

OPTIMAL CONTROL OF NONHOLONOMIC MECHANICAL SYSTEMS

by

Stuart Marcus Rogers

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Applied Mathematics

Department of Mathematical and Statistical Sciences
University of Alberta

© Stuart Marcus Rogers, 2017

Abstract

This thesis investigates the optimal control of two nonholonomic mechanical systems, Suslov's problem and the rolling ball. Suslov's problem is a nonholonomic variation of the classical rotating free rigid body problem, in which the body angular velocity $\boldsymbol{\Omega}(t)$ must always be orthogonal to a prescribed, time-varying body frame vector $\boldsymbol{\xi}(t)$, i.e. $\langle \boldsymbol{\Omega}(t), \boldsymbol{\xi}(t) \rangle = 0$. The motion of the rigid body in Suslov's problem is actuated via $\boldsymbol{\xi}(t)$, while the motion of the rolling ball is actuated via internal point masses that move along rails fixed within the ball. First, by applying Lagrange-d'Alembert's principle with Euler-Poincaré's method, the uncontrolled equations of motion are derived. Then, by applying Pontryagin's minimum principle, the controlled equations of motion are derived, a solution of which obeys the uncontrolled equations of motion, satisfies prescribed initial and final conditions, and minimizes a prescribed performance index. Finally, the controlled equations of motion are solved numerically by a continuation method, starting from an initial solution obtained analytically (in the case of Suslov's problem) or via a direct method (in the case of the rolling ball).

Preface

This thesis contains material that has appeared in a pair of papers, one on Suslov’s problem and the other on rolling ball robots, co-authored with my supervisor, Vakhtang Putkaradze. Chapter 4 contains much of the material published in the paper [1] on Suslov’s problem: Vakhtang Putkaradze and Stuart Rogers, “Constraint Control of Nonholonomic Mechanical Systems,” *Journal of Nonlinear Science*, DOI: 10.1007/s00332-017-9406-1, 2017. Chapter 5 contains much of the material in the paper [2] on rolling ball robots that has been submitted for publication in *Optimal Control, Applications and Methods* and posted online on *arXiv*: Vakhtang Putkaradze and Stuart Rogers, “Optimal Control of a Rolling Ball Robot Actuated by Internal Point Masses,” *arXiv preprint arXiv:1708.03829*, 2017.

Rome wasn't built in a day, but they were laying bricks every hour.

Acknowledgements

I wish to thank my supervisor, Vakhtang Putkaradze, for bringing me to the University of Alberta, suggesting the topics researched in this thesis, and showing me how to apply the calculus of variations to investigate mechanical systems.

There were fruitful discussions with Profs. A.M. Bloch, D.V. Zenkov, M. Leok, and A. Lewis concerning geometric mechanics and optimal control. In particular, Prof D.V. Zenkov suggested the new variation of Suslov's problem investigated in this thesis, in which ξ may vary with time.

Prof. H. Oberle suggested that I try to use the ODE TPBVP solver `bvp4c` for numerically solving the controlled equations of motion. Prof. L.F. Shampine provided advice on using the ODE TPBVP solvers `bvp4c` and `bvp5c`. Prof. F. Mazzia provided information on using the ODE TPBVP solvers `TOM` and `bvptwp`. J. Willkomm provided help on using the automatic differentiation software `ADiMat` and suggested that I investigate the alternative automatic differentiation software `ADiGator`. M. Weinsten provided copious advice on using `ADiGator` and fixed numerous bugs in `ADiGator` that were revealed in the course of this research. Prof. A. Rao fixed a critical bug in the direct method optimal control solver `GPOPS-II` that was revealed in the course of this research. In return, Prof. A. Rao provided me with a free license to use `GPOPS-II` for this research.

I would like to thank the following funding sources: the University of Alberta Doctoral Recruitment Scholarship, the NSERC Discovery Grant, the University of Alberta Centennial Fund, the FGSR Graduate Travel Award, the IGR Travel Award, the GSA Academic Travel Award, the AMS Fall Sectional Graduate Student Travel Grant, and the Alberta Innovates Technology Funding (AITF) which came through the Alberta Centre for Earth Observation Sciences (CEOS).

Finally, I thank all my teachers, both past and present.

Table of Contents

1	Introduction	1
1.A	Motivation and Methodology for this Research	1
1.B	Background on Suslov’s Problem	3
1.C	Background on the Rolling Ball	3
1.D	Contributions and Thesis Summary	5
2	Mechanics	8
2.A	Hamilton’s Principle, Symmetry Reduction, and Euler-Poincaré’s Method	8
2.B	Nonholonomic Constraints and Lagrange-d’Alembert’s Principle	15
2.C	A Simple Nonholonomic Particle	16
3	Optimal Control	22
3.A	Calculus of Variations	22
3.B	Pontryagin’s Minimum Principle	23
3.C	Implementation Details for Solving the Optimal Control ODE TPBVP	26
4	Suslov’s Problem	33
4.A	Suslov’s Optimal Control Problem for an Arbitrary Group	33
4.A.1	Derivation of Suslov’s Uncontrolled Equations of Motion	33
4.A.2	Derivation of Suslov’s Controlled Equations of Motion	35
4.B	Suslov’s Optimal Control Problem for Rigid Body Motion	38

4.B.1	Derivation of Suslov’s Uncontrolled Equations of Motion	38
4.B.2	Controllability and Accessibility of Suslov’s Uncontrolled Equations of Motion	42
4.B.3	Suslov’s Optimal Control Problem	46
4.B.4	Derivation of Suslov’s Controlled Equations of Motion	47
4.C	Numerical Solution of Suslov’s Optimal Control Problem	57
4.C.1	Analytical Solution of a Singular Version of Suslov’s Optimal Control Problem	57
4.C.2	Numerical Solution of Suslov’s Optimal Control Problem via Continuation	58
4.C.3	Numerical Solution of Suslov’s Optimal Control Problem via the Indirect Method and Continuation	59
5	The Rolling Ball	64
5.A	Mechanical System and Motivation	64
5.A.1	Coordinate Systems and Notation	65
5.B	Uncontrolled Equations of Motion	67
5.B.1	Kinetic Energy, Potential Energy, and Lagrangian	67
5.B.2	Rolling Constraint and Lagrange-d’Alembert’s Principle	69
5.B.3	Uncontrolled Equations of Motion for the Rolling Ball	73
5.B.4	Uncontrolled Equations of Motion for the Rolling Disk	77
5.C	Controlled Equations of Motion	84
5.C.1	Controlled Equations of Motion for the Rolling Disk	84
5.C.2	Controlled Equations of Motion for the Rolling Ball	89
5.D	Numerical Solutions of the Controlled Equations of Motion	98
5.D.1	Simulations of the Rolling Disk	99
5.D.2	Simulations of the Rolling Ball	102
6	Conclusions	111
6.A	Summary and Conclusions	111
6.B	Future Work	113

Bibliography	116
Appendices	125
A Survey of Numerical Methods for Solving Optimal Control Problems: Dynamic Programming, the Direct Method, and the Indirect Method	125
B Calculation Connecting Classical and Reduced Costates for Suslov’s Optimal Control Problem	129
C Predictor-Corrector Continuation Method for Solving an ODE TPBVP	132
C.1 Introduction	132
C.2 A Hilbert Space	133
C.3 The Fréchet Derivative and Newton’s Method	133
C.4 The Davidenko ODE IVP	134
C.5 Construct the Tangent	135
C.6 Normalize the Tangent	138
C.7 Construct the Tangent Predictor	138
C.8 Construct the Corrector	139
C.9 Polish the Corrector	142
C.10 Pseudocode for Predictor-Corrector Continuation	143
D Sweep Predictor-Corrector Continuation Method for Solving an ODE TPBVP	146
D.1 Introduction	146
D.2 Construct the Tangent	146
D.3 Determine the Tangent Direction	148
D.4 Sweep along the Tangent	149
D.5 Pseudocode for Sweep Predictor-Corrector Continuation	151
E Quaternions	153

List of Figures

1.1	Examples of real rolling ball robots.	1
1.2	Suslov’s problem studies the motion of a rotating rigid body subject to $\langle \boldsymbol{\Omega}(t), \boldsymbol{\xi}(t) \rangle = 0$. In Suslov’s original formulation, $\boldsymbol{\xi}(t) \equiv \boldsymbol{\xi}$ is fixed (i.e. does not vary with respect to time) in the body frame.	3
1.3	Idealized realization of Suslov’s problem using a pair of diametrically opposed ice skates wedged inside a spherical ice shell. The grey shaded blob represents the rigid body rotating about the fixed point O . At this time instant, the fixed vector $\boldsymbol{\xi}$ lies in the plane of the page and is orthogonal to the rotation axis, which points out of or into the page and which is parallel to $\boldsymbol{\Omega}$	4
1.4	A ball of radius r rolls without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g . The ball’s geometric center, center of mass, and contact point with the flat surface are denoted by GC, CM, and CP, respectively.	5
1.5	Different methods to actuate a rolling ball.	6
2.1	Illustrations of the tangent bundles of several manifolds.	8
2.2	Lagrange-d’Alembert vs vakonomic solutions for a simple nonholonomic particle.	20
4.1	The desired body angular velocity is approximately the projection of a spiral onto the constant kinetic energy ellipsoid.	60
4.2	Numerical solution of the optimal control problem for $\alpha = 1, \beta = .1, \gamma = 1, \eta = 1, \delta = .2$, and free final time. The optimal final time is $b = 11.36$	62
4.3	Numerical solution of the optimal control problem for $\alpha = 1, \beta = .1, \gamma = 1, \eta = .01, \delta = .2$, and free final time. The optimal final time is $b = 9.84$	63

5.1	A ball of radius r and mass m_0 rolls without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g . The ball's center of mass is denoted by m_0 . In addition, the ball contains 2 internal point masses, m_1 and m_2 , that may move within the ball. How must m_1 and m_2 be moved to induce the ball to follow the prescribed trajectory \mathbf{z}_d ?	65
5.2	A ball of radius r and mass m_0 rolls without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g . The ball's geometric center, center of mass, and contact point with the flat surface are denoted by GC, m_0 , and CP, respectively. The ball's motion is actuated by n point masses, each of mass m_i , $1 \leq i \leq n$, that move inside the ball. The spatial frame has origin located at height r above the flat surface and orthonormal axes \mathbf{e}_1 , \mathbf{e}_2 , and \mathbf{e}_3 . The body frame has origin located at the ball's center of mass (denoted by m_0) and orthonormal axes \mathbf{E}_1 , \mathbf{E}_2 , and \mathbf{E}_3 . All vectors inside the ball are expressed with respect to the body frame, while all vectors outside the ball are expressed with respect to the spatial frame.	67
5.3	Each control mass, denoted by m_i , $1 \leq i \leq n$, moves along a control rail fixed inside the ball depicted here by the dashed hoop. The position of the control rail is denoted by ζ_i and is parameterized by θ_i .	77
5.4	A disk of radius r and mass m_0 rolls without slipping in the \mathbf{e}_1 - \mathbf{e}_3 plane. \mathbf{e}_2 and \mathbf{E}_2 are directed into the page and are omitted from the figure. The disk's center of mass is denoted by m_0 . The disk's motion is actuated by n point masses, each of mass m_i , $1 \leq i \leq n$, that move along control rails fixed inside the disk. The control mass depicted here by m_i moves along a circular hoop in the disk that is not centered on the disk's geometric center (GC). The disk's orientation is determined by ϕ , the angle measured counter-clockwise from \mathbf{e}_1 to \mathbf{E}_1 .	78
5.5	Portraits of Newton and Lagrange, progenitors of classical mechanics.	82
5.6	A disk actuated by a single control mass.	83
5.7	Plot of the desired path of the disk's GC. The disk's GC starts from rest at $z = 0$ at time $t = 0$, moves to the right for $0 < t < 2$, and stops at rest at $z = 10$ at time $t = 2$.	87
5.8	Plot of the desired path of the ball's GC. The ball's GC starts from rest at $\mathbf{z} = \begin{bmatrix} 0 & 0 \end{bmatrix}^T$ at time $t = 0$ and stops at rest at $\mathbf{z} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$ at time $t = .5$.	94
5.9	The disk of radius $r = 1$ actuated by 4 control masses, m_1 , m_2 , m_3 , and m_4 , each on its own circular control rail. The control rail radii are $r_1 = .9$, $r_2 = .6\bar{3}$, $r_3 = .3\bar{6}$, and $r_4 = .1$. The location of the disk's CM is denoted by m_0 .	101
5.10	Numerical solutions of the rolling disk optimal control problem (5.103) using 4 control masses for $\beta = 0$, $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = .1$, and fixed initial and final times. The direct method results for $\alpha = 20$ are shown in the left column, while the predictor-corrector continuation indirect method results for $\alpha = 54,866$ are shown in the right column.	104

