**Pose Tracking Control for Spacecraft Proximity Operations
Using the Udwadia-Kalaba Framework**

by
Abin Alex Pothan

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs
in partial fulfillment of the requirements for the degree of

**Master of Applied Science
in
Aerospace Engineering**

Carleton University
Ottawa, Ontario

© 2019
Abin Alex Pothan

Dedicated to the revealer of mysteries unknown.

Abstract

This thesis develops an analytical dynamics-based approach for simultaneous position and orientation tracking control of a chaser spacecraft with respect to an uncontrolled target spacecraft. The control requirements are formulated as holonomic or non-holonomic constraints, which are differentiated to obtain a constraint equation linear in acceleration. Exact real-time control forces are then generated by substituting the control constraints into the Udwadia-Kalaba equation. Three major contributions are presented. Firstly, the complete six-degree-of-freedom formulation of the Udwadia-Kalaba based pose tracking controller is presented. Simulations demonstrate the achievement of the desired objectives in space. Subsequently, a planar pose tracking controller is formulated for both a single and dual chaser configuration. Simulation results highlight the planar position and orientation synchronization with respect to a spinning target. Finally, the controller is experimentally validated in the Spacecraft Proximity Operations Testbed at Carleton University. Results show that the pose tracking control objective is achieved.

Acknowledgements

My deepest gratitude goes to my Prof. Steve Ulrich who, through his experience and focused guidance helped me navigate the complexities of space research. He was generous and dedicated a part of the NSERC Discovery Grant given to the Spacecraft Robotics and Control Laboratory to support my thesis work. He also guided me carefully in places where I got stuck. I would also like to thank Prof. Ashwini Ratnoo from the Indian Institute of Science, who got me interested in the area of guidance and control. He inculcated in me the patience to attack problems and defend my position through the medium of articulated writing. With respect to the experiments performed at SRCL, Alexander Crain's help was noteworthy. In spite of his busy schedule working at the National Research Council, he helped me to run my control algorithm in the lab simulators and to perform the initial set of lab experiments. I would also like to thank Kirk Hovell, who helped fix the issues that came up while running the experiments. Some insightful comments from Prof. Bruce Burlton during the orbital mechanics course presentation also helped me address practical aspects related to control in space. Others who helped me with the work include Bibin Alex Pothen, Jeff Hough, Samuel Lortie, and Yazan Chihabi.

I also take this opportunity to acknowledge the constant support from my fiancée, Nimmy Prasad, who is my personal cheerleader. Her understanding and unwavering commitment is an anchor to my soul. Both my parents and Nimmy's parents have been encouraging me to push my intellectual boundaries while reminding me of the necessity to maintain a healthy lifestyle. I will be indebted to Prem Anand, a structural engineer at NovAerial Robotics and an alumni of Carleton, who recommended me for part-time teaching job to handle the additional expenses incurred during my studies. I would also like to thank my part-time employer Caroline Sohn, the Director of EyeLevel Learning Center of Barrhaven, who helped me get work schedules based on the level of my research and teaching commitments at Carleton University.

Table of Contents

Abstract	iii
Acknowledgements	iv
List of Tables	ix
List of Figures	x
Chapter 1 Introduction	1
1.1 Motivation	2
1.2 Problem Statement	2
1.3 Previous Work	4
1.4 Thesis Objectives	7
1.5 Contributions	7
1.6 Thesis Organization	8
Chapter 2 The Udwadia-Kalaba Framework	9
2.1 Unconstrained Generalized Acceleration	9
2.2 Acceleration Constraint	10
2.3 Gauss's Principle of Least Constraint	13
2.4 The Udwadia-Kalaba Equation	15
2.5 The Moore-Penrose Generalized Inverse	15
2.6 Udwadia-Kalaba Equation and the Gauss's Principle	16
2.6.1 Udwadia-Kalaba Equation Satisfies the Constraint Equation .	17
2.6.2 Udwadia-Kalaba Equation Minimizes the Gaussian	18
2.7 System Constraints	19
2.7.1 Modeling Constraints	20
2.7.2 Control Constraints	21

Chapter 3	Dynamic Model	24
3.1	Frames of Reference	24
3.1.1	ECI frame	25
3.1.2	LVLH Frame	26
3.1.3	Body-fixed Frame	27
3.2	Rotation Matrix	28
3.2.1	Euler-Angle Representation	29
3.2.2	Quaternion Representation	30
3.3	6 DOF Dynamics	32
3.3.1	Determination of Unconstrained Acceleration	32
3.3.2	Orbital Motion	33
3.3.3	Quaternion Tumbling Dynamics	36
3.3.4	Summary of the 6-DOF Dynamics	41
Chapter 4	Spatial Stand-off Pose Tracking	43
4.1	Spatial Problem Formulation	43
4.2	Position Control Constraint	44
4.3	Orientation Control Constraint	51
4.4	Generalized Control Force Generation	56
4.5	Simulation Results	57
4.5.1	Case 1: Pose Tracking in a Purely Inertial Frame	59
4.5.2	Case 2: Pose Tracking on J_2 -Perturbed Orbits	63
Chapter 5	3-DOF Pose Tracking	71
5.1	A Case for Planar Analysis	71
5.2	Planar Formulation of the Problem	72
5.3	Single Chaser Formation Constraint	74
5.4	Control Force Determination	77
5.4.1	Double Integrator Dynamics for Planar Case	78
5.5	Single Chaser Pose Tracking Control Simulation	79
5.6	Dual-Chaser Formation Constraint	82

5.7	Control Force and Dual-Chaser-Target Dynamics	85
5.8	Dual-chaser pose tracking Simulation	85
Chapter 6	Planar Experimental Validation	89
6.1	Introduction	89
6.2	The Spacecraft Proximity Operations Testbed	89
6.2.1	Experiment Phases	91
6.2.2	Spacecraft Platforms	92
6.2.3	Sources of error	93
6.3	Case 1: Pose Tracking Around a Spinning Target	93
6.4	Case 2: Pose Tracking Around a Translating and Spinning Target . .	99
6.5	Case 3: Docking with a Translating and Spinning Target	103
6.6	Summary	109
Chapter 7	Conclusion and Future Work	110
7.1	Thesis Summary	110
7.2	A Note of Practicality	112
7.3	Recommendations for Future Work	112
7.4	Significance of Work	112
Appendix A	Details of Derivations	114
A.1	Proof for Theorem in Chapter 2	114
A.2	Orbital Elements to the Inertial Position and Velocity	117
Appendix B	MATLAB Scripts	118
B.1	Codes for Orbit Rendezvous with J_2 Perturbation	118
B.1.1	init_orbit_rendezvous.m	118
B.1.2	orbit_rendezvous.slx	122
B.1.3	chaser_controller block	123
B.1.4	target_orbit_motion block	126
B.1.5	chaser_orbit_motion block	126

B.1.6	target_dynamics block	127
B.1.7	chaser_dynamics block	128
Bibliography		130

List of Tables

4.1	Target Mass Properties	58
4.2	Chaser Mass Properties	58
4.3	Target Orbital Elements	63
4.4	Chaser Orbital Elements	64
4.5	Chaser Orbital Elements	69

List of Figures

3.1	ECI frame of reference	25
3.2	LVLH frame with respect to the target center of mass	26
3.3	Body-fixed frame of reference	27
3.4	High-level block diagram showing signals and processes	42
4.1	Chaser and target frame in the inertial ECI frame.	43
4.2	Desired position configuration of the chaser.	45
4.3	Desired orientation of the chaser.	52
4.4	Spatial pose tracking simulation showing the trajectory of the tip of the alignment and point of interest vectors.	60
4.5	Case 1: Chaser and target trajectory.	61
4.6	Results for pose tracking in a purely inertial frame	62
4.7	Constraint errors for pose tracking in a purely inertial frame	63
4.8	Case 2: Chaser and target trajectory for one period in orbit.	65
4.9	Case 2: Chaser and target trajectory for five periods in orbit.	65
4.10	Results for pose tracking on J_2 -perturbed orbits.	66
4.11	Constraint errors for pose tracking on J_2 -perturbed orbits.	67
4.12	Results for on-orbit pose tracking with lower control gains	68
4.13	Constraint errors for on-orbit pose tracking with lower control gains	68
4.14	Results for on-orbit pose tracking closer to the target.	70
4.15	Constraint errors for on-orbit pose tracking closer to the target.	70
5.1	Planar problem formulation	73
5.2	Desired pose tracking configuration for single chaser.	74
5.3	Single chaser and target pose tracking trajectory.	80
5.4	Time history of chaser 1 and target states	80
5.5	Control force in single chaser pose tracking.	81

5.6	Angular velocity matching and stand-off separation maintenance in single chaser pose tracking	81
5.7	Desired pose tracking configuration for dual-chaser system. . .	82
5.8	Chaser and target trajectory.	86
5.9	Evolution of Chaser 1, Chaser 2 and target states	86
5.10	Chaser 1 and Chaser 2 control force.	87
5.11	Simultaneous pose tracking using dual chasers	87
6.1	Top view of SPOT along with the spacecraft platforms	90
6.2	The chaser (red) and target (black) spacecraft platform	92
6.3	Experiment on pose tracking around a spinning target	94
6.4	Chaser and target spacecraft trajectory.	95
6.5	Position time history.	96
6.6	Relative separation between chaser and target.	97
6.7	Orientation time history.	97
6.8	Angular velocity.	98
6.9	Control force.	98
6.10	Experiment on pose tracking around a translating and spinning target	99
6.11	Case 2: Chaser and target trajectory.	100
6.12	Case 2: Chaser and target position time history.	101
6.13	Case 2: Stand-off separation.	101
6.14	Case 2: Angular displacement time history.	102
6.15	Case 2: Angular velocity.	103
6.16	Case 2: Control force.	103
6.17	Experiment on docking with a translating and spinning target	104
6.18	Case 3: Chaser and target trajectory.	105
6.19	Case 3: Position time history.	106
6.20	Case 3: Stand-off distance.	107
6.21	Case 3: Angular displacement time history.	107

6.22	Case 3: Angular velocity.	108
6.23	Case 3: Control force.	108
B.1	MATLAB Simulink Block	122

Chapter 1

Introduction

While space activity over the past sixty years has given us greater insight on terrestrial and celestial phenomena, it has filled the lower earth orbits with satellites that are both functional and non-functional. As of January 2018, the European Space Agency estimates¹ that out of the 8650 satellites launched since 1957, about 4,700 are still in orbit, out of which only 1,800 are still functional. Though the number of non-active debris in space may seem harmless as compared to the vast space surrounding earth, the reality paints a grim picture. From the start of the space age, there has been more than 500 break-ups, collisions, explosions, and other unknown events that have resulted in the development of a large number of space debris orbiting earth, out of which around 21,000 are currently being tracked. Using statistical estimates, the number of debris with dimension higher than 10 cm has been determined to be greater than 29,000. The objects with dimension between 1 – 10 cm is around 750,000 while the smaller debris (< 1 cm) whose orbits are difficult to track with high accuracy is around 166 million.

Frequent evasive maneuvers are used to protect the International Space Station (ISS) and other active satellites from potential collisions with such debris. Further, the spacecraft that operates in this region carries Whipple shields² that protect from such small debris traveling at a relative speed of 3 – 18 km/s. A combination of Whipple shield, Nextel/Kevlar stuffed Whipple shield, and multi-shock shield [1] is also used to protect the ISS from such hypervelocity impact with debris of size less than 1 cm. While the impact from smaller debris can be minimized using shields, the large debris collisions can be quite catastrophic. Such collisions can also create thousands of smaller debris that are difficult to track and capture.

¹www.esa.int/Our_Activities/Operations/Space_Debris/Space_debris_by_the_numbers

²https://en.wikipedia.org/wiki/Whipple_shield

1.1 Motivation

In the year 1978, Kessler et al. [2] applied the model used for studying the evolution of the asteroid belt to predict the growth of debris surrounding the earth orbit. Based on their analysis, it was concluded that the presence of defunct satellites that have run out of fuel, rocket motors and debris from fragmentation would result in a non-zero collision probability and an exponential growth of debris surrounding the earth orbit, even if new launches are halted. With major catastrophic events like the anti-satellite missile test of Fengyun 1C by People’s Republic of China in 2007 [3] and the collision of Iridium-33 and Cosmos-2251 satellites in 2009 [4], growth of debris surrounding the earth is a concern. As the collisions become more frequent, it would become riskier and costlier to mitigate the growth of debris. Recently, the European Space Agency reported that the number of objects in lower earth orbit has reached a critical concentration that calls for remediation measures through active debris removal (ADR)³. While a lot of effort has been going on in terms of achieving ADR, the practical application of the methods have been limited due to technical, economic, political and legal challenges [5]. The need to achieve ADR, however, is more important than ever with the increasing rocket launches to cater to the growing demand for enabling communication, weather prediction, etc. Using NASA’s long-term debris evolutionary model, it has been estimated that removal of large space debris at the rate of 5 per year with an operational success rate of 90% is required by the year 2020 to maintain the current low-earth satellite concentration, so that regular rocket launches can continue [6].

1.2 Problem Statement

In the ADR technique, a chaser spacecraft is used to rendezvous and capture a tumbling debris. The debris is then removed by either pulling it down to a lower earth orbit to decay in the atmosphere or by pushing it into a higher graveyard orbit. As an interesting recent development, the European Space Agency’s e.Deorbit mission

³www.esa.int/Our_Activities/Operations/Space_Debris/Active_debris_removal

has refocused its objectives on refueling, refurbishing and reboosting defunct satellites in orbit so that they can be made active again to mitigate the debris problem⁴. Typically, an ADR mission involves a five-phase process of chaser launch, far-range rendezvous followed by a close-range rendezvous, debris capture and removal [7]. In the close-range approach phase, careful control of the relative position and velocity of the chaser spacecraft with respect to the tumbling object must be achieved. This phase can also be adapted for on-orbit servicing or refueling of hundreds of fuel-depleted satellites that are currently in orbit as would be attempted in the e.Deorbit mission in 2023. For the post-capture phase of detumbling, use of tethered spacecraft systems is being considered as a promising option due to the tether flexibility, low weight and ability to handle varying debris size. Misra et al. [8] presented a double-tether system that presented better rotational characteristics as compared to the single tether system. In the recent work by Kirk and Ulrich [9, 10], a method to quickly damp out the relative velocity between the chaser and tumbling object was investigated using a visco-elastic tethers in sub-tether configuration.

In all of these methods, however, it is assumed that the chaser is precisely aligned before the de-tumbling phase is initiated. In order to enable the practical implementation of the above techniques, it is first required to simultaneously and accurately track both the relative translational motion and rotational motion with respect to a target spacecraft. This task is typically referred to as pose tracking or six-degree-of-freedom (6-DOF) motion tracking control problem. Note that the six degrees of freedom originate from the set of three independent positions and orientations that completely define the configuration of a rigid body in space.

⁴http://www.esa.int/Our_Activities/Space_Engineering_Technology/Clean_Space/ESA_s_e.Deorbit_debris_removal_mission_reborn_as_servicing_vehicle

1.3 Previous Work

As mentioned in the previous section, the key to enable autonomous proximity operations is the capability to successfully achieve a proper synchronization of the translational and rotational motion through pose tracking maneuvers. However, the real-time guidance and control of a chaser spacecraft to synchronize its translational and angular motion in a safe and efficient manner so as to capture a target with no relative angular motion at the specified stand-off distance remains a challenging task. This is mostly due to the fact that the majority of the techniques developed for the real-time motion planning with a moving target assume a point mass target [11–18].

Motion synchronization with a drifting and tumbling target is a more difficult task due to the complex dynamics of rotating rigid bodies in space. Specifically, because of the low-frequency coupling between the translational and rotational motion, notable multivariable strategies were previously developed to address this problem.

Traditionally, in operational scenarios, this cross-coupling term is fed as a feed-forward term which is combined to either a linear proportional-integral-derivative (PID) or a robust H_∞ feedback loop [19]. Ma et al. [20] formulated and analyzed the rendezvous control problem of a chaser spacecraft for optimal approach to and alignment with a rotating target. An iterative algorithm for a planar case was proposed to solve the optimal trajectory for minimum time/fuel consumption. However, the iterative nature of the proposed solution drove up its computational cost. An experimental study of close-range rendezvous and docking for a cooperative target was presented by Romano et al. [21]. The target spacecraft was, however, assumed to be active and attitude stabilized using its own thrusters, while the chaser docks on to it. This is similar to the case of the docking of a crewed spacecraft or supply capsule to the International Space Station. Artificial potential functions were used by Zhang et al. [22] in rendezvous and docking guidance to capture a non-cooperative target in the presence of static and dynamic obstacles. The method, however, suffers from the well-known local minima issue associated with potential functions closer to the obstacles. The Canadian Space Agency (CSA) developed a combined prediction

and motion-planning scheme for docking with a drifting and tumbling target [23]. The problem of finding the optimal rendezvous point to null out the relative motion was formulated as solving the system Hamiltonian function for the terminal time by using a Newton-Raphson numerical technique. Unfortunately, the numerical solution of the nonlinear optimization problem prevents the autonomous, real-time, application of this technique on a flight computer. More recently, Wilde et al. [24] experimentally demonstrated an inverse dynamics based trajectory optimization for planar docking of a chaser onto an uncontrolled spinning object. As a limitation, this object was assumed to be spinning along a single axis without translating. Ventura et al. [25] presented a spatial inverse-dynamics based approach for docking onto a tumbling object where the inversion is done in a virtual domain rather than a time domain for the iterative generation of optimal trajectories. Using the current state estimates, the approach trajectory is calculated and the control is updated. Zagaris and Romano [26] proposed a reachability analysis for planar docking onto a spinning target. The translational-only control problem of rendezvous and docking was solved by Dong et al. [27] by encoding the approach and obstacle avoidance constraints into the artificial potential function and sliding mode scheme, following which an adaptive controller was implemented. A full 6-DOF control of both the translational and rotational motion is, however, necessary to achieve precise motion tracking in space.

Using Euler-Lagrange approach, Kristiansen et al. [28] compared the passivity based PD+ and sliding mode control along with the integrator back-stepping approach for 6-DOF pose tracking control. The theoretical comparison and simulation results were presented that showed uniform asymptotic convergence of the controllers around equilibrium points. The work, however, described the position and attitude separately, which led to complex control design and calculations. Sun and Huo [29] proposed an integrated adaptive back-stepping approach that was robust to modeling uncertainties for spacecraft rendezvous by considering the dynamic coupling between the relative translational and rotational motion. A non-linear adaptive pose tracking control law based on the immersion and invariance approach was presented by Sun et al. [30] that implemented the position and attitude control separately. The method,

however, addressed the case of a cooperative target and chaser spacecraft. Terminal sliding mode control was applied for finite time convergence in a distributed attitude control scheme by Zou et al. [31]. Note that the distributed spacecraft were assumed to have full control authority. Further, Wang et al. [32, 33] proposed a sliding mode control strategy that addresses 6-DOF pose tracking control with finite time convergence characteristics that was applied to a spacecraft formation flying problem. In this work, dual-quaternions that integrate the position and attitude information were used to describe the equations of motion. The dual-quaternions have the same form as rotation-only equations of motion expressed in quaternions. Filipe et al. [34] proposed an adaptive robust pose tracking controller based on the 6-DOF model for spacecraft rendezvous and docking using dual-quaternions. Incorporating the field-of-view constraint so as to point the local sensors of the chaser spacecraft to the target, Lee et al. [35] used dual-quaternion based dynamics to obtain a special type of model predictive controller that estimates the future system behavior based on current and previous information. This approach assumed a principal axis spin of the target and hence, addressed only the simplified pose-tracking mechanics. Furthermore, the convex optimization step in the controller was deemed to be computationally heavy. A lower computational effort was obtained for the dual-quaternion based 6-DOF pose tracking approach by Dong et al. [36] that used a modified artificial potential function to achieve the pose-tracking objective while simultaneously addressing the approach path constraint and the field-of-view constraint. From a theoretical standpoint, the local minima issue still persists on account of using potential functions, but the solutions suggested a conditional switching of control gains at such a local trap to allow the trajectory to escape from it. By estimating the velocity using a dual-quaternion based filter, velocity-free controllers were obtained for pose tracking [37, 38]. In more recent approaches, fault-tolerant control [39, 40] and control in the presence of parameter uncertainties [41] under similar mathematical background was presented. While the representation of dynamics using dual quaternions provide advantages in terms of reusing the attitude-only controllers to perform the dual-objective of simultaneous position and orientation control, the underlying control scheme is, however, based on the conventional approaches.

In this context, it is desired to have a simple control methodology that can provide the exact closed-form solution to the problem of simultaneous position and orientation tracking of a passive 6-DOF target undergoing a general tumbling motion. The proposed method should also provide real-time commands that can efficiently maneuver the chaser so that a zero relative motion is achieved with respect to a specific point of interest on the target. A recent advancement in analytical dynamics called the *Udwadia-Kalaba equation* [42] is able to provide exact force commands to maintain a specified system constraint. By enforcing the pose tracking objective as a control constraint on the states of the chaser spacecraft, the desired generalized control forces can be generated in real time.

1.4 Thesis Objectives

This work presents a fundamental contribution to the emerging field of stand-off pose tracking of an uncontrolled object. Specifically, a new close-range pose tracking method that uses recent advancements in analytical dynamics to simultaneously control the linear and rotational motion to precisely position and align the chaser spacecraft with respect to a tumbling object is presented. The Udwadia-Kalaba equation provides an elegant solution for the constrained acceleration, derived from the holonomic and non-holonomic Pfaffian constraints, that can be utilized to generate the exact real-time force commands. The control scheme uses the idea that control is a constraint on the states of a system to construct a second order constraint error dynamics equation that ensures that the states settle onto the desired control constraint surface while obeying the set performance characteristics [43, 44].

1.5 Contributions

The following unique contributions are achieved in this thesis –

- 6 DOF equations for generating exact real-time force commands for spatial pose tracking control around an uncontrolled target is developed using the Udwadia-Kalaba framework and simulations highlight the achievement of the proposed

characteristics.

- Pose tracking control of a spinning object in the planar case using a single and double chaser spacecraft is derived using the Udwadia-Kalaba equation and extensive simulation studies are presented.
- Experimental validation of the uncontrolled object capture using a single chaser in the planar case is demonstrated. By incorporating a varying stand-off distance in the Udwadia-Kalaba equation-based control method, a planar docking is also validated in the experiments.

1.6 Thesis Organization

The remainder of this work is organized as follows: Chapter 2 reviews the Udwadia-Kalaba formulation and describes determination of control force in the presence of modeling and control constraints. The dynamic model used for describing the motion of chaser spacecraft and the tumbling target is presented in Chapter 3. The exact control force formulation for the 6-DOF spatial pose tracking along with simulation results are subsequently presented in Chapter 4. Further, the planar pose tracking problem is solved using the Udwadia-Kalaba equation-based formulation and simulations are shown in Chapter 5. Chapter 6 describes the experimental validation of the Udwadia-Kalaba equation-based stand-off pose tracking control in the planar case. Finally, the work ends with the conclusion and a discussion of potential future work in Chapter 7.

Chapter 2

The Udwadia-Kalaba Framework

This chapter reviews the theoretical framework required for the development of the proposed solution to address the problem of stand-off pose tracking in space. Specifically, a recent advancement in analytical dynamics, the *Udwadia-Kalaba* equation [42] and its application in solving constrained motion is discussed. A recent work by Koganti and Udwadia [44] on real-time control of such complex constrained systems with asymptotically diminishing error dynamics using the UKE framework is also reviewed. The purpose of this section is to build the strong theoretical foundation upon which the reader would find comfort in understanding the Udwadia-Kalaba equation-based stand-off pose tracking formulation. For more detail, the reader is referred to Refs. [42] and [44].

2.1 Unconstrained Generalized Acceleration

Consider a set of point masses $m_1, m_2, \dots, m_n$ with their corresponding position vectors at $\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n$ in the three-dimensional inertial space. Suppose the given forces on the respective point masses are $\mathbf{F}_1, \mathbf{F}_2, \dots, \mathbf{F}_n$, then according to the Newton's law,

$$\mathbf{M}_i \ddot{\mathbf{r}}_i = \mathbf{F}_i, \quad \text{for } i = 1, 2, \dots, n \quad (2.1)$$

where $\mathbf{M}_i$ is a positive definite diagonal matrix with m_i along the diagonals.

Alternatively, the generalized configuration of n particles can be represented as

$$\mathbf{q} = \begin{bmatrix} \mathbf{r}_1^T & \mathbf{r}_2^T & \cdots & \mathbf{r}_n^T \end{bmatrix}^T \quad (2.2)$$

Note that $\mathbf{q}$ is a $3n \times 1$ matrix and hence, the generalized configuration is $3n$ -dimensional. Further, the generalized mass matrix is given as follows

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{0}_{3 \times 3} & \cdots & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{M}_2 & \cdots & \mathbf{0}_{3 \times 3} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \cdots & \mathbf{M}_n \end{bmatrix} \quad (2.3)$$

By composing the individual forces $\mathbf{F}_i$ into the generalized force vector,

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_1^T & \mathbf{F}_2^T & \cdots & \mathbf{F}_n^T \end{bmatrix}^T \quad (2.4)$$

Newton's law of Eq. (2.1) can then be rewritten as

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{F} \quad (2.5)$$

In Eq. (2.5), $\ddot{\mathbf{q}}$ is the generalized acceleration obtained by double-differentiating the generalized configuration vector $\mathbf{q}$ of Eq. (2.2). In the absence of any constraints, the unconstrained acceleration, $\ddot{\mathbf{q}}_u$ is given as

$$\ddot{\mathbf{q}}_u = \mathbf{M}^{-1}\mathbf{F} \quad (2.6)$$

Note that the inverse of the generalized mass matrix $\mathbf{M}$ exists as it is square and positive definite.

2.2 Acceleration Constraint

A common set of restrictions on the states that arise in the context of mechanics is categorized into holonomic and non-holonomic constraints. Though more categories exist, these two classes encompass most of the real-world constraints. A holonomic

constraint is one where the constraint function, ϕ depends on the generalized configuration and time as shown in Eq. (2.7).

$$\phi(\mathbf{q}, t) = 0 \quad (\text{holonomic constraint}) \quad (2.7)$$

Note that a constraint function of the form in Eq. (2.7) reduces the degree of freedom of the originally unconstrained system by one. By considering an infinitesimal change, Eq. (2.7) can be written as

$$\sum_{i=1}^{3n} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \delta q_i + \frac{\partial \phi(\mathbf{q}, t)}{\partial t} \delta t = 0 \quad (2.8)$$

This is known as the *Pfaffian form* of the holonomic constraint and can be integrated to obtain Eq. (2.7). By dividing throughout by δt , an alternate form of Eq. (2.8) is obtained as

$$\sum_{i=1}^{3n} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \dot{q}_i + \frac{\partial \phi(\mathbf{q}, t)}{\partial t} = 0 \quad (2.9)$$

This means that a restriction in the generalized coordinates also constrain the generalized velocities as seen from Eq. (2.9), such that Eq. (2.7) is always satisfied. When the constraint of Eq. (2.8) is non-integrable, it is called a non-holonomic Pfaffian constraint, and it is given by

$$\sum_{i=1}^{3n} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \delta q_i + \frac{\partial \phi(\mathbf{q}, t)}{\partial t} \delta t = 0 \quad (\text{non-integrable}) \quad (2.10)$$

Dividing throughout by δt , the same form as Eq. (2.9) is obtained, which is dependent on $\mathbf{q}$, $\dot{\mathbf{q}}$ and time. This can then be denoted as

$$\psi(\mathbf{q}, \dot{\mathbf{q}}, t) = 0 \quad (\text{non-holonomic constraint}) \quad (2.11)$$

Whether holonomic or non-holonomic, a constraint function can be differentiated to obtain a form that is linear in generalized acceleration. Specifically, differentiating

the holonomic constraint of Eq. (2.7) twice with respect to time yields

$$\begin{aligned} & \sum_{i=1}^{3n} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \ddot{q}_i + \sum_{i=1}^{3n} \sum_{j=1}^{3n} \frac{\partial}{\partial q_j} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \right] \dot{q}_j \dot{q}_i + \sum_{i=1}^{3n} \frac{\partial}{\partial t} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \right] \dot{q}_i \\ & + \sum_{i=1}^{3n} \frac{\partial}{\partial q_i} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial t} \right] \dot{q}_i + \frac{\partial^2 \phi(\mathbf{q}, t)}{\partial t^2} = 0 \end{aligned} \quad (2.12)$$

$$\begin{aligned} \Rightarrow & \sum_{i=1}^{3n} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \ddot{q}_i = - \sum_{i=1}^{3n} \sum_{j=1}^{3n} \frac{\partial}{\partial q_j} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \right] \dot{q}_j \dot{q}_i - \sum_{i=1}^{3n} \frac{\partial}{\partial t} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \right] \dot{q}_i \\ & - \sum_{i=1}^{3n} \frac{\partial}{\partial q_i} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial t} \right] \dot{q}_i - \frac{\partial^2 \phi(\mathbf{q}, t)}{\partial t^2} \\ \Rightarrow & \begin{bmatrix} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_1} & \frac{\partial \phi(\mathbf{q}, t)}{\partial q_2} & \dots & \frac{\partial \phi(\mathbf{q}, t)}{\partial q_{3n}} \end{bmatrix} \ddot{\mathbf{q}} = b_h \end{aligned} \quad (2.13)$$

where

$$b_h = - \sum_{i=1}^{3n} \sum_{j=1}^{3n} \frac{\partial}{\partial q_j} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \right] \dot{q}_j \dot{q}_i - \sum_{i=1}^{3n} \frac{\partial}{\partial t} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial q_i} \right] \dot{q}_i - \sum_{i=1}^{3n} \frac{\partial}{\partial q_i} \left[\frac{\partial \phi(\mathbf{q}, t)}{\partial t} \right] \dot{q}_i - \frac{\partial^2 \phi(\mathbf{q}, t)}{\partial t^2}$$

By setting matrix $\mathbf{A}_h$ as

$$\mathbf{A}_h = \begin{bmatrix} \frac{\partial \phi(\mathbf{q}, t)}{\partial q_1} & \frac{\partial \phi(\mathbf{q}, t)}{\partial q_2} & \dots & \frac{\partial \phi(\mathbf{q}, t)}{\partial q_n} \end{bmatrix} \quad (2.14)$$

the holonomic acceleration constraint of Eq. (2.13) in the matrix form becomes

$$\mathbf{A}_h \ddot{\mathbf{q}} = b_h \quad (\text{holonomic acceleration constraint}) \quad (2.15)$$

Similarly, by differentiating the non-holonomic constraint of Eq. (2.11) once with respect to time, a constraint linear in generalized acceleration is obtained as

$$\sum_{i=1}^{3n} \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q_i} \dot{q}_i + \sum_{i=1}^{3n} \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{q}_i} \ddot{q}_i + \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial t} = 0 \quad (2.16)$$

$$\Rightarrow \sum_{i=1}^{3n} \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{q}_i} \ddot{q}_i = - \sum_{i=1}^{3n} \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q_i} \dot{q}_i - \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial t} \quad (2.17)$$

An alternate matrix form of Eq. (2.17) can be written by defining $\mathbf{A}_{nh}$ and b_{nh} as

$$\mathbf{A}_{nh} = \begin{bmatrix} \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{q}_1} & \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{q}_2} & \dots & \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{q}_{3n}} \end{bmatrix} \quad (2.18)$$

$$b_{nh} = - \sum_{i=1}^{3n} \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial q_i} \dot{q}_i - \frac{\partial \psi(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial t} \quad (2.19)$$

Substituting Eq. (2.18) and Eq. (2.19) into Eq. (2.17), the non-holonomic acceleration constraint can be denoted by

$$\mathbf{A}_{nh} \ddot{\mathbf{q}} = b_{nh} \quad (\text{non-holonomic acceleration constraint}) \quad (2.20)$$

Notice from Eq. (2.15) and Eq. (2.20) that whether holonomic or non-holonomic, the i^{th} constraint can be represented using the same form $\mathbf{A}_i \ddot{\mathbf{q}} = b_i$ where $\mathbf{A}_i$ is an $1 \times n$ matrix and b_i is a scalar.

