_{72}} & \Theta_{\check{q}_{73}} & \Theta_{\check{q}_{74}} & \Theta_{\check{q}_{75}} & \Theta_{\check{q}_{76}} & \Theta_{\check{q}_{77}} & \Theta_{\check{q}_{78}} & \Theta_{\check{\omega}_{71}} & \Theta_{\check{\omega}_{72}} & \Theta_{\check{\omega}_{73}} & \Theta_{\check{\omega}_{74}} & \Theta_{\check{\omega}_{75}} & \Theta_{\check{\omega}_{76}} & \Theta_{\check{\omega}_{77}} & \Theta_{\check{\omega}_{78}} & \Theta_{\check{F}_{71}} & \Theta_{\check{F}_{72}} & \Theta_{\check{F}_{73}} & \Theta_{\check{F}_{74}} & \Theta_{\check{F}_{75}} & \Theta_{\check{F}_{76}} & \Theta_{\check{F}_{77}} & \Theta_{\check{F}_{78}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\Theta_{m_8} & \Theta_{\check{q}_{81}} & \Theta_{\check{q}_{82}} & \Theta_{\check{q}_{83}} & \Theta_{\check{q}_{84}} & \Theta_{\check{q}_{85}} & \Theta_{\check{q}_{86}} & \Theta_{\check{q}_{87}} & \Theta_{\check{q}_{88}} & \Theta_{\check{\omega}_{81}} & \Theta_{\check{\omega}_{82}} & \Theta_{\check{\omega}_{83}} & \Theta_{\check{\omega}_{84}} & \Theta_{\check{\omega}_{85}} & \Theta_{\check{\omega}_{86}} & \Theta_{\check{\omega}_{87}} & \Theta_{\check{\omega}_{88}} & \Theta_{\check{F}_{81}} & \Theta_{\check{F}_{82}} & \Theta_{\check{F}_{83}} & \Theta_{\check{F}_{84}} & \Theta_{\check{F}_{85}} & \Theta_{\check{F}_{86}} & \Theta_{\check{F}_{87}} & \Theta_{\check{F}_{88}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right] \tag{1.60}
$$

$$\boldsymbol{B}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{z}(t) = \begin{pmatrix} \dot{\bar{m}}(t) \\ \dot{\bar{\boldsymbol{q}}}_{\mathtt{B/A}}(t) \\ \dot{\bar{\boldsymbol{q}}}_{d_{\mathtt{B/A}}}(t) \\ \dot{\bar{\boldsymbol{\omega}}}^{\mathrm{B}}_{\mathtt{B/A}}(t) \\ 0 \\ \dot{\bar{\boldsymbol{v}}}^{\mathrm{B}}_{\mathtt{B/A}}(t) \\ 0 \\ \dot{\bar{\boldsymbol{F}}}^{\mathrm{B}}(t) \\ 0 \\ \dot{\bar{\boldsymbol{T}}}^{\mathrm{B}}(t) \\ 0 \\ \ddot{\bar{\boldsymbol{F}}}^{\mathrm{B}}(t) \\ 0 \\ \ddot{\bar{\boldsymbol{T}}}^{\mathrm{B}}(t) \\ 0 \end{pmatrix} - \boldsymbol{A}(t) \begin{pmatrix} \bar{m}(t) \\ \bar{\boldsymbol{q}}_{\mathtt{B/A}}(t) \\ \bar{\boldsymbol{q}}_{d_{\mathtt{B/A}}}(t) \\ \bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathtt{B/A}}(t) \\ 0 \\ \bar{\boldsymbol{v}}^{\mathrm{B}}_{\mathtt{B/A}}(t) \\ 0 \\ \bar{\boldsymbol{F}}^{\mathrm{B}}(t) \\ 0 \\ \bar{\boldsymbol{T}}^{\mathrm{B}}(t) \\ 0 \\ \ddot{\bar{\boldsymbol{F}}}^{\mathrm{B}}(t) \\ 0 \\ \ddot{\bar{\boldsymbol{T}}}^{\mathrm{B}}(t) \\ 0 \end{pmatrix} - \boldsymbol{B}(t) \begin{pmatrix} \ddot{\bar{\boldsymbol{F}}}^{\mathrm{B}}(t) \\ 0 \\ \ddot{\bar{\boldsymbol{T}}}^{\mathrm{B}}(t) \\ 0 \end{pmatrix} \tag{1.61}$$