5.11	Evolution of various parameters during the predictor-corrector continuation indirect method, which starts from the direct method solution, used to solve the rolling disk optimal control problem (5.103). Note the pair of turning points at solutions 14 and 19. The maximum tangent steplength σ_{\max} is fixed to 30.	105
5.12	Numerical solutions of the rolling disk optimal control problem (5.103) using 4 control masses for $\beta = 0$, $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = .1$, and fixed initial and final times. The direct method results for $\alpha = 20$ are shown in the left column, while the predictor-corrector continuation indirect method results for $\alpha = 1,238,285$ are shown in the right column.	106
5.13	Evolution of various parameters during the predictor-corrector continuation indirect method, which starts from the direct method solution, used to solve the rolling disk optimal control problem (5.103). The maximum tangent steplength σ_{\max} is increased linearly after passing the turning points.	107
5.14	The ball of radius $r = 1$ actuated by 3 control masses, m_1 , m_2 , and m_3 , each on its own circular control rail. The control rail radii are $r_1 = .95$, $r_2 = .9$, and $r_3 = .85$. The location of the ball's CM is denoted by m_0	108
5.15	Numerical solutions of the rolling ball optimal control problem (5.141) using 3 control masses for $\alpha = 20$, $\beta = 0$, $\gamma_1 = \gamma_2 = \gamma_3 = 10$, and fixed initial and final times. The obstacle centers are located at $\mathbf{v}_1 = [v_{1,1} \ v_{1,2}]^T = [.3 \ .3]^T$ and $\mathbf{v}_2 = [v_{2,1} \ v_{2,2}]^T = [.7 \ .7]^T$ and the obstacle radii are $\rho_1 = \rho_2 = .2$. The direct method results for obstacle heights at $h_1 = h_2 = .1$ are shown in the left column, while the predictor-corrector continuation indirect method results for obstacle heights at $h_1 = h_2 = 1,495,740$ are shown in the right column. . .	109
5.16	Evolution of various parameters during the predictor-corrector continuation indirect method, which starts from the direct method solution, used to solve the rolling ball optimal control problem (5.141).	110
C.1	Predictor-corrector continuation.	136

List of Tables

3.1	Explanation of shorthand notation for first and second derivatives of \hat{H} used in (3.22) and (3.23).	28
3.2	Equality between Jacobians of boundary condition functions in normalized and un-normalized coordinates.	31
5.1	Initial condition parameter values for the rolling disk. Refer to (5.92) and (5.93).	100
5.2	Final condition parameter values for the rolling disk. Refer to (5.93).	100
5.3	Integrand cost function coefficient values for the rolling disk. Refer to (5.95).	100
5.4	Initial condition parameter values for the rolling ball. Refer to (5.132).	102
5.5	Final condition parameter values for the rolling ball. Refer to (5.133).	102
5.6	Integrand cost function coefficient values for the rolling ball. Refer to (5.135) and (5.140). . .	103

List of Algorithms

C.1	Predictor-Corrector Continuation for Nonlinear ODE TPBVPs. Part 1.	144
C.1	Predictor-Corrector Continuation for Nonlinear ODE TPBVPs. Part 2.	145
D.2	Sweep Predictor-Corrector Continuation for Nonlinear ODE TPBVPs.	152

Chapter 1

Introduction

1.A Motivation and Methodology for this Research

The first six films in the famous *Star Wars* space saga starred the sidekick robot R2-D2, which locomoted via a three-wheeled tripod. However, the seventh film in that saga, *The Force Awakens*, stars a new, next-generation, sidekick robot called BB-8. BB-8, depicted in Figure 1.1a, locomotes via a single rolling ball. To cash in on this new *Star Wars* fan favorite, the toy companies Sphero and Hasbro sell working toy models of BB-8. But rolling ball robots are not just gimmicks used by the entertainment and toy industries. The defense, security, energy, and agricultural industries are also interested in exploiting sensor-equipped rolling ball robots, such as Rosphere shown in Figure 1.1b, for such tasks as surveillance and environmental monitoring. How are rolling ball robots like BB-8 and Rosphere controlled to locomote over a prescribed trajectory?



(a) Sphero's toy incarnation of BB-8, *Star Wars*' next-generation rolling ball robot [3].



(b) Rosphere can be used in agriculture for monitoring crops, © 2013 Emerald [4].

Figure 1.1: Examples of real rolling ball robots.

More specifically, suppose a rolling ball like BB-8 or Rosphere is actuated by some internal mechanism which may be controlled, such as by spinning internal rotors, by swinging an internal pendulum, or by moving internal point masses along rails fixed within the ball. In addition, suppose initial and final conditions, like the ball's initial and final positions and velocities, algebraic (i.e. non-differential) path inequality constraints, like engineering limitations on the internal mechanism's acceleration, and a performance index, such as the

mean error between the ball's actual and prescribed trajectory, are prescribed. How can the ball's internal mechanism be controlled to minimize the prescribed performance index while satisfying the prescribed initial and final conditions and the prescribed algebraic path inequality constraints? This is an example of a so-called optimal control problem. In the absence of algebraic path inequality constraints, this thesis offers a solution to this optimal control problem by applying methods from mechanics and optimal control.

In this thesis, a dynamical system is always assumed to be a system that evolves according to a collection of ordinary differential equations. Optimal control is an optimization technique to derive a control policy for governing a dynamical system that minimizes some prescribed performance index such as a cost function or a cost functional, while satisfying prescribed initial and final conditions, prescribed algebraic path inequality constraints, and the ordinary differential equations that define the dynamical system. An optimal control problem may be solved by dynamic programming, Pontryagin's minimum principle, or the direct method. Dynamic programming, developed by Richard Bellman in the 1950s, formulates the Hamilton-Jacobi-Bellman partial differential equation (HJB PDE). Pontryagin's minimum principle, developed by Lev Pontryagin in the 1950s, formulates an ordinary differential equation two-point boundary value problem (ODE TPBVP). In the literature, Pontryagin's minimum principle is often referred to as the indirect method. The direct method recasts the infinite-dimensional optimal control problem as a finite-dimensional parameter optimization problem, i.e. a nonlinear programming (NLP) problem, by discretizing time and by approximating the unknown controls via an unknown linear combination of a finite set of given basis functions. Because the HJB PDE is difficult to solve due to the curse of dimensionality, this thesis relies on the direct and indirect methods to solve optimal control problems.

However, before optimal control can be applied to the rolling ball or any other dynamical system, its ordinary differential equations of motion must be derived first; henceforth, the ordinary differential equations of motion of a dynamical system will often be referred to as the equations of motion or the uncontrolled equations of motion to distinguish them from the controlled equations of motion which may be obtained by the indirect method. To derive the uncontrolled equations of motion for the rolling ball, methods from mechanics must be utilized. The rolling ball is an example of a nonholonomic (as opposed to a holonomic) mechanical system. A mechanical system is said to be nonholonomic (holonomic) if its physical motion constraints depend on the system's velocity (position). The uncontrolled equations of motion governing a nonholonomic mechanical system are given by Lagrange-d'Alembert's principle, a somewhat nonintuitive method in mechanics developed by Jean d'Alembert in the 18th century. In addition, Euler-Poincaré's method [5], published by Henri Poincaré in 1901, provides a more efficient derivation of the equations of motion of a mechanical system than Hamilton's principle.

Given a nonholonomic mechanical system, such as the rolling ball, the procedure to formulate and solve an optimal control problem in this thesis is as follows. First, the uncontrolled equations of motion are derived by applying Lagrange-d'Alembert's principle with Euler-Poincaré's method, from which an optimal control problem is formulated, given initial and final conditions and a performance index to be minimized. Next, Pontryagin's minimum principle is used to derive the controlled equations of motion, a solution of which obeys the uncontrolled equations of motion, satisfies the prescribed initial and final conditions, and minimizes the prescribed performance index. Finally, the controlled equations of motion are solved numerically by continuation, starting from an initial solution to a simpler optimal control problem obtained via analytics or a direct method.

Before investigating the optimal control of the rolling ball, this thesis first investigates the optimal control of Suslov’s problem because it is a much simpler nonholonomic mechanical system. In summary, this thesis uses the procedure just described to investigate the optimal control of two particular nonholonomic mechanical systems, Suslov’s problem and the rolling ball. The optimal control of other nonholonomic mechanical systems, namely the continuous variable transmission and the Chaplgyin sleigh, has been investigated recently in [6].

1.B Background on Suslov’s Problem

In 1902, Suslov [7] considered the motion of a rigid body that rotates about a fixed point but which is constrained such that its body angular velocity Ω must be orthogonal to a prescribed body frame vector ξ . Such a rigid body, depicted in Figure 1.2 using a coffee mug, is therefore called Suslov’s problem. Mathematically, the constraint for Suslov’s problem is $\langle \Omega(t), \xi(t) \rangle = 0$, and so Suslov’s problem is an example of a nonholonomic mechanical system.

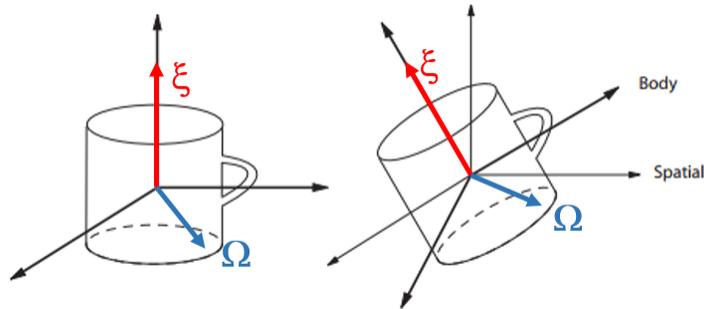


Figure 1.2: Suslov’s problem studies the motion of a rotating rigid body subject to $\langle \Omega(t), \xi(t) \rangle = 0$. In Suslov’s original formulation, $\xi(t) \equiv \xi$ is fixed (i.e. does not vary with respect to time) in the body frame.

In [7], Suslov derived the equations of motion for Suslov’s problem assuming that ξ is fixed. Reference [8] reviews how Suslov’s problem might be physically realized, as illustrated in Figure 1.3, though Suslov’s problem would be difficult to actually build, requiring idealized (i.e. unrealizable) conditions. Even though it is probably not physically realizable, Suslov’s problem is a frequent and active topic in the nonholonomic mechanics literature [9, 10, 11, 12, 13, 8, 14] due to its mathematical simplicity. In fact, Suslov’s problem is perhaps the simplest nontrivial nonholonomic mechanical system. For this reason, Suslov’s problem is investigated in this thesis before attacking the more complicated rolling ball. However, this thesis considers a new variation of Suslov’s problem, in which ξ may vary with time so that ξ may serve as a control, thereby transforming Suslov’s problem from a pure dynamics problem into an optimal control problem.

1.C Background on the Rolling Ball

Consider a ball rolling without slipping on a flat surface in the presence of a uniform gravitational field. Figure 1.4 shows a ball of radius r rolling without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g .