Now, let there be a set of m constraints that act upon the particle system. Then, the constraint matrix in acceleration can be denoted by

$$\mathbf{A} \ddot{\mathbf{q}} = \mathbf{b} \quad (2.21)$$

where $\mathbf{A}$ is an $m \times n$ matrix and $\mathbf{b}$ is an $m \times 1$ matrix given by

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_m \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \quad (2.22)$$

2.3 Gauss's Principle of Least Constraint

Under the effect of constraints, the actual generalized acceleration, $\ddot{\mathbf{q}}$ deviates from the unconstrained acceleration, $\ddot{\mathbf{q}}_u$ of Eq. (2.6). Gauss's Principle [45] states that of

all the possible set of accelerations that satisfy the constraint equation (2.21) at a given time t , the one that materializes minimize the Gaussian, G given by

$$G = (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)^T \mathbf{M} (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u) \quad (2.23)$$

Note that the Gaussian is the normalization of $\Delta \mathbf{q} = \ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u$ with respect to the generalized mass matrix, $\mathbf{M}$.

Since $\mathbf{M}$ is a diagonal matrix with non-negative elements, it can also be written as $\mathbf{M} = \mathbf{M}^{\frac{1}{2}} \mathbf{M}^{\frac{1}{2}}$ where

$$\mathbf{M}^{\frac{1}{2}} = \begin{bmatrix} m_1^{\frac{1}{2}} & 0 & 0 & 0 & \cdots & 0 \\ 0 & m_1^{\frac{1}{2}} & 0 & 0 & \cdots & 0 \\ 0 & 0 & m_1^{\frac{1}{2}} & 0 & \cdots & 0 \\ 0 & 0 & 0 & m_2^{\frac{1}{2}} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & m_n^{\frac{1}{2}} \end{bmatrix} \quad (2.24)$$

Using the fact that $\mathbf{M} = \mathbf{M}^{\frac{1}{2}} \mathbf{M}^{\frac{1}{2}}$ and $\mathbf{M}^{\frac{1}{2}}$ is symmetric, the Gaussian can be rewritten as

$$\begin{aligned} G &= (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)^T \mathbf{M}^{\frac{1}{2}} \mathbf{M}^{\frac{1}{2}} (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u) \\ &= (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)^T (\mathbf{M}^{\frac{1}{2}})^T \mathbf{M}^{\frac{1}{2}} (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u) \\ &= [\mathbf{M}^{\frac{1}{2}} (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)]^T [\mathbf{M}^{\frac{1}{2}} (\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)] \end{aligned} \quad (2.25)$$

The alternate form of the Gaussian given by Eq. (2.25) would be used in Sec. 2.6. To summarize the discussion in this subsection, the actual generalized acceleration under the effect of constraints must fulfill two conditions:

1. $\ddot{\mathbf{q}}$ should satisfy the constraint equation (2.21).
2. $\ddot{\mathbf{q}}$ has to minimize the Gaussian given by Eq. (2.25).

2.4 The Udwadia-Kalaba Equation

In the field of analytical mechanics, the Udwadia-Kalaba equation is a relatively new development that provides an elegant form of the constrained acceleration based on the Moore-Penrose generalized inverse of the constraint matrix while taking into account the given forces. In this section, the fundamental equation is introduced and greater insights will be derived in the following sections through discussion about the MP-generalized inverse. Subsequently, it would be shown that the formulation satisfies Gauss's principle and hence, describes the actual constrained motion.

Given the generalized mass matrix $\mathbf{M}$ and the given force vector $\mathbf{F}$, the unconstrained acceleration $\ddot{\mathbf{q}}_u$ can be obtained by Eq. (2.6). In the presence of m constraints as given by Eq. (2.21), the constrained acceleration is given by

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_u + \mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u) \quad (2.26)$$

Equation (2.26) is the *Udwadia-Kalaba* equation that solves for the constrained acceleration of the system where $^+$ denotes the Moore-Penrose generalized inverse of the constraint matrix $\mathbf{B} = \mathbf{A}\mathbf{M}^{-\frac{1}{2}}$.

2.5 The Moore-Penrose Generalized Inverse

Let $\mathbf{B}$ be an $m \times n$ matrix of rank r . Then, the singular value decomposition of matrix $\mathbf{B}$ is given by

$$\mathbf{B} = \mathbf{W}\mathbf{\Lambda}\mathbf{V}^T \quad (2.27)$$

where $\mathbf{W}_{m \times r}$ and $\mathbf{V}_{n \times r}$ are orthogonal matrices in $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively, and $\mathbf{\Lambda}_{r \times r}$ is a diagonal matrix containing the r singular values of $\mathbf{B}$. The Moore-Penrose generalized inverse of the matrix $\mathbf{B}$ is unique and is given by

$$\mathbf{B}^+ = \mathbf{V}\mathbf{\Lambda}^{-1}\mathbf{W}^T \quad (2.28)$$

where $\mathbf{\Lambda}^{-1}$ implies the reciprocal of individual diagonal elements in the singular matrix. Significance of the Moore-Penrose generalized inverse, $\mathbf{B}^+$, is revealed when the results from satisfying the Moore-Penrose condition is understood. The four conditions are as follows:

1. $\mathbf{B}\mathbf{B}^+\mathbf{B} = \mathbf{B}$
2. $\mathbf{B}^+\mathbf{B}\mathbf{B}^+ = \mathbf{B}^+$
3. $(\mathbf{B}\mathbf{B}^+)^T = (\mathbf{B}\mathbf{B}^+)$
4. $(\mathbf{B}^+\mathbf{B})^T = (\mathbf{B}^+\mathbf{B})$

Using the singular value decomposition results as shown in Eq. (2.27) and (2.28) and the fact that $\mathbf{W}^T\mathbf{W} = \mathbf{V}^T\mathbf{V} = \mathbf{I}_r$, where $\mathbf{I}_r$ is the $r \times r$ identity matrix, the above four conditions can be easily proven.

Theorem 1. *For a given $m \times n$ matrix $\mathbf{B}$ and $m \times 1$ matrix $\mathbf{c}$, consider a possibly inconsistent set of equations of the form*

$$\mathbf{B}\mathbf{x} = \mathbf{c} \tag{2.29}$$

where $\mathbf{x}$ is the unknown. Suppose that there is a matrix $\mathbf{G}$ such that $\mathbf{x} = \mathbf{G}\mathbf{b}$ has the shortest length and when substituted into Eq. (2.29), yields the minimum value for $\|\mathbf{B}\mathbf{x} - \mathbf{c}\|$, then $\mathbf{G} = \mathbf{B}^+$, the Moore-Penrose generalized inverse.

Proof. Proof of this theorem is provided in Appendix A.1. Utility of the *possible inconsistency* of Eq. (2.29) will become clear from the discussion on control constraints in Sec. 2.7.2. The inconsistency has to do with the fact that states may be away from the constraint surface during initialization. $\square$

2.6 Udwadia-Kalaba Equation and the Gauss's Principle

Now that the reader is aware of the Moore-Penrose generalized inverse, it can be proven that the Udwadia-Kalaba equation of (2.26) satisfies the two conditions of

Gauss's Principle of Least Constraint and hence, provide the actual constrained acceleration of the system. The following subsections discuss this aspect.

2.6.1 Udwadia-Kalaba Equation Satisfies the Constraint Equation

From Eq. (2.26),

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_u + \mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u)$$

Pre-multiplying matrix $\mathbf{A}$ to $\ddot{\mathbf{q}}$, Eq. (2.26) becomes

$$\begin{aligned} \mathbf{A}\ddot{\mathbf{q}} &= \mathbf{A}(\ddot{\mathbf{q}}_u + \mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u)) \\ &= \mathbf{A}\ddot{\mathbf{q}}_u + \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} - \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{A}\ddot{\mathbf{q}}_u \\ &= \mathbf{A}\ddot{\mathbf{q}}_u + \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} - \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{A}\mathbf{M}^{-\frac{1}{2}}\mathbf{M}^{\frac{1}{2}}\ddot{\mathbf{q}}_u \\ &= \mathbf{A}\ddot{\mathbf{q}}_u + \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} - \mathbf{A}\mathbf{M}^{-\frac{1}{2}}\ddot{\mathbf{q}}_u \quad (\text{Moore-Penrose condition 1}) \\ &= \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} \end{aligned} \tag{2.30}$$

By expanding the constraint equation (2.21) $\mathbf{A}\ddot{\mathbf{q}} = \mathbf{b}$, pre-multiplying by $\mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+$ and subsequently, using Moore-Penrose condition 1,

$$\begin{aligned} \mathbf{A}\mathbf{M}^{-\frac{1}{2}}\mathbf{M}^{\frac{1}{2}}\ddot{\mathbf{q}} &= \mathbf{b} \\ \implies \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{A}\mathbf{M}^{-\frac{1}{2}}\mathbf{M}^{\frac{1}{2}}\ddot{\mathbf{q}} &= \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} \\ \implies \mathbf{A}\mathbf{M}^{-\frac{1}{2}}\mathbf{M}^{\frac{1}{2}}\ddot{\mathbf{q}} &= \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} \\ \implies \mathbf{A}\ddot{\mathbf{q}} &= \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} \\ \implies \mathbf{b} &= \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} \end{aligned} \tag{2.31}$$

Using Eq. (2.31) in Eq. (2.30),

$$\mathbf{A}\ddot{\mathbf{q}} = \mathbf{A}\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+\mathbf{b} = \mathbf{b} \tag{2.32}$$

Indeed, as shown in Eq. (2.32), the Udwadia-Kalaba equation satisfies the acceleration constraint.

2.6.2 Udwadia-Kalaba Equation Minimizes the Gaussian

Let $\ddot{\mathbf{q}}_p$ be a possible acceleration vector such that $\ddot{\mathbf{q}} + \mathbf{a}$ satisfies the constraint equation. Then, the Gaussian from Eq. (2.25) is given by

$$\begin{aligned} G(\ddot{\mathbf{q}}_p) &= [\mathbf{M}^{\frac{1}{2}}(\ddot{\mathbf{q}}_p - \ddot{\mathbf{q}}_u)]^T [\mathbf{M}^{\frac{1}{2}}(\ddot{\mathbf{q}}_p - \ddot{\mathbf{q}}_u)] \\ &= [\mathbf{M}^{\frac{1}{2}}(\ddot{\mathbf{q}} + \mathbf{a} - \ddot{\mathbf{q}}_u)]^T [\mathbf{M}^{\frac{1}{2}}(\ddot{\mathbf{q}} + \mathbf{a} - \ddot{\mathbf{q}}_u)] \end{aligned} \quad (2.33)$$

Substituting the Udwadia-Kalaba equation in Eq. (2.33) and simplifying,

$$\begin{aligned} G(\ddot{\mathbf{q}}_p) &= [\mathbf{M}^{\frac{1}{2}}(\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u) + \mathbf{a})]^T \mathbf{M}^{\frac{1}{2}}(\mathbf{M}^{-\frac{1}{2}}(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u) + \mathbf{a}) \\ &= ((\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u))^T (\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u) \\ &\quad + [(\mathbf{M}^{\frac{1}{2}}\mathbf{a})^T (\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u)]^T + (\mathbf{M}^{\frac{1}{2}}\mathbf{a})^T (\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u) \\ &\quad + (\mathbf{M}^{\frac{1}{2}}\mathbf{a})^T \mathbf{M}^{\frac{1}{2}}\mathbf{a} \end{aligned} \quad (2.34)$$

Now, the Gaussian of the Udwadia-Kalaba equation-based constraint acceleration is

$$\begin{aligned} G(\ddot{\mathbf{q}}) &= [\mathbf{M}^{\frac{1}{2}}(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)]^T [\mathbf{M}^{\frac{1}{2}}(\ddot{\mathbf{q}} - \ddot{\mathbf{q}}_u)] \\ &= ((\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u))^T (\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b} - \mathbf{A}\ddot{\mathbf{q}}_u) \end{aligned} \quad (2.35)$$

The fact that $\ddot{\mathbf{q}}_p$ and $\ddot{\mathbf{q}}$ satisfy the constraint equation can be exploited to obtain

$$\begin{aligned} \mathbf{A}\ddot{\mathbf{q}}_p &= \mathbf{b} \\ \implies \mathbf{A}\ddot{\mathbf{q}} + \mathbf{A}\mathbf{a} &= \mathbf{b} \\ \implies \mathbf{b} + \mathbf{A}\mathbf{a} &= \mathbf{b} \\ \implies \mathbf{A}\mathbf{a} &= \mathbf{0} \\ \mathbf{A}\mathbf{M}^{-\frac{1}{2}}\mathbf{M}^{\frac{1}{2}}\mathbf{a} &= \mathbf{0} \end{aligned} \quad (2.36)$$

Let $\mathbf{C} = \mathbf{M}^{\frac{1}{2}}\mathbf{a}$. Then, Eq. (2.36) is given by

$$\begin{aligned} \implies \mathbf{B}\mathbf{C} &= \mathbf{0} \\ \implies \mathbf{B}^+\mathbf{B}\mathbf{C}\mathbf{C}^+ &= \mathbf{0} \end{aligned}$$

$$\begin{aligned}
(\mathbf{C}\mathbf{C}^+)^T(\mathbf{B}^+\mathbf{B})^T &= \mathbf{C}\mathbf{C}^+\mathbf{B}^+\mathbf{B} = \mathbf{0} && \text{(transposing)} \\
\implies \mathbf{C}^+\mathbf{C}\mathbf{C}^+\mathbf{B}^+\mathbf{B}\mathbf{B}^+ &= \mathbf{0} \\
\implies \mathbf{C}^+\mathbf{B}^+ &= \mathbf{0}
\end{aligned} \tag{2.37}$$

Since $\mathbf{C}$ is a row vector, a standard result gives the Moore-Penrose generalized inverse as

$$\mathbf{C}^+ = (\mathbf{C}^T\mathbf{C})\mathbf{C}^T \tag{2.38}$$

Then, Eq. (2.37) becomes

$$\begin{aligned}
(\mathbf{C}^T\mathbf{C})\mathbf{C}^T\mathbf{B}^+ &= \mathbf{0} \\
\implies \mathbf{C}^T\mathbf{B}^+ &= \mathbf{0} && \text{since } \mathbf{C}^T\mathbf{C} > 0 \\
\implies (\mathbf{M}^{\frac{1}{2}}\mathbf{a})^T(\mathbf{A}\mathbf{M}^{-\frac{1}{2}})^+ &= \mathbf{0}
\end{aligned} \tag{2.39}$$

Using Eq. (2.35) and (2.39) in Eq. (2.34),

$$G(\ddot{\mathbf{q}}_p) = G(\ddot{\mathbf{q}}) + (\mathbf{M}^{\frac{1}{2}}\mathbf{a})^T\mathbf{M}^{\frac{1}{2}}\mathbf{a} = G(\ddot{\mathbf{q}}) + \mathbf{a}^T\mathbf{M}\mathbf{a} \tag{2.40}$$

Since $\mathbf{M}$ is positive definite, $G(\ddot{\mathbf{q}}_p) > G(\ddot{\mathbf{q}})$ for all $\mathbf{a} \neq \mathbf{0}$. Therefore, the global minimum of G occurs when $\ddot{\mathbf{q}}_p = \ddot{\mathbf{q}}$. By Gauss's Principle of Least Constraint, the acceleration that's materialized is given by the Udwadia-Kalaba equation.

To summarize, when the dynamical system is subjected to a set of constraints, the Udwadia-Kalaba equation provides a constraint force that minimizes the Gaussian, which is the cost function.

2.7 System Constraints

As seen from Sec. 2.2, the Udwadia-Kalaba framework is able to handle either of the holonomic or non-holonomic constraints with ease and therefore, such a categorization is not necessary. A more useful categorization in the case of the Udwadia-Kalaba equation-based methods is to classify them into modeling and control constraints.

2.7.1 Modeling Constraints

A modeling constraint occurs when describing the dynamics of a system in the Udwadia-Kalaba equation-based approach. This could be used either in predicting the exact motion in the dynamics layer simulation or in the control layer for generating guidance commands that take into account the prospective motion of the dynamical system. Suppose that a modeling constraint is given by

$$\phi_m(\mathbf{q}, \dot{\mathbf{q}}, t) = 0 \quad (2.41)$$

The first derivative of Eq (2.41) with respect to time is obtained as

$$\frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \mathbf{q}} \dot{\mathbf{q}} + \frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{\mathbf{q}}} \ddot{\mathbf{q}} + \frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial t} = 0 \quad (2.42)$$

Alternately, Eq. (2.42) can be rearranged to obtain

$$\frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{\mathbf{q}}} \ddot{\mathbf{q}} = -\frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \mathbf{q}} \dot{\mathbf{q}} - \frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial t} \quad (2.43)$$

$$\implies \mathbf{A}_m \ddot{\mathbf{q}} = b_m \quad (2.44)$$

where

$$\mathbf{A}_m = \frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \dot{\mathbf{q}}}, \quad b_m = -\frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial \mathbf{q}} \dot{\mathbf{q}} - \frac{\partial \phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)}{\partial t} \quad (2.45)$$

Note that when the modeling constraint is given by $\phi_m(\mathbf{q}, \dot{\mathbf{q}}, t)$, a function of $\mathbf{q}$, $\dot{\mathbf{q}}$ and t , the constraint function has to be double-differentiated to obtain the form of Eq. (2.44). Using the Udwadia-Kalaba equation, the net generalized acceleration under the effect of the modeling constraints is given by

$$\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_u + \mathbf{M}^{-\frac{1}{2}} (\mathbf{A}_m \mathbf{M}^{-\frac{1}{2}})^+ (b_m - \mathbf{A}_m \ddot{\mathbf{q}}_u) \quad (2.46)$$

An example of the modeling constraint is the unit vector constraint that comes up in the quaternion-based dynamic formulation, which is given by

$$u_1^2 + u_2^2 + u_3^2 + u_4^2 = 1 \quad (2.47)$$

where $u_i, i = 1, 2, 3, 4$ are components of the quaternion vector along the axes of the hyper-complex plane. By differentiating with respect to time, Eq. (2.47) becomes

$$2u_1\dot{u}_1 + 2u_2\dot{u}_2 + 2u_3\dot{u}_3 + 2u_4\dot{u}_4 = 0 \quad (2.48)$$

Further differentiation of Eq. (2.48) yields

$$\begin{aligned} & 2u_1\ddot{u}_1 + 2u_2\ddot{u}_2 + 2u_3\ddot{u}_3 + 2u_4\ddot{u}_4 + 2\dot{u}_1^2 + 2\dot{u}_2^2 + 2\dot{u}_3^2 + 2\dot{u}_4^2 = 0 \\ \implies & u_1\ddot{u}_1 + u_2\ddot{u}_2 + u_3\ddot{u}_3 + u_4\ddot{u}_4 = -\dot{u}_1^2 - \dot{u}_2^2 - \dot{u}_3^2 - \dot{u}_4^2 \\ \implies & \mathbf{u}^T \ddot{\mathbf{u}} = -\mathbf{u}^T \dot{\mathbf{u}} \end{aligned} \quad (2.49)$$

Note that Eq. (2.49) is of the form

$$\mathbf{A}_m \ddot{\mathbf{u}} = b_m \quad (2.50)$$

where $\mathbf{A}_m = \mathbf{u}^T$ and $b_m = -\mathbf{u}^T \dot{\mathbf{u}}$. Using the Udwadia-Kalaba equation of Eq. (2.26), the constrained quaternion acceleration vector is given by

$$\ddot{\mathbf{u}} = \ddot{\mathbf{u}}_u + \mathbf{M}_u^{-\frac{1}{2}} (\mathbf{A}_m \mathbf{M}_u^{-\frac{1}{2}})^+ (b_m - \mathbf{A}_m \ddot{\mathbf{u}}_u) \quad (2.51)$$

where $\mathbf{u}_m$ is the unconstrained quaternion acceleration vector and $\mathbf{M}_u$ is the generalized inertial matrix that will be detailed in the next chapter. The use of this quaternion modeling constraint to formulate the Udwadia-Kalaba equation-based rigid body dynamics layer will also be demonstrated.

2.7.2 Control Constraints

In its fundamental sense, control is a constraint on the state of the system. This idea was exploited in the work by Koganti and Udwadia [44] to generate desired force commands to achieve a specific control action of a multi-body system. Once a constraint equation is formulated, it is easy to incorporate it into the Udwadia-Kalaba equation to generate the exact real-time force commands that can achieve the desired trajectory with the set control performance characteristics.

Suppose that the desired control objective is to get the k^{th} state of a system q_k to the desired state $q_{k,d}$. Then, the i^{th} control constraint i can be defined as

$$\phi_i := q_k - q_{k,d} \quad (2.52)$$

ϕ_i characterizes the deviation of the current state from the desired state. Now, the control action has to act in such a way that the function $\phi_i \rightarrow 0$. The subsequent discussion clarifies as to how this could be done.

To begin, take the first derivative of Eq. (2.52) with respect to time, which is given by

$$\dot{\phi}_i = \dot{q}_k - \dot{q}_{k,d} \quad (2.53)$$

On further differentiating with time, Eq. (2.53) yields

$$\ddot{\phi}_i = \ddot{q}_k - \ddot{q}_{k,d} \quad (2.54)$$

Using Eq. (2.52)-(2.54) to form a second order error dynamics equation in ϕ_i ,

$$\ddot{\phi}_i + \alpha_i \dot{\phi}_i + \gamma_i \phi_i = 0 \quad (2.55)$$

where $\alpha_i > 0$ and $\gamma_i > 0$ are constants. The value of α_i and γ_i could be chosen to achieve the desired performance characteristics.

Note that in the beginning, q_k need not satisfy the control constraint equation. Theorem 1 provided in Sec. 2.5 shows that the pseudo inverse, in this case, provides the shortest distance to the control constraint.

Substituting Eq. (2.52) to (2.54) into Eq. (2.55) and rearranging,

$$\ddot{q}_k = \ddot{q}_{k,d} - \alpha_i(\dot{q}_k - \dot{q}_{k,d}) - \gamma_i(q_k - q_{k,d}) \quad (2.56)$$

In the standard form, the i^{th} control constraint could be written as

$$\mathbf{A}_{c,i}\ddot{\mathbf{q}} = b_{c,i} \quad (2.57)$$

where $\mathbf{A}_{c,i}$ is a $1 \times n$ matrix with the element multiplying with the q_k^{th} term as 1 and the rest 0, $\mathbf{q}$ is the generalized coordinate, and

$$b_{c,i} = \ddot{q}_{k,d} - \alpha_i(\dot{q}_k - \dot{q}_{k,d}) - \gamma_i(\dot{q}_k - \dot{q}_{k,d}) \quad (2.58)$$

For a set of m constraints, Eq. (2.57) is composed into a single control constraint matrix equation as

$$\mathbf{A}_c\ddot{\mathbf{q}} = \mathbf{b}_c \quad (2.59)$$

where $\mathbf{A}_c = \begin{bmatrix} \mathbf{A}_{c,1} & \mathbf{A}_{c,2} & \dots & \mathbf{A}_{c,m} \end{bmatrix}_{m \times n}^T$ and $\mathbf{b}_c = \begin{bmatrix} b_{c,1} & b_{c,2} & \dots & b_{c,m} \end{bmatrix}_{m \times 1}^T$

Using the Udwadia-Kalaba equation of Eq. (2.26), the exact control force to achieve the desired state at the specified performance is then given by

$$\mathbf{F}_c = \mathbf{M}^{\frac{1}{2}}(\mathbf{A}_c\mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b}_c - \mathbf{A}_c\ddot{\mathbf{q}}_u) \quad (2.60)$$

In the following chapter, the dynamical equations of motion will be detailed. The quaternion-based attitude dynamic equations will also be discussed.

Chapter 3

Dynamic Model

Coordinate frames are necessary to express the three-dimensional components of any vector in space. This chapter introduces the frames of reference that are relevant to the pose tracking problem and discusses the formulation of dynamic model. The double-integrator dynamics is introduced for the planar case and subsequently, details of the 6-degree-of-freedom model for the spatial simulation is detailed. The primary references for this chapter include Ref. [46] that discusses the basics of orbital mechanics and Ref. [47] that discusses the quaternion-based tumbling dynamics. At the end of this chapter, the reader would be familiar with the prior work in the existing literature that are useful in building the proposed pose tracking controller as detailed in the subsequent chapters.

3.1 Frames of Reference

A vector in space can be represented using a set of components along a specified frame of reference. The frame, in general, is non-inertial, which means that the origin of the frame accelerates in time. For example, if a frame is fixed on the surface of earth, this frame rotates around the earth's spin axis and hence, it is an accelerating frame. A frame fixed at the center of the earth is also revolving around the sun and hence, in theory, this is an accelerating frame. But, often, as a reasonable approximation wherein the accelerations are negligibly small and the relative times for the analysis as compared to the time period of an event is short, the frame may be assumed as inertial, which simplifies our analysis. Conventionally, in the domain of pose tracking, the Earth-Centered-Inertial (ECI) frame, the Local-Vectical-Local-Horizontal (LVLH) frame and the body-fixed frames are used.

3.1.1 ECI frame

The ECI frame has its origin fixed at the center of the earth. The x -axis points towards the Vernal Equinox line, z -axis along the Earth's spin axis and the y -axis is eastward along the right ascension of the ascending node and completes the triad.

Figure 3.1: ECI frame of reference

Figure 3.1 shows the spatial orientation of the ECI frame with respect to a wire-mesh model of the earth. Due to precession and nutation effects, the equatorial plane and the ecliptic are not strictly fixed with respect to the stars. But, for all practical purposes, these effects are meager for shorter time spans of a few days and hence, can be neglected. The ECI frame can, hence, be assumed as inertial without much adverse effects. As a result, Newton's laws can be applied in the frame. Let the position vector of the center of mass of a target object be $\vec{r}_t$. Then, in the ECI frame,

this is represented by

$$\vec{\mathbf{r}}_t = X\vec{\mathbf{I}}_x + Y\vec{\mathbf{I}}_y + Z\vec{\mathbf{I}}_z \quad (3.1)$$

where X , Y and Z are components of the position vector along $\vec{\mathbf{I}}_x$, $\vec{\mathbf{I}}_y$ and $\vec{\mathbf{I}}_z$, respectively.

3.1.2 LVLH Frame

The LVLH frame is an accelerating frame of reference whose origin is fixed on the center of mass of the orbiting target object in space and is defined by its three unit vectors $\{\vec{\mathbf{L}}_x, \vec{\mathbf{L}}_y, \vec{\mathbf{L}}_z\}$. This reference frame has its $\vec{\mathbf{L}}_x$ unit vector in the direction of inertial target position vector $\vec{\mathbf{r}}_t$, its $\vec{\mathbf{L}}_z$ unit vector perpendicular to the orbital plane and in the direction of the target orbital angular momentum vector $\vec{\mathbf{h}}_t$, and its $\vec{\mathbf{L}}_y$ in the orbital plane completing the triad, that is,

$$\vec{\mathbf{L}}_x = \frac{\vec{\mathbf{r}}_t}{r_t} \quad \vec{\mathbf{L}}_y = \frac{\vec{\mathbf{h}}_t}{h_t} \quad \vec{\mathbf{L}}_z = \vec{\mathbf{L}}_y \times \vec{\mathbf{L}}_x \quad (3.2)$$

where $r_t = \|\vec{\mathbf{r}}_t\|$ and $h_t = \|\vec{\mathbf{h}}_t\|$.

Figure 3.2: LVLH frame with respect to the target center of mass

The LVLH reference frame as shown in Fig. 3.2 has its origin at the target object.

Taking the components along each of these coordinate axes as x , y and z , respectively, the position vector of a point with respect to the LVLH frame is given by

$$\vec{\mathbf{r}}_h = x\vec{\mathbf{L}}_x + y\vec{\mathbf{L}}_y + z\vec{\mathbf{L}}_z \quad (3.3)$$

3.1.3 Body-fixed Frame

A body-fixed frame as in Fig. 3.3 has its origin at the center of mass of the object and it is used to describe the rotation of the object.

Figure 3.3: Body-fixed frame of reference

Note that the body being considered, in the context of this work, could be a target that is tumbling uncontrollably along all three axes while translating on orbit, or a controlled chaser spacecraft that is attempting to pose track the uncontrolled target. Fixing the orientation of the three mutually perpendicular axes with respect to the body depends on the spacecraft structure, its payload, and other mission-related

considerations. A vector on body-fixed frame of the object p is given by

$$\vec{\mathbf{r}}_p = x_p \vec{\mathbf{b}}_x + y_p \vec{\mathbf{b}}_y + z_p \vec{\mathbf{b}}_z \quad (3.4)$$

where x_p , y_p and z_p are components of vector $\vec{\mathbf{r}}_p$ along the body-fixed axes $\vec{\mathbf{b}}_x$, $\vec{\mathbf{b}}_y$ and $\vec{\mathbf{b}}_z$, respectively.

Conventional approaches assume one of the spacecraft (usually the target) as a point mass for simplifying the relative equations of motion and are expressed in the LVLH frame. This is a reasonable choice when assuming an attitude-stabilized target. However, the primary interest, in the context of this work, lies in a simultaneous position and angular velocity control of a 6-DOF chaser with respect to a 6-DOF tumbling and translating target. Hence, the control equations would be solved for in the ECI frame and transformed into the body-fixed chaser frame to generate the exact control forces to track the tumbling 6-DOF target.