1.4.2. Virtual Control

Linearisation of the dynamics can lead to the introduction of infeasibility in the solution space even if the original nonlinear problem is feasible. This situation is called *artificial infeasibility*. This undesirable infeasiblity prevents convergence as it obstructs the iteration process. Mao, Szmuk, and Acıkmeşe 2016 suggest the use of a *virtual control*, $\boldsymbol{v}(t)$ that works as an addition to the actual control input to make the infeasible region infeasible. An unconstrained virtual control ensures that any state in the feasible region of the original problem is reachable in finite time when just the control input $\boldsymbol{u}(t)$ is not enough to do so. The virtual control can be inferred as a an artificial or synthetic acceleration that acts on the SC, to steer it to any state in the feasible region. The linearised dynamics therefore becomes

$$\dot{\boldsymbol{d}}(t) = \boldsymbol{A}(t)\boldsymbol{d}(t) + \boldsymbol{B}(t)\boldsymbol{e}(t) + \boldsymbol{D}(t)\boldsymbol{v}(t) + \boldsymbol{C}(t) \tag{1.62}$$

where, $\boldsymbol{D}(t)$ can be chosen based on $\boldsymbol{A}(t)$ such that both of them $(\boldsymbol{A}(.),\ \boldsymbol{D}(.))$ are controllable.

The virtual control is unconstrained, which means there is no restriction on its need, which needs to be taken care of to ensure that the control is still within the real bounds and can be achieved by the SC subject to the original nonlinear constraints. This is done by *penalising* the cost function with an additional term of the form

$$\gamma(.) := \operatorname*{ess\,sup}_{t\in[0,t_f]} \|.(t)\|_1 \tag{1.63}$$

where, $\|.\|_1$ is the first norm on $\mathbb{R}^n$ given by $\|c(t)\|_1 = \sum_{i=1}^{n} |c(t)|$. The term used for penalising is $\boldsymbol{Ev}$ and hence the penalty function becomes,

$$\gamma(\boldsymbol{Dv}) := \operatorname*{ess\,sup}_{t\in[0,t_f]} \|\boldsymbol{D}(t)\boldsymbol{v}(t)\|_1 \tag{1.64}$$

The new penalised cost function after linearising therefore becomes,

$$\text{minimise} - z(t_f) + w_v\gamma(\boldsymbol{Dv}) \tag{1.65}$$

where, w_v is the penalty weight for the virtual control.

1.4.3. Trust Regions

Linearising nonlinear functions leads to the concern of another problem arising, which is that it might render the problem unbounded. When a large deviation is allowed to occur from the linearising point, the linear approximation may not be capable of capturing the nonlinearity. To understand this we take simple example problem,—.

Trust region as the name suggests, helps define the region within which the linear approximation can be trusted to capture the nonlinearity. This ensures that the linearised trajectory does not diverge from the nominal trajectory attained in the previous succession by a significant amount. The trust region needs to be co

$$\delta x(t)^T \delta x(t) + \delta u(t)^T \delta u(t) \leq \eta(t) \tag{1.66}$$

The new penalised cost function after linearising therefore becomes,

$$\text{minimise} - z(t_f) + w_v\gamma(Ev) + w_\eta\gamma(\eta) \tag{1.67}$$

where, w_η is the penalty weight for the trust region.

1.5. Discretisation

The differential of the state represents continuous time dynamics and this needs to be converted to finite dimensional parameter optimisation problem. The trajectory from initial to final state can therefore be discretised to K points in the total time. The final time, t_f can therefore be divided in K nodes and the time at each node is indexed by $k \in \mathbb{N}$ and can be given as:

$$t_k \triangleq \left(\frac{k}{K-1}\right)t_f, \quad k \in [0, K) \tag{1.68}$$

The initial time can be $t_0 = 0\ s$ without any loss of generality since the problem is not explicitly time dependent.

1.5.1. Control Parametrisation

Parametrisation of the control vector is needed to map the control variable in finite time for the solver to give optimal controls to be used in continuous time by the actual system. This can be done via a zero-order (ZOH) or first-order hold on the control. Other methods include the use of Chebyshev polynomials but to keep it simple we approach the problem with ZOH. The continuous time control can be represented as a linear combination of basis functions, ϑ_k:

$$\boldsymbol{u}(t) = \sum_{k=1}^{K} \boldsymbol{u}_k \, \vartheta_k(t) \tag{1.69}$$

where, $\boldsymbol{u}_k \triangleq \boldsymbol{u}(t_k) \; \forall \; k \in [0, K)$ are the optimal controls given by the solver. For ZOH, the basis functions are defined as:

$$\vartheta_k(t) = \begin{cases} 1 & \text{when } t = [t_{k-1}, t_k] \\ 0 & \text{otherwise} \end{cases} \tag{1.70}$$