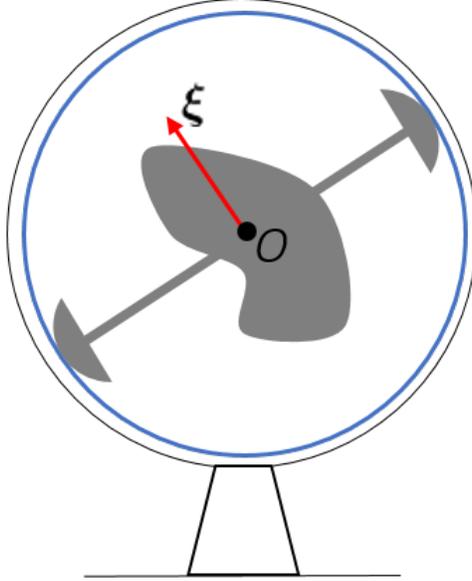


Figure 1.3: Idealized realization of Suslov's problem using a pair of diametrically opposed ice skates wedged inside a spherical ice shell. The grey shaded blob represents the rigid body rotating about the fixed point O . At this time instant, the fixed vector ξ lies in the plane of the page and is orthogonal to the rotation axis, which points out of or into the page and which is parallel to Ω .

There are several terminologies in the literature to describe a ball rolling without slipping on a flat surface in the presence of a uniform gravitational field, depending on its mass distribution and the location of its center of mass. A Chaplygin sphere is a ball with an inhomogeneous mass distribution, but with its center of mass located at the ball's geometric center [15]. A Chaplygin top is a ball with an inhomogeneous mass distribution, but with its center of mass not located at the ball's geometric center [15]. Reference [16] does not distinguish between these two cases, calling a Chaplygin ball a ball with an inhomogeneous mass distribution, regardless of the location of its center of mass; as a special case of a Chaplygin ball, [16] calls a Chaplygin concentric sphere a ball with an inhomogeneous mass distribution with its center of mass coinciding with the ball's geometric center. Thus, the Chaplygin concentric sphere (used by [16]) and the Chaplygin sphere (used by [15]) are different terms for the same mechanical system. Note that a ball with a homogeneous mass distribution (in a uniform gravitational field) necessarily has its center of mass at the ball's geometric center, and is therefore not very interesting. In this thesis, these terminologies are not used, rather the mechanical system is referred to simply as a ball or a rolling ball, regardless of its mass distribution (homogeneous vs inhomogeneous) and regardless of the location of its center of mass (at the ball's geometric center vs not at the ball's geometric center).

In this thesis, the motion of the rolling ball is investigated assuming both static and dynamic internal structure. The dynamics of the rolling ball with static internal structure was first solved analytically by Chaplygin for the cylindrically symmetric rolling ball, i.e. a ball such that the line joining the ball's center of mass and geometric center forms an axis of symmetry, in 1897 [17] and for the Chaplygin sphere in 1903 [18], though dynamical properties of the cylindrically symmetric rolling ball were previously investigated by Routh [19] and Jellet [20]. The dynamics of the rolling ball with dynamic internal structure is still an active topic in the nonholonomic mechanics literature [21, 22, 15, 23, 24, 25, 26, 27, 28].

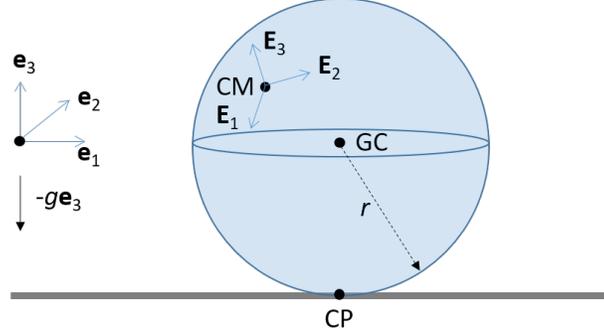


Figure 1.4: A ball of radius r rolls without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g . The ball's geometric center, center of mass, and contact point with the flat surface are denoted by GC, CM, and CP, respectively.

Many methods have been proposed (and some realized) to actuate a rolling ball, such as illustrated in Figure 1.5. This thesis considers a rolling ball actuated by internal point masses that move along rails fixed within the ball, such as depicted in Figure 1.5c. Actuating the rolling ball by moving internal point masses along general control rails has not been considered yet in the literature; [28] considers a very special case where 6 magnets each move inside its own linear tube as shown in Figure 1.5b. Prior to [28] and this thesis, controlling the motion of a nonholonomic mechanical system by moving internal point masses has been studied previously in [29], which investigates the controlled motion of the Chaplygin sleigh actuated by a single internal point mass.

In a comprehensive review of nonholonomic optimal control, [30] briefly discusses the optimal control of a rolling ball, where an external control force pushes the ball's geometric center. While several papers [21, 22, 15, 23, 24, 25, 26, 27] have investigated methods to control the rolling ball, none have used the optimal control methods investigated in this thesis.

1.D Contributions and Thesis Summary

The key contributions of this thesis are listed below.

- The uncontrolled equations of motion are derived for a variation of Suslov's problem where ξ is permitted to vary with time; in Suslov's original formulation, ξ was assumed to be fixed. Controllability of this variation of Suslov's problem is demonstrated. Controlled equations of motion for this variation of Suslov's problem are derived, a solution of which obeys the uncontrolled equations of motion, satisfies prescribed initial and final conditions, and minimizes a prescribed performance index. A singular case of the controlled equations of motion is solved analytically. The controlled equations of motion are solved numerically by a monotonic continuation method, starting from the analytical solution to the singular case.
- The uncontrolled equations of motion are derived for a rolling ball actuated by internal point masses that move along rails fixed within the ball. By using automatic differentiation, controlled equations

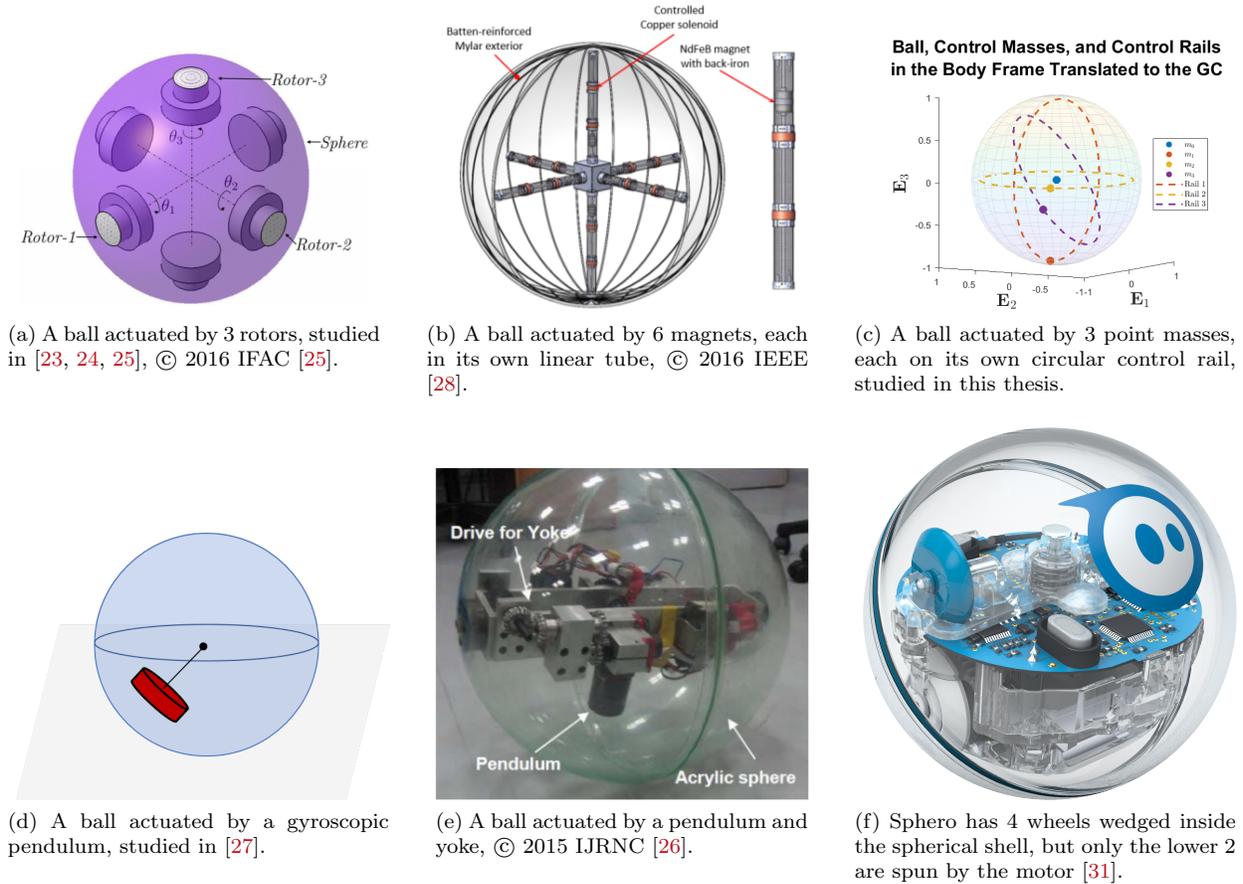


Figure 1.5: Different methods to actuate a rolling ball.

of motion for this rolling ball are constructed numerically, a solution of which obeys the uncontrolled equations of motion, satisfies prescribed initial and final conditions, and minimizes a prescribed performance index. The controlled equations of motion are solved numerically by a predictor-corrector continuation method, starting from an initial solution provided by a direct method.

- Jacobians of the ordinary differential equations (ODEs) and boundary conditions (BCs) which constitute the controlled equations of motion (i.e. an ODE TPBVP) corresponding to the optimal control of a dynamical system are derived. These Jacobians are useful for numerically solving the controlled equations of motion.
- Algorithms for solving an ODE TPBVP by predictor-corrector continuation are developed and were implemented in MATLAB to numerically solve the controlled equations of motion for the rolling ball. There are not very many predictor-corrector continuation methods publicly available for solving dynamical systems. The idea of using a monotonic continuation ODE TPBVP solver in conjunction with a predictor-corrector continuation method to advance (or “sweep”) as far along the tangent as possible is new.

The thesis is organized as follows. Chapter 2 reviews methods from mechanics needed to derive the uncontrolled equations of motion for holonomic and nonholonomic mechanical systems. Chapter 3 reviews the

theory of optimal control needed to derive the controlled equations of motion for a generic dynamical system given initial and final conditions, given a performance index to be minimized, and in the absence of path inequality constraints. Chapter 4 derives the uncontrolled and controlled equations of motion and numerically solves the controlled equations of motion for Suslov's problem. Chapter 5 derives the uncontrolled equations of motion and numerically constructs and solves the controlled equations of motion for the rolling ball actuated by internal point masses that move along rails fixed within the ball. Chapter 6 summarizes the results of the thesis and discusses topics for future work. Appendix A provides a brief survey of methods to numerically solve optimal control problems. Appendix B validates a claim concerning the controlled equations of motion for Suslov's problem. Appendices C and D develop algorithms for numerically solving an ODE TPBVP via predictor-corrector continuation; these algorithms are utilized to numerically solve the controlled equations of motion for the rolling ball. Appendix E reviews quaternions, which are utilized to formulate the optimal control problem for the rolling ball.

Chapter 2

Mechanics

This chapter reviews several principles from mechanics that are useful for developing the uncontrolled equations of motion for rigid bodies. Hamilton's principle and Euler-Poincaré's method are reviewed in Section 2.A, while Lagrange-d'Alembert's principle is reviewed in Section 2.B. By studying a simple nonholonomic particle in Section 2.C, it is demonstrated that the nonintuitive Lagrange-d'Alembert's principle gives different equations of motion than the more intuitive vakonomic approach. Euler-Poincaré's method and Lagrange-d'Alembert's principle are later utilized to derive the uncontrolled equations of motion for Suslov's problem in Chapter 4 and for the rolling ball in Chapter 5.