3.2 Rotation Matrix

In spacecraft attitude dynamics and control problems, orientation of the body-fixed chaser frame is described with respect to the ECI frame of reference. Sensors such as star trackers directly measure this rotation or attitude matrix. This section discusses the rotation matrices used for transforming between frames of references. Two formulations of the rotation are relevant to our discussion. First is the Euler angle-based representation, which is used for the planar simulation and planar experimental validation of the pose tracking approach developed in this thesis. This is because of the fact that most motion capture systems used for planar validation of the guidance algorithms give their representations in an Euler angle form. The second formulation uses the quaternion-based representation which is relevant to the 6-DOF simulation of pose tracking. Quaternions avoid a frequently encountered singularity issue faced by the Euler angle representation.

3.2.1 Euler-Angle Representation

Suppose the coordinate transformation from the frame A to frame B is given by the $3 - 2 - 1$ rotation sequence of the frame by yaw (θ_z), pitch (θ_y) and roll (θ_x) angles, respectively. Then, the transformations that converts vector components form A to B is given by

$$\begin{aligned} \begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & \sin \theta_x \\ 0 & -\sin \theta_x & \cos \theta_x \end{bmatrix} \begin{bmatrix} \cos \theta_y & 0 & -\sin \theta_y \\ 0 & 1 & 0 \\ \sin \theta_y & 0 & \cos \theta_y \end{bmatrix} \begin{bmatrix} \cos \theta_z & \sin \theta_z & 0 \\ -\sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_A \\ y_A \\ z_A \end{bmatrix} \\ &= \mathbf{R}_{BA} \begin{bmatrix} x_A \\ y_A \\ z_A \end{bmatrix} \end{aligned} \quad (3.5)$$

where $\mathbf{R}_{BA}$ represents the rotation matrix that transforms the components from frame A to B and is given by

$$\mathbf{R}_{BA} = \begin{bmatrix} \cos \theta_y \cos \theta_z & \cos \theta_y \sin \theta_z & -\sin \theta_y \\ \sin_x \sin_y \cos_z - \cos_x \sin_z & \sin \theta_x \sin \theta_y \sin \theta_z + \cos \theta_x \cos \theta_z & \sin \theta_x \cos \theta_y \\ \cos \theta_x \sin \theta_y \cos \theta_z + \sin \theta_x \sin \theta_z & \cos \theta_x \sin \theta_y \sin \theta_z - \sin \theta_x \cos \theta_y & \cos \theta_x \cos \theta_y \end{bmatrix} \quad (3.6)$$

The relationship between the Euler angle rates ($\dot{\theta}_x, \dot{\theta}_y, \dot{\theta}_z$) and the body-fixed angular velocity is given by

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta_y \\ 0 & \cos \theta_x & \sin \theta_x \cos \theta_y \\ 0 & -\sin \theta_x & \cos \theta_x \cos \theta_y \end{bmatrix} \begin{bmatrix} \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} \quad (3.7)$$

By inverting the transformation matrix in Eq. (3.7), the matrix that converts Euler rates to the body-fixed angular rates is obtained as

$$\begin{bmatrix} \dot{\theta}_x \\ \dot{\theta}_y \\ \dot{\theta}_z \end{bmatrix} = \begin{bmatrix} 1 & \sin \theta_x \tan \theta_y & \cos \theta_x \tan \theta_y \\ 0 & \cos \theta_x & -\sin \theta_x \\ 0 & \sin \theta_x \sec \theta_y & \cos \theta_x \sec \theta_x \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (3.8)$$

Note that for values of $\theta_y = \pm\pi/2$, the transformation matrix of Eq. (3.8) is undefined. Therefore, the Euler angle representations suffer from the issue of singularity and cannot represent some orientations. In order to overcome the singularity issue, quaternions were introduced, which is discussed in the next section.

3.2.2 Quaternion Representation

Quaternion is a hyper-complex number that parametrizes general tumbling as a rotation about a unit vector, $\hat{\mathbf{a}}$ through an angle ϕ . They are also called the *Euler parameters* and is represented by

$$\begin{aligned}\vec{\mathbf{u}} &= \cos\left(\frac{\theta}{2}\right) + \sin\left(\frac{\theta}{2}\right)\hat{\mathbf{a}} \\ &= \cos\left(\frac{\theta}{2}\right) + a_x \sin\left(\frac{\theta}{2}\right)\hat{\mathbf{I}} + a_y \sin\left(\frac{\theta}{2}\right)\hat{\mathbf{J}} + a_z \sin\left(\frac{\theta}{2}\right)\hat{\mathbf{K}} \\ &= u_0 + u_1\hat{\mathbf{I}} + u_2\hat{\mathbf{J}} + u_3\hat{\mathbf{K}}\end{aligned}\tag{3.9}$$

In component form, the quaternion can be represented as

$$\mathbf{u} = \begin{bmatrix} u_0 & u_1 & u_2 & u_3 \end{bmatrix}^T\tag{3.10}$$

The Euclidean norm of the quaternion vector is given by

$$\|\mathbf{u}\| = \sqrt{u_0^2 + u_1^2 + u_2^2 + u_3^2}\tag{3.11}$$

$$= \sqrt{\cos^2\left(\frac{\theta}{2}\right) + a_x^2 \sin^2\left(\frac{\theta}{2}\right) + a_y^2 \sin^2\left(\frac{\theta}{2}\right) + a_z^2 \sin^2\left(\frac{\theta}{2}\right)}\tag{3.12}$$

$$= 1\tag{3.13}$$

Hence, the quaternion is a unit vector. The rotation matrix used to transform the components of a vector from the body-fixed frame, B to the inertial ECI frame, I is given by

$$\mathbf{R}_{IB} = \begin{bmatrix} 2u_0^2 - 1 + 2u_1^2 & 2u_1u_2 - 2u_0u_3 & 2u_1u_3 + 2u_0u_2 \\ 2u_1u_2 + 2u_0u_3 & 2u_0^2 - 1 + 2u_2^2 & 2u_2u_3 - 2u_0u_1 \\ 2u_1u_3 - 2u_0u_2 & 2u_2u_3 + 2u_0u_1 & 2u_0^2 - 1 + 2u_3^2 \end{bmatrix}\tag{3.14}$$

By taking the time derivative, the quaternion vector becomes

$$\dot{\mathbf{u}} = \dot{u}_0 + \dot{u}_1 \hat{\mathbf{I}} + \dot{u}_2 \hat{\mathbf{J}} + \dot{u}_3 \hat{\mathbf{K}} \quad (3.15)$$

$$= \left(\frac{\dot{\theta}}{2}\right) \left(-\sin\left(\frac{\theta}{2}\right) + a_x \cos\left(\frac{\theta}{2}\right) \hat{\mathbf{I}} + a_y \cos\left(\frac{\theta}{2}\right) \hat{\mathbf{J}} + a_z \cos\left(\frac{\theta}{2}\right) \hat{\mathbf{K}} \right) \quad (3.16)$$

Now, an interesting result is obtained when taking the scalar product of the quaternion vector and its time derivative.

$$\mathbf{u} \cdot \dot{\mathbf{u}} = \left(\frac{\dot{\theta}}{2}\right) \left(\sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right) \right) \left(-1 + a_x^2 + a_y^2 + a_z^2 \right) \quad (3.17)$$

$$= \left(\frac{\dot{\theta}}{2}\right) \left(\sin\left(\frac{\theta}{2}\right) \cos\left(\frac{\theta}{2}\right) \right) \left(-1 + 1 \right) = 0 \quad (3.18)$$

$$\implies u_0 \dot{u}_0 + u_1 \dot{u}_1 + u_2 \dot{u}_2 + u_3 \dot{u}_3 = 0 \quad (3.19)$$

The body-fixed angular velocity is given by

$$\begin{bmatrix} 0 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 2u_0 & 2u_1 & 2u_2 & 2u_3 \\ -2u_1 & 2u_0 & 2u_3 & -2u_2 \\ -2u_2 & -2u_3 & 2u_0 & 2u_1 \\ -2u_3 & 2u_2 & -2u_1 & 2u_0 \end{bmatrix} \begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \end{bmatrix} \quad (3.20)$$

$$\implies \boldsymbol{\Omega} = 2\mathbf{E}\dot{\mathbf{u}} \quad (3.21)$$

where

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 \\ \omega \end{bmatrix} \quad (3.22)$$

and

$$\mathbf{E} = \begin{bmatrix} u_0 & u_1 & u_2 & u_3 \\ -u_1 & u_0 & u_3 & -u_2 \\ -u_2 & -u_3 & u_0 & u_1 \\ -u_3 & u_2 & -u_1 & u_0 \end{bmatrix} \quad (3.23)$$

Note that the first row of the $\boldsymbol{\Omega}$ vector is 0 because of the property from Eq. (3.19).

Further, ω is the body-fixed angular rate vector. The transformation from the body-fixed rates to the quaternion derivative can then be obtained by inverting the $\mathbf{E}$

matrix. The fact that Euclidean norm of the quaternion vector is unity can be used to obtain

$$\begin{bmatrix} \dot{u}_0 \\ \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} u_0 & -u_1 & -u_2 & -u_3 \\ u_1 & u_0 & -u_3 & u_2 \\ u_2 & u_3 & u_0 & -u_1 \\ u_3 & -u_2 & u_1 & u_0 \end{bmatrix} \begin{bmatrix} 0 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \frac{1}{2} \mathbf{E}^T \boldsymbol{\Omega} \quad (3.24)$$

Note that, $\mathbf{E}^T = \mathbf{E}^{-1}$. Therefore, $\mathbf{E}$ is an orthogonal matrix.

3.3 6 DOF Dynamics

The complete dynamics of the system is described by its translational and rotational components. Motion under the effect of the gravitational pull of the Earth and the perturbations is described in the translational part in the inertial ECI frame. Meanwhile, the tumbling dynamics with respect to the ECI frame is described using the quaternion based rotational dynamics equations.

3.3.1 Determination of Unconstrained Acceleration

Suppose that a force vector $\mathbf{F}_i$ defined in the inertial Cartesian frame acts on a system with mass matrix $\mathbf{M}_i$. If no additional constraints are considered, Newton's law relates the acceleration vector in the inertial frame, $\mathbf{a}_i$, by

$$\mathbf{M}_i \mathbf{a}_i = \mathbf{F}_i \quad (3.25)$$

$$\implies \mathbf{a}_i = \mathbf{M}_i^{-1} \mathbf{F}_i \quad (3.26)$$

Note that $\mathbf{a}_i$ is the unconstrained acceleration vector in the inertial frame.

When the system is more general and defined with a set of generalized coordinates $\mathbf{q}$, the generalized acceleration can be determined using the Langrangian equation given as

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_j} \right) - \frac{\partial \mathcal{L}}{\partial q_j} = Q_j \quad \text{for } j = 1, 2, \dots, n \quad (3.27)$$

where Q_j is the j^{th} generalized given force, q_j is the j^{th} generalized coordinate of the system and $\mathcal{L}$ is the Lagrangian or the kinetic potential of the system defined by $\mathcal{L} = T - V$. Here, T is the total kinetic energy of the system and V is the generalized potential energy of the system. When solved for each of the generalized coordinates, Equation (3.27) in matrix form becomes

$$\mathbf{M}(\mathbf{q}, t)\ddot{\mathbf{q}}_u = \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, t) \quad (3.28)$$

The unconstrained acceleration is obtained by inverting the mass matrix $\mathbf{M}(\mathbf{q}, t)$.

$$\ddot{\mathbf{q}}_u = \mathbf{M}(\mathbf{q}, t)^{-1}\mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, t) \quad (3.29)$$

3.3.2 Orbital Motion

The acceleration of a body with position vector $\mathbf{r}$ under the effect of gravitational pull of the Earth is expressed in the inertial frame by

$$\ddot{\mathbf{r}} = -\frac{GM}{r^3}\mathbf{r} \quad (3.30)$$

where G is the gravitational constant, M is the mass of earth and $r = \|\mathbf{r}\|$. Alternately, by defining the gravitational constant as $\mu = GM$,

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^3}\mathbf{r} \quad (3.31)$$

Equation (3.31), referred to as the two-body equation of motion, can be integrated twice to obtain the instantaneous state vector $\mathbf{r}(t)$ at time t . To do the numerical integration, the initial position $\mathbf{r}_0$ vector and the initial velocity $\dot{\mathbf{r}}_0$ vector need to be specified. Specifying these six initial conditions is equivalent to uniquely defining an orbit with respect to the inertial frame of reference. Alternately, to uniquely define a spacecraft orbit about the Earth, six other scalars, called the six “classical orbital elements” are used. A major advantage of using these as opposed to defining $\mathbf{r}_0$ and $\dot{\mathbf{r}}_0$ is that the orbital elements are much more intuitive to define an orbit. While the first five of the elements fully describe the size, shape and orientation of the orbit in

three dimensional space, the sixth one relates to position of the spacecraft in the orbit.

The six orbital elements are –

- a - semi-major axis
- e - eccentricity
- i - inclination
- Ω - right ascension of ascending node
- ω - argument of perigee
- θ - true anomaly

The elements a and e define the size and shape of the orbit in the orbit plane, respectively, and θ relates to the orbital position. The period of an orbit may be calculated from the semi-major axis as

$$t = \frac{2\pi}{\sqrt{\mu}} a^{\frac{3}{2}} \quad (3.32)$$

The elements i and Ω specify the orientation of the orbital plane with respect to the ECI reference frame, whereas the element ω defines the orientation of the orbit in its orbital plane. The inclination i corresponds to the angle between the orbital plane and the equatorial plane and is given as the angle between $\vec{\mathbf{h}}$ and $\vec{\mathbf{I}}_z$, where $\vec{\mathbf{h}}$ is the orbital angular momentum vector. The line of intersection between the equatorial and orbital planes is called the line of nodes. The ascending node is the point on the line of nodes where the orbit crosses from the southern to the northern hemisphere. In other words, this is the point where the orbit crosses from southern to the northern hemisphere. The right ascension of the ascending node, denoted by Ω , corresponds to the angle between the $\vec{\mathbf{I}}_x$ unit vector and the ascending node direction. Finally, the argument of perigee ω is used to orient the orbit in its plane, or, in other words, to locate the direction of the perigee. Specifically, ω is the angle between the direction of the ascending node and the direction of the perigee which is itself specified by the

eccentricity vector. The conversion from the orbital elements to the inertial position and velocity of the spacecraft is shown in Sec. A.2.

In reality, the Earth is an oblate spheroid with an equatorial diameter that exceeds the polar diameter by about 20 km. The resulting equatorial bulge is the most significant contributor to the gravitational perturbations affecting the orbit of a spacecraft. This J_2 perturbation is about three orders of magnitude smaller than the central attraction, but it is nevertheless easily noticed on the orbits of spacecraft. Under J_2 perturbation of the orbit, there would be an additional acceleration given by

$$\mathbf{a}_{J_2} = -\frac{3}{2}J_2\frac{\mu R^2}{r^4} \begin{bmatrix} \left(1 - 5\left(\frac{Z}{r}\right)^2\right)\frac{X}{r} \\ \left(1 - 5\left(\frac{Z}{r}\right)^2\right)\frac{Y}{r} \\ \left(3 - 5\left(\frac{Z}{r}\right)^2\right)\frac{Z}{r} \end{bmatrix} \quad (3.33)$$

where, $J_2 = 1.0826269 \times 10^{-3}$ is the coefficient of second zonal harmonic of Earth's potential. Accounting for this dominant orbital perturbation, the net gravitational acceleration of a spacecraft in orbit is given by

$$\begin{aligned} \ddot{\mathbf{r}} &= -\frac{\mu}{r^3}\mathbf{r} + \mathbf{a}_{J_2} \\ \Rightarrow \begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} &= -\frac{\mu}{r^3} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \frac{3}{2}J_2\frac{\mu R^2}{r^4} \begin{bmatrix} \left(1 - 5\left(\frac{Z}{r}\right)^2\right)\frac{X}{r} \\ \left(1 - 5\left(\frac{Z}{r}\right)^2\right)\frac{Y}{r} \\ \left(3 - 5\left(\frac{Z}{r}\right)^2\right)\frac{Z}{r} \end{bmatrix} \end{aligned} \quad (3.35)$$

3.3.3 Quaternion Tumbling Dynamics

A Langrangian-based derivation of the quaternion dynamics and the simplification of resulting equation after enforcing the quaternion unit vector modeling constraint was detailed by Udwadia and Schutte [47]. Using Eq. (3.21), the rotational kinetic energy of the body is given by

$$T_u = \frac{1}{2} \boldsymbol{\Omega}^T \mathbf{J} \boldsymbol{\Omega} \quad (3.36)$$

where $\mathbf{J}$ is the augmented inertia matrix given by

$$\mathbf{J} = \begin{bmatrix} J_0 & 0 & 0 & 0 \\ 0 & J_x & 0 & 0 \\ 0 & 0 & J_y & 0 \\ 0 & 0 & 0 & J_z \end{bmatrix} \quad (3.37)$$

Note that, J_x , J_y and J_z are principal moments of inertia along the x , y and z axes, respectively and $J_0 > 0$ is chosen arbitrarily. It can be seen in the subsequent derivation that an arbitrary value of J_0 does not affect the dynamics in any way. Alternately, $\mathbf{J}$ can also be written as

$$\mathbf{J} = \begin{bmatrix} J_0 & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 1} & \tilde{\mathbf{J}} \end{bmatrix} \quad (3.38)$$

where

$$\tilde{\mathbf{J}} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix} \quad (3.39)$$

Using Eq. (3.21) in the kinetic energy equation (3.36),

$$T_u = 2\dot{\mathbf{u}}^T \mathbf{E}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}} \quad (3.40)$$

The Lagrangian equation is given by

$$\frac{d}{dt} \left(\frac{\partial T_u}{\partial \dot{\mathbf{u}}} \right) - \frac{\partial T_u}{\partial \mathbf{u}} = \boldsymbol{\Gamma}_u \quad (3.41)$$

where $\mathbf{\Gamma}_u$ is the generalized quaternion force vector. When solving the Langrangian equation, Eq. (3.41) results in

$$4\mathbf{E}^T \mathbf{J} \mathbf{E} \ddot{\mathbf{u}} = \mathbf{\Gamma}_u - 8\dot{\mathbf{E}}^T \hat{\mathbf{J}} \mathbf{E} \dot{\mathbf{u}} - 4\mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}} \quad (3.42)$$

$$\implies \mathbf{M}_u \ddot{\mathbf{u}} = \mathbf{Q}_u \quad (3.43)$$

where $\mathbf{M}_u = 4\mathbf{E}^T \mathbf{J} \mathbf{E}$ and $\mathbf{Q}_u = \mathbf{\Gamma}_u - 8\dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}} - 4\mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}}$. The unconstrained acceleration, $\mathbf{a}_u$ is given by rearranging Eq. (3.43) as

$$\mathbf{a}_u = \mathbf{M}_u^{-1} \mathbf{Q}_u = (4\mathbf{E}^T \mathbf{J} \mathbf{E})^{-1} (\mathbf{\Gamma}_u - 8\dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}} - 4\mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}}) \quad (3.44)$$

$$= (\mathbf{E}^{-1} \mathbf{J}^{-1} (\mathbf{E}^T)^{-1}) \left(\frac{\mathbf{\Gamma}_u}{4} - 2\dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}} - \mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}} \right) \quad (3.45)$$

$$= (\mathbf{E}^T \mathbf{J}^{-1} \mathbf{E}) \left(\frac{\mathbf{\Gamma}_u}{4} - 2\dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}} - \mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}} \right) \quad (3.46)$$

In the previous derivation, the quaternion vector was assumed to be unconstrained. But, in practice, the unit vector constraint needs to be enforced on the quaternion vector, which is given by

$$\|\mathbf{u}\|^2 = u_0^2 + u_1^2 + u_2^2 + u_3^2 = 1 \quad (3.47)$$

Double-differentiating and rearranging to get the quaternion acceleration vector to the left hand side,

$$2u_0\ddot{u}_0 + 2u_1\ddot{u}_1 + 2u_2\ddot{u}_2 + 2u_3\ddot{u}_3 = -2(\dot{u}_0^2 + \dot{u}_1^2 + \dot{u}_2^2 + \dot{u}_3^2) \quad (3.48)$$

$$\implies 2\mathbf{u}^T \ddot{\mathbf{u}} = -2\dot{\mathbf{u}}^T \dot{\mathbf{u}} \quad (3.49)$$

$$\implies \mathbf{u}^T \ddot{\mathbf{u}} = -\dot{\mathbf{u}}^T \dot{\mathbf{u}} \quad (3.50)$$

$$\implies \mathbf{A}_u \ddot{\mathbf{u}} = b_u \quad (3.51)$$

where $\mathbf{A}_u = \mathbf{u}^T$ and $b_u = -\dot{\mathbf{u}}^T \dot{\mathbf{u}}$. Using the fundamental equation (2.26),

$$\ddot{\mathbf{u}} = \mathbf{a}_u + \mathbf{M}_u^{-1} (\mathbf{M}_u^{\frac{1}{2}} (\mathbf{A}_u \mathbf{M}_u^{-\frac{1}{2}})^+ (b_u - \mathbf{A}_u \mathbf{a}_u)) \quad (3.52)$$

Let

$$\mathbf{E}_1 = \begin{bmatrix} -u_1 & u_0 & u_3 & -u_2 \\ -u_2 & -u_3 & u_0 & u_1 \\ -u_3 & u_2 & -u_1 & u_0 \end{bmatrix} \quad (3.53)$$

Using Eq. (3.10) and (3.53) in Eq. (3.23),

$$\mathbf{E} = \begin{bmatrix} \mathbf{u}^T \\ \mathbf{E}_1 \end{bmatrix} \quad (3.54)$$

Post-multiplying by $\mathbf{u}$ in Eq. (3.54),

$$\mathbf{E}\mathbf{u} = \begin{bmatrix} \mathbf{u}^T \mathbf{u} \\ \mathbf{E}_1 \mathbf{u} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (3.55)$$

Pre-multiplying Eq. (3.55) by $\mathbf{E}^T \mathbf{J}^{-1}$ and using the inverse of Eq. (3.38),

$$\mathbf{E}^T \mathbf{J}^{-1} \mathbf{E} \mathbf{u} = \begin{bmatrix} \mathbf{u} & \mathbf{E}_1^T \end{bmatrix} \begin{bmatrix} \frac{1}{J_0} & \mathbf{0}_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & \tilde{\mathbf{J}}^{-1} \end{bmatrix} \begin{bmatrix} 1 \\ \mathbf{0}_{3 \times 1} \end{bmatrix} = \frac{\mathbf{u}}{J_0} \quad (3.56)$$

Taking the transpose of Eq. (3.56),

$$\mathbf{u}^T \mathbf{E}^T \mathbf{J}^{-1} \mathbf{E} = \frac{\mathbf{u}^T}{J_0} \quad (3.57)$$

Using the fact that $\mathbf{A}_u = \mathbf{u}^T$ and Eq. (3.46), the term $\mathbf{A}_u \mathbf{a}_u$ in Eq. (3.52) is given by

$$\mathbf{A}_u \mathbf{a}_u = \mathbf{u}^T \mathbf{E}^T \mathbf{J}^{-1} \mathbf{E} \left(\frac{\Gamma_u}{4} - 2 \dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}} - \mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}} \right) \quad (3.58)$$

Using Eq. (3.57) in Eq. (3.58),

$$\mathbf{A}_u \mathbf{a}_u = \frac{\mathbf{u}^T \Gamma_u}{4J_0} - \frac{2\mathbf{u}^T \dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}}}{J_0} - \frac{\mathbf{u}^T \mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}}}{J_0} \quad (3.59)$$

The component $\frac{2\mathbf{u}^T \dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}}}{J_0}$ in Eq. (3.59) can be expressed as

$$\frac{2\mathbf{u}^T \dot{\mathbf{E}}^T \mathbf{J} \mathbf{E} \dot{\mathbf{u}}}{J_0} = -\frac{(2\mathbf{E}_1 \dot{\mathbf{u}})^T \tilde{\mathbf{J}} (2\mathbf{E}_1 \dot{\mathbf{u}})}{2J_0} = -\frac{\omega^T \tilde{\mathbf{J}} \omega}{2J_0} \quad (3.60)$$

Further, the term $\frac{\mathbf{u}^T \mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}}}{J_0}$ in Eq. (3.59) is alternately represented as

$$\frac{\mathbf{u}^T \mathbf{E}^T \mathbf{J} \dot{\mathbf{E}} \dot{\mathbf{u}}}{J_0} = \frac{4}{J_0} \mathbf{u}^T \begin{bmatrix} \mathbf{u} & \mathbf{E}_1^T \end{bmatrix} \begin{bmatrix} J_0 & \mathbf{0}_{1 \times 3} \\ \mathbf{0}_{3 \times 1} & \tilde{\mathbf{J}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}} \\ \dot{\mathbf{E}}_1 \end{bmatrix} \dot{\mathbf{u}} \quad (3.61)$$

$$= \frac{1}{J_0} \mathbf{u}^T \begin{bmatrix} J_0 \mathbf{u} & \mathbf{E}_1^T \tilde{\mathbf{J}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}}^T \dot{\mathbf{u}} \\ \dot{\mathbf{E}}_1 \dot{\mathbf{u}} \end{bmatrix} \quad (3.62)$$

$$= \frac{1}{J_0} \mathbf{u}^T \begin{bmatrix} J_0 \mathbf{u} & \mathbf{E}_1^T \tilde{\mathbf{J}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{u}}^T \dot{\mathbf{u}} \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (3.63)$$

$$= \frac{1}{J_0} J_0 \mathbf{u}^T \mathbf{u} \dot{\mathbf{u}}^T \dot{\mathbf{u}} \quad (3.64)$$

$$= \dot{\mathbf{u}}^T \dot{\mathbf{u}} \quad (3.65)$$

Substituting Eq. (3.60) and Eq. (3.65) into Eq. (3.59),

$$\mathbf{A}_u \mathbf{a}_u = \frac{\mathbf{u}^T \boldsymbol{\Gamma}_u}{4J_0} + \frac{\omega^T \tilde{\mathbf{J}} \omega}{2J_0} - \dot{\mathbf{u}}^T \dot{\mathbf{u}} \quad (3.66)$$

But $-\dot{\mathbf{u}}^T \dot{\mathbf{u}} = b_u$ and substituting this into Eq. (3.66) and rearranging,

$$b_u - \mathbf{A}_u \mathbf{a}_u = -\frac{\mathbf{u}^T \boldsymbol{\Gamma}_u}{4J_0} - \frac{\omega^T \tilde{\mathbf{J}} \omega}{2J_0} \quad (3.67)$$

Since $\mathbf{A}_u = \mathbf{u}^T$ is a row vector, a property of the Moore-Penrose generalized-inverse [42] allows to write the term $\mathbf{M}_u^{\frac{1}{2}} (\mathbf{A}_u \mathbf{M}_u^{-\frac{1}{2}})^+$ as

$$\begin{aligned} \mathbf{M}_u^{\frac{1}{2}} (\mathbf{A}_u \mathbf{M}_u^{-\frac{1}{2}})^+ &= \mathbf{A}_u^T (\mathbf{A}_u \mathbf{M}_u^{-1} \mathbf{A}_u^T)^{-1} \\ &= (\mathbf{u}^T)^T (\mathbf{u}^T (4\mathbf{E}^T \mathbf{J} \mathbf{E})^{-1} \mathbf{u})^{-1} \\ &= 4\mathbf{u} (\mathbf{u}^T \mathbf{E}^T \mathbf{J}^{-1} \mathbf{E} \mathbf{u}) \end{aligned} \quad (3.68)$$

Using the result from Eq. (3.57), Eq. (3.68) becomes

$$\mathbf{M}_{\mathbf{u}}^{\frac{1}{2}}(\mathbf{A}_{\mathbf{u}}\mathbf{M}_{\mathbf{u}}^{-\frac{1}{2}})^+ = 4J_0\mathbf{u}(\mathbf{u}^T\mathbf{u}) = 4J_0\mathbf{u} \quad (3.69)$$

From Eq. (3.67) and Eq. (3.69), $\mathbf{M}_{\mathbf{u}}^{\frac{1}{2}}(\mathbf{A}_{\mathbf{u}}\mathbf{M}_{\mathbf{u}}^{-\frac{1}{2}})^+(b_{\mathbf{u}} - \mathbf{A}_{\mathbf{u}}\mathbf{a}_{\mathbf{u}})$ in Eq. (3.52) becomes

$$\mathbf{M}_{\mathbf{u}}^{\frac{1}{2}}(\mathbf{A}_{\mathbf{u}}\mathbf{M}_{\mathbf{u}}^{-\frac{1}{2}})^+(b_{\mathbf{u}} - \mathbf{A}_{\mathbf{u}}\mathbf{a}_{\mathbf{u}}) = -\mathbf{u}\mathbf{u}^T\boldsymbol{\Gamma}_{\mathbf{u}} - 2(\omega^T\tilde{\mathbf{J}}\omega)\mathbf{u} \quad (3.70)$$

Substituting Eq. (3.70) into Eq. (3.52), $\ddot{\mathbf{u}}$ becomes

$$\begin{aligned} \ddot{\mathbf{u}} &= \mathbf{a}_{\mathbf{u}} - \mathbf{M}_{\mathbf{u}}^{-1}(\mathbf{u}\mathbf{u}^T\boldsymbol{\Gamma}_{\mathbf{u}} + 2(\omega^T\tilde{\mathbf{J}}\omega)) \\ &= (\mathbf{E}^T\mathbf{J}^{-1}\mathbf{E})\left(\frac{\boldsymbol{\Gamma}_{\mathbf{u}}}{4} - 2\dot{\mathbf{E}}^T\mathbf{J}\mathbf{E}\dot{\mathbf{u}} - \mathbf{E}^T\mathbf{J}\dot{\mathbf{E}}\dot{\mathbf{u}}\right) - (4\mathbf{E}^T\mathbf{J}^{-1}\mathbf{E})^{-1}(\mathbf{u}\mathbf{u}^T\boldsymbol{\Gamma}_{\mathbf{u}} + 2(\omega^T\tilde{\mathbf{J}}\omega)\mathbf{u}) \\ &= (\mathbf{E}^T\mathbf{J}^{-1}\mathbf{E})(\mathbf{I} - \mathbf{u}\mathbf{u}^T)\frac{\boldsymbol{\Gamma}}{4} - 2\mathbf{E}^T\mathbf{J}^{-1}\mathbf{E}\dot{\mathbf{E}}^T\mathbf{J}\mathbf{E}\dot{\mathbf{u}} - \mathbf{E}^T\mathbf{J}^{-1}\mathbf{E}\mathbf{E}^T\mathbf{J}\dot{\mathbf{E}}\dot{\mathbf{u}} \\ &\quad \mathbf{E}^T\mathbf{J}^{-1}\mathbf{E}\frac{\omega^T\tilde{\mathbf{J}}\omega\mathbf{u}}{2} \end{aligned} \quad (3.71)$$

By multiplying the matrices $\mathbf{E}$ and $\dot{\mathbf{E}}$, it can be proved that

$$\mathbf{E}\dot{\mathbf{E}} = \frac{1}{2} \begin{bmatrix} 0 & -\omega^T \\ \omega & \tilde{\omega} \end{bmatrix} \quad (3.72)$$

where

$$\tilde{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (3.73)$$

Using Eq. (3.23) and (3.56) with the additional property in Eq. (3.72), Eq. (3.71) becomes

$$\ddot{\mathbf{u}} = -\frac{1}{2}\mathbf{E}_1^T\tilde{\mathbf{J}}^{-1}\tilde{\omega}\tilde{\mathbf{J}}\omega - (\dot{\mathbf{u}}^T\dot{\mathbf{u}})\mathbf{u} + \frac{\mathbf{E}_1^T\tilde{\mathbf{J}}^{-1}\mathbf{E}_1\boldsymbol{\Gamma}_{\mathbf{u}}}{4} \quad (3.74)$$

Assuming a zero net external force ($\boldsymbol{\Gamma}_{\mathbf{u}} = \mathbf{0}$), the resulting quaternion acceleration vector under the unit vector constraint is given by

$$\ddot{\mathbf{u}} = -\frac{1}{2}\mathbf{E}_1^T\tilde{\mathbf{J}}^{-1}\tilde{\omega}\tilde{\mathbf{J}}\omega - (\dot{\mathbf{u}}^T\dot{\mathbf{u}})\mathbf{u} \quad (3.75)$$

where , $\mathbf{J} = \begin{bmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{bmatrix}$ and $\tilde{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$

3.3.4 Summary of the 6-DOF Dynamics

The generalized displacement vector of the coupled motion is defined as

$$\mathbf{q} = \begin{bmatrix} \mathbf{r}^T & \mathbf{u}^T \end{bmatrix}^T \quad (3.76)$$

The generalized acceleration vector is given by

$$\ddot{\mathbf{q}} = \begin{bmatrix} \mathbf{a}_g + \mathbf{a}_{J2} \\ -\frac{1}{2}\mathbf{E}_1^T \mathbf{J}^{-1} \tilde{\omega} \mathbf{J} \omega - (\dot{\mathbf{u}}^T \dot{\mathbf{u}}) \mathbf{u} \end{bmatrix} \quad (3.77)$$

This describes the orbital and attitude dynamics of the chaser spacecraft when no control is applied to it. In other words, Eq. (3.77) is the expression for the unconstrained acceleration. When acceleration constraints of the form in Eq. (2.21) are applied to the orbital and/or attitude dynamics, the resulting motion is modified. The generalized constraint force vector is determined using Eq. (2.60). When the constraints are derived from the control objectives, then the generalized control force is given by

$$\mathbf{Q}^C = \begin{bmatrix} \mathbf{F}^C \\ \mathbf{\Gamma}_u^C \end{bmatrix} \quad (3.78)$$

The generalized quaternion torque force vector $\mathbf{\Gamma}_u^C$ is related to the physically applied body torque $\mathbf{\Gamma}^C$ by

$$\mathbf{\Gamma}^C = \frac{1}{2} \mathbf{E}_1 \mathbf{\Gamma}_u^C \quad (3.79)$$

Note that Eq. (3.77) is the dynamics model used for both the control design purpose and for the simulation of the actual motion of both spacecraft in orbit. A high-level MATLAB Simulink block diagram that highlights the connection between various signals and processes is shown in Fig. 3.4.