The above definition means that the basis functions are active ($\vartheta = 1$) from t_{k-1} to t_k but not including t_k. This keeps the optimal control, u_k during the interval and deactivates it at the next time step. Since the descent problem is a minimum furl problem, a bang-bang control profile is expected and hence ZOH is suitable for the algorithm. However since ZOH is a stepping function, errors are introduced since the discontinuities are not physically possible to be implemented by the actual system. Figure **??** represents how the control output looks like with respect to time. The definition for FOH is given as

$$\vartheta_k(t) = \begin{cases} \dfrac{t_{k+1} - t}{t_{k+1} - t_k} & \text{when } t = [t_{k-1}, t_k] \\ \dfrac{t - t_k}{t_{k+1} - t_k} & \text{when } t = [t_k, t_{k+1}] \\ 0 & \text{otherwise} \end{cases} \tag{1.71}$$

The pattern of FOH is shown in Figure **??**.

1.6. Extended Convex Guidance

The powered descent guidance algorithm can prove to be challenging because it usually treats the translational and rotational dynamics of the SC separately. This makes it difficult to deal with situations where both these dynamics need to be combined for some mission requirement. Certain constraints like the one on the line of sight of the SC require the SC to keep the target in sight while descent which make it dependent on both the orientation and position of the SC. In this section we will parameterise all such constraints that need to be met while the powered descent phase is in action. Finally in the problem formulation, these constraints will be converted to DQ format.

1.6.1. Need for Extended Constraints

We have the following requirements for the safety of the SC:

- The SC needs to avoid hazards like boulders or rough or highly sloped terrains near the landing site or those that may have not been observed during mapping.

- The SC needs to follow a trajectory such that no part of the SC comes in contact with the target surface before end of flight.

- The SC needs to maintain a certain altitude at the end of flight, so that just the sample collector hardware is in contact with the surface which also requires the enforcing of no subterranean flight to prevent any kind of damage to the hardware or SC.

- Also depending in the resolution of the mapping images, the landing site may need to be changed by HDA and retargeting to a nearby safe site is required. This can lead to drastic changes in the trajectory and there should be a constraint for robustness of the optimiser performance.

1.6.2. Glide-Slope Constraint

The guidance problem for the powered descent is a TPBVP, which means it finds an optimal trajectory from the initial state of the SC to a required final state or landing site. The control constraints discussed in the previous section take care of the feasibility of the trajectory from the point of view of the propulsion system. But we need to implement certain path constraints since this is a TAG descent.

In lieu with all the above requirements, the *glide-slope constraint* can be established, which enforces on the optimiser software a safe reachable space for the SC. The glide slope is nothing but a cone about the normal from the surface of the landing site with its vertex at the landing site location on the surface, which guides the trajectory of the SC to ensure no contact with the surface before the end of flight. In other words this means that the elevation angle of the SC with respect to the plane of the surface must always be greater than the glide-slope angle, θ. Geometrically this is represented in the Figure **??** and mathematically can be represented as

$$\boldsymbol{r}^{\mathtt{A}}_{\mathtt{B/A}} \cdot \boldsymbol{z}_{\mathtt{A}} \geq \|\boldsymbol{r}^{\mathtt{A}}_{\mathtt{B/A}}\| \cos\phi \quad \forall t \in [t_0, t_f] \tag{1.72}$$

This takes care of the second and third requirements mentioned above. Now with the HDA system in the loop, retargetings might me required on the detection of hazards. One such trigger could be the presence of an unobserved boulder[6] near the landing site. This can be taken care of by simply increasing the glide-slope angle by the amount required as shown in Figure **??** and resolving the optimisation problem. The same treatment is implemented for a surface with increasing slope close to the landing site.

$$\phi_h = \arctan\left(\frac{h + \delta h}{\sqrt{x_b^2 + y_b^2}}\right) \tag{1.73}$$

The inability of this constraint to safeguard the SC arises in situations where the HDA software detects one or more hazards at the landing the site for example rocky terrain or an undetected crater (could be the rim or slope or pit). In this case a new trajectory to a nearby safe landing site needs to be solved for. This will be discussed in detailed in Section 8.8, where we discuss the implementation of the logic of optimised landing on a new safe site.

Implementation

As discussed in the earlier sections the constraint needs to be converted to SOCP form

1.6.3. Attitude Constraints

The rotational manoeuvres of the SC need to be restricted depending on the requirement of the onboard instruments. For altitude measurements for example via RADAR the attitude of the SC should be within a certain angle with respect to the normal to the surface being observed for accurate measurements as well as within a certain angle with respect to the boresight of the RADAR or optical camera due to sensor hardware limitations. Now, the constraint for accurate measurements is an angle about the surface normal(ground fixed frame) whereas the LOS constraint is about the body fixed frame. The visual representation of both these constraints is shown in the Figure **??**.