2.A Hamilton's Principle, Symmetry Reduction, and Euler-Poincaré's Method

Hamilton's Principle A mechanical system consists of a configuration space, which is a manifold M with tangent bundle $TM = \bigcup_{q \in M} T_q M$, and a Lagrangian $L(q, \dot{q}) : TM \rightarrow \mathbb{R}$, $(q, \dot{q}) \in TM$. Figure 2.1 illustrates the tangent bundles of several manifolds. The equations of motion are given by Hamilton's principle (also

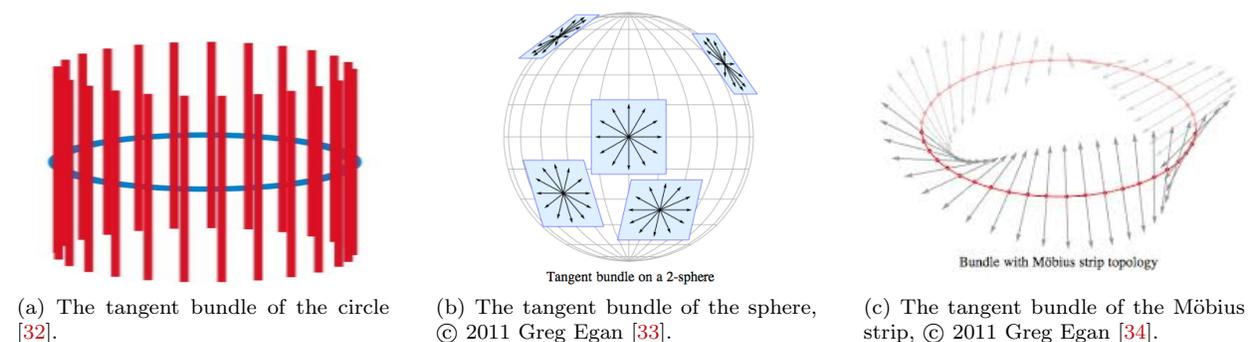


Figure 2.1: Illustrations of the tangent bundles of several manifolds.

called the variational principle of stationary action) which states that

$$\delta \int_a^b L(q, \dot{q}) dt = 0, \quad \delta q(a) = \delta q(b) = 0, \quad (2.1)$$

for all smooth functions $\delta q(t)$ defined for $a \leq t \leq b$ and that vanish at the endpoints (i.e. $\delta q(a) = \delta q(b) = 0$). Pushing the variational derivative inside the integral, integrating by parts, and enforcing the vanishing endpoint conditions $\delta q(a) = \delta q(b) = 0$ yields

$$\begin{aligned} \delta \int_a^b L(q, \dot{q}) dt &= \int_a^b \delta L(q, \dot{q}) dt = \int_a^b \left[\frac{\partial L(q, \dot{q})}{\partial q} \delta q + \frac{\partial L(q, \dot{q})}{\partial \dot{q}} \delta \dot{q} \right] dt \\ &= \int_a^b \left[\frac{\partial L(q, \dot{q})}{\partial q} \delta q - \frac{d}{dt} \frac{\partial L(q, \dot{q})}{\partial \dot{q}} \delta q \right] dt + \left. \frac{\partial L(q, \dot{q})}{\partial \dot{q}} \delta q \right|_a^b \\ &= \int_a^b \left[\frac{\partial L(q, \dot{q})}{\partial q} - \frac{d}{dt} \frac{\partial L(q, \dot{q})}{\partial \dot{q}} \right] \delta q dt. \end{aligned} \quad (2.2)$$

Insisting that $\delta \int_a^b L(q, \dot{q}) dt = 0$ for all such smooth functions δq produces the Euler-Lagrange equations of motion:

$$\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = 0. \quad (2.3)$$

Recall that a Lie group is a smooth manifold which is also a group and for which the group operations of multiplication and inversion are smooth functions [16]. In the case when there is an intrinsic symmetry in the configuration space, in particular when $M = G$, a Lie group, and when there is an appropriate invariance of the Lagrangian with respect to G , these Euler-Lagrange equations, defined on the tangent bundle of the group TG (i.e. depending on both g and \dot{g}) are cumbersome to use.

Free Rigid Body For example, consider the case of a rigid body rotating about a fixed point with no external torques, so that $G = SO(3)$, $g = \Lambda \in SO(3) = G$, and the Lagrangian is $L(\Lambda, \dot{\Lambda})$. This mechanical system is called a free rigid body. The Euler-Lagrange equations are

$$\frac{\partial L}{\partial \Lambda} - \frac{d}{dt} \frac{\partial L}{\partial \dot{\Lambda}} = 0, \quad \Lambda^T \Lambda = I, \quad (2.4)$$

where $I \in \mathbb{R}^{3 \times 3}$ is the 3×3 identity matrix. Equation (2.4) involves 9 ordinary differential equations with 6 constraints, and (2.4) is highly counterintuitive to use. Thus, the Euler-Poincaré description of motion [5], or Euler-Poincaré's method, is exploited to handle this situation and substantially simplify the Euler-Lagrange equations of motion which result from Hamilton's principle. Assuming that the Lagrangian is invariant with respect to rotations on the left, which corresponds to the description of the equations of motion in the body frame, the symmetry-reduced Lagrangian should be of the form $\ell(\Lambda^{-1} \dot{\Lambda})$.

Since $\Lambda \in SO(3)$, $\Lambda^{-1} \Lambda = I$ and $\Lambda^{-1} = \Lambda^T$, so that

$$\begin{aligned} (\Lambda^{-1} \dot{\Lambda})^\cdot &= \Lambda^{-1} \ddot{\Lambda} + (\Lambda^{-1})^\cdot \dot{\Lambda} = \Lambda^{-1} \ddot{\Lambda} + (\Lambda^T)^\cdot \dot{\Lambda} = \Lambda^{-1} \ddot{\Lambda} + \dot{\Lambda}^T \dot{\Lambda} = \Lambda^{-1} \ddot{\Lambda} + (\Lambda^T \dot{\Lambda})^T \\ &= \Lambda^{-1} \ddot{\Lambda} + (\Lambda^{-1} \dot{\Lambda})^T = 0. \end{aligned} \quad (2.5)$$

Hence $\Lambda^{-1}\delta\Lambda = -(\Lambda^{-1}\delta\Lambda)^\top$, and so $\Lambda^{-1}\delta\Lambda \in \mathfrak{so}(3)$. The isomorphic mapping from the column vectors in \mathbb{R}^3 to the Lie algebra $\mathfrak{so}(3)$, i.e. skew-symmetric matrices, is defined using the hat map $\wedge : \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$ as

$$\widehat{\boldsymbol{\omega}} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \quad (2.6)$$

and the inverse mapping from $\mathfrak{so}(3)$ to the column vectors in \mathbb{R}^3 is defined using the caron map $\vee : \mathfrak{so}(3) \rightarrow \mathbb{R}^3$ as

$$\begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}^\vee = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \boldsymbol{\omega}. \quad (2.7)$$

Since the hat map $\wedge : \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$ and its inverse $\vee : \mathfrak{so}(3) \rightarrow \mathbb{R}^3$ give isomorphisms between $\mathfrak{so}(3)$ and \mathbb{R}^3 and since $\Lambda^{-1}\dot{\Lambda} \in \mathfrak{so}(3)$, the symmetry-reduced Lagrangian should also be of the form $\ell(\boldsymbol{\Omega})$, where $\boldsymbol{\Omega} \equiv [\Lambda^{-1}\dot{\Lambda}]^\vee \in \mathbb{R}^3$. The variation of $\boldsymbol{\Omega}$ is computed as follows:

$$\begin{aligned} \delta\boldsymbol{\Omega} &= \left(\delta\Lambda^{-1}\dot{\Lambda} + \Lambda^{-1}\delta\dot{\Lambda} \right)^\vee = \left(-\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\Lambda} + (\dot{\boldsymbol{\Sigma}})^\wedge - (\Lambda^{-1})^\cdot \delta\Lambda \right)^\vee \\ &= \left(-\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\Lambda} + (\dot{\boldsymbol{\Sigma}})^\wedge + \Lambda^{-1}\dot{\Lambda}\Lambda^{-1}\delta\Lambda \right)^\vee = \left(-\widehat{\boldsymbol{\Sigma}}\widehat{\boldsymbol{\Omega}} + (\dot{\boldsymbol{\Sigma}})^\wedge + \widehat{\boldsymbol{\Omega}}\widehat{\boldsymbol{\Sigma}} \right)^\vee \\ &= \dot{\boldsymbol{\Sigma}} + \left(\widehat{\boldsymbol{\Omega}}\widehat{\boldsymbol{\Sigma}} - \widehat{\boldsymbol{\Sigma}}\widehat{\boldsymbol{\Omega}} \right)^\vee = \dot{\boldsymbol{\Sigma}} + \boldsymbol{\Omega} \times \boldsymbol{\Sigma}, \end{aligned} \quad (2.8)$$

where $\boldsymbol{\Sigma} \equiv (\Lambda^{-1}\delta\Lambda)^\vee \in \mathbb{R}^3$. Under the hat map isomorphism, the variations $\boldsymbol{\Sigma}$ lie in the Lie algebra $\mathfrak{so}(3)$. Taking the variation of the action integral, pushing the variational derivative inside the integral, integrating by parts, and enforcing the endpoint conditions $\boldsymbol{\Sigma}(a) = \boldsymbol{\Sigma}(b) = \mathbf{0}$ yields

$$\begin{aligned} \delta \int_a^b \ell(\boldsymbol{\Omega}) dt &= \int_a^b \delta\ell(\boldsymbol{\Omega}) dt = \int_a^b \left\langle \frac{\delta\ell}{\delta\boldsymbol{\Omega}}, \delta\boldsymbol{\Omega} \right\rangle dt = \int_a^b \left\langle \frac{\delta\ell}{\delta\boldsymbol{\Omega}}, \dot{\boldsymbol{\Sigma}} + \boldsymbol{\Omega} \times \boldsymbol{\Sigma} \right\rangle dt \\ &= - \int_a^b \left\langle \left(\frac{d}{dt} + \boldsymbol{\Omega} \times \right) \frac{\delta\ell}{\delta\boldsymbol{\Omega}}, \boldsymbol{\Sigma} \right\rangle dt + \left\langle \frac{\delta\ell}{\delta\boldsymbol{\Omega}}, \boldsymbol{\Sigma} \right\rangle \Big|_a^b = - \int_a^b \left\langle \left(\frac{d}{dt} + \boldsymbol{\Omega} \times \right) \frac{\delta\ell}{\delta\boldsymbol{\Omega}}, \boldsymbol{\Sigma} \right\rangle dt. \end{aligned} \quad (2.9)$$

Insisting that $\delta \int_a^b \ell(\boldsymbol{\Omega}) dt = 0$ for all smooth variations $\boldsymbol{\Sigma}$ that vanish at the endpoints generates the well-known equations of motion for the free rigid body:

$$\frac{d}{dt} \frac{\delta\ell}{\delta\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \frac{\delta\ell}{\delta\boldsymbol{\Omega}} = \mathbf{0}. \quad (2.10)$$

Note that in the above derivation, the functional derivative notation $\frac{\delta\ell}{\delta\boldsymbol{\Omega}}$ is used rather than the partial derivative notation $\frac{\partial\ell}{\partial\boldsymbol{\Omega}}$. The former is used if the Lagrangian depends functionally (e.g. involving a derivative or integral) rather than pointwise on its argument. If the Lagrangian depends only pointwise on its argument, such as is the case for the free rigid body and heavy top (to be discussed next), the two notations agree. For the free rigid body, the symmetry-reduced Lagrangian is $l(\boldsymbol{\Omega}) = \frac{1}{2} \langle \mathbb{I}\boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle$, $\frac{\delta\ell}{\delta\boldsymbol{\Omega}} = \mathbb{I}\boldsymbol{\Omega}$, and the equations of motion (2.10) become $\dot{\boldsymbol{\Omega}} = \mathbb{I}^{-1} [(\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}]$. By multiplying (2.10) by Λ and using the identity $\dot{\Lambda} = \Lambda\widehat{\boldsymbol{\Omega}}$, the

equations of motion for the free rigid body may be expressed in conservation law form:

$$\frac{d}{dt} \left[\Lambda \frac{\delta \ell}{\delta \mathbf{\Omega}} \right] = \mathbf{0} \Leftrightarrow \Lambda \frac{\delta \ell}{\delta \mathbf{\Omega}} = \text{const.} \quad (2.11)$$