Figure 3.4: High-level block diagram showing signals and processes

Chapter 4

Spatial Stand-off Pose Tracking

Based on the prior work in literature, the previous chapters build a strong mathematical foundation that is relevant to the pose tracking control formulation. From this chapter onwards, the derivation, simulation study and validation of the proposed pose tracking controller is presented. This chapter focuses on developing the control method that can be used to capture a specific point of interest on the target frame of reference that is undergoing a general translating and tumbling motion. Detailed derivation of the control constraints are provided and the resulting control force is applied onto the full six degree-of-freedom model of the chaser. Spatial simulation studies of both the chaser and target motion highlight the operational characteristics of the Udwadia-Kalaba equation-based pose tracking control.

4.1 Spatial Problem Formulation

The problem of stand-off pose tracking is formulated and solved in the inertial ECI frame using the Udwadia-Kalaba framework. To begin, variables and some transformations that are useful to formulate the spatial problem are defined.

Figure 4.1: Chaser and target frame in the inertial ECI frame.

The red chaser frame and the black target frame are located in the inertial ECI frame of reference as shown in Fig. 4.1. The position vector of the target frame origin is $\mathbf{r}_t = [X_t \ Y_t \ Z_t]^T$ and its orientation with respect to the inertial ECI frame is described by the quaternion vector $\mathbf{u}_t = [u_{0,t} \ u_{1,t} \ u_{2,t} \ u_{3,t}]^T$. A point of interest on the target has a position vector $\mathbf{p}_t = [p_{x,t} \ p_{y,t} \ p_{z,t}]^T$ that is fixed with respect to the target body-fixed frame of reference origin. In a similar way, the chaser frame's origin is located at $\mathbf{r}_c = [X_c \ Y_c \ Z_c]^T$ with respect to the inertial frame and its orientation is described by the quaternion vector $\mathbf{u}_c = [u_{0,c} \ u_{1,c} \ u_{2,c} \ u_{3,c}]^T$. A vector $\mathbf{p}_t = [p_{x,c} \ p_{y,c} \ p_{z,c}]^T$ is defined as the alignment vector on the chaser frame, the utility of which is to maintain the alignment of the field-of-view of computer vision sensors towards the target. The rotation matrix that transforms the components of a vector from the target frame to the inertial frame is represented by $\mathbf{R}_{IT}$ while the transformation from the chaser to the inertial frame is given by $\mathbf{R}_{IC}$. Using Eq. (3.14), $\mathbf{R}_{IT}$ and $\mathbf{R}_{IC}$ are written in terms of the quaternion components as

$$\mathbf{R}_{IT} = \begin{bmatrix} 2u_{0,t}^2 - 1 + 2u_{1,t}^2 & 2u_{1,t}u_{2,t} - 2u_{0,t}u_{3,t} & 2u_{1,t}u_{3,t} + 2u_{0,t}u_{2,t} \\ 2u_{1,t}u_{2,t} + 2u_{0,t}u_{3,t} & 2u_{0,t}^2 - 1 + 2u_{2,t}^2 & 2u_{2,t}u_{3,t} - 2u_{0,t}u_{1,t} \\ 2u_{1,t}u_{3,t} - 2u_{0,t}u_{2,t} & 2u_{2,t}u_{3,t} + 2u_{0,t}u_{1,t} & 2u_{0,t}^2 - 1 + 2u_{3,t}^2 \end{bmatrix} \quad (4.1)$$

$$\mathbf{R}_{IC} = \begin{bmatrix} 2u_{0,c}^2 - 1 + 2u_{1,c}^2 & 2u_{1,c}u_{2,c} - 2u_{0,c}u_{3,c} & 2u_{1,c}u_{3,c} + 2u_{0,c}u_{2,c} \\ 2u_{1,c}u_{2,c} + 2u_{0,c}u_{3,c} & 2u_{0,c}^2 - 1 + 2u_{2,c}^2 & 2u_{2,c}u_{3,c} - 2u_{0,c}u_{1,c} \\ 2u_{1,c}u_{3,c} - 2u_{0,c}u_{2,c} & 2u_{2,c}u_{3,c} + 2u_{0,c}u_{1,c} & 2u_{0,c}^2 - 1 + 2u_{3,c}^2 \end{bmatrix} \quad (4.2)$$

The pose tracking objective is to simultaneously position the origin of the chaser's body-fixed reference frame at the specified stand-off separation L_0 with respect to the target origin while inertially aligning $\mathbf{p}_c$ along the target point of interest vector $\mathbf{p}_t$.

4.2 Position Control Constraint

The desired final configuration of the chaser is shown by the dashed red frame in Fig. 4.2. The control has to act in order to maneuver the chaser through one of the possible trajectories to the final configuration. Note that in the final location, origin

of the chaser frame is at a position $\bar{\mathbf{p}}_t$ from the target origin.

Figure 4.2: Desired position configuration of the chaser.

This vector is defined to be in the same direction as $\mathbf{p}_t$, with a norm corresponding to the stand-off distance L_0 , that is

$$\bar{\mathbf{p}}_t = L_0 \cdot \frac{\mathbf{p}_t}{\|\mathbf{p}_t\|} \quad (4.3)$$

In order to build the control equations, the constraint function needs to be defined from the desired final configuration. The control acts to align the chaser position vector $\mathbf{r}_c$ at the tip of vector $\bar{\mathbf{p}}_t$. The final position configuration is described by

$$\mathbf{r}_c = \mathbf{r}_t + \mathbf{R}_{IT} \left(L_0 \cdot \frac{\mathbf{p}_t}{\|\mathbf{p}_t\|} \right) \quad (4.4)$$

By replacing $L_0 \cdot \frac{\mathbf{p}_t}{\|\mathbf{p}_t\|}$ with the vector $\bar{\mathbf{p}}_t = \begin{bmatrix} \bar{p}_{x,t} & \bar{p}_{y,t} & \bar{p}_{z,t} \end{bmatrix}^T$, Eq. 4.4 becomes

$$\mathbf{r}_c = \mathbf{r}_t + \mathbf{R}_{IT} \bar{\mathbf{p}}_t \quad (4.5)$$

The position constraint function $\Phi_{\mathbf{r}}$ is defined as

$$\Phi_{\mathbf{r}} = \mathbf{r}_t + \mathbf{R}_{IT}\bar{\mathbf{p}}_t - \mathbf{r}_c \quad (4.6)$$

The first derivative of Eq. (4.6) is given by

$$\dot{\Phi}_{\mathbf{r}} = \dot{\mathbf{r}}_t + \dot{\mathbf{R}}_{IT}\bar{\mathbf{p}}_t + \mathbf{R}_{IT}\dot{\bar{\mathbf{p}}}_t - \dot{\mathbf{r}}_c \quad (4.7)$$

Since $\bar{\mathbf{p}}_t$ is fixed with respect to the body-fixed target frame, its time variation in this frame, $\dot{\bar{\mathbf{p}}}_t = \mathbf{0}$. Note that, $\dot{\bar{\mathbf{p}}}_t$ is not the inertial derivative. Therefore, the Eq. (4.7) becomes,

$$\dot{\Phi}_{\mathbf{r}} = \dot{\mathbf{r}}_t + \dot{\mathbf{R}}_{IT}\bar{\mathbf{p}}_t - \dot{\mathbf{r}}_c \quad (4.8)$$

Now, the objective is to find the rotation matrix derivative $\dot{\mathbf{R}}_{IT}$ and express it in a form that is easier to code and implement in the controller. By taking the first derivative with respect to time, Eq. (4.1) yields

$$\dot{\mathbf{R}}_{IT} = \begin{bmatrix} 4u_{0,t}\dot{u}_{0,t} + 4u_{1,t}\dot{u}_{1,t} & 2u_{1,t}\dot{u}_{2,t} + 2u_{2,t}\dot{u}_{1,t} & 2u_{1,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{1,t} \\ -2u_{0,t}\dot{u}_{3,t} - 2u_{3,t}\dot{u}_{0,t} & +2u_{0,t}\dot{u}_{2,t} + 2u_{2,t}\dot{u}_{0,t} & \\ 2u_{1,t}\dot{u}_{2,t} + 2u_{2,t}\dot{u}_{1,t} & 4u_{0,t}\dot{u}_{0,t} + 4u_{2,t}\dot{u}_{2,t} & 2u_{2,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{2,t} \\ +2u_{0,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{0,t} & -2u_{0,t}\dot{u}_{1,t} - 2u_{1,t}\dot{u}_{0,t} & \\ 2u_{1,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{1,t} & 2u_{2,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{2,t} & 4u_{0,t}\dot{u}_{0,t} + 4u_{3,t}\dot{u}_{3,t} \\ -2u_{0,t}\dot{u}_{2,t} - 2u_{2,t}\dot{u}_{0,t} & +2u_{0,t}\dot{u}_{1,t} + 2u_{1,t}\dot{u}_{0,t} & \end{bmatrix} \quad (4.9)$$

As it would be seen, Eq. (4.9) is useful in deriving the quaternion double-derivative, which is necessary to write the acceleration constraint equation. However, this expression is quite large and cumbersome from the perspective of writing the controller codes. It is also prone to coding errors and results in bugs. Note that for the acceleration constraint, a derivative of Eq. (4.9) with respect to time is required, which would result in a more complicated form. In order to enhance its readability and avoid the potential bugs, this matrix has to be expressed in an alternate form using

a set of column vectors $\dot{\mathbf{R}}_{1,IT}$, $\dot{\mathbf{R}}_{2,IT}$ and $\dot{\mathbf{R}}_{3,IT}$ as

$$\dot{\mathbf{R}}_{IT} = \begin{bmatrix} \dot{\mathbf{R}}_{1,IT} & \dot{\mathbf{R}}_{2,IT} & \dot{\mathbf{R}}_{3,IT} \end{bmatrix} \quad (4.10)$$

where the column vectors are defined and expressed in a simpler form in the subsequent derivation.

The first column vector $\dot{\mathbf{R}}_{1,IT}$ can be expressed as

$$\begin{aligned} \dot{\mathbf{R}}_{1,IT} &= \begin{bmatrix} 4u_{0,t}\dot{u}_{0,t} + 4u_{1,t}\dot{u}_{1,t} \\ 2u_{1,t}\dot{u}_{2,t} + 2u_{2,t}\dot{u}_{1,t} \\ + 2u_{0,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{0,t} \\ 2u_{1,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{1,t} \\ - 2u_{0,t}\dot{u}_{2,t} - 2u_{2,t}\dot{u}_{0,t} \end{bmatrix} \\ &= \begin{bmatrix} 4u_{0,t} & 4u_{1,t} & 0 & 0 \\ 2u_{3,t} & 2u_{2,t} & 2u_{1,t} & 2u_{0,t} \\ -2u_{2,t} & 2u_{3,t} & -2u_{0,t} & 2u_{1,t} \end{bmatrix} \begin{bmatrix} \dot{u}_{0,t} \\ \dot{u}_{1,t} \\ \dot{u}_{2,t} \\ \dot{u}_{3,t} \end{bmatrix} \\ &= \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t \end{aligned} \quad (4.11)$$

where $\mathbf{L}_{1,IT}$ is given by

$$\mathbf{L}_{1,IT} = \begin{bmatrix} 4u_{0,t} & 4u_{1,t} & 0 & 0 \\ 2u_{3,t} & 2u_{2,t} & 2u_{1,t} & 2u_{0,t} \\ -2u_{2,t} & 2u_{3,t} & -2u_{0,t} & 2u_{1,t} \end{bmatrix} \quad (4.12)$$

Following a similar approach, the second column vector is written as

$$\begin{aligned}
\dot{\mathbf{R}}_{2,IT} &= \begin{bmatrix} 2u_{1,t}\dot{u}_{2,t} + 2u_{2,t}\dot{u}_{1,t} \\ -2u_{0,t}\dot{u}_{3,t} - 2u_{3,t}\dot{u}_{0,t} \\ 4u_{0,t}\dot{u}_{0,t} + 4u_{2,t}\dot{u}_{2,t} \\ 2u_{2,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{2,t} \\ +2u_{0,t}\dot{u}_{1,t} + 2u_{1,t}\dot{u}_{0,t} \end{bmatrix} \\
&= \begin{bmatrix} -2u_{3,t} & 2u_{2,t} & 2u_{1,t} & -2u_{0,t} \\ 4u_{0,t} & 0 & 4u_{2,t} & 0 \\ 2u_{1,t} & 2u_{0,t} & 2u_{3,t} & 2u_{2,t} \end{bmatrix} \begin{bmatrix} \dot{u}_{0,t} \\ \dot{u}_{1,t} \\ \dot{u}_{2,t} \\ \dot{u}_{3,t} \end{bmatrix} \\
&= \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t
\end{aligned} \tag{4.13}$$

where $\mathbf{L}_{2,IT}$ is given as

$$\mathbf{L}_{2,IT} = \begin{bmatrix} -2u_{3,t} & 2u_{2,t} & 2u_{1,t} & -2u_{0,t} \\ 4u_{0,t} & 0 & 4u_{2,t} & 0 \\ 2u_{1,t} & 2u_{0,t} & 2u_{3,t} & 2u_{2,t} \end{bmatrix} \tag{4.14}$$

Finally, the column matrix $\dot{\mathbf{R}}_{3,IT}$ is given by

$$\begin{aligned}
\dot{\mathbf{R}}_{3,IT} &= \begin{bmatrix} 2u_{1,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{1,t} \\ +2u_{0,t}\dot{u}_{2,t} + 2u_{2,t}\dot{u}_{0,t} \\ 2u_{2,t}\dot{u}_{3,t} + 2u_{3,t}\dot{u}_{2,t} \\ -2u_{0,t}\dot{u}_{1,t} - 2u_{1,t}\dot{u}_{0,t} \\ 4u_{0,t}\dot{u}_{0,t} + 4u_{3,t}\dot{u}_{3,t} \end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
&= \begin{bmatrix} 2u_{2,t} & 2u_{3,t} & 2u_{0,t} & 2u_{1,t} \\ -2u_{1,t} & -2u_{0,t} & 2u_{3,t} & 2u_{2,t} \\ 4u_{0,t} & 0 & 0 & 4u_{3,t} \end{bmatrix} \begin{bmatrix} \dot{u}_{0,t} \\ \dot{u}_{1,t} \\ \dot{u}_{2,t} \\ \dot{u}_{3,t} \end{bmatrix} \\
&= \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t
\end{aligned} \tag{4.15}$$

where the matrix $\mathbf{L}_{3,IT}$ is defined as

$$\mathbf{L}_{3,IT} = \begin{bmatrix} 2u_{2,t} & 2u_{3,t} & 2u_{0,t} & 2u_{1,t} \\ -2u_{1,t} & -2u_{0,t} & 2u_{3,t} & 2u_{2,t} \\ 4u_{0,t} & 0 & 0 & 4u_{3,t} \end{bmatrix} \tag{4.16}$$

Using Eq. (4.11), (4.13) and (4.15) in Eq. (4.10),

$$\dot{\mathbf{R}}_{IT} = \begin{bmatrix} \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t & \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t & \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t \end{bmatrix} \tag{4.17}$$

Substituting Eq. (4.17) into Eq. (4.8),

$$\begin{aligned}
\dot{\Phi}_{\mathbf{r}} &= \dot{\mathbf{r}}_t + \begin{bmatrix} \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t & \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t & \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t \end{bmatrix} \bar{\mathbf{p}}_t - \dot{\mathbf{r}}_c \\
&= \dot{\mathbf{r}}_t + \begin{bmatrix} \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t & \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t & \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t \end{bmatrix} \begin{bmatrix} \bar{p}_{x,t} \\ \bar{p}_{y,t} \\ \bar{p}_{z,t} \end{bmatrix} - \dot{\mathbf{r}}_c \\
&= \dot{\mathbf{r}}_t + \bar{p}_{x,t} \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t + \bar{p}_{y,t} \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t + \bar{p}_{z,t} \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t - \dot{\mathbf{r}}_c
\end{aligned} \tag{4.18}$$

The time derivative of $\dot{\Phi}_{\mathbf{r}}$ results in

$$\begin{aligned}
\ddot{\Phi}_{\mathbf{r}} &= \ddot{\mathbf{r}}_t + \bar{p}_{x,t} (\mathbf{L}_{1,IT} \ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{1,IT} \dot{\mathbf{u}}_t) \\
&\quad + \bar{p}_{y,t} (\mathbf{L}_{2,IT} \ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{2,IT} \dot{\mathbf{u}}_t) \\
&\quad + \bar{p}_{z,t} (\mathbf{L}_{3,IT} \ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{3,IT} \dot{\mathbf{u}}_t) - \ddot{\mathbf{r}}_c \\
&= \ddot{\mathbf{r}}_t + (\bar{p}_{x,t} \mathbf{L}_{1,IT} + \bar{p}_{y,t} \mathbf{L}_{2,IT} + \bar{p}_{z,t} \mathbf{L}_{3,IT}) \ddot{\mathbf{u}}_t \\
&\quad + (\bar{p}_{x,t} \dot{\mathbf{L}}_{1,IT} + \bar{p}_{y,t} \dot{\mathbf{L}}_{2,IT} + \bar{p}_{z,t} \dot{\mathbf{L}}_{3,IT}) \dot{\mathbf{u}}_t - \ddot{\mathbf{r}}_c
\end{aligned} \tag{4.19}$$

where

$$\dot{\mathbf{L}}_{1,IT} = \begin{bmatrix} 4\dot{u}_{0,t} & 4\dot{u}_{1,t} & 0 & 0 \\ 2\dot{u}_{3,t} & 2\dot{u}_{2,t} & 2\dot{u}_{1,t} & 2\dot{u}_{0,t} \\ -2\dot{u}_{2,t} & 2\dot{u}_{3,t} & -2\dot{u}_{0,t} & 2\dot{u}_{1,t} \end{bmatrix} \quad (4.20)$$

$$\dot{\mathbf{L}}_{2,IT} = \begin{bmatrix} -2\dot{u}_{3,t} & 2\dot{u}_{2,t} & 2\dot{u}_{1,t} & -2\dot{u}_{0,t} \\ 4\dot{u}_{0,t} & 0 & 4\dot{u}_{2,t} & 0 \\ 2\dot{u}_{1,t} & 2\dot{u}_{0,t} & 2\dot{u}_{3,t} & 2\dot{u}_{2,t} \end{bmatrix} \quad (4.21)$$

$$\dot{\mathbf{L}}_{3,IT} = \begin{bmatrix} 2\dot{u}_{2,t} & 2\dot{u}_{3,t} & 2\dot{u}_{0,t} & 2\dot{u}_{1,t} \\ -2\dot{u}_{1,t} & -2\dot{u}_{0,t} & 2\dot{u}_{3,t} & 2\dot{u}_{2,t} \\ 4\dot{u}_{0,t} & 0 & 0 & 4\dot{u}_{3,t} \end{bmatrix} \quad (4.22)$$

Combining the position, velocity and acceleration constraints, a second order differential form is obtained as

$$\ddot{\Phi}_{\mathbf{r}} + \alpha_{\mathbf{r}} \dot{\Phi}_{\mathbf{r}} + \gamma_{\mathbf{r}} \Phi_{\mathbf{r}} = \mathbf{0} \quad (4.23)$$

where $\alpha_{\mathbf{r}} = \text{diag}(\begin{bmatrix} \alpha_x & \alpha_y & \alpha_z \end{bmatrix})$ and $\gamma_{\mathbf{r}} = \text{diag}(\begin{bmatrix} \gamma_x & \gamma_y & \gamma_z \end{bmatrix})$ are positive definite control gain diagonal matrices. Using Eqs. (4.6), (4.18) and (4.19) in Eq. (4.23) results in

$$\begin{aligned} \ddot{\mathbf{r}}_t + (\bar{p}_{x,t} \mathbf{L}_{1,IT} + \bar{p}_{y,t} \mathbf{L}_{2,IT} + \bar{p}_{z,t} \mathbf{L}_{3,IT}) \ddot{\mathbf{u}}_t + (\bar{p}_{x,t} \dot{\mathbf{L}}_{1,IT} + \bar{p}_{y,t} \dot{\mathbf{L}}_{2,IT} + \bar{p}_{z,t} \dot{\mathbf{L}}_{3,IT}) \dot{\mathbf{u}}_t - \ddot{\mathbf{r}}_c \\ + \alpha_{\mathbf{r}} (\dot{\mathbf{r}}_t + \bar{p}_{x,t} \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t + \bar{p}_{y,t} \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t + \bar{p}_{z,t} \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t - \dot{\mathbf{r}}_c) + \gamma_{\mathbf{r}} (\mathbf{r}_t + \mathbf{R}_{IT} \bar{\mathbf{p}}_t - \mathbf{r}_c) = \mathbf{0} \end{aligned} \quad (4.24)$$

Rearranging Eq. (4.24), $\ddot{\mathbf{r}}_c$ is given by

$$\begin{aligned} \ddot{\mathbf{r}}_c = \ddot{\mathbf{r}}_t + (\bar{p}_{x,t} \mathbf{L}_{1,IT} + \bar{p}_{y,t} \mathbf{L}_{2,IT} + \bar{p}_{z,t} \mathbf{L}_{3,IT}) \ddot{\mathbf{u}}_t + (\bar{p}_{x,t} \dot{\mathbf{L}}_{1,IT} + \bar{p}_{y,t} \dot{\mathbf{L}}_{2,IT} + \bar{p}_{z,t} \dot{\mathbf{L}}_{3,IT}) \dot{\mathbf{u}}_t \\ + \alpha_{\mathbf{r}} (\dot{\mathbf{r}}_t + \bar{p}_{x,t} \mathbf{L}_{1,IT} \dot{\mathbf{u}}_t + \bar{p}_{y,t} \mathbf{L}_{2,IT} \dot{\mathbf{u}}_t + \bar{p}_{z,t} \mathbf{L}_{3,IT} \dot{\mathbf{u}}_t - \dot{\mathbf{r}}_c) + \gamma_{\mathbf{r}} (\mathbf{r}_t + \mathbf{R}_{IT} \bar{\mathbf{p}}_t - \mathbf{r}_c) \end{aligned} \quad (4.25)$$

The right hand side of Eq. (4.25) is a column vector. Let us define $\mathbf{b}_{\mathbf{r}}$ as

$$\mathbf{b}_{\mathbf{r}} = \ddot{\mathbf{r}}_t + (\bar{p}_{x,t} \mathbf{L}_{1,IT} + \bar{p}_{y,t} \mathbf{L}_{2,IT} + \bar{p}_{z,t} \mathbf{L}_{3,IT}) \ddot{\mathbf{u}}_t + (\bar{p}_{x,t} \dot{\mathbf{L}}_{1,IT} + \bar{p}_{y,t} \dot{\mathbf{L}}_{2,IT} + \bar{p}_{z,t} \dot{\mathbf{L}}_{3,IT}) \dot{\mathbf{u}}_t$$

$$+\boldsymbol{\alpha}_r(\dot{\mathbf{r}}_t + \bar{p}_{x,t}\mathbf{L}_{1,IT}\dot{\mathbf{u}}_t + \bar{p}_{y,t}\mathbf{L}_{2,IT}\dot{\mathbf{u}}_t + \bar{p}_{z,t}\mathbf{L}_{3,IT}\dot{\mathbf{u}}_t - \dot{\mathbf{r}}_c) + \boldsymbol{\gamma}_r(\mathbf{r}_t + \mathbf{R}_{IT}\bar{\mathbf{p}}_t - \mathbf{r}_c) \quad (4.26)$$

Then, using Eq. (4.26) in Eq. (4.25),

$$\begin{aligned} \ddot{\mathbf{r}}_c &= \mathbf{b}_r \\ \implies \mathbf{A}_r \ddot{\mathbf{r}}_c &= \mathbf{b}_r \end{aligned} \quad (4.27)$$

where $\mathbf{A}_r$ is a 3×3 identity matrix. Eq. (4.27) is the acceleration control constraint for the chaser stand-off distance maintenance along the target point of interest vector, as observed from the inertial frame.

4.3 Orientation Control Constraint

The final orientation of the chaser body-fixed reference frame with respect to that of the target should be such that the point of interest vector $\mathbf{p}_t$ must align along the alignment vector $\mathbf{p}_c$, as observed from the inertial ECI frame. The unit vector along the direction of $\mathbf{p}_t$ is given by

$$\hat{\mathbf{p}}_t = \frac{\mathbf{p}_t}{\|\mathbf{p}_t\|} \quad (4.28)$$

Note that $\hat{\mathbf{p}}_t$ is defined with respect to the target body-fixed reference frame. Similarly, the alignment unit vector $\hat{\mathbf{p}}_c$ is defined in the chaser frame of reference as

$$\hat{\mathbf{p}}_c = \frac{\mathbf{p}_c}{\|\mathbf{p}_c\|} \quad (4.29)$$

The control objective is to inertially align $\hat{\mathbf{p}}_c$ along $\hat{\mathbf{p}}_t$ as observed from Fig. 4.3.