An important constraint on the attitude comes from safety of the SC which intends to prevent the upside-down flight of the SC at all times. But this will already be taken care of by the combination of LOS and accurate measurement constraints.

Implementation

1.6.4. Thrust Direction Constraint

Thrust pointing/direction constraint as the name suggests is the constraint in the body fixed frame for the direction of thrust from the propulsion system. The importance of this constraint is that it couples the feasibility of the physical trajectory to the SC hardware capabilities. propulsion system. In case of a rigidly fixed thruster system to the SC body, this constraint is the same as the attitude constraint since the direction of the commanded thrust will determine the attitude. For the mission scenario of TAG descent, a gimballed thruster would prove to be more beneficial. This is because the SC has a sample collector in the extended position whilst thrusting to successfully hover over the surface in the final phase of descent which would require the thruster to avoid disturbing the surface from where the sample needs to be collected.

[6] Mapping will detect the tall boulders, only boulders of small height need to be considered. Implementing the presence of tall boulders would make the safe glide slope angle too large and make the landing scenario infeasible.

If the SC is treated as a point mass and the net thrust is represented by a single thrust vector, then the net thrust on the SC could be pointed arbitrarily. With a rigid body SC, an arbitrary net thrust is not possible since the propulsion thruster cannot have any possible orientation with respect to the SC structure. The restrictions come from the SC design as well as the safety concern for onboard instruments. This can translated to the thrust direction constraint where the net thrust vector would lie within a cone about the gimbal axis of the propulsion system.

1.6.5. Stacking Equations

1.7. Algorithm

A

Mathematical Properties

Matrices

Quaternions

$\boldsymbol{a},\ \boldsymbol{b}$ & $\boldsymbol{c}$ are quaternions and $\gamma \in \mathbb{R}$

$$\begin{aligned}
\boldsymbol{a} \otimes (\boldsymbol{b} + \boldsymbol{c}) &= \boldsymbol{a} \otimes \boldsymbol{b} + \boldsymbol{a} \otimes \boldsymbol{c} \\
(\boldsymbol{a} \otimes \boldsymbol{b})^* &= \boldsymbol{b}^* \otimes \boldsymbol{a}^* \\
(\gamma \boldsymbol{a}) \otimes \boldsymbol{b} &= \boldsymbol{a} \otimes \gamma \boldsymbol{b} + \gamma (\boldsymbol{a} \otimes \boldsymbol{b}) \\
\boldsymbol{a} \otimes (\boldsymbol{b} \otimes \boldsymbol{c}) &= (\boldsymbol{a} \otimes \boldsymbol{b}) \otimes \boldsymbol{c} \\
\boldsymbol{a}^T (\boldsymbol{b} \otimes \boldsymbol{c}) &= \boldsymbol{c}^T (\boldsymbol{b}^* \otimes \boldsymbol{a}) = \boldsymbol{b}^T (\boldsymbol{a} \otimes \boldsymbol{c}^*)
\end{aligned}$$

Unit Quaternion Triple Identity

$$(\boldsymbol{t} \otimes \boldsymbol{q})^T (\boldsymbol{y} \otimes \boldsymbol{q}) = (\boldsymbol{q} \otimes \boldsymbol{t})^T (\boldsymbol{q} \otimes \boldsymbol{y}) = \boldsymbol{t}^T \boldsymbol{y} = \boldsymbol{y}^T \boldsymbol{t}$$

B

Verification using the Comet 67P Mapping

C

Verification using a 6-DOF Mars Descent

Szmuk, Utku, and Açıkmeşe 2017 have used successive convexification for the non-convex minimum-fuel OCP for 6 DOF rocket powered Mars landing Scenario. We use this case to verify our SCvx algorithm. It is a fixed final time problem and in the inertial and body reference frames. The vehicle has a single engine and is capable of commanding various thrust magnitudes at different engine gimbal angles. The gravity, COM and inertia are assumed to remain constant. The state vector and its differential is given by:

$$\boldsymbol{x}(t) = \begin{pmatrix} m(t) \\ \boldsymbol{r}^{\mathtt{I}}(t) \\ \boldsymbol{v}^{\mathtt{I}}(t) \\ \boldsymbol{q}_{\mathtt{B/I}}(t) \\ \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t) \\ \boldsymbol{T}^{\mathtt{B}}(t) \\ \dot{\boldsymbol{T}}^{\mathtt{B}}(t) \end{pmatrix} \quad \dot{\boldsymbol{x}}(t) = \begin{pmatrix} -\alpha \|\boldsymbol{T}^{\mathtt{B}}(t)\| \\ \boldsymbol{v}^{\mathtt{I}}(t) \\ \boldsymbol{a}^{\mathtt{I}}(t) \\ \dot{\boldsymbol{q}}_{\mathtt{B/I}}(t) \\ \dot{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}}(t) \\ \dot{\boldsymbol{T}}^{\mathtt{B}}(t) \\ \boldsymbol{u}(t) \end{pmatrix} = \begin{pmatrix} -\alpha \|\boldsymbol{T}^{\mathtt{B}}(t)\| \\ \boldsymbol{v}^{\mathtt{I}}(t) \\ \frac{1}{m(t)} \boldsymbol{C}_{\mathtt{B/I}} \boldsymbol{T}^{\mathtt{B}}(t) + \boldsymbol{g}^{\mathtt{I}} \\ \frac{1}{2} \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t) \otimes \boldsymbol{q}_{\mathtt{B/I}}(t) \\ J^{-1} [\boldsymbol{r}^{\mathtt{B}}_{e} \times] \boldsymbol{T}^{\mathtt{B}}(t) - [\boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t) \times] J \boldsymbol{\omega}^{\mathtt{B}}_{\mathtt{B/I}}(t)) \\ \dot{\boldsymbol{T}}^{\mathtt{B}}(t) \\ \boldsymbol{u}(t) \end{pmatrix} \tag{C.1}$$

The non-linear dynamics are linearised by Taylor series to get the form as discussed in Eq. 7.42 to 7.46. The linearised equations are given below:

$$\begin{aligned} f_{\boldsymbol{T}^{\mathtt{B}},t} &= \left(\|\bar{\boldsymbol{T}}^{\mathtt{B}}(t)\| + \frac{\bar{\boldsymbol{T}}^{\mathtt{B}}(t)}{\|\bar{\boldsymbol{T}}^{\mathtt{B}}(t)\|} (\boldsymbol{T}^{\mathtt{B}}(t) - \bar{\boldsymbol{T}}^{\mathtt{B}}(t) \right) \\ f_{\boldsymbol{a}^{\mathtt{I}},t} &= \bar{\boldsymbol{a}}^{\mathtt{I}}(t) + \frac{\partial \boldsymbol{a}^{\mathtt{I}}}{\partial \boldsymbol{x}} \bigg|_{\bar{\boldsymbol{x}}(t)} (\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)) \\ f_{\dot{\boldsymbol{q}}_{\mathtt{B/I}},t} &= \dot{\bar{\boldsymbol{q}}}_{\mathtt{B/I}}(t) + \frac{\partial \dot{\boldsymbol{q}}_{\mathtt{B/I}}}{\partial \boldsymbol{x}} \bigg|_{\bar{\boldsymbol{x}}(t)} (\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)) \\ f_{\dot{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}},t} &= \dot{\bar{\boldsymbol{\omega}}}^{\mathtt{B}}_{\mathtt{B/I}}(t) + \frac{\partial \dot{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}}}{\partial \boldsymbol{x}} \bigg|_{\bar{\boldsymbol{x}}(t)} (\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t)) \end{aligned} \tag{C.2}$$

Now we will derive the linearisation matrices for continuous time, for which we first derive the partial differentials with respect to state and controls.

Translational Acceleration:

$$\begin{aligned} \boldsymbol{a}^{\mathtt{I}}(t) &= \frac{1}{m(t)} \boldsymbol{C}_{\mathtt{B/I}}(t) \boldsymbol{T}^{\mathtt{B}}(t) + \boldsymbol{g}^{\mathtt{I}} \\ &= \frac{1}{m(t)} \Big[\boldsymbol{q}^{*}_{\mathtt{B/I}}(t) \otimes \boldsymbol{T}^{\mathtt{B}}(t) \otimes \Big] \boldsymbol{q}_{\mathtt{B/I}}(t) + \boldsymbol{g}^{\mathtt{I}} \\ &= \frac{1}{m(t)} \Big[\boldsymbol{T}^{\mathtt{B}}(t) \otimes \boldsymbol{q}_{\mathtt{B/I}}(t) \odot \Big] \boldsymbol{q}^{*}_{\mathtt{B/I}}(t) + \boldsymbol{g}^{\mathtt{I}} \end{aligned} \tag{C.3}$$