Heavy Top As another application of Euler-Poincaré's method, consider the heavy top, which is a rigid body of mass m rotating with a fixed point of support in a uniform gravitational field with gravitational acceleration g . Let $\boldsymbol{\chi}$ denote the vector in the body frame from the fixed point of support to the heavy top's center of mass. To compute the equations of motion for the heavy top, another advected variable $\boldsymbol{\Gamma} = \Lambda^{-1} \hat{\boldsymbol{z}}$ must be introduced. $\boldsymbol{\Gamma}$ represents the motion of the unit vector $\hat{\boldsymbol{z}}$ along the spatial vertical axis, as seen from the body frame. $\dot{\boldsymbol{\Gamma}} = (\Lambda^{-1} \hat{\boldsymbol{z}})' = -\Lambda^{-1} \dot{\Lambda} \Lambda^{-1} \hat{\boldsymbol{z}} = -\widehat{\boldsymbol{\Omega}} \boldsymbol{\Gamma} = \boldsymbol{\Gamma} \times \boldsymbol{\Omega}$ and $\delta \boldsymbol{\Gamma} = \delta (\Lambda^{-1} \hat{\boldsymbol{z}}) = -\Lambda^{-1} \delta \Lambda \Lambda^{-1} \hat{\boldsymbol{z}} = -\widehat{\boldsymbol{\Sigma}} \boldsymbol{\Gamma} = \boldsymbol{\Gamma} \times \boldsymbol{\Sigma}$. The heavy top's reduced Lagrangian is $l(\boldsymbol{\Omega}, \boldsymbol{\Gamma}) = \frac{1}{2} \langle \mathbb{I} \boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle - \langle mg \boldsymbol{\chi}, \boldsymbol{\Gamma} \rangle$. Taking the variation of the action integral, pushing the variational derivative inside the integral, integrating by parts, and enforcing the endpoint conditions $\boldsymbol{\Sigma}(a) = \boldsymbol{\Sigma}(b) = \mathbf{0}$ yields

$$\begin{aligned} \delta \int_a^b l(\boldsymbol{\Omega}, \boldsymbol{\Gamma}) dt &= \int_a^b \delta l(\boldsymbol{\Omega}, \boldsymbol{\Gamma}) dt = \int_a^b [\langle \mathbb{I} \boldsymbol{\Omega}, \delta \boldsymbol{\Omega} \rangle - \langle mg \boldsymbol{\chi}, \delta \boldsymbol{\Gamma} \rangle] dt \\ &= \int_a^b \left[\langle \mathbb{I} \boldsymbol{\Omega}, \dot{\boldsymbol{\Sigma}} + \boldsymbol{\Omega} \times \boldsymbol{\Sigma} \rangle - \langle mg \boldsymbol{\chi}, \boldsymbol{\Gamma} \times \boldsymbol{\Sigma} \rangle \right] dt \\ &= \int_a^b \left\langle -\frac{d}{dt} (\mathbb{I} \boldsymbol{\Omega}) + (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} + mg \boldsymbol{\Gamma} \times \boldsymbol{\chi}, \boldsymbol{\Sigma} \right\rangle dt + \langle \mathbb{I} \boldsymbol{\Omega}, \boldsymbol{\Sigma} \rangle \Big|_a^b \\ &= \int_a^b \left\langle -\frac{d}{dt} (\mathbb{I} \boldsymbol{\Omega}) + (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} + mg \boldsymbol{\Gamma} \times \boldsymbol{\chi}, \boldsymbol{\Sigma} \right\rangle dt. \end{aligned} \quad (2.12)$$

Insisting that $\delta \int_a^b l(\boldsymbol{\Omega}, \boldsymbol{\Gamma}) dt = 0$ for all smooth variations $\boldsymbol{\Sigma}$ that vanish at the endpoints generates the equations of motion for the heavy top:

$$\begin{aligned} \dot{\boldsymbol{\Omega}} &= \mathbb{I}^{-1} [(\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} + mg \boldsymbol{\Gamma} \times \boldsymbol{\chi}], \\ \dot{\boldsymbol{\Gamma}} &= \boldsymbol{\Gamma} \times \boldsymbol{\Omega}. \end{aligned} \quad (2.13)$$

AD, Ad, ad, Ad*, and ad* In order to consider mechanics on general groups, adjoint and coadjoint operations are defined as follows. Consider a Lie group G with Lie algebra \mathfrak{g} , dual Lie algebra \mathfrak{g}^* , and a pairing $\langle \cdot, \cdot \rangle : \mathfrak{g}^* \times \mathfrak{g} \rightarrow \mathbb{R}$. The ADjoint operation $\text{AD} : G \times G \rightarrow G$ is defined by

$$\text{AD}_g h = ghg^{-1} \quad \forall g, h \in G. \quad (2.14)$$

The Adjoint operation $\text{Ad} : G \times \mathfrak{g} \rightarrow \mathfrak{g}$ is defined by taking a smooth curve $h(t)$ with $h(0) = e$ and $\dot{h}(0) = \eta \in \mathfrak{g}$ (arbitrary and fixed) and computing

$$\text{Ad}_g \eta := \left. \frac{d}{dt} \right|_{t=0} \text{AD}_g h(t) = g \eta g^{-1} \quad \forall g \in G, \quad \forall \eta \in \mathfrak{g}. \quad (2.15)$$

The adjoint operation $\text{ad} : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g}$ is defined by taking a smooth curve $g(t)$ with $g(0) = e$ and $\dot{g}(0) = \xi \in \mathfrak{g}$ (arbitrary and fixed) and computing

$$\text{ad}_\xi \eta := \left. \frac{d}{dt} \right|_{t=0} \text{Ad}_{g(t)} \eta = \xi \eta - \eta \xi = [\xi, \eta] \quad \forall \xi, \eta \in \mathfrak{g}, \quad (2.16)$$

where $[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{g}$ is the Lie bracket defined by

$$[\xi, \eta] = \xi \eta - \eta \xi \quad \forall \xi, \eta \in \mathfrak{g}. \quad (2.17)$$

The coAdjoint operation $\text{Ad}^* : G \times \mathfrak{g}^* \rightarrow \mathfrak{g}^*$ is defined by

$$\langle \text{Ad}_g^* \mu, \eta \rangle = \langle \mu, \text{Ad}_g \eta \rangle \quad \forall g \in G, \quad \forall \mu \in \mathfrak{g}^*, \quad \forall \eta \in \mathfrak{g}. \quad (2.18)$$

The coadjoint operation $\text{ad}^* : \mathfrak{g} \times \mathfrak{g}^* \rightarrow \mathfrak{g}^*$ is defined by

$$\langle \text{ad}_\xi^* \mu, \eta \rangle = \langle \mu, \text{ad}_\xi \eta \rangle \quad \forall \xi, \eta \in \mathfrak{g}, \quad \forall \mu \in \mathfrak{g}^*. \quad (2.19)$$

Euler-Poincaré's Method More generally, if the Lagrangian $L : TG \rightarrow \mathbb{R}$ is left-invariant, i.e. $L(hg, h\dot{g}) = L(g, \dot{g}) \quad \forall (g, \dot{g}) \in TG, \quad \forall h \in G$, we can define the *symmetry-reduced Lagrangian* through the symmetry reduction $\ell = \ell(g^{-1}\dot{g}) = \ell(\xi) = L(e, \xi)$, where $\xi = g^{-1}\dot{g}$. Then, the equations of motion (2.3) are equivalent to the Euler-Poincaré equations of motion obtained from the variational principle

$$\delta \int_a^b \ell(\xi) dt = 0, \quad \text{for variations} \quad \delta \xi = \dot{\eta} + \text{ad}_\xi \eta, \quad \forall \eta(t) : \eta(a) = \eta(b) = 0. \quad (2.20)$$

The variations $\eta(t)$, assumed to be sufficiently smooth, are sometimes called *free variations*. Applying the variational principle (2.20) gives

$$\begin{aligned} \delta \int_a^b \ell(\xi) dt &= \int_a^b \left\langle \frac{\delta \ell}{\delta \xi}, \delta \xi \right\rangle dt = \int_a^b \left\langle \frac{\delta \ell}{\delta \xi}, \dot{\eta} + \text{ad}_\xi \eta \right\rangle dt \\ &= \int_a^b \left\langle -\frac{d}{dt} \frac{\delta \ell}{\delta \xi} + \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi}, \eta \right\rangle dt + \left\langle \frac{\delta \ell}{\delta \xi}, \eta \right\rangle \Big|_a^b \\ &= \int_a^b \left\langle -\frac{d}{dt} \frac{\delta \ell}{\delta \xi} + \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi}, \eta \right\rangle dt = 0, \end{aligned} \quad (2.21)$$

which yields the Euler-Poincaré equations of motion:

$$\frac{d}{dt} \frac{\delta \ell}{\delta \xi} - \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi} = 0. \quad (2.22)$$

For right-invariant Lagrangians, i.e. $L(gh, \dot{g}h) = L(g, \dot{g}) \quad \forall h \in G$, the Euler-Poincaré equations of motion (2.22) change by altering the sign in front of ad_ξ^* from minus to plus. For the free rigid body, $\xi = \Omega$ and $\text{ad}_\Omega^* \frac{\delta \ell}{\delta \Omega} = -\Omega \times \frac{\delta \ell}{\delta \Omega}$, so that the free rigid body equations of motion (2.10) derived earlier agree with the Euler-Poincaré equations of motion (2.22).

Next it will be shown that (2.22) implies conservation of angular momentum. Letting $\alpha \in \mathfrak{g}$ be arbitrary

and constant in time and letting $t_0 \in \mathbb{R}$ be an arbitrary time, observe that

$$\begin{aligned}
\left\langle \frac{d}{dt} \Big|_{t_0} \text{Ad}_{g^{-1}}^* \frac{\delta \ell}{\delta \xi}, \alpha \right\rangle &= \frac{d}{dt} \Big|_{t_0} \left\langle \text{Ad}_{g^{-1}}^* \frac{\delta \ell}{\delta \xi}, \alpha \right\rangle = \frac{d}{dt} \Big|_{t_0} \left\langle \frac{\delta \ell}{\delta \xi}, \text{Ad}_{g^{-1}} \alpha \right\rangle \\
&= \left\langle \frac{d}{dt} \Big|_{t_0} \frac{\delta \ell}{\delta \xi}, \text{Ad}_{g^{-1}} \alpha \right\rangle + \left\langle \frac{\delta \ell}{\delta \xi}, \frac{d}{dt} \Big|_{t_0} \text{Ad}_{g^{-1}} \alpha \right\rangle \\
&= \left\langle \text{Ad}_{g^{-1}}^* \left[\frac{d}{dt} \Big|_{t_0} \frac{\delta \ell}{\delta \xi} \right], \alpha \right\rangle + \left\langle \frac{\delta \ell}{\delta \xi}, -\text{ad}_\xi [\text{Ad}_{g^{-1}} \alpha] \right\rangle \\
&= \left\langle \text{Ad}_{g^{-1}}^* \left[\frac{d}{dt} \Big|_{t_0} \frac{\delta \ell}{\delta \xi} \right], \alpha \right\rangle - \left\langle \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi}, \text{Ad}_{g^{-1}} \alpha \right\rangle \\
&= \left\langle \text{Ad}_{g^{-1}}^* \left[\frac{d}{dt} \Big|_{t_0} \frac{\delta \ell}{\delta \xi} \right], \alpha \right\rangle - \left\langle \text{Ad}_{g^{-1}}^* \left[\text{ad}_\xi^* \frac{\delta \ell}{\delta \xi} \right], \alpha \right\rangle \\
&= \left\langle \text{Ad}_{g^{-1}}^* \left[\frac{d}{dt} \Big|_{t_0} \frac{\delta \ell}{\delta \xi} - \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi} \right], \alpha \right\rangle = \langle \text{Ad}_{g^{-1}}^* 0, \alpha \rangle = 0,
\end{aligned} \tag{2.23}$$

where (2.22) is used in the second to last equality. In the fourth equality of (2.23), the following result is used