In the following discussion, the orientation control constraint would be formulated and the orienting acceleration constraint would be derived. In the desired final configuration of the frame orientation, the following equation is satisfied:

$$\mathbf{R}_{IT}\hat{\mathbf{p}}_t = \mathbf{R}_{IC}\hat{\mathbf{p}}_c \quad (4.30)$$

Figure 4.3: Desired orientation of the chaser.

where $\mathbf{R}_{IT}$, $\mathbf{R}_{IC}$, $\hat{\mathbf{p}}_t$ and $\hat{\mathbf{p}}_c$ are given by Eq. (4.1), (4.2), (4.28) and (4.29), respectively. The orientation constraint function which is based on the quaternions $\mathbf{u}_t$ and $\mathbf{u}_c$ can hence be written as

$$\Phi_{\mathbf{u}} = \mathbf{R}_{IT}\hat{\mathbf{p}}_t - \mathbf{R}_{IC}\hat{\mathbf{p}}_c \quad (4.31)$$

The first derivative of the orientation constraint of Eq. (4.31) with respect to time is given by

$$\dot{\Phi}_{\mathbf{u}} = \dot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t + \mathbf{R}_{IT}\dot{\hat{\mathbf{p}}}_t - \dot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c - \mathbf{R}_{IC}\dot{\hat{\mathbf{p}}}_c \quad (4.32)$$

Since the unit vectors along the direction of the point of interest and alignment vectors in their respective body-fixed frames are fixed, $\dot{\hat{\mathbf{p}}}_t = \dot{\hat{\mathbf{p}}}_c = \mathbf{0}$. As before, note that $\dot{\hat{\mathbf{p}}}_t$ and $\dot{\hat{\mathbf{p}}}_c$ are not the inertial derivatives. Therefore, Eq. (4.32) is reduced to

$$\dot{\Phi}_{\mathbf{u}} = \dot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c \quad (4.33)$$

The rotation matrix derivative $\dot{\mathbf{R}}_{IT}$ is given by Eq. (4.17) whereas $\dot{\mathbf{R}}_{IC}$ is given by

$$\dot{\mathbf{R}}_{IC} = \begin{bmatrix} \mathbf{L}_{1,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{2,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{3,IC}\dot{\mathbf{u}}_c \end{bmatrix} \quad (4.34)$$

In Eq. (4.34), $\dot{\mathbf{u}}_c$ denotes the chaser's quaternion derivative while $\mathbf{L}_{1,IC}$, $\mathbf{L}_{2,IC}$ and $\mathbf{L}_{3,IC}$ are functions of $\mathbf{u}_c = \begin{bmatrix} u_{0,c} & u_{1,c} & u_{2,c} & u_{3,c} \end{bmatrix}^T$ and are given as

$$\mathbf{L}_{1,IC} = \begin{bmatrix} 4u_{0,c} & 4u_{1,c} & 0 & 0 \\ 2u_{3,c} & 2u_{2,c} & 2u_{1,c} & 2u_{0,c} \\ -2u_{2,c} & 2u_{3,c} & -2u_{0,c} & 2u_{1,c} \end{bmatrix} \quad (4.35)$$

$$\mathbf{L}_{2,IC} = \begin{bmatrix} -2u_{3,c} & 2u_{2,c} & 2u_{1,c} & -2u_{0,c} \\ 4u_{0,c} & 0 & 4u_{2,c} & 0 \\ 2u_{1,c} & 2u_{0,c} & 2u_{3,c} & 2u_{2,c} \end{bmatrix} \quad (4.36)$$

$$\mathbf{L}_{3,IC} = \begin{bmatrix} 2u_{2,c} & 2u_{3,c} & 2u_{0,c} & 2u_{1,c} \\ -2u_{1,c} & -2u_{0,c} & 2u_{3,c} & 2u_{2,c} \\ 4u_{0,c} & 0 & 0 & 4u_{3,c} \end{bmatrix} \quad (4.37)$$

Now, the second derivative with respect to time of Eq. (4.31) is given by

$$\ddot{\Phi}_{\mathbf{u}} = \ddot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t + \dot{\mathbf{R}}_{IT}\dot{\hat{\mathbf{p}}}_t - \ddot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c - \dot{\mathbf{R}}_{IC}\dot{\hat{\mathbf{p}}}_c \quad (4.38)$$

As the point of interest vector and alignment vector are fixed in their respective body frames, their derivatives become zero and Eq. (4.32) is simplified to

$$\ddot{\Phi}_{\mathbf{u}} = \ddot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \ddot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c \quad (4.39)$$

The double derivative of the rotation matrix $\mathbf{R}_{IT}$ in Eq. (4.39) is given by

$$\ddot{\mathbf{R}}_{IT} = \begin{bmatrix} \mathbf{L}_{1,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{1,IT}\dot{\mathbf{u}}_t & \mathbf{L}_{2,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{2,IT}\dot{\mathbf{u}}_t & \mathbf{L}_{3,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{3,IT}\dot{\mathbf{u}}_t \end{bmatrix} \quad (4.40)$$

where $\dot{\mathbf{L}}_{1,IT}$, $\dot{\mathbf{L}}_{2,IT}$ and $\dot{\mathbf{L}}_{3,IT}$ are given by Eq. (4.20), (4.21) and (4.22), respectively. Further, $\ddot{\mathbf{R}}_{IC}$ from Eq. (4.39) is obtained as

$$\ddot{\mathbf{R}}_{IC} = \begin{bmatrix} \mathbf{L}_{1,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{1,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{2,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{2,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{3,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{3,IC}\dot{\mathbf{u}}_c \end{bmatrix} \quad (4.41)$$

where $\dot{\mathbf{L}}_{1,IC}$, $\dot{\mathbf{L}}_{2,IC}$ and $\dot{\mathbf{L}}_{3,IC}$ are given by

$$\dot{\mathbf{L}}_{1,IC} = \begin{bmatrix} 4\dot{u}_{0,c} & 4\dot{u}_{1,c} & 0 & 0 \\ 2\dot{u}_{3,c} & 2\dot{u}_{2,c} & 2\dot{u}_{1,c} & 2\dot{u}_{0,c} \\ -2\dot{u}_{2,c} & 2\dot{u}_{3,c} & -2\dot{u}_{0,c} & 2\dot{u}_{1,c} \end{bmatrix} \quad (4.42)$$

$$\dot{\mathbf{L}}_{2,IC} = \begin{bmatrix} -2\dot{u}_{3,c} & 2\dot{u}_{2,c} & 2\dot{u}_{1,c} & -2\dot{u}_{0,c} \\ 4\dot{u}_{0,c} & 0 & 4\dot{u}_{2,c} & 0 \\ 2\dot{u}_{1,c} & 2\dot{u}_{0,c} & 2\dot{u}_{3,c} & 2\dot{u}_{2,c} \end{bmatrix} \quad (4.43)$$

$$\dot{\mathbf{L}}_{3,IC} = \begin{bmatrix} 2\dot{u}_{2,c} & 2\dot{u}_{3,c} & 2\dot{u}_{0,c} & 2\dot{u}_{1,c} \\ -2\dot{u}_{1,c} & -2\dot{u}_{0,c} & 2\dot{u}_{3,c} & 2\dot{u}_{2,c} \\ 4\dot{u}_{0,c} & 0 & 0 & 4\dot{u}_{3,c} \end{bmatrix} \quad (4.44)$$

Combining Eqs. (4.31), (4.33) and (4.39) into a second order differential form,

$$\ddot{\Phi}_{\mathbf{u}} + \alpha_{\mathbf{u}}\dot{\Phi}_{\mathbf{u}} + \gamma_{\mathbf{u}}\Phi_{\mathbf{u}} = \mathbf{0} \quad (4.45)$$

Substituting Eqs. (4.31), (4.33) and (4.39) into Eq. (4.45) results in

$$\ddot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \ddot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c + \alpha_{\mathbf{u}}(\dot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c) + \gamma_{\mathbf{u}}(\mathbf{R}_{IT}\hat{\mathbf{p}}_t - \mathbf{R}_{IC}\hat{\mathbf{p}}_c) = \mathbf{0} \quad (4.46)$$

Substituting for $\ddot{\mathbf{R}}_{IT}$ and $\ddot{\mathbf{R}}_{IC}$ from Eq. (4.40) and (4.41), respectively, into Eq. (4.46), the equation is rearranged to keep the quaternion double-derivative of the chaser frame on the left hand side as

$$\begin{aligned} & \begin{bmatrix} \mathbf{L}_{1,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{1,IT}\dot{\mathbf{u}}_t & \mathbf{L}_{2,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{2,IT}\dot{\mathbf{u}}_t & \mathbf{L}_{3,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{3,IT}\dot{\mathbf{u}}_t \end{bmatrix} \hat{\mathbf{p}}_t \\ & - \begin{bmatrix} \mathbf{L}_{1,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{1,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{2,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{2,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{3,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{3,IC}\dot{\mathbf{u}}_c \end{bmatrix} \hat{\mathbf{p}}_c \\ & + \alpha_{\mathbf{u}}(\dot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c) + \gamma_{\mathbf{u}}(\mathbf{R}_{IT}\hat{\mathbf{p}}_t - \mathbf{R}_{IC}\hat{\mathbf{p}}_c) = \mathbf{0} \\ \Rightarrow & \begin{bmatrix} \mathbf{L}_{1,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{1,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{2,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{2,IC}\dot{\mathbf{u}}_c & \mathbf{L}_{3,IC}\ddot{\mathbf{u}}_c + \dot{\mathbf{L}}_{3,IC}\dot{\mathbf{u}}_c \end{bmatrix} \hat{\mathbf{p}}_c = \\ & \begin{bmatrix} \mathbf{L}_{1,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{1,IT}\dot{\mathbf{u}}_t & \mathbf{L}_{2,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{2,IT}\dot{\mathbf{u}}_t & \mathbf{L}_{3,IT}\ddot{\mathbf{u}}_t + \dot{\mathbf{L}}_{3,IT}\dot{\mathbf{u}}_t \end{bmatrix} \hat{\mathbf{p}}_t \\ & + \alpha_{\mathbf{u}}(\dot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c) + \gamma_{\mathbf{u}}(\mathbf{R}_{IT}\hat{\mathbf{p}}_t - \mathbf{R}_{IC}\hat{\mathbf{p}}_c) \\ \Rightarrow & \begin{bmatrix} \mathbf{L}_{1,IC}\ddot{\mathbf{u}}_c & \mathbf{L}_{2,IC}\ddot{\mathbf{u}}_c & \mathbf{L}_{3,IC}\ddot{\mathbf{u}}_c \end{bmatrix} \hat{\mathbf{p}}_c + \begin{bmatrix} \dot{\mathbf{L}}_{1,IC}\dot{\mathbf{u}}_c & \dot{\mathbf{L}}_{2,IC}\dot{\mathbf{u}}_c & \dot{\mathbf{L}}_{3,IC}\dot{\mathbf{u}}_c \end{bmatrix} \hat{\mathbf{p}}_c = \end{aligned}$$

$$\begin{aligned}
& \left[\mathbf{L}_{1,IT} \ddot{\mathbf{u}}_t \quad \mathbf{L}_{2,IT} \ddot{\mathbf{u}}_t \quad \mathbf{L}_{3,IT} \ddot{\mathbf{u}}_t \right] \hat{\mathbf{p}}_t + \left[\dot{\mathbf{L}}_{1,IT} \dot{\mathbf{u}}_t \quad \dot{\mathbf{L}}_{2,IT} \dot{\mathbf{u}}_t \quad \dot{\mathbf{L}}_{3,IT} \dot{\mathbf{u}}_t \right] \hat{\mathbf{p}}_t \\
& + \alpha_u (\dot{\mathbf{R}}_{IT} \hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC} \hat{\mathbf{p}}_c) + \gamma_u (\mathbf{R}_{IT} \hat{\mathbf{p}}_t - \mathbf{R}_{IC} \hat{\mathbf{p}}_c)
\end{aligned} \tag{4.47}$$

Suppose the body-frame components of the point of interest vector and the alignment vector are $\hat{\mathbf{p}}_t = [\hat{p}_{x,t} \quad \hat{p}_{y,t} \quad \hat{p}_{z,t}]^T$ and $\hat{\mathbf{p}}_c = [\hat{p}_{x,c} \quad \hat{p}_{y,c} \quad \hat{p}_{z,c}]^T$, respectively. Substituting into Eq. (4.47),

$$\begin{aligned}
& \hat{p}_{x,c} \mathbf{L}_{1,IC} \ddot{\mathbf{u}}_c + \hat{p}_{y,c} \mathbf{L}_{2,IC} \ddot{\mathbf{u}}_c + \hat{p}_{z,c} \mathbf{L}_{3,IC} \ddot{\mathbf{u}}_c + \hat{p}_{x,c} \dot{\mathbf{L}}_{1,IC} \dot{\mathbf{u}}_c + \hat{p}_{y,c} \dot{\mathbf{L}}_{2,IC} \dot{\mathbf{u}}_c + \hat{p}_{z,c} \dot{\mathbf{L}}_{3,IC} \dot{\mathbf{u}}_c = \\
& \hat{p}_{x,t} \mathbf{L}_{1,IT} \ddot{\mathbf{u}}_t + \hat{p}_{y,t} \mathbf{L}_{2,IT} \ddot{\mathbf{u}}_t + \hat{p}_{z,t} \mathbf{L}_{3,IT} \ddot{\mathbf{u}}_t + \hat{p}_{x,t} \dot{\mathbf{L}}_{1,IT} \dot{\mathbf{u}}_t + \hat{p}_{y,t} \dot{\mathbf{L}}_{2,IT} \dot{\mathbf{u}}_t + \hat{p}_{z,t} \dot{\mathbf{L}}_{3,IT} \dot{\mathbf{u}}_t \\
& + \alpha_u (\dot{\mathbf{R}}_{IT} \hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC} \hat{\mathbf{p}}_c) + \gamma_u (\mathbf{R}_{IT} \hat{\mathbf{p}}_t - \mathbf{R}_{IC} \hat{\mathbf{p}}_c)
\end{aligned} \tag{4.48}$$

Now, Eq. (4.48) is rearranged to keep the term with chaser's quaternion double derivative to one side of the equation.

$$\begin{aligned}
\implies (\hat{p}_{x,c} \mathbf{L}_{1,IC} + \hat{p}_{y,c} \mathbf{L}_{2,IC} + \hat{p}_{z,c} \mathbf{L}_{3,IC}) \ddot{\mathbf{u}}_c &= (\hat{p}_{x,t} \mathbf{L}_{1,IT} + \hat{p}_{y,t} \mathbf{L}_{2,IT} + \hat{p}_{z,t} \mathbf{L}_{3,IT}) \ddot{\mathbf{u}}_t \\
&+ (\hat{p}_{x,t} \dot{\mathbf{L}}_{1,IT} + \hat{p}_{y,t} \dot{\mathbf{L}}_{2,IT} + \hat{p}_{z,t} \dot{\mathbf{L}}_{3,IT}) \dot{\mathbf{u}}_t \\
&- (\hat{p}_{x,c} \dot{\mathbf{L}}_{1,IC} + \hat{p}_{y,c} \dot{\mathbf{L}}_{2,IC} + \hat{p}_{z,c} \dot{\mathbf{L}}_{3,IC}) \dot{\mathbf{u}}_c \\
&+ \alpha_u (\dot{\mathbf{R}}_{IT} \hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC} \hat{\mathbf{p}}_c) \\
&+ \gamma_u (\mathbf{R}_{IT} \hat{\mathbf{p}}_t - \mathbf{R}_{IC} \hat{\mathbf{p}}_c)
\end{aligned} \tag{4.49}$$

Alternately, Eq. (4.49) can be written as

$$\mathbf{A}_u \ddot{\mathbf{u}}_c = \mathbf{b}_u \tag{4.50}$$

where

$$\mathbf{A}_u = \hat{p}_{x,c} \mathbf{L}_{1,IC} + \hat{p}_{y,c} \mathbf{L}_{2,IC} + \hat{p}_{z,c} \mathbf{L}_{3,IC} \tag{4.51}$$

$$\begin{aligned}
\mathbf{b}_u &= (\hat{p}_{x,t} \mathbf{L}_{1,IT} + \hat{p}_{y,t} \mathbf{L}_{2,IT} + \hat{p}_{z,t} \mathbf{L}_{3,IT}) \ddot{\mathbf{u}}_t \\
&+ (\hat{p}_{x,t} \dot{\mathbf{L}}_{1,IT} + \hat{p}_{y,t} \dot{\mathbf{L}}_{2,IT} + \hat{p}_{z,t} \dot{\mathbf{L}}_{3,IT}) \dot{\mathbf{u}}_t \\
&- (\hat{p}_{x,c} \dot{\mathbf{L}}_{1,IC} + \hat{p}_{y,c} \dot{\mathbf{L}}_{2,IC} + \hat{p}_{z,c} \dot{\mathbf{L}}_{3,IC}) \dot{\mathbf{u}}_c
\end{aligned}$$

$$\begin{aligned}
& +\alpha_{\mathbf{u}}(\dot{\mathbf{R}}_{IT}\hat{\mathbf{p}}_t - \dot{\mathbf{R}}_{IC}\hat{\mathbf{p}}_c) \\
& +\gamma_{\mathbf{u}}(\mathbf{R}_{IT}\hat{\mathbf{p}}_t - \mathbf{R}_{IC}\hat{\mathbf{p}}_c)
\end{aligned} \tag{4.52}$$

Eq. (4.50) is the quaternion constraint equation that can be used to generate control torque that can place the chaser's alignment vector along the target point of interest vector.

4.4 Generalized Control Force Generation

The position and orientation control constraints from Eq. (4.27) and (4.50) can be composed into a single matrix form as

$$\begin{bmatrix} \mathbf{A}_{\mathbf{r}} & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{4 \times 3} & \mathbf{A}_{\mathbf{u}} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{r}}_c \\ \ddot{\mathbf{u}}_c \end{bmatrix} = \begin{bmatrix} \mathbf{b}_{\mathbf{r}} \\ \mathbf{b}_{\mathbf{u}} \end{bmatrix} \tag{4.53}$$

$$\implies \mathbf{A}_{\mathbf{q}_c} \ddot{\mathbf{q}}_c = \mathbf{b}_{\mathbf{q}_c} \tag{4.54}$$

where $\mathbf{A}_{\mathbf{q}_c} = \begin{bmatrix} \mathbf{A}_{\mathbf{r}} & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{4 \times 3} & \mathbf{A}_{\mathbf{u}} \end{bmatrix}$, $\ddot{\mathbf{q}}_c = \begin{bmatrix} \ddot{\mathbf{r}}_c \\ \ddot{\mathbf{u}}_c \end{bmatrix}$ and $\mathbf{b}_{\mathbf{q}_c} = \begin{bmatrix} \mathbf{b}_{\mathbf{r}} \\ \mathbf{b}_{\mathbf{u}} \end{bmatrix}$

The generalized mass matrix of the chaser spacecraft is given by

$$\mathbf{M}_{\mathbf{q}_c} = \begin{bmatrix} m_c \mathbb{I}_{3 \times 3} & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{4 \times 3} & \mathbf{E}_c^T \mathbf{J}_c \mathbf{E}_c \end{bmatrix} \tag{4.55}$$

where m_c is the mass of the chaser, $\mathbf{E}_c$ is the transformation matrix relating quaternion derivatives to the body rates and $\mathbf{J}_c$ is the augmented inertia matrix. The term $\mathbf{E}_c$ is found by using Eq. (3.23) as

$$\mathbf{E}_c = \begin{bmatrix} u_{0,c} & u_{1,c} & u_{2,c} & u_{3,c} \\ -u_{1,c} & u_{0,c} & u_{3,c} & -u_{2,c} \\ -u_{2,c} & -u_{3,c} & u_{0,c} & u_{1,c} \\ -u_{3,c} & u_{2,c} & -u_{1,c} & u_{0,c} \end{bmatrix} \tag{4.56}$$

Using Eq. (3.37), $\mathbf{J}_c$ is obtained as

$$\mathbf{J}_c = \begin{bmatrix} J_{0,c} & 0 & 0 & 0 \\ 0 & J_{x,c} & 0 & 0 \\ 0 & 0 & J_{y,c} & 0 \\ 0 & 0 & 0 & J_{z,c} \end{bmatrix} \quad (4.57)$$

Suppose a given generalized force $\mathbf{Q}_c$ acts on the system, then, the unconstrained acceleration is

$$\ddot{\mathbf{q}}_{c,u} = \mathbf{M}_{\mathbf{q}_c}^{-1} \mathbf{Q}_c \quad (4.58)$$

When a control constraint of the form in Eq. (4.54) is applied on the system, resulting generalized force $\mathbf{Q}_{c,con}$ is given by Eq. (2.60) as

$$\mathbf{Q}_{c,con} = \mathbf{M}_{\mathbf{q}_c}^{\frac{1}{2}} (\mathbf{A}_{\mathbf{q}_c} \mathbf{M}_{\mathbf{q}_c}^{-\frac{1}{2}})^+ (\mathbf{b}_{\mathbf{q}_c} - \mathbf{A}_{\mathbf{q}_c} \ddot{\mathbf{q}}_{c,u}) \quad (4.59)$$

Note that, the two-body effects and perturbations may be incorporated into the generalized given force vector $\mathbf{Q}_c$ to build a controller that takes into account the orbital motion of the chaser. The target's orbital effects may be incorporated into $\mathbf{b}_r$ and $\mathbf{b}_u$ from Eq. (4.26) and (4.52), respectively.

4.5 Simulation Results

This section presents the simulation study for a general case of 6-DOF pose tracking in proximity operations. In the first case, the orbital effects are ignored and the simulation is performed in the inertial frame. The Udwadia-Kalaba equation is solved in the inertial reference frame that demonstrates pose tracking relative to a specific point of interest on the target spacecraft. Finally, the simulation of the pose tracking control considering a two-body motion in a J_2 perturbed Earth orbit is presented.

The target mass properties that are used for all the simulations are the same and is listed in Table 4.1. Corresponding chaser inertial parameters are given in Table 4.2. Note that the additional parameter J_0 required in the augmented inertia matrix

Table 4.1: Target Mass Properties

Parameter	Value
m_t	1 kg
$J_{x,t}$	1.3626 kg · m ²
$J_{y,t}$	1.5333 kg · m ²
$J_{z,t}$	0.3848 kg · m ²

Table 4.2: Chaser Mass Properties

Parameter	Value
m_c	1 kg
$J_{x,c}$	1.3626 kg · m ²
$J_{y,c}$	1.5333 kg · m ²
$J_{z,c}$	0.3848 kg · m ²

in the quaternion tumbling dynamics can be chosen as any positive value. In this work, the choices are $J_{0,c} = 1$ and $J_{0,t} = 1$. The complete six degree-of-freedom dynamics of Eq. (3.77) is used for the final case of on-orbit simulation with J_2 perturbation. Note that in the first two cases, the orbital effects are ignored while still a 6-DOF simulation is performed. Starting from the initial conditions, the controller has to position the chaser frame origin at a stand-off distance along the target point of interest vector, $\bar{\mathbf{p}}_t$, while placing the chaser alignment vector $\bar{\mathbf{p}}_c$ along $\bar{\mathbf{p}}_t$ as observed from the inertial frame. The desired point of interest to be captured on the target frame is arbitrarily chosen as

$$\bar{\mathbf{p}}_t = \begin{bmatrix} -1.5 & 0.7 & -1 \end{bmatrix}^T \text{ m} \quad (4.60)$$

Note that, the components of $\bar{\mathbf{p}}_t$ are defined with respect to the target frame and it's magnitude, $\|\bar{\mathbf{p}}_t\| = 1.9339$ m is the desired stand-off distance with respect to the target frame's origin. Alignment vector on the chaser frame is chosen as the x -axis of its body-fixed reference frame, that is

$$\bar{\mathbf{p}}_c = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T \text{ m} \quad (4.61)$$

Generalized control forces are given by the Eq. (4.59) in Sec. 4.4 as

$$\mathbf{Q}_{c,con} = \mathbf{M}_{\mathbf{q}_c}^{\frac{1}{2}} (\mathbf{A}_{\mathbf{q}_c} \mathbf{M}_{\mathbf{q}_c}^{-\frac{1}{2}})^+ (\mathbf{b}_{\mathbf{q}_c} - \mathbf{A}_{\mathbf{q}_c} \ddot{\mathbf{q}}_{c,u}) \quad (4.62)$$

Different control gains are chosen for each simulation and the details are provided in the appropriate sections.

4.5.1 Case 1: Pose Tracking in a Purely Inertial Frame

In this case, the target is assumed to be initially located at (2, 3, 4) m with respect to an inertial frame of reference. Initial velocity of the target is $(-0.1, -0.2, 0.2)$ m/s. The body-frame is rotated along the axis (0.2449, 0.4809, 0.8419) m with respect to the inertial frame by a clockwise angle of $+30^\circ$. Then, the corresponding initial quaternion vector of the target is obtained from Eq. (3.9) as

$$\mathbf{u}_{0,t} = \begin{bmatrix} 0.9659 & 0.0634 & 0.1245 & 0.2179 \end{bmatrix}^T \quad (4.63)$$

Let us also consider an initial body-fixed angular rate of (0.1, 0, 0) rad/s for the target frame. Using Eq. (3.24), the initial quaternion derivative of the target is

$$\dot{\mathbf{u}}_{0,t} = \begin{bmatrix} -0.0032 & 0.0483 & 0.0109 & -0.0062 \end{bmatrix}^T \quad (4.64)$$

Now, let us consider the origin of the chaser frame to be located at (1, 1, 1) m with a zero initial velocity relative to the inertial frame. The orientation of the chaser frame is initially the same as that of the inertial frame and the corresponding quaternion vector is

$$\mathbf{u}_{0,c} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T \quad (4.65)$$

The body-rates are assumed to be zero and hence, the initial quaternion derivative is zero as well.

$$\dot{\mathbf{u}}_{0,c} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \quad (4.66)$$

The control force from Eq. (4.59) that is used to attain the desired pose tracking objective was obtained by selecting the control gains α_r , γ_r , α_u and γ_u as,

$$\alpha_r = 0.1 \cdot \mathbb{I}_{3 \times 3} \quad \gamma_r = 0.05 \cdot \mathbb{I}_{3 \times 3} \quad \alpha_u = \mathbb{I}_{3 \times 3} \quad \gamma_u = 0.5 \cdot \mathbb{I}_{3 \times 3} \quad (4.67)$$

where $\mathbb{I}_{3 \times 3}$ is the identity matrix. Note that these gains were tuned by trial and error. Simulation is performed with a variable-step ode45 integrator in MATLAB R2018b Simulink. The simulation parameters of relative error tolerance is set at 10^{-13} while the absolute error tolerance is set at 10^{-16} .

Figure 4.4: Spatial pose tracking simulation showing the trajectory of the tip of the alignment and point of interest vectors.

Figure 4.4 presents the various stages in the pose tracking for a simulation period of 100 s. The blue frame is the inertial reference frame. The body-fixed black Cartesian coordinate is the target frame, while the red reference frame represents the chaser. Dotted lines represent the trajectory of the tip of $\bar{\mathbf{p}}_t$ and $\bar{\mathbf{p}}_c$. Starting from the above motioned initial conditions as shown in Fig. 4.4(a), the target frame undergoes a general tumbling motion. Meanwhile, the chaser alignment vector attempts to align along the target point of interest. In Fig. 4.4(i), it is seen that the chaser accurately tracks the target.

Figure 4.5: Case 1: Chaser and target trajectory.

The corresponding trajectory of the body-frame origins of the target and chaser are plotted in Fig. 4.5. As no external forces act on the target, its trajectory is a straight line. Note that, the motion is with respect to a pure inertial frame and no orbital effects are included. The corresponding stand-off distance as a function of time is presented in the top-left plot in Fig. 4.6. As observed from the plot, the stand-off distance settles onto the desired value of 1.9339 m. In order to know if the second objective of aligning the target is met, the components of the point of interest and alignment unit vectors expressed in the inertial frame are plotted in the top-right of Fig. 4.6.

Figure 4.6: Results for pose tracking in a purely inertial frame

Note that all components of the chaser alignment vector settle to the corresponding components of the point of interest vector. Hence, in the limiting case, the desired orientation is reached. The force applied on the chaser to control its position is shown in the bottom-left of Fig. 4.6. The physically-applied control torque is calculated from the generalized quaternion force vector using Eq. (3.79) and it is plotted as shown in the bottom-right of Fig. 4.6. By its fundamental nature, the Udwadia-Kalaba framework tends to minimize the control cost. The corresponding constraint errors as a function of time is presented in Fig. 4.7. Note that the orientation constraint error plot shows the evolution of inertial axes components. This is due to the fact that, the orientation constraint is defined based on the body-fixed position vectors in the target and chaser frame, expressed in their inertial components (see Eq. (4.30)).

Figure 4.7: Constraint errors for pose tracking in a purely inertial frame

4.5.2 Case 2: Pose Tracking on J_2 -Perturbed Orbits

The on-orbit simulation is performed with respect to the ECI frame. The effect of Earth's gravity forces and the oblateness that causes the J_2 perturbation (Sec. 3.3.2) on the body's motion is considered. The initial position and velocity of the target and chaser are determined based on the orbital elements discussed in Sec. 3.3.2. Tables 4.3 and 4.4 present the orbital elements of the target and chaser, respectively.

Table 4.3: Target Orbital Elements

Orbital Elements	Value
e	0.05
a	7500 km
i	30°
Ω	10°
ω	20°
θ_0	0°

Using Eq. (A.15) and (A.16), the classical orbital elements can be converted into the position and velocities of the bodies, respectively.

The initial positions and velocities are expressed in SI units and hence, all the orbital elements need to be converted to their appropriate units before calculation. The following initial position and velocity vectors are obtained from the orbital elements

Table 4.4: Chaser Orbital Elements

Orbital Elements	Value
e	0.06
a	7550 km
i	31°
Ω	10.5°
ω	20.6°
θ_0	-2°

in Tables 4.3 and 4.4.

$$\mathbf{r}_{0,t} = \begin{bmatrix} 6.22712 & 3.24098 & 1.21845 \end{bmatrix}^T \times 10^6 \text{ m} \quad (4.68)$$

$$\mathbf{v}_{0,t} = \begin{bmatrix} -3.66458 & 5.68722 & 3.60103 \end{bmatrix}^T \times 10^3 \text{ m/s} \quad (4.69)$$

$$\mathbf{r}_{0,c} = \begin{bmatrix} 6.26030 & 3.13372 & 1.16591 \end{bmatrix}^T \times 10^6 \text{ m} \quad (4.70)$$

$$\mathbf{v}_{0,c} = \begin{bmatrix} -3.57547 & 5.70795 & 3.76376 \end{bmatrix}^T \times 10^3 \text{ m/s} \quad (4.71)$$

Using Eq. (3.32), the period of the target orbit can be determined as $t = 6464.03$ s. The control gains are

$$\boldsymbol{\alpha}_{\mathbf{r}} = 2 \cdot \mathbb{I}_{3 \times 3} \quad \boldsymbol{\gamma}_{\mathbf{r}} = 0.5 \cdot \mathbb{I}_{3 \times 3} \quad \boldsymbol{\alpha}_{\mathbf{u}} = 2\mathbb{I}_{3 \times 3} \quad \boldsymbol{\gamma}_{\mathbf{u}} = 0.5 \cdot \mathbb{I}_{3 \times 3} \quad (4.72)$$

The simulation is run for one full-period t and the corresponding orbital trajectory of the tumbling target and the controlled chaser is plotted in Fig. 4.8.

The zoomed in portion of this graph shows that starting from a point away from the target, the chaser trajectory settles closer to that of the target. Further, the trajectory with five orbital periods is provided in Fig. 4.9, which shows a slight drift in the orbit due to the J_2 perturbation, as observed from the zoomed in portion of graph. These simulations were carried out using a fixed-step Runge-Kutta solver with a time-step of 0.1 s in MATLAB R2018b Simulink. As the pose tracking step is accomplished much faster than the orbital period, it is not possible to see the transient stabilization zone and the orbital effect of J_2 perturbation by plotting the stand-off separation with respect to the full orbit period.

Figure 4.8: Case 2: Chaser and target trajectory for one period in orbit.

Figure 4.9: Case 2: Chaser and target trajectory for five periods in orbit.

Hence, a zoomed-in box of the trajectory is provided in Figs. 4.8 and 4.9. The subsequent figures are shown for the first 200 s to observe the transient pose tracking phase. The stand-off separation as a function of time is presented in top-left of Fig. 4.10. Note that, this graph shows the theoretical convergence of the trajectory from around 130 km to the specified stand-off separation of 1.9339 m. There would, however, be practical considerations such as those with addressing the control saturation, wherein

Figure 4.10: Results for pose tracking on J_2 -perturbed orbits.

the thrusters of the chaser spacecraft saturate at its maximum thrust capability when a control force beyond its capacity is demanded. However, this aspect is deemed to be beyond the scope of this work. Nevertheless, the plots demonstrate the functionality and usefulness of the Udwadia-Kalaba framework to generate trajectories even in cases where there is significant variations in the initial conditions during the start of close-range proximity operations. The orientation time history is observed from the top-right of Fig. 4.10 in which the components of the chaser alignment vector $\bar{\mathbf{p}}_c$ aligns with the point of interest vector $\bar{\mathbf{p}}_t$ in the inertial frame. The associated control force to result in and maintain the desired stand-off is presented in the bottom-left of Fig. 4.10. The initial force is very high and it can be reduced by making sure that the pose tracking maneuvers are initiated only when the chaser is in close vicinity of the target, as opposed to several hundreds of kilometers away (130 km in this case). Another option is to reduce the position control gains α_r and γ_r so that the force demands are lower. The convergence would, however, occur over a longer period of

time due to the lower control gain values.