We find the partial differentials below:

$$\begin{aligned}
\Psi_m &= \left.\frac{\partial \boldsymbol{a}^{\mathrm{I}}}{\partial m}\right|_{\bar{\boldsymbol{x}}} = \frac{1}{\bar{m}^2(t)}\bar{\boldsymbol{C}}_{\mathrm{B/I}}(t)\bar{\boldsymbol{T}}^{\mathrm{B}} \\
\Psi_q &= \left.\frac{\partial \boldsymbol{a}^{\mathrm{I}}}{\partial \boldsymbol{q}_{\mathrm{B/I}}}\right|_{\bar{\boldsymbol{x}}} = \frac{1}{\bar{m}(t)}\left[\bar{\boldsymbol{q}}^*_{\mathrm{B/I}}(t)\otimes\bar{\boldsymbol{T}}^{\mathrm{B}}(t)\otimes\right] + \frac{1}{m(t)}\left[\bar{\boldsymbol{T}}^{\mathrm{B}}(t)\otimes\bar{\boldsymbol{q}}_{\mathrm{B/I}}(t)\odot\right](-1\ -1\ -1\ 1)^{\mathrm{T}} \\
\Psi_T &= \left.\frac{\partial \boldsymbol{a}^{\mathrm{I}}}{\partial \boldsymbol{T}^{\mathrm{B}}}\right|_{\bar{\boldsymbol{x}}} = \frac{1}{\bar{m}(t)}\bar{\boldsymbol{C}}_{\mathrm{B/I}}(t)
\end{aligned} \tag{C.4}$$

Attitude Kinematics:

$$\begin{aligned}
\dot{\boldsymbol{q}}_{\mathrm{B/I}}(t) &= \frac{1}{2}\left[\boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}(t)\otimes\right]\boldsymbol{q}_{\mathrm{B/I}}(t) \\
&= \frac{1}{2}\left[\boldsymbol{q}_{\mathrm{B/I}}(t)\odot\right]\begin{pmatrix}\boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}(t)\\ 0\end{pmatrix}
\end{aligned} \tag{C.5}$$

We find the partial differentials below:

$$\begin{aligned}
\Upsilon_q &= \left.\frac{\partial \dot{\boldsymbol{q}}_{\mathrm{B/I}}}{\partial \boldsymbol{q}_{\mathrm{B/I}}}\right|_{\bar{\boldsymbol{x}}} = \frac{1}{2}\left[\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t)\otimes\right] \\
\Upsilon_\omega &= \left.\frac{\partial \dot{\boldsymbol{q}}_{\mathrm{B/I}}}{\partial \boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}}\right|_{\bar{\boldsymbol{x}}} = \frac{1}{2}\left[\bar{\boldsymbol{q}}_{\mathrm{B/I}}(t)\odot\right]
\end{aligned} \tag{C.6}$$

Attitude Dynamics:

$$\begin{aligned}
\dot{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t) &= J^{-1}\Big([\boldsymbol{r}^{\mathrm{B}}_e\times]\boldsymbol{T}^{\mathrm{B}}(t) - \left[\boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}(t)\times\right]J\boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}(t)\Big) \\
&= J^{-1}\Big([\boldsymbol{r}^{\mathrm{B}}_e\times]\boldsymbol{T}^{\mathrm{B}}(t) + \left[J\boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}(t)\times\right]\boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}(t)\Big)
\end{aligned} \tag{C.7}$$

We find the partial differentials below:

$$\begin{aligned}
\Theta_\omega &= \left.\frac{\partial \dot{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}}{\partial \boldsymbol{\omega}^{\mathrm{B}}_{\mathrm{B/I}}}\right|_{\bar{\boldsymbol{x}}} = J^{-1}\Big(-\left[\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t)\times\right]J + \left[J\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t))\times\right]\Big) \\
\Theta_T &= \left.\frac{\partial \dot{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}}{\partial \boldsymbol{T}^{\mathrm{B}}}\right|_{\bar{\boldsymbol{x}}} = J^{-1}[\boldsymbol{r}^{\mathrm{B}}_e\times]
\end{aligned} \tag{C.8}$$

The compiled $\boldsymbol{A}(t)$ matrix is given on the next page along with the control matrix. Since the state differential has controls as the last vector, it can be directly computed as given in $\boldsymbol{B}(t)$. And for $\boldsymbol{z}(t)$, the continuous time matrix is as follows:

$$\boldsymbol{z}(t) = \begin{pmatrix} -\alpha\|\bar{\boldsymbol{T}}^{\mathrm{B}}(t)\| \\ \bar{\boldsymbol{v}}^{\mathrm{I}}(t) \\ \frac{1}{\bar{m}(t)}\bar{\boldsymbol{C}}_{\mathrm{B/I}}\bar{\boldsymbol{T}}^{\mathrm{B}}(t) + \boldsymbol{g}^{\mathrm{I}} \\ \frac{1}{2}\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t)\otimes\bar{\boldsymbol{q}}_{\mathrm{B/I}}(t) \\ J^{-1}[\boldsymbol{r}^{\mathrm{B}}_e\times]\bar{\boldsymbol{T}}^{\mathrm{B}}(t) - [\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t)\times]J\bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t)) \\ \dot{\bar{\boldsymbol{T}}}^{\mathrm{B}}(t) \\ \bar{\boldsymbol{u}}(t) \end{pmatrix} - \boldsymbol{A}(t)\begin{pmatrix} \bar{m}(t) \\ \bar{\boldsymbol{r}}^{\mathrm{I}}(t) \\ \bar{\boldsymbol{v}}^{\mathrm{I}}(t) \\ \bar{\boldsymbol{q}}_{\mathrm{B/I}}(t) \\ \bar{\boldsymbol{\omega}}^{\mathrm{B}}_{\mathrm{B/I}}(t) \\ \bar{\boldsymbol{T}}^{\mathrm{B}}(t) \\ \dot{\bar{\boldsymbol{T}}}^{\mathrm{B}}(t) \end{pmatrix} - \boldsymbol{B}(t)\bar{\boldsymbol{u}}(t) \tag{C.9}$$

From TPBVP problem formulation, we have the initial and final required state values which give us the $\boldsymbol{x}_0$ and $\boldsymbol{x}_f$ for a closed time, t_f problem. The in between states and state differential solutions can be found by discretisation. For discretisation the total time is divided into K nodes and the solutions at these points is calculated as per the Eq.7. to 7. The discretisation is given in page 8.

$$\boldsymbol{A}(t)=\begin{bmatrix}
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & -\alpha(\frac{\bar{\boldsymbol{T}}^{\mathtt{B}}(t)_1}{\|\bar{\boldsymbol{T}}^{\mathtt{B}}(t)\|}) & -\alpha(\frac{\bar{\boldsymbol{T}}^{\mathtt{B}}(t)_2}{\|\bar{\boldsymbol{T}}^{\mathtt{B}}(t)\|}) & -\alpha(\frac{\bar{\boldsymbol{T}}^{\mathtt{B}}(t)_3}{\|\bar{\boldsymbol{T}}^{\mathtt{B}}(t)\|}) & 0&0&0\\
0 & 0&0&0&1&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&1&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&1 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&0\\
\frac{1}{\bar{m}^2}(\bar{\boldsymbol{C}}_{\mathtt{B/I}}(t)\bar{\boldsymbol{T}}^{\mathtt{B}})_1 & 0&0&0&0&0&0 & \Psi_{q_{11}}&\Psi_{q_{12}}&\Psi_{q_{13}}&\Psi_{q_{14}} & 0&0&0 & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{11}} & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{12}} & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{13}} & 0&0&0\\
\frac{1}{\bar{m}^2}(\bar{\boldsymbol{C}}_{\mathtt{B/I}}(t)\bar{\boldsymbol{T}}^{\mathtt{B}})_2 & 0&0&0&0&0&0 & \Psi_{q_{21}}&\Psi_{q_{22}}&\Psi_{q_{23}}&\Psi_{q_{24}} & 0&0&0 & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{21}} & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{22}} & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{23}} & 0&0&0\\
\frac{1}{\bar{m}^2}(\bar{\boldsymbol{C}}_{\mathtt{B/I}}(t)\bar{\boldsymbol{T}}^{\mathtt{B}})_3 & 0&0&0&0&0&0 & \Psi_{q_{31}}&\Psi_{q_{32}}&\Psi_{q_{33}}&\Psi_{q_{34}} & 0&0&0 & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{31}} & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{32}} & \frac{1}{\bar{m}}\bar{\boldsymbol{C}}_{\mathtt{B/I}_{33}} & 0&0&0\\
0 & 0&0&0&0&0&0 & 0 & \bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_3} & -\bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_2} & \bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_1} & \bar{\boldsymbol{q}}_{\mathtt{B/I}_4} & -\bar{\boldsymbol{q}}_{\mathtt{B/I}_3} & \bar{\boldsymbol{q}}_{\mathtt{B/I}_2} & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&0 & -\bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_3} & 0 & \bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_1} & \bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_2} & \bar{\boldsymbol{q}}_{\mathtt{B/I}_3} & \bar{\boldsymbol{q}}_{\mathtt{B/I}_4} & -\bar{\boldsymbol{q}}_{\mathtt{B/I}_1} & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&0 & \bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_2} & -\bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_1} & 0 & \bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_3} & -\bar{\boldsymbol{q}}_{\mathtt{B/I}_2} & \bar{\boldsymbol{q}}_{\mathtt{B/I}_1} & \bar{\boldsymbol{q}}_{\mathtt{B/I}_4} & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&0 & -\bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_1} & -\bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_2} & -\bar{\boldsymbol{\omega}}^{\mathtt{B}}_{\mathtt{B/I}_3} & 0 & -\bar{\boldsymbol{q}}_{\mathtt{B/I}_1} & -\bar{\boldsymbol{q}}_{\mathtt{B/I}_2} & -\bar{\boldsymbol{q}}_{\mathtt{B/I}_3} & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & \Theta_{\omega_{11}} & \Theta_{\omega_{12}} & \Theta_{\omega_{13}} & \Theta_{T_{11}} & \Theta_{T_{12}} & \Theta_{T_{13}} & 0&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & \Theta_{\omega_{21}} & \Theta_{\omega_{22}} & \Theta_{\omega_{23}} & \Theta_{T_{21}} & \Theta_{T_{22}} & \Theta_{T_{23}} & 0&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & \Theta_{\omega_{31}} & \Theta_{\omega_{32}} & \Theta_{\omega_{33}} & \Theta_{T_{31}} & \Theta_{T_{32}} & \Theta_{T_{33}} & 0&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 1&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&1&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&1\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&0\\
0 & 0&0&0&0&0&0 & 0&0&0&0 & 0&0&0 & 0&0&0 & 0&0&0
\end{bmatrix},\quad \boldsymbol{B}(t)=\begin{bmatrix}
0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\0&0&0\\1&0&0\\0&1&0\\0&0&1
\end{bmatrix} \tag{C.10}$$