$$\begin{aligned}
\frac{d}{dt} \Big|_{t_0} \text{Ad}_{g^{-1}} \alpha &= \frac{d}{dt} \Big|_{t=t_0} \text{Ad}_{g(t)^{-1}} \alpha = \frac{d}{dt} \Big|_{t=t_0} \text{Ad}_{g(t)^{-1}g(t_0)} [\text{Ad}_{g(t_0)^{-1}} \alpha] \\
&= \frac{d}{dt} \Big|_{t=t_0} \{g(t)^{-1}g(t_0) [\text{Ad}_{g(t_0)^{-1}} \alpha] g(t_0)^{-1}g(t)\} \\
&= \left\{ -g(t)^{-1}\dot{g}(t)g(t)^{-1}g(t_0) [\text{Ad}_{g(t_0)^{-1}} \alpha] g(t_0)^{-1}g(t) \right. \\
&\quad \left. + g(t)^{-1}g(t_0) [\text{Ad}_{g(t_0)^{-1}} \alpha] g(t_0)^{-1}\dot{g}(t) \right\} \Big|_{t=t_0} \\
&= -g(t_0)^{-1}\dot{g}(t_0) [\text{Ad}_{g(t_0)^{-1}} \alpha] + [\text{Ad}_{g(t_0)^{-1}} \alpha] g(t_0)^{-1}\dot{g}(t_0) \\
&= -\xi(t_0) [\text{Ad}_{g(t_0)^{-1}} \alpha] + [\text{Ad}_{g(t_0)^{-1}} \alpha] \xi(t_0) \\
&= -\text{ad}_{\xi(t_0)} [\text{Ad}_{g(t_0)^{-1}} \alpha],
\end{aligned} \tag{2.24}$$

using the property $\text{Ad}_g \text{Ad}_h \eta = g(h\eta h^{-1})g^{-1} = (gh)\eta(gh)^{-1} = \text{Ad}_{(gh)}\eta \quad \forall g, h \in G, \quad \forall \eta \in \mathfrak{g}$ in the second equality. Since $\alpha \in \mathfrak{g}$ is arbitrary, (2.23) implies that $\frac{d}{dt} \Big|_{t_0} \text{Ad}_{g^{-1}}^* \frac{\delta \ell}{\delta \xi} = 0$. Since $t_0 \in \mathbb{R}$ is an arbitrary time, $\frac{d}{dt} \text{Ad}_{g^{-1}}^* \frac{\delta \ell}{\delta \xi} = 0$, thereby proving conservation of angular momentum.

Hamilton-Pontryagin's Principle An alternative to Euler-Poincaré's method is Hamilton-Pontryagin's principle, which says that the equations of motion may be obtained from the variational principle

$$\delta \hat{S}(\xi, g, \dot{g}) = \delta \int_a^b \hat{\ell}(\xi, g, \dot{g}) dt = 0 \tag{2.25}$$

for all variations of g such that $\delta g(a) = \delta g(b) = 0$, where \hat{S} is the constrained action integral

$$\hat{S}(\xi, g, \dot{g}) = \int_a^b \hat{\ell}(\xi, g, \dot{g}) dt = \int_a^b [\ell(\xi) + \langle \mu, g^{-1}\dot{g} - \xi \rangle] dt \tag{2.26}$$

and $\hat{\ell}$ is the augmented, reduced Lagrangian

$$\hat{\ell}(\xi, g, \dot{g}) = \ell(\xi) + \langle \mu, g^{-1}\dot{g} - \xi \rangle. \quad (2.27)$$

To see that Hamilton-Pontryagin's principle gives the same equations of motion as Euler-Poincaré's method, define $\eta = g^{-1}\delta g$. Since $\xi = g^{-1}\dot{g}$,

$$\delta\xi = \delta(g^{-1}\dot{g}) = -g^{-1}\delta g g^{-1}\dot{g} + g^{-1}\delta\dot{g} = -\eta\xi + g^{-1}\delta\dot{g}. \quad (2.28)$$

Since $\eta = g^{-1}\delta g$,

$$\dot{\eta} = (g^{-1}\delta\dot{g})' = -g^{-1}\dot{g}g^{-1}\delta g + g^{-1}(\delta\dot{g})' = -\xi\eta + g^{-1}(\delta\dot{g})'. \quad (2.29)$$

Subtracting (2.29) from (2.28) gives

$$\delta\xi - \dot{\eta} = \xi\eta - \eta\xi = \text{ad}_\xi\eta. \quad (2.30)$$

Now compute the variation of \hat{S} :

$$\begin{aligned} \delta\hat{S} &= \delta \int_a^b \hat{\ell}(\xi, g, \dot{g}) dt = \delta \int_a^b [\ell(\xi) + \langle \mu, g^{-1}\dot{g} - \xi \rangle] dt \\ &= \int_a^b \left[\left\langle \frac{\delta\hat{\ell}}{\delta\xi} - \mu, \delta\xi \right\rangle + \langle \delta\mu, g^{-1}\dot{g} - \xi \rangle + \langle \mu, \delta(g^{-1}\dot{g}) \rangle \right] dt \\ &= \int_a^b \left[\left\langle \frac{\delta\hat{\ell}}{\delta\xi} - \mu, \delta\xi \right\rangle + \langle \delta\mu, g^{-1}\dot{g} - \xi \rangle + \langle \mu, \dot{\eta} + \text{ad}_\xi\eta \rangle \right] dt \\ &= \int_a^b \left[\left\langle \frac{\delta\hat{\ell}}{\delta\xi} - \mu, \delta\xi \right\rangle + \langle \delta\mu, g^{-1}\dot{g} - \xi \rangle + \langle -\dot{\mu} + \text{ad}_\xi^*\mu, \eta \rangle \right] dt + \langle \mu, \eta \rangle \Big|_a^b \\ &= \int_a^b \left[\left\langle \frac{\delta\hat{\ell}}{\delta\xi} - \mu, \delta\xi \right\rangle + \langle \delta\mu, g^{-1}\dot{g} - \xi \rangle + \langle -\dot{\mu} + \text{ad}_\xi^*\mu, \eta \rangle \right] dt, \end{aligned} \quad (2.31)$$

since $\delta(g^{-1}\dot{g}) = \dot{\eta} + \text{ad}_\xi\eta$ by (2.30) and because $\eta(a) = \eta(b) = 0$ (since $\delta g(a) = \delta g(b) = 0$). $\delta\hat{S} = 0$ gives the Hamilton-Pontryagin equations of motion

$$\frac{\partial\hat{\ell}}{\partial\xi} = \mu, \quad g^{-1}\dot{g} - \xi, \quad \dot{\mu} = \text{ad}_\xi^*\mu, \quad (2.32)$$

which agree with the Euler-Poincaré equations of motion (2.22). Even though Hamilton-Pontryagin's principle could be utilized, this thesis instead relies on Euler-Poincaré's method to derive the equations of motion for Suslov's problem in Chapter 4 and for the rolling ball in Chapter 5.

Euler-Poincaré's Method with an Advected Parameter Let V be a vector space. Suppose the Lagrangian L depends on a parameter in the dual space V^* , so that the general Lagrangian has the form $L : TG \times V^* \rightarrow \mathbb{R}$. For a parameter $\alpha_0 \in V^*$, suppose that the Lagrangian $L_{\alpha_0} : TG \rightarrow \mathbb{R}$ defined by $L_{\alpha_0}(g, \dot{g}) = L(g, \dot{g}, \alpha_0) \quad \forall (g, \dot{g}) \in TG$ is left-invariant, i.e. $L(hg, h\dot{g}, h\alpha_0) = L(g, \dot{g}, \alpha_0) \quad \forall (g, \dot{g}) \in TG, \quad \forall h \in G$. Then we can define the *symmetry-reduced Lagrangian* through the symmetry reduction $\ell = \ell(g^{-1}\dot{g}, g^{-1}\alpha_0) = \ell(\xi, \alpha) = L(e, \xi, \alpha)$, where $\xi = g^{-1}\dot{g}$ and $\alpha = g^{-1}\alpha_0$. Euler-Poincaré's method with

an advected parameter says that the equations of motion are obtained from the variational principle

$$\delta \int_a^b \ell(\xi, \alpha) dt = 0, \quad \text{for variations } \delta\xi = \dot{\eta} + \text{ad}_\xi \eta, \delta\alpha = -\eta\alpha, \quad \forall \eta(t) : \eta(a) = \eta(b) = 0. \quad (2.33)$$

Before applying this variational principle, the diamond operation \diamond is defined. $\diamond : V \times V^* \rightarrow \mathfrak{g}^*$ is defined by

$$\langle v \diamond w, \xi \rangle = \langle w, \xi v \rangle \quad \forall v \in V, \quad \forall w \in V^*, \quad \forall \xi \in \mathfrak{g}. \quad (2.34)$$

$\diamond : V^* \times V \rightarrow \mathfrak{g}^*$ is defined by

$$\langle w \diamond v, \xi \rangle = -\langle v \diamond w, \xi \rangle = -\langle w, \xi v \rangle \quad \forall v \in V, \quad \forall w \in V^*, \quad \forall \xi \in \mathfrak{g}. \quad (2.35)$$

Applying the variational principle (2.33) gives

$$\begin{aligned} \delta \int_a^b \ell(\xi, \alpha) dt &= \int_a^b \left[\left\langle \frac{\delta \ell}{\delta \xi}, \delta \xi \right\rangle + \left\langle \frac{\delta \ell}{\delta \alpha}, \delta \alpha \right\rangle \right] dt \\ &= \int_a^b \left[\left\langle \frac{\delta \ell}{\delta \xi}, \dot{\eta} + \text{ad}_\xi \eta \right\rangle + \left\langle \frac{\delta \ell}{\delta \alpha}, -\eta\alpha \right\rangle \right] dt \\ &= \int_a^b \left[\left\langle -\frac{d}{dt} \frac{\delta \ell}{\delta \xi} + \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi}, \eta \right\rangle + \left\langle \frac{\delta \ell}{\delta \alpha} \diamond \alpha, \eta \right\rangle \right] dt + \left. \left\langle \frac{\delta \ell}{\delta \xi}, \eta \right\rangle \right|_a^b \\ &= \int_a^b \left\langle -\frac{d}{dt} \frac{\delta \ell}{\delta \xi} + \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi} + \frac{\delta \ell}{\delta \alpha} \diamond \alpha, \eta \right\rangle dt = 0, \end{aligned} \quad (2.36)$$

which yields the Euler-Poincaré equations of motion with an advected parameter:

$$\frac{d}{dt} \frac{\delta \ell}{\delta \xi} - \text{ad}_\xi^* \frac{\delta \ell}{\delta \xi} - \frac{\delta \ell}{\delta \alpha} \diamond \alpha = 0. \quad (2.37)$$

The most direct application of the Euler-Poincaré equations of motion with an advected parameter is the heavy top, where the advected parameter is the gravity vector expressed in the heavy top's body frame. For the heavy top, $\xi = \Omega$, $\alpha = \Gamma$, $l(\Omega, \Gamma) = \frac{1}{2} \langle \mathbb{I}\Omega, \Omega \rangle - \langle mg\chi, \Gamma \rangle$, $\frac{\delta \ell}{\delta \Omega} = \mathbb{I}\Omega$, $\text{ad}_\Omega^* \frac{\delta \ell}{\delta \Omega} = -\Omega \times \mathbb{I}\Omega$, $\frac{\delta \ell}{\delta \Gamma} = -mg\chi$, and $\frac{\delta \ell}{\delta \Gamma} \diamond \Gamma = -mg\chi \times \Gamma$. Plugging all these identities into (2.37) recovers the previously derived heavy top equations of motion (2.13).