Figure 4.11: Constraint errors for pose tracking on J_2 -perturbed orbits.

The bottom-right plot in Fig. 4.10 presents the corresponding control torques. Fig. 4.11 shows that the constraint error diminishes asymptotically to zero.

Case 2.1: On-orbit Pose Tracking with Lower Control Gains

With the aim of attaining lower control forces and torques as compared to Case 2, simulations were performed after reducing the control gains α_r , γ_r , α_u and γ_u as,

$$\alpha_r = 0.02 \cdot \mathbb{I}_{3 \times 3} \quad \gamma_r = 0.01 \cdot \mathbb{I}_{3 \times 3} \quad \alpha_u = 0.2 \cdot \mathbb{I}_{3 \times 3} \quad \gamma_u = 0.1 \cdot \mathbb{I}_{3 \times 3} \quad (4.73)$$

All the other parameters remain the same as in Case 2. The results for pose tracking with reduced gains are plotted in Fig. 4.12.

The top-left plot in Fig. 4.12 presents the time history of the stand-off separation which shows a larger settling time of around 500 s. The orienting of the alignment and point of interest vectors also take place slowly as shown in the top-right of Fig. 4.12 due to the lower control gains. From the bottom-left and bottom-right plots in Fig. 4.12, the force and torque demands are lower than that of Case 2. The corresponding position and orientation constraint errors are plotted in Fig. 4.13

Figure 4.12: Results for on-orbit pose tracking with lower control gains

Figure 4.13: Constraint errors for on-orbit pose tracking with lower control gains

Case 2.2: On-orbit Pose Tracking Closer to the Target

In this case, the simulations are performed with the chaser closer to the target spacecraft. Orbital elements of the target spacecraft are taken to be the same as in Table 4.3. The new set of orbital elements for the chaser spacecraft are given in Table 4.5.

Table 4.5: Chaser Orbital Elements

Orbital Elements	Value
e	0.05
a	7500.01 km
i	30.001°
Ω	10°
ω	20.001°
θ_0	0.001°

Using Eqs. (A.15) and (A.16), the corresponding initial position and velocity components of the chaser can be calculated as

$$\mathbf{r}_{0,c} = \begin{bmatrix} 6.22702 & 3.24115 & 1.2186 \end{bmatrix}^T \times 10^6 \text{ m} \quad (4.74)$$

$$\mathbf{v}_{0,c} = \begin{bmatrix} -3.6648 & 5.68703 & 3.60109 \end{bmatrix}^T \times 10^3 \text{ m/s} \quad (4.75)$$

Simulations were carried out to obtain the performance characteristics for pose tracking in close-proximity.

The top-left and top-right plots in Fig. 4.14 show similar settling characteristics for the stand-off separation and orientation maintenance as in Case 2. However, the control forces and torques are now much lower as compared to Case 2 as plotted in the bottom-left and bottom-right of Fig. 4.14, respectively. Finally, Fig. 4.15 presents the plot of position and orientation constraint error variation with time that shows asymptotic convergence.

Figure 4.14: Results for on-orbit pose tracking closer to the target.

Figure 4.15: Constraint errors for on-orbit pose tracking closer to the target.

Chapter 5

3-DOF Pose Tracking

This chapter provides a detailed derivation of the planar formulation of the Udwadia-Kalaba equation-based stand-off pose tracking control method. Firstly, the scenario of target tracking using a single chaser spacecraft is presented. A common problem encountered in tethered active debris removal process is the difficulty in maintaining the tether tension. Pulling the target using chasers located at diametrically opposite ends would ensure that the tension is maintained throughout the debris stabilization process and hence, a relevant dual chaser configuration is introduced to enable this approach. After introducing the necessary aspects of the problem formulation, the desired final configuration of the chaser with respect to the target spacecraft is defined as a constraint function in the generalized position coordinates. Subsequently, this positional constraint is differentiated once and then twice to obtain generalized velocity and acceleration constraints. By composing the positional, velocity and acceleration constraints into a second order differential form, positive coefficients guarantee the asymptotic convergence of the states to the desired final configuration. This differential equation can be rearranged to obtain a form that is linear in acceleration and inserted into the Udwadia-Kalaba equation to generate control force commands that obey the set convergence performance while taking into account the inherent dynamics of the system. Simulation studies are performed to demonstrate the pose tracking capability for single and dual chaser scenarios.

5.1 A Case for Planar Analysis

For performing an actual pose tracking with respect to a tumbling and translating target in space, the full 6-six-degree-of-freedom control strategy is desired. A planar formulation, as provided in this chapter, is primarily done as a stepping stone to ultimately performing the experimental validation. It is to be noted that planar

simplification does not affect generality of the control methodology [26]. Hence, a planar rigid body motion of both chasers and the target is considered. The control equations are derived based on the general case of a translating and spinning target. The motion is simulated using a planar double-integrator dynamics as given by the form discussed in Sec. 5.4.1,

$$\begin{bmatrix} \ddot{x}_p \\ \ddot{y}_p \\ \ddot{\theta}_p \end{bmatrix} = \begin{bmatrix} m_p & 0 & 0 \\ 0 & m_p & \\ 0 & 0 & J_p \end{bmatrix}^{-1} \begin{bmatrix} F_{xp}^i \\ F_{yp}^i \\ \tau_p^p \end{bmatrix} \quad (5.1)$$

for the p^{th} body. Note that the derived control equations would also lie on the plane. As it can be seen, this chapter would form the foundation for subsequent discussion on the experimental validation of the control methodology.

5.2 Planar Formulation of the Problem

The double integrator model given by Eq. 5.1 is representative of the equations of relative motion under the assumption that: (1) the target is on a circular orbit, (2) the relative distance of chaser is small as compared to the distance between the target and the Earth, and (3) that the time scale at which proximity operations take place is faster than the orbital period of either spacecraft [48]. The relative velocities between the chasers and debris are assumed to be relatively small so as to attain the capture without control saturation. Note that the relative velocity can be kept to a low value by using an accurate far-range rendezvous guidance. The external perturbing forces or torques are ignored on account of a larger magnitude of the control force or torque. Using the recent advancements in machine vision based pose-estimation in space [49], the position and velocity information of the target can be calculated with sufficiently high accuracy. Though not discussed here, the Udwadia-Kalaba framework can also provide accurate control under bounded errors in the estimated target parameters [50]. In the case of the dual-chaser pose tracking, the pose data from the cameras fixed on each chaser can be used to predict the pose of the target with more accuracy.

An inertial $x - y$ Cartesian frame of reference is defined with the chasers and target lying on it. Body-fixed frames of reference are established on each body in such a way that their origins pass through the body center-of-mass and reference unit vectors point along the principal axes of inertia. Orientation of the body-fixed frames are defined with respect to the inertial frame.

Figure 5.1: Planar problem formulation

Suppose that the chasers and target are located in a bounded subspace $\mathcal{S} \subseteq \mathbb{R}^2$, as illustrated in Fig. 5.1. Let the bodies be located at (x_p, y_p) with an orientation θ_p with respect to the positive direction of the inertial x -axis, where the subscript $p = c1, c2$ and t represent Chaser 1, Chaser 2, or target, respectively. The corresponding linear velocity and acceleration vectors are $(\dot{x}_p, \dot{y}_p)$ and $(\ddot{x}_p, \ddot{y}_p)$, respectively. Angular velocity is given by $\dot{\theta}_p$ and the corresponding out-of-plane angular accelerations is denoted by $\ddot{\theta}_p$. The p^{th} body has a mass of m_k and an out-of-plane inertia of J_p . As the net external force acting in close-range pose tracking is small, the target velocity is assumed to be fixed.

The objective is to define a constraint function of the form $\Phi = \mathbf{0}$ and to subsequently derive the acceleration constraint of the form $\mathbf{A}_c \ddot{\mathbf{q}} = \mathbf{b}_c$, so that the generalized force $\mathbf{Q}_c$ can be determined from the Udwadia-Kalaba equation. In the first stage of the problem formulation, it is desired to get Chaser 1 to synchronize with

the target angular velocity while maintaining a constant stand-off distance L_0 at a given relative orientation with respect to the target. In second stage of the problem, the chasers are desired to synchronize their angular velocities and rendezvous with the target while maintaining a diametrically opposite location with respect to each other and centered at the target center-of-mass. The control action ensures that the control constraint $\mathbf{A}_c \ddot{\mathbf{q}} = \mathbf{b}_c$ is satisfied by the application of the control force $\mathbf{Q}_c$ on the chaser spacecraft. The generalized acceleration of the chaser is given by

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}\mathbf{Q} + \mathbf{M}^{-1}\mathbf{Q}_c \quad (5.2)$$

where

$$\mathbf{Q}_c = \mathbf{M}^{\frac{1}{2}}(\mathbf{A}_c \mathbf{M}^{-\frac{1}{2}})^+(\mathbf{b}_c - \mathbf{A}_c \mathbf{M}^{-1}\mathbf{Q}) \quad (5.3)$$

Note that, Eq. (5.3) is the Udwadia-Kalaba equation.

5.3 Single Chaser Formation Constraint

To begin, the desired final configuration is chosen from which the constraint function is built. For Chaser 1, the final configuration is as shown in Fig. 5.2.

Figure 5.2: Desired pose tracking configuration for single chaser.

In the terminal stage of the pose tracking, the states of Chaser 1 - $\begin{bmatrix} x_{c1} & y_{c1} & \theta_{c1} \end{bmatrix}^T$

are related to the corresponding target states - $\begin{bmatrix} x_t & y_t & \theta_t \end{bmatrix}^T$ by the relation,

$$\begin{bmatrix} x_{c1} \\ y_{c1} \\ \theta_{c1} \end{bmatrix} = \begin{bmatrix} x_t + L_0 \cos \theta_t \\ y_t + L_0 \sin \theta_t \\ \theta_t + \pi \end{bmatrix} \quad (5.4)$$

The first constraint equation can be developed by using the fact that x -component in Eq. (5.4) must be satisfied as $t \rightarrow \infty$. Hence, the first constraint function is defined as

$$\phi_1 = x_{c1} - x_t - L_0 \cos \theta_t \quad (5.5)$$

Note that the constraint function is composed of terms in the generalized position. When this constraint is satisfied, $\phi_1 = 0$. Initially, the constraint may not be satisfied and the proposed controller has to ensure that $\phi_1 \rightarrow 0$ as $t \rightarrow \infty$. The first derivative of Eq. (5.5) gives the corresponding constraint in the generalized velocity as

$$\dot{\phi}_1 = \dot{x}_{c1} - \dot{x}_t + L_0 \sin \theta_t \dot{\theta}_t \quad (5.6)$$

Double-differentiating Eq. (5.6) with respect to time yields the constraint in the generalized acceleration as

$$\ddot{\phi}_1 = \ddot{x}_{c1} - \ddot{x}_t + L_0 \sin \theta_t \ddot{\theta}_t + L_0 \cos \theta_t \dot{\theta}_t^2 \quad (5.7)$$

Proceeding in a similar manner, the constraint function based on the y -component is defined as

$$\phi_2 = y_{c1} - y_t - L_0 \sin \theta_t \quad (5.8)$$

Velocity constraint $\dot{\phi}_2$ is then derived as

$$\dot{\phi}_2 = \dot{y}_{c1} - \dot{y}_t - L_0 \cos \theta_t \dot{\theta}_t \quad (5.9)$$

The subsequent double-differentiation yields

$$\ddot{\phi}_2 = \ddot{y}_{c1} - \ddot{y}_t - L_0 \cos \theta_t \ddot{\theta}_t + L_0 \sin \theta_t \dot{\theta}_t^2 \quad (5.10)$$

Finally, the orienting constraint is define using the third row of Eq. (5.4) and is given by

$$\phi_3 = \theta_{c1} - \theta_t - \pi \quad (5.11)$$

Differentiating (5.11) with respect to time,

$$\dot{\phi}_3 = \dot{\theta}_{c1} - \dot{\theta}_t \quad (5.12)$$

Double-differentiating (5.11) with respect to time gives

$$\ddot{\phi}_3 = \ddot{\theta}_{c1} - \ddot{\theta}_t \quad (5.13)$$

Now, the i^{th} control constraint, where $i = 1, 2$ and 3 , could be composed into a second order differential equation as

$$\ddot{\phi}_i + \alpha_i \dot{\phi}_i + \gamma_i \phi_i = 0 \quad (5.14)$$

where the constants $\alpha_i, \gamma_i > 0$ determine the performance of i^{th} controller action. Substituting Eq. (5.5)-(5.7) into Eq. (5.14), the first control constraint is then given as

$$\ddot{\phi}_1 + \alpha_1 \dot{\phi}_1 + \gamma_1 \phi_1 = 0$$

$$\begin{aligned} \Rightarrow \ddot{x}_{c1} &= \ddot{x}_t - L_0 \sin \theta_t \ddot{\theta}_t - L_0 \cos \theta_t \dot{\theta}_t^2 \\ &\quad - \alpha_1 (\dot{x}_{c1} - \dot{x}_t + L_0 \sin \theta_t \dot{\theta}_t) \\ &\quad - \gamma_1 (x_{c1} - x_t - L_0 \cos \theta_t) \end{aligned} \quad (5.15)$$

$$= b_1 \quad (5.16)$$

where the values of α_1 and γ_1 affects the convergence characteristics along the x -component of the inertial frame. By taking the y -component constraint equations (5.8)-(5.10) and substituting into the second order differential equation (5.14),

$$\ddot{\phi}_2 + \alpha_2 \dot{\phi}_2 + \gamma_2 \phi_2 = 0$$

$$\begin{aligned}
\Rightarrow \ddot{y}_{c1} &= \ddot{y}_t + L_0 \cos \theta_t \ddot{\theta}_t - L_0 \sin \theta_t \dot{\theta}_t^2 \\
&\quad - \alpha_2 (\dot{y}_{c1} - \dot{y}_t - L_0 \cos \theta_t \dot{\theta}_t) \\
&\quad - \gamma_2 (y_{c1} - y_t - L_0 \sin \theta_t)
\end{aligned} \tag{5.17}$$

$$= b_2 \tag{5.18}$$

Finally, the orientation constraint in the angular acceleration is obtained by substituting Eq. (5.11)-(5.13) into Eq. (5.14) to obtain

$$\ddot{\phi}_3 + \alpha_3 \dot{\phi}_3 + \gamma_3 \phi_3 = 0$$

$$\Rightarrow \ddot{\theta}_{c1} = \ddot{\theta}_t - \alpha_3 (\dot{\theta}_{c1} - \dot{\theta}_t) - \gamma_3 (\theta_{c1} - \theta_t - \pi) \tag{5.19}$$

$$= b_3 \tag{5.20}$$

In the matrix form, Eq. (5.16), (5.18) and (5.20) can be re-written as

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{x}_{c1} \\ \ddot{y}_{c1} \\ \ddot{\theta}_{c1} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{5.21}$$

$$\Rightarrow \mathbf{A}_{c1} \ddot{\mathbf{q}}_{c1} = \mathbf{b}_{c1} \tag{5.22}$$

where

$$\mathbf{A}_{c1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{b}_{c1} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{5.23}$$

5.4 Control Force Determination

The exact generalized control force required to achieve the formation can be calculated using Eq. (2.60) as

$$\mathbf{F}_{c1} = \mathbf{M}_{c1}^{\frac{1}{2}} (\mathbf{A}_{c1} \mathbf{M}_{c1}^{-\frac{1}{2}})^+ (\mathbf{b}_{c1} - \mathbf{A}_{c1} \ddot{\mathbf{q}}_{u,c1}) \tag{5.24}$$

where $\mathbf{M}_{c1} = \text{diag}([m_{c1} \ m_{c1} \ J_{c1}])$ and $\ddot{\mathbf{q}}_{u,c1}$ is the net unconstrained acceleration of Chaser 1. Since the external forces and torques are ignored in the planar case as discussed in Sec. 5.2, the unconstrained acceleration $\ddot{\mathbf{q}}_{u,c1} = \mathbf{0}_{3 \times 1}$. Then, Eq. (5.24) is reduced to

$$\mathbf{F}_{c1} = \mathbf{M}_{c1}^{\frac{1}{2}}(\mathbf{A}_{c1}\mathbf{M}_{c1}^{-\frac{1}{2}})^+\mathbf{b}_{c1} \quad (5.25)$$

5.4.1 Double Integrator Dynamics for Planar Case

In the planar case of the pose tracking operation, it is assumed that both the target and chaser spacecraft lie on a plane with their spin axes perpendicular to the plane. Though this is a highly simplified version of the 6-DOF model, it does not affect the generality of the process followed for developing the control methodology [26]. Hence, a simplified planar case could be used on planar testbeds like the Spacecraft Proximity Operations Testbed at Carleton University or the Proximity Operations System Testbed at the Canadian Space Agency to easily validate the control algorithm.

Location and orientation of the body-fixed frames on the target and chaser are specified with respect to an inertial coordinate frame. Let the body p be of mass m_p and out-of-plane inertia be J_p . The location of the body in the inertial frame is (x_p, y_p) with an orientation θ_p . Under the effect of an inertial force (F_{xp}^i, F_{yp}^i) and body-fixed moment τ_p^p wherein the superscripts represent the frame with respect to which the vector component is expressed, the double-derivative of state is given by

$$\begin{bmatrix} \ddot{x}_p \\ \ddot{y}_p \\ \ddot{\theta}_p \end{bmatrix} = \begin{bmatrix} m_p & 0 & 0 \\ 0 & m_p & 0 \\ 0 & 0 & J_p \end{bmatrix}^{-1} \begin{bmatrix} F_{xp}^i \\ F_{yp}^i \\ \tau_p^p \end{bmatrix} \quad (5.26)$$

The inertial force can be expressed in the body-fixed frame using the Euler-angle based rotation matrix as discussed in Sec. 3.2.1. By choosing $\theta_x = 0$, $\theta_y = 0$ and $\theta_z = \theta_p$ in Eq. (3.5) and ignoring the z -component on account of the planar consideration,

$$\begin{bmatrix} F_{xp}^p \\ F_{yp}^p \end{bmatrix} = \begin{bmatrix} \cos \theta_p & \sin \theta_p \\ -\sin \theta_p & \cos \theta_p \end{bmatrix} \begin{bmatrix} F_{xp}^i \\ F_{yp}^i \end{bmatrix} \quad (5.27)$$

Substituting Eq. (5.25) into the second order dynamics as given by Eq. (5.1),

$$\begin{bmatrix} \ddot{x}_{c1} \\ \ddot{y}_{c1} \\ \ddot{\theta}_{c1} \end{bmatrix} = \begin{bmatrix} m_{c1} & 0 & 0 \\ 0 & m_{c1} & 0 \\ 0 & 0 & J_{c1} \end{bmatrix}^{-1} \mathbf{F}_{c1} \quad (5.28)$$

Since the target does not have any control authority and other external forces are ignored, the target motion is based on the equation

$$\begin{bmatrix} \ddot{x}_t \\ \ddot{y}_t \\ \ddot{\theta}_t \end{bmatrix} = \begin{bmatrix} m_t & 0 & 0 \\ 0 & m_t & 0 \\ 0 & 0 & J_t \end{bmatrix}^{-1} \mathbf{0} = \mathbf{0}_{3 \times 1} \quad (5.29)$$

5.5 Single Chaser Pose Tracking Control Simulation

Simulations are performed to demonstrate the proposed single chaser pose tracking. The mass of chaser and target is considered to be of the same value of 1 kg and the out-of-plane inertia of the bodies is taken as $0.5 \text{ kg} \cdot \text{m}^2$. The desired stand-off separation between the target and chaser is chosen as $L_0 = 10 \text{ m}$. For the i^{th} control constraint, where $i = 1, 2$, and 3 , the gains in Eq. (5.14) are chosen to be $\alpha_i = 2$ and $\gamma_i = 1$ for critically damped behavior of the control action. Note that in the real implementation, the gains would have to be adjusted so as to fit the performance requirement. Equation (5.28) can be used to simulate the effect of control action on the chaser.

The origin of the body-fixed chaser 1 frame is considered to be initially located at $(5, 2.5) \text{ m}$ in the inertial coordinates. The chaser frame is assumed to be oriented with respect to the positive direction of the inertial x -axis at an angle of $\pi \text{ rad}$. Additionally, an initial velocity of $(0.5, 0.5) \text{ m/s}$ is imparted on chaser 1. The initial out-of-plane angular velocity of the spacecraft is 0.9 rad/s . Let the target be initially located at the origin of the inertial frame and have a constant velocity of $(-1, 0.3) \text{ m/s}$. It is also considered to be spinning at a constant rate of 1.2 rad/s , starting from an initial orientation of 0 rad with respect to the inertial frame.

Figure 5.3: Single chaser and target pose tracking trajectory.

Figure 5.3 presents the planar pose tracking of a single chaser around a target. In order to keep track of the orientation, the x -axis of the body-fixed frame is marked on both chaser 1 and target. It can be seen that in the final configuration, the orientation vectors from the body frames point towards each other as observed from the inertial reference frame. Hence, the third condition from Eq. (5.4), $\theta_{c1} = \theta_t + \pi$, is satisfied.

Figure 5.4: Time history of chaser 1 and target states

The time history of the x and y components of the position vector for both the uncontrolled target and the controlled Chaser 1 is plotted in Fig. 5.4(a). The target

position variation has a constant slope, while the stand-off maintenance control action results in a cyclic motion for the Chaser 1 position. Figure 5.4(b) presents the angular displacements of the target and Chaser 1. Notice that the angular displacement of the Chaser 1 is π rad more than the target angular displacement. The chaser motion, in the limiting case, shows the same slope as that of the target and in the absence of external perturbations, the slopes are constant.

Figure 5.5: Control force in single chaser pose tracking.

(a) Absolute angular velocity.

(b) Stand-off separation.

Figure 5.6: Angular velocity matching and stand-off separation maintenance in single chaser pose tracking

The control forces and torques acting on the chaser as shown in Fig. 5.5 ensure that

the corresponding relative separation also settles to the desired stand-off distance of $L_0 = 10$ m as observed from Fig. 5.6(b), while simultaneously matching their angular velocities. Since there are no external perturbations, the angular velocity of the target remains fixed at 1.2 rad/s and as shown in Fig. 5.6(a), the chaser 1 angular velocity also settles to this value.

5.6 Dual-Chaser Formation Constraint

Considering the second chaser with the desired configuration set as shown in Fig. 5.7.

Figure 5.7: Desired pose tracking configuration for dual-chaser system.

The final configuration of Chaser 2 based on the target generalized position vector would then be written as,

$$\begin{bmatrix} x_{c2} \\ y_{c2} \\ \theta_{c2} \end{bmatrix} = \begin{bmatrix} x_t - L_0 \cos \theta_t \\ y_t - L_0 \sin \theta_t \\ \theta_t \end{bmatrix} \quad (5.30)$$

The x -component of Eq. (5.30) which can be composed into the fourth constraint function as

$$\phi_4 = x_{c2} - x_t + L_0 \cos \theta_t \quad (5.31)$$

The first differentiation of Eq. (5.31) with respect to time yields the fourth velocity constraint,

$$\dot{\phi}_4 = \dot{x}_{c2} - \dot{x}_t - L_0 \sin \theta_t \dot{\theta}_t \quad (5.32)$$

Subsequent derivative with respect to time of Eq. (5.32) gives

$$\ddot{\phi}_4 = \ddot{x}_{c2} - \ddot{x}_t - L_0 \sin \theta_t \ddot{\theta}_t - L_0 \cos \theta_t \dot{\theta}_t^2 \quad (5.33)$$

Using the second row of Eq. (5.30), the fifth constraint equation ϕ_5 is defined as

$$\phi_5 = y_{c2} - y_t + L_0 \sin \theta_t \quad (5.34)$$

First and second derivatives of Eq. (5.34) are

$$\dot{\phi}_5 = \dot{y}_{c2} - \dot{y}_t + L_0 \cos \theta_t \dot{\theta}_t \quad (5.35)$$

and

$$\ddot{\phi}_5 = \ddot{y}_{c2} - \ddot{y}_t + L_0 \cos \theta_t \ddot{\theta}_t - L_0 \sin \theta_t \dot{\theta}_t^2 \quad (5.36)$$

Finally, the sixth constraint equation is defined as

$$\phi_6 = \theta_{c2} - \theta_t \quad (5.37)$$

The first derivatives are obtained as

$$\dot{\phi}_6 = \dot{\theta}_{c2} - \dot{\theta}_t \quad (5.38)$$

Second derivative of the constraint equation (5.37) give

$$\ddot{\phi}_6 = \ddot{\theta}_{c2} - \ddot{\theta}_t \quad (5.39)$$

Substituting Eqs. (5.31)-(5.33) into Eq. (5.14) and rearranging to keep controlled acceleration on the left hand side,

$$\begin{aligned}\ddot{x}_{c2} &= \ddot{x}_t + L_0 \sin \theta_t \ddot{\theta}_t + L_0 \cos \theta_t \dot{\theta}_t^2 \\ &\quad - \alpha_4(\dot{x}_{c2} - \dot{x}_t - L_0 \sin \theta_t \dot{\theta}_t) \\ &\quad - \gamma_4(x_{c2} - x_t + L_0 \cos \theta_t)\end{aligned}\tag{5.40}$$

$$= b_4\tag{5.41}$$

Using Eq. (5.34)-(5.36) in Eq. (5.14), the controlled y -acceleration of Chaser 2 is given by

$$\ddot{y}_{c2} = \ddot{y}_t - L_0 \cos \theta_t \ddot{\theta}_t + L_0 \sin \theta_t \dot{\theta}_t^2\tag{5.42}$$

$$- \alpha_5(\dot{y}_{c2} - \dot{y}_t + L_0 \cos \theta_t \dot{\theta}_t)\tag{5.43}$$

$$- \gamma_5(y_{c2} - y_t + L_0 \sin \theta_t)$$

$$= b_5\tag{5.44}$$

The controlled angular acceleration is given by substituting Eq. (5.37)-(5.39) into Eq. (5.14) to give

$$\begin{aligned}\ddot{\theta}_{c2} &= \ddot{\theta}_t - \alpha_6(\dot{\theta}_{c2} - \dot{\theta}_t) - \gamma_6(\theta_{c2} - \theta_t) \\ &= b_6\end{aligned}\tag{5.45}$$

Note that, since the target is passive, the control action only affects the chaser spacecraft. Hence, the target states are part of the right hand side b_i for $i = 1, 2 \dots 6$ in Eqs. (5.16), (5.18), (5.20) and (5.41)-(5.45).

In matrix form, Eqs. (5.21),(5.41)-(5.45) are written the standard control constraint form as

$$\mathbf{A}_c \ddot{\mathbf{q}} = \mathbf{b}_c\tag{5.46}$$

where

$$\begin{aligned}\mathbf{A}_c &= \mathbf{I}_{6 \times 6} \\ \ddot{\mathbf{q}} &= \begin{bmatrix} \ddot{x}_{c1} & \ddot{y}_{c1} & \ddot{\theta}_{c1} & \ddot{x}_{c2} & \ddot{y}_{c2} & \ddot{\theta}_{c2} \end{bmatrix}^\top \\ \mathbf{b}_c &= \begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \end{bmatrix}^\top\end{aligned}\tag{5.47}$$

5.7 Control Force and Dual-Chaser-Target Dynamics

Using Eq. (5.46) in Eq. (2.60), the generalized control force command to be applied on the chasers are given by

$$\mathbf{F}_c = \mathbf{M}_c^{\frac{1}{2}} (\mathbf{A}_c \mathbf{M}_c^{-\frac{1}{2}})^+ (\mathbf{b}_c - \mathbf{A}_c \mathbf{M}_c^{-1} \mathbf{F}_u)\tag{5.48}$$

where $\mathbf{M}_c = \text{diag}(\begin{bmatrix} m_{c1} & m_{c1} & J_{c1} & m_{c2} & m_{c2} & J_{c2} \end{bmatrix})$ is the generalized mass matrix. Since the externally applied generalized forces are considered negligible, Eq. (5.48) is reduced to

$$\mathbf{F}_c = \mathbf{M}_c^{\frac{1}{2}} (\mathbf{A}_c \mathbf{M}_c^{-\frac{1}{2}})^+ \mathbf{b}_c\tag{5.49}$$

The second order dynamics for the simulation is then given by applying Eq. (5.1) to both chasers and combining them into a matrix form as

$$\begin{bmatrix} \ddot{x}_{c1} & \ddot{y}_{c1} & \ddot{\theta}_{c1} & \ddot{x}_{c2} & \ddot{y}_{c2} & \ddot{\theta}_{c2} \end{bmatrix}^T = \mathbf{M}_c^{-1} \mathbf{F}_c\tag{5.50}$$

As discussed in Sec. 5.4, the target is assumed to have no applied forces on it and hence, its net acceleration is simply zero, that is,

$$\begin{bmatrix} \ddot{x}_t & \ddot{y}_t & \ddot{\theta}_t \end{bmatrix}^T = \mathbf{0}_{3 \times 1}\tag{5.51}$$

5.8 Dual-chaser pose tracking Simulation

In this scenario, the system parameters and the initial states for both Chaser 1 and the target are assumed to be the same as in the single-chaser pose tracking simulations of Sec. 5.5. Chaser 2 is considered to have a mass of $m_{c2} = 1$ kg and out-of-plane inertia of $J_{c2} = 0.5$ kg.m². The desired stand-off separation between the target and chaser

is defined to be $L_0 = 10$ m. The control gains are chosen as $\alpha_i = 2$ and $\gamma_i = 1$ where $i = 4, 5$, and 6 . As compared to the previous single-chaser pose tracking, Chaser 2 is desired to align at a diametrically opposite location with respect to Chaser 1 with the target at the center.

Figure 5.8: Chaser and target trajectory.

Figure 5.9: Evolution of Chaser 1, Chaser 2 and target states

The origin of body-fixed frame of the Chaser 2 is initially positioned at (10,0) m with an alignment of 0 rad with respect to the inertial frame. The initial linear and

angular velocities are $(0.1, -1)$ m/s and 0.5 rad/s, respectively. Figure 5.8 presents the dual-chaser pose tracking with respect to the spinning target. As it can be seen, the body-fixed x -unit vector on Chaser 2 aligns along the x -axis of the target frame in the inertial reference. The evolution of the x and y components of the three bodies are plotted in Fig. 5.9(a) in which Chaser 2 shows a periodic curve centered at the target position component due to the circumnavigation control. Figure 5.9(b) presents the angular displacement of all the bodies with respect to time. Note that Chaser 2 follows the angular displacement of the target.

Figure 5.10: Chaser 1 and Chaser 2 control force.

Figure 5.11: Simultaneous pose tracking using dual chasers

Control forces and torques shown in Fig. 5.10 ensure that all the control constraints

as governed by Eq. (5.46) are met using the control action given by Eq. (5.48). The simultaneous angular velocity synchronization and stand-off distance maintenance can be seen from Fig. 5.11(a) and Fig. 5.11(b), respectively.