for $k \in [0, K]$

$$
\begin{aligned}
\bar{m}_k &= \left(\frac{K-k}{K}\right) m_{wet} + \left(\frac{k}{K}\right) m_{dry} \\
\bar{\boldsymbol{r}}^{\mathtt{I}}_k &= \left(\frac{k-k}{K}\right) \boldsymbol{r}^{\mathtt{I}}_0 + \left(\frac{k}{K}\right) \boldsymbol{r}^{\mathtt{I}}_f \\
\bar{\boldsymbol{v}}^{\mathtt{I}}_k &= \left(\frac{K-k}{K}\right) \boldsymbol{v}^{\mathtt{I}}_0 + \left(\frac{k}{K}\right) \boldsymbol{v}^{\mathtt{I}}_f \\
\bar{\boldsymbol{q}}_{\mathtt{B/I}_k} &= [0\ 0\ 0\ 1]^{\mathtt{T}} \\
\bar{\boldsymbol{T}}^{\mathtt{B}}_k &= \bar{m}_k \boldsymbol{g} \boldsymbol{e}_{u,\mathtt{I}}
\end{aligned}
\tag{C.11}
$$

Bibliography

[1] Açıkmeşe, B., Carson, J. M., and Blackmore, L. “Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem”. *IEEE Transactions on Control Systems Technology* Vol. 21.No. 6 (2013), pp. 2104–2113. DOI: 10.1109/TCST.2012.2237346.

[2] Açıkmeşe, B. and Ploen, S. R. “Convex programming approach to powered descent guidance for Mars landing”. *Journal of Guidance, Control and Dynamics* Vol. 30.No. 5 (2007), pp. 1353–1366. DOI: 10.1109/TCST.2012.2237346.

[3] Boyd, S. and Vandenberghe, L. *Convex Optimization.* Vol. 25. Cambridge University Press, 2010. DOI: 10.1080/10556781003625177.

[4] Mao, Y., Szmuk, M., and Acıkmeşe, B. “Successive convexification of non-convex optimal control problems and its convergence properties”. *2016 IEEE 55th Conference on Decision and Control, CDC 2016* (2016), pp. 3636–3641. DOI: 10.1109/CDC.2016.7798816.

[5] Razgus, B. “Relative Navigation in Asteroid Missions: Dual Quaternion Approach”. MA thesis. Technical University of Delft, 2017.

[6] Surovik, D. A. and Scheeres, D. J. “Autonomous Maneuver Planning at Small Bodies via Mission Objective Reachability Analysis”. *AIAA/AAS Astrodynamics Specialist Conference* (2014), pp. 1–12. DOI: 10.2514/6.2014-4147.

[7] Szmuk, M. and Açıkmeşe, B. “Successive Convexification for Mars 6-DoF Rocket Powered Landing with Free-Final-Time”. *AIAA Guidance, Navigation, and Control Conference* (2018).

[8] Szmuk, M., Utku, E., and Açıkmeşe, B. “Successive Convexification for Mars 6-DoF Powered Descent Landing Guidance”. *AIAA Guidance, Navigation, and Control Conference.* (2017).

[9] Xinfu, L. and Ping, L. “Solving Nonconvex Optimal Control Problems by Convex Optimization”. *Journal of Guidance, Control, and Dynamics* Vol. 37.No. 3 (2014).