2.B Nonholonomic Constraints and Lagrange-d'Alembert's Principle

Suppose a mechanical system having configuration space M , a manifold of dimension n , must satisfy $m < n$ constraints that are linear in velocity. To express these velocity constraints formally, the notion of a distribution is needed. Given the manifold M , a distribution \mathcal{D} on M is a subset of the tangent bundle $TM = \bigcup_{q \in M} T_q M$: $\mathcal{D} = \bigcup_{q \in M} \mathcal{D}_q$, where $\mathcal{D}_q \subset T_q M$ and $m = \dim \mathcal{D}_q < \dim T_q M = n$ for each $q \in M$. A curve $q(t) \in M$ satisfies the constraints if $\dot{q}(t) \in \mathcal{D}_{q(t)}$. Lagrange-d'Alembert's principle states that the

equations of motion are determined by

$$\delta \int_a^b L(q, \dot{q}) dt = 0 \Leftrightarrow \int_a^b \left[\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} \right] \delta q dt = 0 \Leftrightarrow \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} \in \mathcal{D}_q^\circ \quad (2.38)$$

for all smooth variations $\delta q(t)$ of the curve $q(t)$ such that $\delta q(t) \in \mathcal{D}_{q(t)}$ for all $a \leq t \leq b$ and such that $\delta q(a) = \delta q(b) = 0$, and for which $\dot{q}(t) \in \mathcal{D}_{\dot{q}(t)}$ for all $a \leq t \leq b$. If one writes the nonholonomic constraint in local coordinates as $\sum_{i=1}^n A(q)_i^j \dot{q}^i = 0$, $j = 1, \dots, m < n$, then (2.38) is written in local coordinates as

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}^i} - \frac{\partial L}{\partial q^i} = \sum_{j=1}^m \lambda_j A(q)_i^j, \quad i = 1, \dots, n, \quad \sum_{i=1}^n A(q)_i^j \dot{q}^i = 0, \quad (2.39)$$

where the λ_j are Lagrange multipliers enforcing $\sum_{i=1}^n A(q)_i^j \delta q^i = 0$, $j = 1, \dots, m$. Aside from Lagrange-d'Alembert's approach, there is also an alternative *vakonomic* approach to derive the equations of motion for nonholonomic mechanical systems. Simply speaking, the vakonomic approach relies on substituting the constraint into the Lagrangian before taking variations or, equivalently, enforcing the constraints using the appropriate Lagrange multiplier method. The next section illustrates the differences in the two approaches by studying a simple nonholonomic particle. In particular, it is shown that the two methods yield different equations of motion for this particle. In general, it is an experimental fact that all known nonholonomic mechanical systems obey the equations of motion resulting from Lagrange-d'Alembert's principle [35].

2.C A Simple Nonholonomic Particle

Consider a free particle with unit mass moving in space subject to the nonholonomic constraint

$$\dot{z} = y\dot{x}. \quad (2.40)$$

This problem was introduced and studied by Bates and Śniatycki in [36]. The particle's Lagrangian is

$$l = \frac{1}{2} (\dot{x}^2 + \dot{y}^2 + \dot{z}^2), \quad (2.41)$$

the particle's action integral is

$$S = \int_a^b l dt = \int_a^b \frac{1}{2} (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) dt, \quad (2.42)$$

and the variation of the particle's action integral is

$$\delta S = \delta \int_a^b l dt = \int_a^b \delta l dt = \int_a^b [\dot{x}\delta\dot{x} + \dot{y}\delta\dot{y} + \dot{z}\delta\dot{z}] dt = - \int_a^b [\ddot{x}\delta x + \ddot{y}\delta y + \ddot{z}\delta z] dt. \quad (2.43)$$

Lagrange-d'Alembert's Approach In Lagrange-d'Alembert's approach, the constraint $\dot{z} = y\dot{x}$ implies the variational constraint $\delta z = y\delta x$. Lagrange-d'Alembert's principle states that the equations of motion are given by $\delta S = 0$ for all variations δx , δy , and δz (i.e. Hamilton's principle) subject to the variational constraint $\delta z = y\delta x$ and the original constraint $\dot{z} = y\dot{x}$ [30]. Using the method of Lagrange multipliers to

simultaneously enforce the conditions $\delta S = 0$ and $\delta z = y\delta x$, the equations of motion must satisfy

$$\begin{aligned}
0 &= \delta S + \int_a^b \lambda [\delta z - y\delta x] dt \\
&= - \int_a^b [\ddot{x}\delta x + \ddot{y}\delta y + \ddot{z}\delta z] dt + \int_a^b \lambda [\delta z - y\delta x] dt \\
&= - \int_a^b [(\ddot{x} + \lambda y)\delta x + \ddot{y}\delta y + (\ddot{z} - \lambda)\delta z] dt, \\
\dot{z} &= y\dot{x},
\end{aligned} \tag{2.44}$$

for all variations δx , δy , and δz and where λ is a Lagrange multiplier. Thus, the equations of motion are

$$\begin{aligned}
\ddot{x} + \lambda y &= 0, \\
\ddot{y} &= 0, \\
\ddot{z} - \lambda &= 0, \\
\dot{z} &= y\dot{x}.
\end{aligned} \tag{2.45}$$

These equations may be simplified by using the equation $\ddot{z} - \lambda = 0$ to eliminate λ and by then using the original constraint $\dot{z} = y\dot{x}$ to eliminate z ; after these simplifications the equations of motion only depend on x and y . Since $\ddot{z} - \lambda = 0$, $\ddot{z} = \lambda$ and so the equation $\ddot{x} + \lambda y = 0$ becomes $\ddot{x} + \ddot{z}y = 0$. Having eliminated λ , the equations of motion become

$$\begin{aligned}
\ddot{x} + \ddot{z}y &= 0, \\
\ddot{y} &= 0, \\
\dot{z} &= y\dot{x}.
\end{aligned} \tag{2.46}$$

The original constraint $\dot{z} = y\dot{x}$ implies that $\ddot{z} = \dot{y}\dot{x} + y\ddot{x}$. Substituting $\ddot{z} = \dot{y}\dot{x} + y\ddot{x}$ into $\ddot{x} + \ddot{z}y = 0$ yields $\ddot{x} + (\dot{y}\dot{x} + y\ddot{x})y = 0$. The equations of motion simplify to

$$\begin{aligned}
\ddot{x} + \frac{y}{1+y^2}\dot{x}\dot{y} &= 0, \\
\ddot{y} &= 0.
\end{aligned} \tag{2.47}$$

For this simple problem, a more direct derivation of the above equations of motion, avoiding introduction of the Lagrange multiplier, is achieved by substituting the variational constraint $\delta z = y\delta x$ in for δz in the variation of the action integral. Making this substitution, the variation of the action integral becomes

$$\delta S = - \int_a^b [\ddot{x}\delta x + \ddot{y}\delta y + \ddot{z}\delta z] dt = - \int_a^b [\ddot{x}\delta x + \ddot{y}\delta y + \ddot{z}y\delta x] dt = - \int_a^b [(\ddot{x} + \ddot{z}y)\delta x + \ddot{y}\delta y] dt. \tag{2.48}$$

Applying Hamilton's principle (i.e. demanding that $\delta S = 0$ for all variations δx and δy) and using the original constraint $\dot{z} = y\dot{x}$ recovers the previously obtained equations of motion

$$\begin{aligned}
\ddot{x} + \frac{y}{1+y^2}\dot{x}\dot{y} &= 0, \\
\ddot{y} &= 0.
\end{aligned} \tag{2.49}$$

These equations of motion may be solved analytically. Since $\ddot{y} = 0$,

$$y(t) = ct + d, \quad (2.50)$$

for integration constants c and d . Using this result, the equation $\ddot{x} + \frac{y}{1+y^2}\dot{x}\dot{y} = 0$ becomes

$$\ddot{x} + \frac{ct + d}{1 + (ct + d)^2}c\dot{x} = 0. \quad (2.51)$$

If $c = 0$,

$$x(t) = \alpha t + \beta, \quad (2.52)$$

for integration constants α and β . If $c \neq 0$,

$$x(t) = B \ln \left[ct + d + \sqrt{1 + (ct + d)^2} \right] + E, \quad (2.53)$$

for integration constants B and E . If the particle's initial conditions $x(a)$, $\dot{x}(a)$, $y(a)$, and $\dot{y}(a)$ are given at time $t = a$, then the integration constants may be readily determined. $c = \dot{y}(a)$ and $d = y(a) - \dot{y}(a)a$. If $c = 0$, $\alpha = \dot{x}(a)$ and $\beta = x(a) - \dot{x}(a)a$. If $c \neq 0$, $B = \frac{\dot{x}(a)\sqrt{1+y(a)^2}}{\dot{y}(a)}$ and $E = x(a) - \frac{\dot{x}(a)\sqrt{1+y(a)^2}}{\dot{y}(a)} \ln \left[y(a) + \sqrt{1 + y(a)^2} \right]$. Since $\dot{y} = c$ is constant, the Lagrange-d'Alembert solution conserves the y -component of momentum, which is contrary to a naïve application of Noether's theorem. The Lagrangian (2.41) and nonholonomic constraint (2.40) for this particle are invariant under the translational transformations $\rho_1(x, y, z) = (x + C_1, y, z)$ and $\rho_3(x, y, z) = (x, y, z + C_3)$ for constants C_1 and C_3 . But the nonholonomic constraint (2.40) is not invariant under the translational transformation $\rho_2(x, y, z) = (x, y + C_2, z)$ for a constant C_2 . Based on these observations, (wrong application of) Noether's theorem would predict that the particle's x - and z -momenta, p_x and p_z , should be conserved and that the particle's y -momentum, p_y , should not be conserved, which is not in agreement with (2.53), (2.50), and (2.46). That is, (2.53) says that \dot{x} is nonconstant, (2.50) says that $\dot{y} = c$ is constant, and (2.46) says that \dot{z} , which equals $y\dot{x}$, is nonconstant.

Vakonomic Approach In the vakonomic approach, the constraint $\dot{z} = y\dot{x}$ is added to the Lagrangian via the method of Lagrange multipliers to obtain the modified Lagrangian

$$\tilde{l} = l + \lambda(\dot{z} - y\dot{x}) = \frac{1}{2}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \lambda(\dot{z} - y\dot{x}), \quad (2.54)$$

for a Lagrange multiplier λ [30, 37]. The modified action integral is

$$\begin{aligned} \tilde{S} &= \int_a^b \tilde{l} dt = \int_a^b \left[\frac{1}{2}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \lambda(\dot{z} - y\dot{x}) \right] dt = \int_a^b \frac{1}{2}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) dt + \int_a^b \lambda(\dot{z} - y\dot{x}) dt \\ &= S + \int_a^b \lambda(\dot{z} - y\dot{x}) dt. \end{aligned} \quad (2.55)$$

The variation of the modified action integral is

$$\begin{aligned}
\delta\tilde{S} &= \delta \int_a^b \tilde{l} dt = \delta S + \delta \int_a^b [\lambda (\dot{z} - y\dot{x})] dt \\
&= \delta S + \int_a^b \delta [\lambda (\dot{z} - y\dot{x})] dt \\
&= \int_a^b [\dot{x}\delta\dot{x} + \dot{y}\delta\dot{y} + \dot{z}\delta\dot{z}] dt + \int_a^b \delta [\lambda (\dot{z} - y\dot{x})] dt \\
&= \int_a^b [\dot{x}\delta\dot{x} + \dot{y}\delta\dot{y} + \dot{z}\delta\dot{z} + \lambda (\delta\dot{z} - y\delta\dot{x} - \delta y\dot{x}) + \delta\lambda (\dot{z} - y\dot{x})] dt \\
&= \int_a^b [-\ddot{x}\delta x - \ddot{y}\delta y - \ddot{z}\delta z - \dot{\lambda}\delta z - \lambda\dot{x}\delta y + (\dot{\lambda}y + \lambda\dot{y})\delta x + (\dot{z} - y\dot{x})\delta\lambda] dt \\
&= \int_a^b [(-\ddot{x} + \dot{\lambda}y + \lambda\dot{y})\delta x - (\ddot{y} + \lambda\dot{x})\delta y - (\ddot{z} + \dot{\lambda})\delta z + (\dot{z} - y\dot{x})\delta\lambda] dt.
\end{aligned} \tag{2.56}$$

Demanding that $\delta\tilde{S} = 0$ for all variations δx , δy , δz , and $\delta\lambda$ (i.e. applying Hamilton's principle) yields the equations of motion

$$\begin{aligned}
-\ddot{x} + \dot{\lambda}y + \lambda\dot{y} &= 0, \\
\ddot{y} + \lambda\dot{x} &= 0, \\
\ddot{z} + \dot{\lambda} &= 0, \\
\dot{z} - y\dot{x} &= 0.
\end{aligned} \tag{2.57}$$

The equation $\ddot{z} + \dot{\lambda} = 0$ implies that $\lambda = -\dot{z} + \tilde{c}$, for an integration constant \tilde{c} . Taking $\tilde{c} = 0$, $\lambda = -\dot{z}$. Making the substitutions $\dot{\lambda} = -\ddot{z}$ and $\lambda = -\dot{z}$ eliminates λ from the equations of motion, simplifying them to

$$\begin{aligned}
-\ddot{x} - \ddot{z}y - \dot{z}\dot{y} &= 0, \\
\ddot{y} - \dot{z}\dot{x} &= 0, \\
\dot{z} - y\dot{x} &= 0.
\end{aligned} \tag{2.58}$$