Chapter 6

Planar Experimental Validation

This chapter presents a detailed experimental study of the Udwadia-Kalaba based pose tracking controller that performs simultaneous positioning and angular orientation tracking with respect to the target spacecraft. A single chaser pose tracking and docking with respect to a target spacecraft is demonstrated in this chapter. The facility being used is the Spacecraft Proximity Operations Testbed ((SPOT) at Carleton university's Spacecraft Robotics and Control Laboratory. This facility provides a planar gravity offset and presents a means for testing control algorithms used for proximity operations in space. The experimental results show an accurate tracking of the target states by the chaser spacecraft during the close-range rendezvous phase. Finally, the docking phase is also validated.

6.1 Introduction

In this chapter, the control force given by Eq. (5.25) is used for generating the thruster commands in the SPOT software. It is important to have a clear understanding of the SPOT system that includes air-refilling stations, chaser and target platforms, the granite table, the motion capture systems and the SPOT 2.0 software to know the flexibility and limitations of the experimental setup. While a detailed discussion on the SPOT setup can be found in the work by Hovell [51], this chapter provides a relevant brief discussion that could in some cases help to interpret results that may be affected by it.

6.2 The Spacecraft Proximity Operations Testbed

The Spacecraft Proximity Operations Testbed (SPOT), as shown in Fig. 6.1, is a 3-DOF planar test facility used in the validation of methods used in close-proximity

space operations.

Figure 6.1: Top view of SPOT along with the spacecraft platforms

It consists of a smooth granite table on which the target and chaser spacecraft platforms float using air bearings. This provides an almost friction-free surface which approximates the environment of space. The experiment zone is a plane of length 3.51155 m and width 2.41935 m that is tracked by the PhaseSpace motion capture system. Each platform is equipped with a pressurized air tank that powers up the thrusters and flotation system. Before the start of an experiment, it is necessary to ensure that the tanks are sufficiently filled in order to successfully complete the experiment. The compressor allows the tank to be filled up to 4500 psi in about 10 minutes. Once the tanks are ready, it is required to ensure that the granite surface is wiped and cleaned using Isopropyl alcohol so that the floatation can take place. Using the SPOT 2.0 software which is written using MATLAB Simulink 2017b, the experiment is simulated first before implementing it on the real platforms. Once a satisfactory simulation is achieved, the control block can be transmitted into the Raspberry Pi 3. After the platforms are placed on the table and the experiment is initiated, the SPOT system automatically performs the experiment in seven phases.

When an experiment is running, the thrusters are fired based on the signals from a control mixer block that tracks the commanded force and torque demands provided by the control algorithm to be tested. The PhaseSpace cameras mounted above the granite slab transmits the state information of the platforms onto the ground station computers. This information is further transmitted wirelessly into the Raspberry Pi 3 computer that is mounted on the platforms that generate the desired thruster firing commands. Once the experiment ends, the experimental data can be recovered and stored as a .mat file, which can be used to plot the data.

6.2.1 Experiment Phases

Once the platforms are placed onto the granite table, the experiment is initiated. The following seven phases are encountered while executing the experiment:

1. Phase #0 – This is the synchronization phase that temporarily switches off the flotation system to take in accurate initial data of the chaser and target states from the PhaseSpace motion capture system. All control forces and torques are set to zero. The phase lasts for 10 seconds.
2. Phase #1 – The float is initiated in both platforms as there is continuous state data streaming for a span of 5 seconds. Note that no control forces are applied to the platforms during this phase.
3. Phase #2 – Using a proportional-derivative controller, the platforms are moved into their initial positions from which the actual experiment is initiated. This phase lasts for 20 seconds.
4. Phase #3 – This is the experiment phase that is run over a time of 131 seconds. The translation and/or spinning of the target is initiated as required. The control algorithm to be tested is applied on the chaser spacecraft, which performs the required pose tracking control.
5. Phase #4 – Using a PD controller, the chaser and target platforms are returned to their respective initial position within 30 seconds.

6. Phase #5 – The PD control is further used to correct any misalignments at the homing position for 5 seconds.
7. Phase #6 – Finally, the flotation and actuation systems are stopped. This ends the experiment.

Note that, the experimental data is collected for all the phases, but the data that is relevant for the pose tracking control is in Phase #3. This data can be recovered and plotted to visualize the control performance.

6.2.2 Spacecraft Platforms

The red platform represents the chaser spacecraft while the target spacecraft is the black platform as shown in Fig. 6.2.

Figure 6.2: The chaser (red) and target (black) spacecraft platform

The red chaser is the heavier of the two platforms and has a mass of 16.9478 kg and an out-of-plane inertial of 0.2709 kg.m^2 . The black target platform has a mass of 12.3341 kg and an out-of-plane inertia of 0.1880 kg.m^2 . Three air bearings support each of the platforms on the granite slab and ensure that an almost friction-free surface is achieved. Each platform is equipped with a total of eight thrusters that are distributed on the four sides. Firing of the thrusters are sequenced using a control mixer in the Raspberry Pi 3 flight computer to perform the translational and attitude control of the spacecraft platform. Normally-closed solenoid valves are used to

control the opening and closing of the thrusters to generate the desired control force and torque demands. Note that as the air gets expelled out through the thrusters, both the target and chaser starts getting lighter, and, as a result, their inertias decrease. This variation does not affect the pose tracking objective of positioning and aligning with the target as positive-definitiveness of generalized mass matrix is still maintained.

6.2.3 Sources of error

It is important to understand the sources of error that may affect the experimental results. This helps to interpret some of the anomalous behavior which may be observed in the experiment. Some of the common sources of errors include –

1. Momentary loss of signals between the motion capture system and the controller results in the build up of errors in the data that would affect the commanded forces and it's trajectory.
2. Dust particles on the surface of the granite table would affect the motion of air bearings. Cleaning the surface thoroughly before the experiment helps avoid this issue.
3. The firing of thrusters may not produce the exact control demand. This is due to the inherent time delays or slight misalignments in the thruster locations that is not accounted for in the control mixer block.
4. Due to the impulsive firing, there may be vibrations on the device that could affect the measurements provided by the overhead motion capture system.
5. As the air is expelled, the center of mass of the spacecraft shifts gradually. This can affect the control torque.

6.3 Case 1: Pose Tracking Around a Spinning Target

Figure 6.3 presents the overhead camera feed of the experiment phase. The target spacecraft is the black platform which is characterized by a white nozzle that mimics

a fuel-depleted satellite. The black target and red chaser spacecraft are initially located at positions shown in Fig. 6.3(a). As the experiment begins, a spin is initiated on the target spacecraft in Phase #3, while its position is maintained at a specific location on the table. This simulates the condition in which the chaser and target would initially have a zero-relative velocity. In real applications, this case would be similar to the case of capturing a debris using precise far-range rendezvous that aligns the chaser's terminal trajectory exactly along the target velocity.

Figure 6.3: Experiment on pose tracking around a spinning target

Figures 6.3(a)-6.3(i) presents the still shots of various stages in the experimental

phase. Control forces that are applied on the chaser maintains it at a stand-off separation at a specific orientation with respect to the target. Simultaneously, the chaser attempts to spin at the same rate as the target body. As a result, the chaser and target are rotating in a synchronous orientation with respect to each other, and in the limiting case, it looks as if the two bodies together execute a rigid body motion. The experimental results are plotted and the graphs are interpreted in the subsequent discussion.

Figure 6.4: Chaser and target spacecraft trajectory.

The actual trajectories of the chaser and target spacecraft are presented in Fig. 6.4. Before the experiment began, the SPOT system performed the Phase 0-2 over a period of 35 seconds and maintained the origin of the chaser frame at (2.15578, 1.20968) m in the inertial frame. Similarly, the target frame was initially positioned at (1.35577, 1.20968) m. The orientation of the chaser frame with respect to the positive x -axis of the inertial frame is maintained at π rad, while that of the target remained at 0 rad. Following this, the experiment began and ran for 131 s. Over the period, the chaser was required to maintain the specified stand-off distance of 0.7 m with respect to the target as observed from Fig. 6.4.

Figure 6.5: Position time history.

The corresponding variations in the x and y components are plotted in Fig. 6.5. Note that the system tries to keep the target at its initial position of $(1.35577, 1.20968)$ m throughout the experiment. There are, however, minor deviations from the desired location as imbalances in the thruster firing for maintaining spin causes the translational motion to be affected. The x -component of the chaser traces an approximate cosine curve centered at the x -location of target, while the y -component maps out a sine curve with its mean at the target y -location. This is due to the circular motion of the chaser around the target. The amplitude of both these curves would be the specified stand-off distance of 0.7 m. Maintenance of this relative separation can be seen from Fig. 6.6. The deviations observed in the time ranges above 60 s could be the result a loss of signal or an angle wrap-around issue in the SPOT software. The variation of the angular orientation of the chaser and target is plotted in Fig. 6.7. The target starts at an orientation of 0 rad and continues the rotational motion until it reaches 2π rad to complete one full axial rotation. The chaser follows this orientation with an angular phase shift of π rad as the objective is to point towards the target. Therefore, starting at an angular orientation of π rad, the chaser turned until it reached 3π rad. Since both bodies start from rest, the angular velocity starts from 0 rad/s.

Figure 6.6: Relative separation between chaser and target.

Figure 6.7: Orientation time history.

The time history of the angular velocity is plotted in Fig. 6.8. The target was commanded to rotate at a constant angular velocity of 0.0523599 rad/s. Note that the chaser has no information about this commanded rate of the target. It simply follows the commanded forces that are calculated based on the current body states and aligns its angular velocity with that of the target as observed from Fig. 6.8.

Figure 6.8: Angular velocity.

Figure 6.9: Control force.

The deviations that are seen in the graph could be the result of one of the sources of error mentioned in Sec. 6.2.3. From Fig. 6.5, the position components of chaser near the time that this anomaly occurs seem to be located at the left-most point on the circular trajectory in Fig. 6.4. For the SPOT system, this location corresponds to the point where the sensed angle switches between $\pm\pi$ rad in the software. Hence,

the most probable cause for this error is that the angle switching is unaccounted for in the software. Improving the SPOT software system to avoid such bugs would be an important priority in the future experiments. Finally, the commanded forces as a function of time is plotted in Fig. 6.9. Note that these commands are calculated from the chaser and target state information using Eq. (5.25). The corresponding control action ensures that the simultaneous stand-off distance maintenance along with the orientation control is achieved in the pose tracking control around a spinning target.

6.4 Case 2: Pose Tracking Around a Translating and Spinning Target

Figure 6.10: Experiment on pose tracking around a translating and spinning target

In the pose tracking experiment with the translating and spinning target spacecraft, both the bodies began near one of the shorter edges of the SPOT granite table. Translating motion was established for the target along the length of table. The simulations were run to make sure that either platforms do not fall off the table. Once a satisfactory simulation was achieved, the control codes were exported into the platforms and the experiment was initiated. The stills of the video captured from the overhead camera while the experiment was running are presented in Fig. 6.10. As the experiment proceeds, it can be seen that the target spacecraft translates along the length of the table while rotating around its axis. The chaser is required to maintain the stand-off distance while facing the target throughout its operation. During the start of the experiment, the chaser was commanded to the initial location (3.05578, 1.20968) m while the target was instructed to be positioned at the location (2.25577, 1.20968) m. The initial orientation of both body frames with respect to the inertial reference are the same as in Case 1.

Figure 6.11: Case 2: Chaser and target trajectory.

Time history of the trajectory of chaser and target frame origins is plotted in Fig. 6.11. As observed, the chaser controlled its position in order to maintain the stand-off separation with respect to the translating and spinning target. Cartesian components of the position of both platforms as a function of time are shown in Fig. 6.12.

Figure 6.12: Case 2: Chaser and target position time history.

Considering the horizontal axis first, the chaser's x -component varies in a periodic manner while being centered around the target's x -component, which is linear and has a negative slope. Note that, the sign of the slope depends on the velocity component along the axis. In this case, the velocity is directed along the length of table in the direction of the negative x -axis and hence, the sign.

Figure 6.13: Case 2: Stand-off separation.

The y -component of the target is a straight line parallel to the horizontal axis and the corresponding chaser component approximates a sine curve centered at the target's y -axis location. The stand-off separation being maintained varies as plotted in Fig. 6.13. There is a minor deviation being observed in the stand-off separation between 60 – 80 s and 110 – 130 s time intervals. This is potentially due to incorrect sensor data that may have been fed into the system by any of the error sources listed in Sec. 6.2.3. Note that, the controller immediately compensates for this error as soon as good data is available.

Figure 6.14: Case 2: Angular displacement time history.

The time varying angular displacement of both platforms is presented in Fig. 6.14. The chaser followed the target while maintaining an offset in angular displacement of π rad with respect to that of the target. The corresponding angular velocity is plotted in Fig. 6.15 which shows the chaser settling onto the target angular velocity. The signals are affected by sensor noise, which in the case of the 60 – 80 s time interval seem to create minor deviations that are subsequently corrected by the controller. Correcting these issues related to SPOT would be important future work. The corresponding control forces and torque that simultaneously maintains the stand-off separation and orientation is plotted in Fig. 6.16.

Figure 6.15: Case 2: Angular velocity.

Figure 6.16: Case 2: Control force.

6.5 Case 3: Docking with a Translating and Spinning Target

In the special case when the chaser alignment with respect to the target results in the exact alignment of their docking ports, the method can be adapted to perform the docking maneuver. The idea is to apply the Udwadia-Kalaba equation-based planar

pose tracking method and gradually reduce L_0 to the docking distance. As an important practical consideration, L_0 reduction has to be initiated only after the correct alignment of a chaser with respect to the target so that the closing trajectory can be precisely controlled without any control saturation. Further, L_0 has to be reduced in small increments rather than as a sudden reduction to the desired stand-off distance. This is required to avoid overshoots that could make the chaser to bump into the target and push it away before the docking takes place or in some cases, even damage the docking port.

Figure 6.17: Experiment on docking with a translating and spinning target

In the experiment, the mating port on the chaser is fixed in such a way that orienting its body frame at π rad more than the target frame would result in a perfect alignment of the mating port of chaser with that of the target nozzle. Still shots of the live video feed of the pose tracking and docking experiment is presented in Fig. 6.17. The target and chaser are initially located closer to the shorter edge of the granite table. The target was made to translate along longer edge in the direction of the negative x -axis. Using the state input, the flight computer calculated the control force commands and sent it to the chaser platform to perform the pose tracking and docking. Once the docking maneuver was completed, the experiment was stopped at the instance shown in Fig. 6.17(i). The non-linear dynamics of the combined translation and rotational motion has not been incorporated into the chaser control system for pose tracking and docking. Hence, continuing to exert control forces might have caused the destabilization of the chaser-target combined system.

Figure 6.18: Case 3: Chaser and target trajectory.

The trajectory of the chaser and target from the beginning of the experimental Phase #3 to the pose tracking and docking control termination is shown in Fig. 6.18. The initial position and orientation of both the chaser and the target are the same as in Case 2 presented in Sec. 6.4. Unlike the previous cases, this case presents a docking

maneuver that ran for a period of 86 s. Following the initiation of Phase #3, the docking command was provided at 20 s. The stand-off separation was reduced from 0.7 m to the docking distance of 0.45 m as a linear function of time over a period of 40 s.

Figure 6.19: Case 3: Position time history.

Components of the position vector of the target and chaser varies as shown in Fig. 6.19. Beginning from the y -component of 1.20968 m, the target was commanded to move to 1.30968 m. A constant velocity was imparted along the negative x -axis. The chaser spacecraft followed the target based on the state information and performed the pose tracking and docking control. The corresponding position components of the chaser vary while being centered at the target position components. Variation of the stand-off distance with respect to time is plotted in Fig. 6.20. The docking maneuver can be observed between 20 – 60 s in this plot wherein the distance changes from 0.7 m to 0.45 m. The angular displacement plots are shown in Fig. 6.21. The target frame is initially aligned at 0 rad and moves up to $\frac{3\pi}{2}$. The chaser follows this displacement with a phase difference of π rad. Starting from an initial alignment of π rad, the chaser reached close to $\frac{5\pi}{2}$ when the experiment is stopped. Time history of the angular velocity variation is presented in Fig. 6.22. Note that, the chaser tracks the

target angular velocity and settles onto it.

Figure 6.20: Case 3: Stand-off distance.

Figure 6.21: Case 3: Angular displacement time history.

The data is quite noisy because of any of the sources listed in Sec. 6.2.3. In spite of the noise, it can be seen that the chaser controller accurately positions and orients the chaser as observed from Fig. 6.20 and Fig. 6.21. Finally, Fig. 6.23 presents the

control forces that are used to achieve the pose tracking and docking.

Figure 6.22: Case 3: Angular velocity.

Figure 6.23: Case 3: Control force.

Note that the angular velocities and the control forces tend to diverge towards the end of the experimental phase. This is due to the fact that the body configuration changes after docking while the pose tracking controller continues to act. Note that

the controller doesn't take into account the significant change in the system parameters due to the combined body motion. In future work, a docking feedback switch can be used on the chaser side to stop the controller once the docking takes place.

6.6 Summary

In this chapter, the performance of the Udwadia-Kalaba equation-based planar pose tracking control was analyzed based on data from experiments performed on Spacecraft Proximity Operations Testbed in Carleton University's Spacecraft Robotics and Control Laboratory. The first case of rendezvous onto a spinning target demonstrated satisfactory achievement of the desired control objective of simultaneous stand-off position maintenance and angular alignment. Subsequently, the target was made to simultaneously spin and translate while the chaser spacecraft pose tracked this motion. Successful pose tracking was demonstrated through the experimental results. Finally, by slowly varying the stand-off distance as a linear function of time, the docking maneuver was performed using the Udwadia-Kalaba equation-based pose tracking controller. From the angular velocity plots, it was noted that the SPOT software needs to be updated so as to eliminate the angle wrap-around issue.

Chapter 7

Conclusion and Future Work

This chapter focuses on summarizing the contributions, discussing some limitations and deriving insights on the potential future work. The chapter concludes with a short discussion on the significance of the work in the research community.

7.1 Thesis Summary

This thesis proposed an automated pose tracking control strategy based on the Udwadia-Kalaba framework. This proposed solution was proven to be simple while inherently accounting for the dynamics of the system. Spatial stand-off distance maintenance and alignment control was formulated and demonstrated for both an inertial case without any orbit effects and the on-orbit case with J_2 perturbation. The six DOF simulations highlighted the exact capture of the target point of interest even in the case of large misalignments in the initial positioning with respect to the target. Further, simultaneous position and angular orientation control of a single chaser and a dual chaser system in close-range pose tracking with respect to a spinning target was formulated in the planar case. The planar simulations demonstrated the asymptotic convergence of the chasers to the specified stand-off distance with respect to the target, while aligning with it. Based on the planar formulation, the experimental validation of the Udwadia-Kalaba equation-based guidance controller was demonstrated at the Spacecraft Proximity Operations Testbed in Carleton University's Spacecraft Robotics and Control Laboratory.

The three major contributions of the proposed work are detailed as follows –

- The first major contribution of this work was the development of the spatial stand-off pose tracking controller using the Udwadia-Kalaba framework. The

main advantage of this method was demonstrated in the pose tracking operation, beginning at a relatively far location with respect to the target that is in a slightly elliptical orbit. The controller ensured that the specified stand-off distance at the desired orientation is achieved, beginning from any initial condition. The corresponding plots from the position and orientation control demonstrated the achievement of the pose tracking objective. The effect of controller gains and initial distance from target on the control force and torque demands were also studied in on-orbit simulations.

- Subsequently, the planar formulation of the pose tracking controller was presented in chapter 5. Here, the Udwadia-Kalaba framework was used to develop a planar positioning and orienting controller. The proposed method could be used for applications that desire pose tracking onto an object with no control authority, like a spacecraft that has ran out of fuel or an asteroid that needs to be mined, etc. By aligning another chaser at a diametrically opposite end using the Udwadia-Kalaba equation-based planar control, a configuration useful in inertially stabilizing a centrally located target was generated. Simulation results with the planar dynamical system showed the precise simultaneous position and angular velocity tracking.
- The final contribution of this work was the planar validation of the Udwadia-Kalaba equation-based stand-off pose tracking controller using Carleton University's Spacecraft Proximity Operations Testbed (SPOT). This discussion was presented in chapter 6. In addition to demonstrating the planar pose tracking in the case of a tumbling and translating target, docking maneuver was also presented in the special case when aligning a pair of body-fixed vectors aligned their docking ports. As an interesting observation, the controller stabilized the system in-spite of deviations from the desired configuration due to loss of signal or noisy signals in SPOT. The experiment confirmed the validity of this approach in maintaining the stand-off distance while pointing towards a location of interest on the target.

7.2 A Note of Practicality

While the proposed method is powerful, it does have limitations, some of which are listed below.

- In real applications, the chaser spacecraft may not be able to provide the demanded thrust. Aspects such as those related to the control saturation were not addressed in this work.
- The pose tracking controller kept running even after the docking had taken place in experiment Case 3. As the controller did not incorporate the dynamical model of the combined system, it was seen to be creating undesired motion.
- The motion capture system was used throughout all experiments to provide the chaser with real-time information about the target dynamical states. In an actual operational scenario, such states would have to be provided to the chaser through the use of computer vision sensors and software. This aspect was deemed to be beyond the scope of this work.

7.3 Recommendations for Future Work

The primary focus in the future would be to incorporate the control saturation aspects into the Udwadia-Kalaba equation-based approach. Determining the control parameters based on the desired performance in an adaptive case could also be explored. In the context of practical implementation of the proposed work, the effect of parameter uncertainties, noise and orbital perturbations on the control performance would have to be studied in detail. Finally, due to the need for judicious use of the spacecraft fuel, one of the future directions would include the determination of minimum-fuel solutions in the spatial case.

7.4 Significance of Work

The significance of this work is demonstrated through the publication that has resulted from this research.

American Control Conference 2019 – Pothén, A. A., and Ulrich, S., “Close-Range Rendezvous with a Moving Target Spacecraft using Udwadia-Kalaba Equation, American Control Conference, Philadelphia, PA, 10-12 Jul, 2019. (accepted)

In addition to this, a manuscript is being prepared for the Journal of Guidance, Control, and Dynamics.

Appendix A

Details of Derivations

This appendix presents the proof for theorem presented in chapter 2 and the conversion of orbital elements to the corresponding inertial position and velocity of the spacecraft useful in chapter 4.

A.1 Proof for Theorem in Chapter 2

For a given $m \times n$ matrix $\mathbf{B}$ and $m \times 1$ matrix $\mathbf{c}$, consider a possibly inconsistent set of equations of the form

$$\mathbf{B}\mathbf{x} = \mathbf{c} \quad (\text{A.1})$$

where $\mathbf{x}$ is unknown. Suppose that we have a matrix $\mathbf{G}$ such that $\mathbf{x} = \mathbf{G}\mathbf{b}$ has the shortest length and when substituted into Eq. (2.29), yields the minimum value for $\|\mathbf{B}\mathbf{x} - \mathbf{c}\|$, then $\mathbf{G} = \mathbf{B}^+$, the Moore-Penrose generalized inverse.

Proof. The steps on finding an $\mathbf{x}$ that minimizes Eq. (2.29) and the steps required to prove that $\mathbf{G}\mathbf{b}$ is the shortest length are provided in Chapter 2 of Ref. [42]. Here, for the sake of completeness, we present the detailed proof. To begin with, we need to find an $\mathbf{x}$ that minimizes $\|\mathbf{B}\mathbf{x} - \mathbf{c}\|$

Let

$$\mathbb{Z}(\mathbf{x}) = \|\mathbf{B}\mathbf{x} - \mathbf{c}\|^2 = (\mathbf{B}\mathbf{x} - \mathbf{c})^T (\mathbf{B}\mathbf{x} - \mathbf{c}) \quad (\text{A.2})$$

Then, we can write

$$\begin{aligned} \mathbf{B}\mathbf{x} - \mathbf{c} &= \mathbf{B}\mathbf{x} - \mathbf{B}\mathbf{G}\mathbf{b} + \mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c} \\ &= \mathbf{B}(\mathbf{x} - \mathbf{G}\mathbf{b}) + (\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}) \\ &= \mathbf{B}\mathbf{y} + (\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}) \quad \text{where } \mathbf{y} = \mathbf{x} - \mathbf{G}\mathbf{b} \end{aligned} \quad (\text{A.3})$$

Taking the dot product of the two components $\mathbf{B}\mathbf{y}$ and $(\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c})$,

$$\begin{aligned}
 (\mathbf{B}\mathbf{y})^T(\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}) &= \mathbf{y}^T\mathbf{B}^T(\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}) \\
 &= \mathbf{y}^T\mathbf{B}^T((\mathbf{B}\mathbf{G})^T\mathbf{b} - \mathbf{c}) \quad (\text{Moore-Penrose condition 3}) \\
 &= \mathbf{y}^T\mathbf{B}^T(\mathbf{B}\mathbf{G})^T\mathbf{b} - \mathbf{y}^T\mathbf{B}^T\mathbf{c} \\
 &= \mathbf{y}^T(\mathbf{B}\mathbf{G}\mathbf{B})^T\mathbf{b} - \mathbf{y}^T\mathbf{B}^T\mathbf{c} \\
 &= \mathbf{y}^T\mathbf{B}^T\mathbf{b} - \mathbf{y}^T\mathbf{B}^T\mathbf{c} \quad (\text{Moore-Penrose condition 1}) \\
 &= 0
 \end{aligned} \tag{A.4}$$

Hence, $\mathbf{B}\mathbf{y}$ and $(\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c})$ are orthogonal to each other. Then, Eq. (A.3) and Eq. (A.4) allow us to write

$$\|\mathbf{B}\mathbf{x} - \mathbf{c}\|^2 = \|\mathbf{B}\mathbf{y}\|^2 + \|\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}\|^2 \tag{A.5}$$

$$= \|\mathbf{B}\mathbf{x} - \mathbf{B}\mathbf{G}\mathbf{b}\|^2 + \|\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}\|^2 \tag{A.6}$$

When $\mathbf{x} = \mathbf{G}\mathbf{b} + (\mathbf{I} - \mathbf{G}\mathbf{B})\mathbf{k}$ for an arbitrary vector $\mathbf{k}$,

$$\begin{aligned}
 \mathbf{B}\mathbf{x} - \mathbf{B}\mathbf{G}\mathbf{b} &= \mathbf{B}(\mathbf{G}\mathbf{b} + (\mathbf{I} - \mathbf{G}\mathbf{B})\mathbf{k}) - \mathbf{B}\mathbf{G}\mathbf{b} \\
 &= \mathbf{B}\mathbf{G}\mathbf{b} + \mathbf{B}(\mathbf{I} - \mathbf{G}\mathbf{B})\mathbf{k} - \mathbf{B}\mathbf{G}\mathbf{b} \\
 &= \mathbf{B}(\mathbf{I} - \mathbf{G}\mathbf{B}) \\
 &= \mathbf{B}\mathbf{I} - \mathbf{B}\mathbf{G}\mathbf{B} = \mathbf{B} - \mathbf{B} = \mathbf{0} \quad (\text{Moore-Penrose condition 1})
 \end{aligned} \tag{A.7}$$

Hence, the minimum of $\|\mathbf{B}\mathbf{x} - \mathbf{c}\| = \|\mathbf{B}\mathbf{G}\mathbf{b} - \mathbf{c}\|$ is obtained at $\mathbf{x} = \mathbf{G}\mathbf{b} + (\mathbf{I} - \mathbf{G}\mathbf{B})\mathbf{k}$ for an arbitrary vector $\mathbf{k}$ when Moore-Penrose generalized inverse conditions 1 and 3 are satisfied.

Now, the second part of the proof is to show that $\mathbf{x} = \mathbf{G}\mathbf{b}$ has the shortest length. We have

$$\mathbf{x} = \mathbf{G}\mathbf{b} + (\mathbf{I} - \mathbf{G}\mathbf{B})\mathbf{k} \tag{A.8}$$

Now, taking the dot product of $\mathbf{Gb}$ and $(\mathbf{I} - \mathbf{GB})\mathbf{k}$, we have

$$\begin{aligned}
 (\mathbf{Gb})^T(\mathbf{I} - \mathbf{GB})\mathbf{k} &= \mathbf{b}^T\mathbf{G}^T(\mathbf{I} - \mathbf{GB})\mathbf{k} \\
 &= \mathbf{b}^T\mathbf{G}^T\mathbf{k} - \mathbf{b}^T\mathbf{G}^T\mathbf{GB}\mathbf{k} \\
 &= \mathbf{b}^T\mathbf{G}^T\mathbf{k} - \mathbf{b}^T\mathbf{G}^T(\mathbf{GB})^T\mathbf{k} \quad (\text{Moore-Penrose condition 4}) \\
 &= \mathbf{b}^T\mathbf{G}^T\mathbf{k} - \mathbf{b}^T(\mathbf{GBG})^T\mathbf{k} \\
 &= \mathbf{b}^T\mathbf{G}^T\mathbf{k} - \mathbf{b}^T\mathbf{G}^T\mathbf{k} \quad (\text{Moore-Penrose condition 2}) \\
 &= 0
 \end{aligned} \tag{A.9}$$

Therefore, $\mathbf{Gb}$ and $(\mathbf{I} - \mathbf{GB})\mathbf{k}$ are orthogonal. From Eq. (A.8) and Eq. (A.9),

$$\|\mathbf{x}\|^2 = \|\mathbf{Gb}\|^2 + \|(\mathbf{I} - \mathbf{GB})\mathbf{k}\|^2 \tag{A.10}$$

Note that from Eq. (A.10) the minimum value of $\|\mathbf{x}\| = \|\mathbf{Gb}\|$ is obtained when $\mathbf{k} = 0$. Hence, $\mathbf{x} = \mathbf{Gb}$ when $\mathbf{G}$ satisfies the Moore-Penrose conditions 2 and 4.

Since $\mathbf{G}$ satisfies all the Moore-Penrose conditions, $\mathbf{G} = \mathbf{B}^+$ □

A.2 Orbital Elements to the Inertial Position and Velocity

The perifocal reference frame is a Cartesian coordinate system fixed in space and centered at the focus of the orbit. It has its x -unit vector, $\hat{\mathbf{i}}$ pointed along the perigee, its z -unit vector, $\hat{\mathbf{k}}$ pointed perpendicular to the orbital plane and $\hat{\mathbf{j}}$ completes the triad. The position vector components of a point in the perifocal frame is given by

$$\mathbf{r}_P = \frac{h^2}{\mu} \frac{1}{1 + e \cos \theta} \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix} \quad (\text{A.11})$$

The corresponding velocity vector is given by

$$\mathbf{v}_P = \frac{\mu}{h} \begin{bmatrix} -\sin \theta \\ e + \cos \theta \\ 0 \end{bmatrix} \quad (\text{A.12})$$

Note that h can be calculated from the classical orbit elements as

$$h = \sqrt{\mu a(1 - e^2)} \quad (\text{A.13})$$

Once the position and velocity vectors are calculated from Eq. (A.11) and (A.12), respectively, they may be transformed from the perifocal coordinates to the inertial ECI components using the osculating elements as

$$\mathbf{R}_{EP} = \begin{bmatrix} -\sin \Omega \cos i \sin \omega + \cos \Omega \cos \omega & -\sin \Omega \cos i \cos \omega - \cos \Omega \sin \omega & \sin \Omega \sin i \\ \cos \Omega \cos i \sin \omega + \sin \Omega \cos \omega & \cos \Omega \cos i \cos \omega - \sin \Omega \sin \omega & -\cos \Omega \sin i \\ \sin i \sin \omega & \sin i \cos \omega & \cos i \end{bmatrix} \quad (\text{A.14})$$

Then, the position and velocity vectors in the ECI frame are given by

$$\mathbf{r}_I = \mathbf{R}_{EP} \mathbf{r}_P \quad (\text{A.15})$$

$$\mathbf{v}_I = \mathbf{R}_{EP} \mathbf{v}_P \quad (\text{A.16})$$

Appendix B

MATLAB Scripts

This appendix provides the scripts developed in MATLAB R2018b for building and testing the automated rendezvous controller.

B.1 Codes for Orbit Rendezvous with J_2 Perturbation

B.1.1 init_orbit_rendezvous.m

```
% PROGRAM FOR SIMULTANEOUS POSITION AND VELOCITY CAPTURE OF TUMBLING
% POI ON-ORBIT SIMULATION
% In this MATLAB script, the chaser spacecraft captures the target at
% the point of interest in space with an varying pose. In addition to
% matching the position with the point of interest, the chaser
% attempts to match its angular velocity along its local x-axis with
% component of angular velocity along the point of interest vector.
%
% Author: Abin Alex Pothen
% Email: abinalexpothen@gmail.com
% Date: February 6, 2019
% Modified: March 28, 2019

close all

%% orbital parameters
mu = 398600e9;      % gravitational parameter (m^3/s^2)

%% initial target state definition
% define initial position vector of target C.O.M.
r0_t = 1e6*[6.2271 3.2410 1.2184]';

% define initial velocity vector of target C.O.M.
rd0_t = 1e3*[-3.6646 5.6872 3.6010]';

% define initial axis of rotation of target frame
rotation_axis_t = [0.2449 0.4809 0.8419]';
```

```

% normalize the rotation axis
rotation_axis_t = rotation_axis_t/norm(rotation_axis_t);

disp('The initial axis of rotation of target is : ');
disp(rotation_axis_t);

% define initial angle of rotation about initial axis of rotation of
% target frame
theta_t = 30; % in degrees

% define the initial target quaternion vector
u0_t = [cosd(theta_t/2) sind(theta_t/2)*rotation_axis_t']';

% define initial body-fixed angular velocity of target
omega0_t = [0.1 0 0]';

Omega0_t = [0 -omega0_t(1) -omega0_t(2) -omega0_t(3);
omega0_t(1) 0 omega0_t(3) -omega0_t(2);
omega0_t(2) -omega0_t(3) 0 omega0_t(1);
omega0_t(3) omega0_t(2) -omega0_t(1) 0];
% Pg. 264, Quaternions & Rotation
% Seq. by Kuipers

ud0_t = 0.5*Omega0_t*u0_t;

% define point of interest position vector in target frame
poi_rt = [-1.5 0.7 -1]';

% define mass of target C.O.M.
mt = 1;
% define target mass matrix
M_t = diag([mt mt mt]);

% define target principal moment of inertia
J_t = diag([1.3626 1.5333 0.3848]);

J0 = 1; % additional parameter that comes up in quaternion dynamics.
% see Udwadia 2010. Note that this parameter SHOULD NOT be zero.
% Any other value doesn't affect the dynamics.

% quaternion moment of inertia matrix of target
Jq_t = [J0 zeros(1,3);

```

```

zeros(3,1) J_t];

% initial target position vector
q0_t = [r0_t' u0_t']';

% initial target velocity vector
qd0_t = [rd0_t' ud0_t'];

%% define chaser states and parameters

% initial chaser position
r0_c = 1e6*[6.2603 3.1337 1.1659]';

% initial chaser velocity
rd0_c = 1e3*[-3.5755 5.7080 3.7638]';

% define initial axis of rotation of chaser frame
rotation_axis_c = [0 0 1]';
% normalize the rotation axis
rotation_axis_c = rotation_axis_c/norm(rotation_axis_c);

disp('The initial axis of rotation of chaser is : ');
disp(rotation_axis_c);

% define initial angle of rotation about initial axis of rotation of
% target frame
theta_c = 0; % in degrees

% define the initial chaser quaternion vector
u0_c = [cosd(theta_c/2) sind(theta_c/2)*rotation_axis_c']';

% define initial body-fixed angular velocity of chaser
omega0_c = [0 0 0]';

Omega0_c = [0 -omega0_c(1) -omega0_c(2) -omega0_c(3);
omega0_c(1) 0 omega0_c(3) -omega0_c(2);
omega0_c(2) -omega0_c(3) 0 omega0_c(1);
omega0_c(3) omega0_c(2) -omega0_c(1) 0];
% Pg. 264, Quaternions & Rotation
% Seq. by Kuipers

ud0_c = 0.5*Omega0_c*u0_c;

```



```

% define alignment vector in chaser frame
rc_align = [1 0 0]';

% define mass of target C.O.M.
mc = 1;

% define target mass matrix
M_c = diag([mc mc mc]);

% define target principal moment of inertia
J_c = diag([1.3626 1.5333 0.3848]);

% already defined
% J0 = 1; % additional parameter that comes up in quaternion dynamics.
% see Udwadia 2010
% Note that this parameter SHOULD NOT be zero. Any other value
% doesn't affect the dynamics.

% quaternion moment of inertia matrix of target
Jq_c = [J0 zeros(1,3);
zeros(3,1) J_c];

% initial target position vector
q0_c = [r0_c' u0_c']';

% initial target velocity vector
qd0_c = [rd0_c' ud0_c'];

%% generate dynamics simulation data
TimePeriod = 6464.03;
sim('orbit_rendezvous.slx')

```

B.1.3 chaser_controller block

```

function [Qcon_c, F_con, T_con] = chaser_controller(q_c,qd_c,q_t,qd_t,
qdd_t,Qext_c,Jq_c,M_c,poi_rt,rc_align)

% control parameters
alpha_position = 0.1*diag([1 1 1]);
gamma_position = 0.1*diag([0.5 0.5 0.5]);
alpha_align = diag([1 1 1]);
gamma_align = diag([0.5 0.5 0.5]);

% abstract states -> real states
Fext_c = Qext_c(1:3,1); % applied force at C.O.M
Gamma_c = Qext_c(4:7,1); % quaternion torque
% may not be used right away

r_c = q_c(1:3,1); % position of chaser frame
r_t = q_t(1:3,1); % position of target frame

rd_c = qd_c(1:3,1); % velocity of chaser frame
rd_t = qd_t(1:3,1); % velocity of target frame

rdd_t = qdd_t(1:3,1); % acceleration of target frame

u_t = q_t(4:7,1); % quaternion of target frame
u_c = q_c(4:7,1); % quaternion of chaser frame

ud_t = qd_t(4:7,1); % quaternion derivative of target frame
ud_c = qd_c(4:7,1); % quaternion derivative of chaser frame

udd_t = qdd_t(4:7,1); % quaternion double derivative of target frame

%% position control constraint generation

% quaternion based rotation matrix from target frame -> inertial frame
R_et = [2*u_t(1)^2-1+2*u_t(2)^2    2*u_t(2)*u_t(3)-2*u_t(1)*u_t(4) ...
         2*u_t(2)*u_t(4)+2*u_t(1)*u_t(3);
         2*u_t(2)*u_t(3)+2*u_t(1)*u_t(4)    2*u_t(1)^2-1+2*u_t(3)^2 ...
         2*u_t(3)*u_t(4)-2*u_t(1)*u_t(2);
         2*u_t(2)*u_t(4)-2*u_t(1)*u_t(3)    2*u_t(3)*u_t(4)+2*u_t(1)*u_t(2) ...
         2*u_t(1)^2-1+2*u_t(4)^2];

% matrices used in defining rotation derivative matrix

```

```

L1_t = [4*u_t(1) 4*u_t(2) 0 0;
2*u_t(4) 2*u_t(3) 2*u_t(2) 2*u_t(1);
-2*u_t(3) 2*u_t(4) -2*u_t(1) 2*u_t(2)];

L2_t = [-2*u_t(4) 2*u_t(3) 2*u_t(2) -2*u_t(1);
4*u_t(1) 0 4*u_t(3) 0;
2*u_t(2) 2*u_t(1) 2*u_t(4) 2*u_t(3)];

L3_t = [2*u_t(3) 2*u_t(4) 2*u_t(1) 2*u_t(2);
-2*u_t(2) -2*u_t(1) 2*u_t(4) 2*u_t(3);
4*u_t(1) 0 0 4*u_t(4)];

% quaternion based rotation matrix derivative target -> earth
Rd_et = [L1_t*ud_t L2_t*ud_t L3_t*ud_t];

% intermediate matrices to define rotation double dvt target -> earth
Ldt_udt = 4*[ud_t(1)^2+ud_t(2)^2 ud_t(2)*ud_t(3)-ud_t(1)*ud_t(4)...
ud_t(2)*ud_t(4)+ud_t(1)*ud_t(3);
ud_t(2)*ud_t(3)+ud_t(1)*ud_t(4) ud_t(1)^2+ud_t(3)^2 ud_t(3)*ud_t(4)...
-ud_t(1)*ud_t(2);
ud_t(2)*ud_t(4)-ud_t(1)*ud_t(3) ud_t(1)*ud_t(2)+ud_t(3)*ud_t(4)...
ud_t(1)^2+ud_t(4)^2];

% quaternion based rotation matrix double derivative target -> earth
Rdd_et = [L1_t*udd_t L2_t*udd_t L3_t*udd_t] + Ldt_udt;

% quaternion based rotation matrix from chaser -> inertial frame
R_ec = [2*u_c(1)^2-1+2*u_c(2)^2 2*u_c(2)*u_c(3)-2*u_c(1)*u_c(4) ...
2*u_c(2)*u_c(4)+2*u_c(1)*u_c(3);
2*u_c(2)*u_c(3)+2*u_c(1)*u_c(4) 2*u_c(1)^2-1+2*u_c(3)^2...
2*u_c(3)*u_c(4)-2*u_c(1)*u_c(2);
2*u_c(2)*u_c(4)-2*u_c(1)*u_c(3) 2*u_c(3)*u_c(4)+2*u_c(1)*u_c(2) ...
2*u_c(1)^2-1+2*u_c(4)^2];

% matrices used in defining rotation derivative matrix
L1_c = [4*u_c(1) 4*u_c(2) 0 0;
2*u_c(4) 2*u_c(3) 2*u_c(2) 2*u_c(1);
-2*u_c(3) 2*u_c(4) -2*u_c(1) 2*u_c(2)];

L2_c = [-2*u_c(4) 2*u_c(3) 2*u_c(2) -2*u_c(1);
4*u_c(1) 0 4*u_c(3) 0;
2*u_c(2) 2*u_c(1) 2*u_c(4) 2*u_c(3)];

```

```

L3_c = [2*u_c(3) 2*u_c(4) 2*u_c(1) 2*u_c(2);
-2*u_c(2) -2*u_c(1) 2*u_c(4) 2*u_c(3);
4*u_c(1) 0 0 4*u_c(4)];

% quaternion based rotation matrix derivative target -> earth
Rd_ec = [L1_c*ud_c L2_c*ud_c L3_c*ud_c];

% intermediate matrices to define rotation double dvt target -> earth
Ldc_udc = 4*[ud_c(1)^2+ud_c(2)^2 ud_c(2)*ud_c(3)-ud_c(1)*ud_c(4)...
ud_c(2)*ud_c(4)+ud_c(1)*ud_c(3);
ud_c(2)*ud_c(3)+ud_c(1)*ud_c(4) ...
ud_c(1)^2+ud_c(3)^2 ud_c(3)*ud_c(4)...
-ud_c(1)*ud_c(2);
ud_c(2)*ud_c(4)-ud_c(1)*ud_c(3)...
ud_c(1)*ud_c(2)+ud_c(3)*ud_c(4) ...
ud_c(1)^2+ud_c(4)^2];

% The position constraint R.H.S matrix
B_pos = rdd_t + Rdd_et*poi_rt...
+alpha_position*(rd_t+Rd_et*poi_rt-rd_c)...
+gamma_position*(r_t+R_et*poi_rt-r_c);

% angular velocity constraint R.H.S. matrix
r_poi_uvec = poi_rt/norm(poi_rt); % point of interest unit vector
rc_align = rc_align/norm(rc_align); % alignment unit vector

B_ang = -Ldc_udc*rc_align + Rdd_et*r_poi_uvec ...
+ alpha_align*(Rd_et*r_poi_uvec-Rd_ec*rc_align)...
+ gamma_align*(R_et*r_poi_uvec-R_ec*rc_align);

% rotational constraint A-matrix
Arot = rc_align(1)*L1_c+rc_align(2)*L2_c+rc_align(3)*L3_c;

% generalized mass matrix
E_c = [u_c';
-u_c(2) u_c(1) u_c(4) -u_c(3);
-u_c(3) -u_c(4) u_c(1) u_c(2);
-u_c(4) u_c(3) -u_c(2) u_c(1)];

Mgen_c = [M_c zeros(3,4);
zeros(4,3) E_c'*Jq_c*E_c];

```

```
% control constraint
A_c = [diag([1 1 1]) zeros(3,4);
zeros(3,3) Arot];
b_c = [B_pos;B_ang];

au_c = Mgen_c\Qext_c;

% Generalized force
Qcon_c = (A_c')*pinv(A_c*(Mgen_c)^(-1)*A_c')*(b_c-A_c*au_c);

E1 = [-u_c(2) u_c(1) u_c(4) -u_c(3);
-u_c(3) -u_c(4) u_c(1) u_c(2);
-u_c(4) u_c(3) -u_c(2) u_c(1)];

F_con = Qcon_c(1:3,1);
T_con = E1*Qcon_c(4:7,1);
```

B.1.4 target_orbit_motion block

```
function Qext_t = target_orbit_motion(q_t,M_t,mu,J2, R)

r_t = q_t(1:3,1);

r_norm = norm(r_t);
rdd_t = -mu*r_t/(r_norm^3);
a_J2 = -1.5*J2*mu*R^2*[(1-5*(r_t(3)/r_norm)^2)*r_t(1)/r_norm;...
(1-5*(r_t(3)/r_norm)^2)*r_t(2)/r_norm;...
(3-5*(r_t(3)/r_norm)^2)*r_t(3)/r_norm]/r_norm^4;

Qr = M_t*(rdd_t+a_J2);
Qu = [0 0 0 0]';

Qext_t = [Qr; Qu];
```

B.1.5 chaser_orbit_motion block

```
function Qext_c = chaser_orbit_motion(q_c,M_c,mu,J2, R)

r_c = q_c(1:3,1);
r_norm = norm(r_c);
rdd_c = -mu*r_c/(r_norm^3);
a_J2 = -1.5*J2*mu*R^2*[(1-5*(r_c(3)/r_norm)^2)*r_c(1)/r_norm;...
```

```

(1-5*(r_c(3)/r_norm)^2)*r_c(2)/r_norm;...
(3-5*(r_c(3)/r_norm)^2)*r_c(3)/r_norm]/r_norm^4;

Qr = M_c*(rdd_c+a_J2);

Qu = [0 0 0 0]';

Qext_c = [Qr; Qu];

```

B.1.6 target_dynamics block

```

function qdd_t = target_dynamics(q_t,qd_t,Qext_t,Jq_t,M_t)

%% abstract states -> real states
F_t = Qext_t(1:3,1); % applied force at target C.O.M

u_t = q_t(4:7,1); % quaternion of target frame
ud_t = qd_t(4:7,1); % quaternion derivative of target frame
Gamma_t = Qext_t(4:7,1); % quaternion torque

E_t = [u_t';
-u_t(2) u_t(1) u_t(4) -u_t(3);
-u_t(3) -u_t(4) u_t(1) u_t(2);
-u_t(4) u_t(3) -u_t(2) u_t(1)];

%% generalized matrices
% generalized mass matrix for target
Mgen_t = [M_t zeros(3,4);
zeros(4,3) E_t'*Jq_t*E_t];

Ed_t = [ud_t';
-ud_t(2) ud_t(1) ud_t(4) -ud_t(3);
-ud_t(3) -ud_t(4) ud_t(1) ud_t(2);
-ud_t(4) ud_t(3) -ud_t(2) ud_t(1)];

% generalized given force
Q_t = [F_t; Gamma_t/4-2*Ed_t'*Jq_t*E_t*ud_t...
-Jq_t(1,1)*(ud_t')*ud_t*u_t];
% J(1,1) is equal to J0

% unconstrained acceleration
au_t = Mgen_t\Q_t;

```

```

% constraint vectors
A_t = [0 0 0 u_t'];
b_t = -(u_t')*u_t;

% constrained acceleration
qdd_t = au_t + (Mgen_t)^(-1)*(A_t')*(A_t*(Mgen_t)^(-1)...
*A_t')^(-1)*(b_t-A_t*au_t);

```

B.1.7 chaser_dynamics block

```

function qdd_c = chaser_dynamics(q_c,qd_c,Qext_c,Jq_c,M_c,Qcon_c)

%% abstract states -> real states
Fext_c = Qext_c(1:3,1); % applied force at C.O.M
Fcon_c = Qcon_c(1:3,1); % control force at C.O.M.

u_c = q_c(4:7,1); % quaternion of chaser frame
ud_c = qd_c(4:7,1); % quaternion derivative of chaser frame
Gamma_c = Qext_c(4:7,1); % quaternion torque
Gammacon_c = Qcon_c(4:7,1); % quaternion control torque

E_c = [u_c';
-u_c(2) u_c(1) u_c(4) -u_c(3);
-u_c(3) -u_c(4) u_c(1) u_c(2);
-u_c(4) u_c(3) -u_c(2) u_c(1)];

%% generalized matrices
% generalized mass matrix
Mgen_c = [M_c zeros(3,4);
zeros(4,3) E_c'*Jq_c*E_c];

Ed_c = [ud_c';
-ud_c(2) ud_c(1) ud_c(4) -ud_c(3);
-ud_c(3) -ud_c(4) ud_c(1) ud_c(2);
-ud_c(4) ud_c(3) -ud_c(2) ud_c(1)];

% generalized given force
Q_c = [(Fext_c+Fcon_c); (Gamma_c+Gammacon_c)/4...
-2*Ed_c'*Jq_c*E_c*ud_c-Jq_c(1,1)*(ud_c')*ud_c*u_c]; ...
% J(1,1) is equal to J0

```



```

% unconstrained acceleration
au_c = Mgen_c\Q_c;

% constraint vectors
A_c = [0 0 0 u_c'];
b_c = -(ud_c')*ud_c;

% constrained acceleration
qdd_c = au_c + (Mgen_c)^(-1)*(A_c')*(A_c*(Mgen_c)^(-1)...
*A_c')^(-1)*(b_c-A_c*au_c);

```

Bibliography

- [1] Christiansen, E. L., *Meteoroid/Debris Shielding*, NASA Johnson Space Center Houston, Texas, 2003.
- [2] Kessler, D. J. and Cour-Palais, B. G., “Collision Frequency of Artificial Satellites: The Creation of a Debris Belt,” *Journal of Geophysical Research*, Vol. 83, 1960, pp. 2637–2646.
- [3] N. L. Johnson, E. Stansbery, D. O. W. K. J. A. and Shoots, D., *History of On-orbit Satellite Fragmentations 14th Edition*, NASA Johnson Space Center Houston, Texas, 2008.
- [4] Kelso, T. S., “Analysis of the Iridium 33-Cosmos 2251 Collision,” *AAS/AIAA Astrodynamics Specialist Conference*, Pittsburgh, PA, 2009, AAS paper 09-368.
- [5] Ansdell, M., “Active Space Debris Removal: Needs, Implications, and Recommendations for Today’s Geopolitical Environment,” *Journal of Public and International Affairs*, Vol. 21, No. 1, 2010.
- [6] J.-C. Liou, N. L. J. and Hill, N. M., “A Parametric Study on Using Active Debris Removal for LEO Environment Remediation,” *Acta Astronautica*, Vol. 66, No. 5, 2010.
- [7] M. Shan, J. G. and Gill, E., “Review and Comparison of Active Space Debris Capturing and Removal Methods,” *Progress in Aerospace Sciences*, Vol. 80, 2016, pp. 18–32.
- [8] Misra, A. K. and Diamond, G. S., “Dynamics of a Subsatellite System Supported by Two Tethers,” *Journal of Guidance, Control, and Dynamics*, Vol. 9, No. 1, 1986, pp. 12–16.
- [9] Hovell, K. and Ulrich, S., “Attitude Stabilization of an Uncooperative Spacecraft in an Orbital Environment Using Visco-elastic Tethers,” *AIAA Guidance, Navigation, and Control Conference*, AIAA, San Diego, California, 2016, p. AIAA paper 0641.
- [10] Hovell, K. and Ulrich, S., “Postcapture Dynamics and Experimental Validation of Subtethered Space Debris,” *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 2, 2018, pp. 519–525.
- [11] Z. Lin, V. Z. and Patel, R., “On-Line Robot Trajectory Planning for Catching a Moving Object,” *IEEE International Conference on Robotics and Automation*, IEEE, Piscataway, NJ, 1989, pp. 1726–1731.

- [12] R. Sharma, J.-Y. H. and Cucka, P., "Dynamic Robot Manipulation Using Visual Tracking," *IEEE International Conference on Robotics and Automation*, IEEE, Piscataway, NJ, 1992, pp. 1844–1849.
- [13] Lei, M. and Ghosh, B., "Visually Guided Robotic Tracking and Grasping of a Moving Object," *IEEE Conference on Decision and Control*, IEEE, Piscataway, NJ, 1993, pp. 1604–1609.
- [14] Chen, Y. and Watson, L. T., "Optimal Trajectory Planning for a Space Robot Docking with a Moving Target via Homotopy Algorithms," *Journal of Robotic Systems*, Vol. 12, No. 8, 1995, pp. 531–540.
- [15] Zhang, M. and Buehler, M., "Sensor-Based Online Trajectory Generation for Smoothly Grasping Moving Objects," *IEEE International Symposium on Intelligent Control*, IEEE, Piscataway, NJ, 1994, pp. 141–146.
- [16] Fitz-Coy, N. and Liu, M. C., "New Modified Proportional Navigation Scheme for Rendezvous and Docking with Tumbling Targets: The Planar Case," *Symposium on Flight Mechanics/Estimation Theory*, NASA, Maryland, 1995, pp. 243–252.
- [17] E. A. Croft, R. G. F. and Benhabib, B., "Optimal Rendezvous-Point Selection for Robotic Inteception of Moving Objects," *IEEE Transaction on Systems, Man, and Cybernetics*, Vol. 28, No. 2, 1998, pp. 192–204.
- [18] M. Mehrandezh, N. M. Sela, R. G. F. and Benhabib, B., "Robotic Interception of Moving Objects Using an Augmented Ideal Proportional Navigation Guidance Technique," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 30, No. 3, 2000, pp. 238–250.
- [19] Fehse, W., *Automated Rendezvous and Docking of Spacecraft*, Cambridge University Press, New York, 2003.
- [20] Z. Ma, O. M. and Shashikanth, B. N., "Optimal Approach to and Alignment with a Rotating Rigid Body for Capture," *Journal of the Astronautical Sciences*, Vol. 55, No. 4, 2007, pp. 407–419.
- [21] M. Romano, D. A. F. and Shay, T. J., "Laboratory Experimentation of Autonomous Spacecraft Approach and Docking to a Collaborative Target," *Journal of Spacecraft and Rockets*, Vol. 44, No. 1, 2007, pp. 164–173.
- [22] D. Zhang, S. S. and Pei, R., "Safe Guidance for Autonomous Rendezvous and Docking with a Non-Cooperative Target," *AIAA Guidance, Navigation, and Control Conference*, AIAA, Toronto, 2010, pp. AIAA paper 2010–7592.
- [23] Aghili, F., "A Prediction and Motion-Planning Scheme for Visually Guided Robotic Capturing of Free-Floating Tumbling Objects with Uncertain Dynamics," *IEEE Transactions on Robotics*, Vol. 28, No. 3, 2012, pp. 634–649.

- [24] M. Wilde, M. Ciarci, A. G. and Romano, M., “Experimental Characterization of Inverse Dynamics Guidance in Docking with a Rotating Target,” *Journal of Guidance, Control and Dynamics*, Vol. 39, No. 6, 2016, pp. 1173–1187.
- [25] J. Ventura, M. Ciarci, M. R. and Ulrich, W., “Fast and Near-Optimal Guidance for Docking to Uncontrolled Spacecraft,” *Journal of Guidance, Control and Dynamics*, Vol. 40, No. 12, 2017, pp. 3138–3154.
- [26] Zagaris, C. and Romano, M., “Reachability Analysis of Planar Spacecraft Docking with Rotating Body in Close Proximity,” *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 6, 2018, pp. 1416–1422.
- [27] H. Dong, Q. H. and Akella, M. R., “Safety Control for Spacecraft Autonomous Rendezvous and Docking Under Motion Constraints,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 7, 2017, pp. 1680–1692.
- [28] R. Kristiansen, P. J. N. and Gravdahl, J. T., “Spacecraft Coordination Control in 6DOF: Integrator Backstepping vs Passivity-based Control,” *Automatica*, Vol. 44, No. 11, 2008, pp. 2896–2901.
- [29] Sun, L. and Huo, W., “6-DOF Integrated Adaptive Backstepping Control for Spacecraft Proximity Operations,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 51, No. 3, 2015, pp. 2433–2443.
- [30] L. Sun, W. H. and Jiao, Z., “Robust Nonlinear Adaptive Relative Pose Control for Cooperative Spacecraft During Rendezvous and Proximity Operations,” *IEEE Transactions on Control Systems Technology*, Vol. 25, No. 5, 2017, pp. 1840–1847.
- [31] Zou, A.-M. and Kumar, K. D., “Distributed Attitude Coordination Control for Spacecraft Formation Flying,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 48, No. 2, 2012, pp. 1329–1346.
- [32] J. Wang, H. Liang, Z. S. S. Z. and Liu, M., “Finite-Time Control for Spacecraft Formation with Dual-Number-Based Description,” *Journal of Guidance, Control, and Dynamics*, Vol. 35, 2012, pp. 950–962.
- [33] Wang, J. and Sun, Z., “6-DOF Robust Adaptive Terminal Sliding Mode Control for Spacecraft Formation Flying,” *Acta Astronautica*, Vol. 73, 2012, pp. 76–87.
- [34] Filipe, N. and Tsiotras, P., “Adaptive Position and Attitude-Tracking Controller for Satellite Proximity Operations Using Dual Quaternions,” *Journal of Guidance, Control and Dynamics*, Vol. 38, No. 4, 2014, pp. 566–577.
- [35] Lee, U. and Mesbahi, M., “Dual Quaternion based Spacecraft Rendezvous with Rotational and Translational Field of View Constraints,” *AIAA/AAS Astrodynamics Specialist Conference*, AIAA SPACE Forum, San Diego, CA, 2014, pp. AIAA 2014–4362.

- [36] H. Dong, Q. H. and Akella, M. R., “Dual-Quaternion-Based Spacecraft Autonomous Rendezvous and Docking Under Six-Degree-of-Freedom Motion Constraints,” *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 5, 2018, pp. 1150–1162.
- [37] Gui, H. and Vukovich, G., “Finite-time Output-feedback Position and Attitude Tracking of a Rigid Body,” *Automatica*, Vol. 74, 2016, pp. 270–278.
- [38] N. Filipe, A. V. and Tsiotras, P., “Pose Tracking Without Linear and Angular Velocity Feedback Using Dual Quaternions,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 52, No. 1, 2016, pp. 411–422.
- [39] H. Dong, Q. Hu, M. I. F. and Ma, G., “Dual-Quaternion-Based Fault-Tolerant Control for Spacecraft Tracking With Finite-Time Convergence,” *IEEE Transactions on Control Systems Technology*, Vol. 25, No. 4, 2017, pp. 1231–1242.
- [40] Gui, H. and de Ruiter, A. H. J., “Adaptive Fault-Tolerant Spacecraft Pose Tracking With Control Allocation,” *IEEE Transactions on Control Systems Technology*, Vol. 27, No. 2, 2019, pp. 479–494.
- [41] Gui, H. and Vukovich, G., “Dual-Quaternion-based Adaptive Motion Tracking of Spacecraft with Reduced Control Effort,” *Nonlinear Dynamics*, Vol. 83, No. 1, 2016.
- [42] Udwadia, F. E. and Kalaba, R. E., *Analytical Dynamics: A New Approach*, Cambridge University Press, New York, 1996.
- [43] Udwadia, F. E. and Schutte, A. D., “A Unified Approach to Rigid Body Rotational Dynamics and Control,” *Proceedings of the Royal Society*, Vol. 468, No. 2138, 2011, pp. 395–414.
- [44] Koganti, P. B. and Udwadia, F. E., “Dynamics and Precision Control of Tumbling Multibody Systems,” *Journal of Guidance, Control and Dynamics*, Vol. 40, No. 3, 2017, pp. 584–602.
- [45] Gauss, C. F., “Über ein neues allgemeines Grundgesetz der Mechanik,” *Journal für die reine und angewandte Mathematik*, Vol. 1829, No. 4, 1829, pp. 232–235.
- [46] Curtis, H. D., *Orbital Mechanics for Engineering Students 3rd Edition*, Butterworth-Heinemann, Oxford, UK, 2013.
- [47] Udwadia, F. and Schutte, A., “An Alternative Derivation of the Quaternion Equations of Motion for Rigid-Body Rotational Dynamics,” *Journal of Applied Mechanics*, Vol. 77, No. 4, 2010, pp. 044505.
- [48] S. Ulrich, A. S.-O. and Barkana, I., “Passivity-Based Adaptive Control of Robotic Spacecraft for Proximity Operations Under Uncertainties,” *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 6, 2016, pp. 1441–1450.

- [49] J.-F. Shi, S. Ulrich, S. R. and Anctil, M., “Uncooperative Spacecraft Pose Estimation Using an Infrared Camera During Proximity Operations,” *AIAA Space Forum*, AIAA, Pasadena, California, 2015.
- [50] F. E. Udwadia, T. W. and Cho, H., “Methodology for Satellite Formation-Keeping in the Presence of System Uncertainties,” *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 5, 2014, pp. 1611–1624.
- [51] Hovell, K., “Detumbling Space Debris Using Tethers,” *Master’s thesis, Carleton University, 1125 Colonel By Drive, Ottawa, Ontario, Canada*, 2017.