The equation $\dot{z} - y\dot{x} = 0$ implies that $\dot{z} = y\dot{x} + y\ddot{x}$. Making the substitutions $\dot{z} = y\dot{x} + y\ddot{x}$ and $\ddot{z} = \dot{y}\dot{x} + y\ddot{x} + y\dot{x}\dot{x}$ eliminates z from the equations of motion, simplifying them to

$$\begin{aligned}
-\ddot{x} - (\dot{y}\dot{x} + y\ddot{x})y - y\dot{x}\dot{y} &= 0, \\
\ddot{y} - y\dot{x}\dot{x} &= 0,
\end{aligned} \tag{2.59}$$

which is equivalent to

$$\begin{aligned}
\ddot{x} + \frac{2y}{1+y^2}\dot{x}\dot{y} &= 0, \\
\ddot{y} - y\dot{x}^2 &= 0.
\end{aligned} \tag{2.60}$$

For this simple problem, a more direct derivation of the equations of motion substitutes the constraint $\dot{z} = y\dot{x}$

into the Lagrangian. With this substitution, the Lagrangian becomes

$$l = \frac{1}{2} (\dot{x}^2 + \dot{y}^2 + \dot{z}^2) = \frac{1}{2} (\dot{x}^2 + \dot{y}^2 + y^2 \dot{x}^2) = \frac{1}{2} \left((1 + y^2) \dot{x}^2 + \dot{y}^2 \right). \quad (2.61)$$

The action integral is

$$S = \int_a^b l dt = \int_a^b \frac{1}{2} \left((1 + y^2) \dot{x}^2 + \dot{y}^2 \right) dt. \quad (2.62)$$

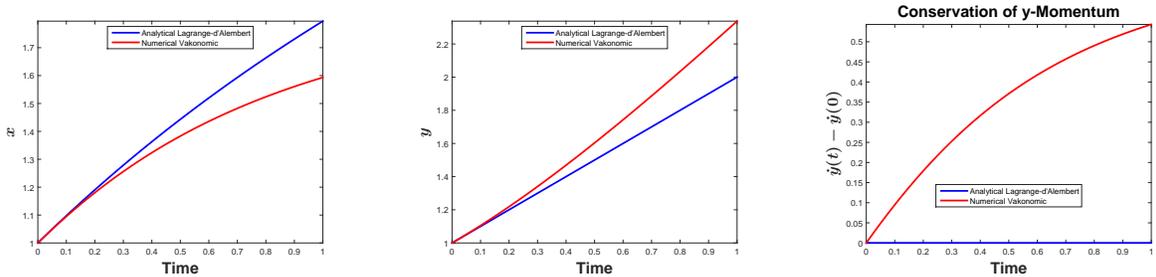
The variation of the action integral is

$$\begin{aligned} \delta S &= \delta \int_a^b l dt = \int_a^b \delta l dt = \int_a^b \left[(1 + y^2) \dot{x} \delta \dot{x} + y \delta y \dot{x}^2 + \dot{y} \delta \dot{y} \right] dt \\ &= \int_a^b \left[-\frac{d}{dt} \left[(1 + y^2) \dot{x} \right] \delta x + y \dot{x}^2 \delta y - \ddot{y} \delta y \right] dt \\ &= \int_a^b \left[-\left((1 + y^2) \ddot{x} + 2y \dot{y} \dot{x} \right) \delta x + (y \dot{x}^2 - \ddot{y}) \delta y \right] dt. \end{aligned} \quad (2.63)$$

Demanding that $\delta S = 0$ for all variations δx and δy (i.e. applying Hamilton's principle) recovers the previously obtained equations of motion

$$\begin{aligned} \ddot{x} + \frac{2y}{1 + y^2} \dot{x} \dot{y} &= 0, \\ \ddot{y} - y \dot{x}^2 &= 0. \end{aligned} \quad (2.64)$$

Comparison Figure 2.2 compares the solutions of the Lagrange-d'Alembert and vakonomic equations of motion during the time interval $0 \leq t \leq 1$ assuming that the initial conditions at time $t = 0$ are $x(0) = 1$, $\dot{x}(0) = 1$, $y(0) = 1$, and $\dot{y}(0) = 1$. The MATLAB routine `ode45` was used to numerically solve the vakonomic equations of motion using the default error tolerances. Figures 2.2a and 2.2b show that the x - and y -components of the solutions of the Lagrange-d'Alembert and vakonomic equations of motion disagree. Figure 2.2c shows that the Lagrange-d'Alembert solution conserves the y -component of momentum, while the vakonomic solution does not.



(a) The x -component of the Lagrange-d'Alembert and vakonomic solutions disagree.

(b) The y -component of the Lagrange-d'Alembert and vakonomic solutions disagree.

(c) The Lagrange-d'Alembert solution conserves the y -component of momentum, while the vakonomic solution does not.

Figure 2.2: Lagrange-d'Alembert vs vakonomic solutions for a simple nonholonomic particle.

While the vakonomic approach does not give the correct equations of motion for a mechanical system with

nonholonomic constraints, the vakonomic approach does give the correct equations of motion for a mechanical system with holonomic constraints. The vakonomic approach is also applicable in optimal control, where a performance index must be minimized subject to satisfying equations of motion, which may have been obtained through the vakonomic or Lagrange-d'Alembert approaches depending on whether the mechanical constraints are holonomic or nonholonomic. When the vakonomic approach is applied to an optimal control problem as part of Pontryagin's Minimum Principle (discussed in Chapter 3), the resulting equations are called the controlled equations of motion.

Chapter 3

Optimal Control

This chapter reviews Pontryagin’s minimum principle, which provides necessary conditions that a solution of an optimal control problem must satisfy. In this thesis, these necessary conditions, in the context of describing the optimal control of rigid bodies, are referred to as the controlled equations of motion. Section 3.A briefly reviews some notation from the calculus of variations needed to understand the derivation of Pontryagin’s minimum principle, after which Pontryagin’s minimum principle is developed in Section 3.B. In Section 3.C, the controlled equations of motion and their Jacobians are expressed in normalized coordinates, which is useful for the numerical solution of the controlled equations of motion. Pontryagin’s minimum principle is utilized in Chapter 4 to derive the controlled equations of motion for Suslov’s problem. Chapter 5 utilizes the formulas derived in Section 3.C to construct the controlled equations of motion (and their Jacobians) for the rolling ball via automatic differentiation, so that the controlled equations of motion may be solved numerically.

3.A Calculus of Variations

Before proceeding with Pontryagin’s minimum principle, some terminology from the calculus of variations is briefly reviewed. Suppose that y is a time-dependent function, w is a time-independent variable, and Q is a scalar-valued function or functional that depends on y and w . The variation of y is $\delta y \equiv \left. \frac{\partial y}{\partial \epsilon} \right|_{\epsilon=0}$, the differential of y is $dy \equiv \delta y + \dot{y}dt = \left. \frac{\partial y}{\partial \epsilon} \right|_{\epsilon=0} + \left. \frac{\partial y}{\partial t} \right|_{\epsilon=0} dt$, and the differential of w is $dw \equiv \left. \frac{dw}{d\epsilon} \right|_{\epsilon=0}$, where ϵ represents an independent “variational” variable. The variation of Q with respect to y is $\delta_y Q \equiv \frac{\partial Q}{\partial y} \delta y$, while the differential of Q with respect to w is $d_w Q \equiv \frac{\partial Q}{\partial w} dw$. The total differential (or for brevity “the differential”) of Q is $dQ \equiv \delta_y Q + d_w Q = \frac{\partial Q}{\partial y} \delta y + \frac{\partial Q}{\partial w} dw$. Colloquially, the variation of Q with respect to y means the change in Q due to a small change in y , the differential of Q with respect to w means the change in Q due to a small change in w , and the total differential of Q means the change in Q due to small changes in y and w . The extension to vectors of time-dependent functions and time-independent variables is straightforward. If \mathbf{y} is a vector of time-dependent functions, \mathbf{w} is a vector of time-independent variables, and Q is a scalar-valued function or functional depending on \mathbf{y} and \mathbf{w} , then the variation of \mathbf{y} is $\delta \mathbf{y} \equiv \left. \frac{\partial \mathbf{y}}{\partial \epsilon} \right|_{\epsilon=0}$, the

differential of \mathbf{y} is $d\mathbf{y} \equiv \delta\mathbf{y} + \dot{\mathbf{y}}dt = \left. \frac{\partial\mathbf{y}}{\partial\epsilon} \right|_{\epsilon=0} + \left. \frac{\partial\mathbf{y}}{\partial t} \right|_{\epsilon=0} dt$, the differential of \mathbf{w} is $d\mathbf{w} \equiv \left. \frac{d\mathbf{w}}{d\epsilon} \right|_{\epsilon=0}$, the variation of Q with respect to \mathbf{y} is $\delta_{\mathbf{y}}Q \equiv \frac{\partial Q}{\partial\mathbf{y}}\delta\mathbf{y}$, the differential of Q with respect to \mathbf{w} is $d_{\mathbf{w}}Q \equiv \frac{\partial Q}{\partial\mathbf{w}}d\mathbf{w}$, and the total differential (or for brevity “the differential”) of Q is $dQ \equiv \delta_{\mathbf{y}}Q + d_{\mathbf{w}}Q = \frac{\partial Q}{\partial\mathbf{y}}\delta\mathbf{y} + \frac{\partial Q}{\partial\mathbf{w}}d\mathbf{w}$.

To illustrate these definitions, consider the integral I of the function $F(t, \mathbf{y}(t))$ with respect to t with free upper limit of integration b and free lower limit of integration a

$$I = \int_a^b F(t, \mathbf{y}(t)) dt. \quad (3.1)$$

Applying the above definitions and using the Fundamental Theorem of Calculus, the differential of I is

$$\begin{aligned} dI &= \delta_{\mathbf{y}} \int_a^b F(t, \mathbf{y}(t)) dt + \frac{\partial}{\partial b} \left[\int_a^b F(t, \mathbf{y}(t)) dt \right] db + \frac{\partial}{\partial a} \left[\int_a^b F(t, \mathbf{y}(t)) dt \right] da \\ &= \int_a^b \delta_{\mathbf{y}} F(t, \mathbf{y}(t)) dt + F(b, \mathbf{y}(b)) db - F(a, \mathbf{y}(a)) da \\ &= \int_a^b \frac{\partial F}{\partial\mathbf{y}} \delta\mathbf{y} dt + [F(t, \mathbf{y}(t)) dt]_a^b, \end{aligned} \quad (3.2)$$

which is Leibnitz’s Rule.

3.B Pontryagin’s Minimum Principle

This section derives necessary conditions, called Pontryagin’s minimum principle, which a solution to an optimal control problem must satisfy. In the literature, application of Pontryagin’s minimum principle to solve an optimal control problem is called the indirect method. The derivation presented here follows [38]. Suppose a dynamical system has state $\mathbf{x} \in \mathbb{R}^n$ and control $\mathbf{u} \in \mathbb{R}^m$ and the control \mathbf{u} is sought that minimizes the performance index

$$J \equiv p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \int_a^b L(t, \mathbf{x}, \mathbf{u}, \mu) dt \quad (3.3)$$

subject to the system dynamics defined for $a \leq t \leq b$

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu), \quad (3.4)$$

the prescribed initial conditions at time $t = a$

$$\boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \quad (3.5)$$

and the prescribed final conditions at time $t = b$

$$\boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}. \quad (3.6)$$

p is a scalar-valued function called the endpoint cost function, L is a scalar-valued function called the integrand cost function, \mathbf{x} and \mathbf{f} are $n \times 1$ vector-valued functions, \mathbf{u} is an $m \times 1$ vector-valued function, $\boldsymbol{\sigma}$ is a $k_1 \times 1$ vector-valued function, and $\boldsymbol{\psi}$ is a $k_2 \times 1$ vector-valued function. The initial time a and final time

b may be prescribed or free. μ is a prescribed scalar parameter which may be exploited to numerically solve this problem via continuation. More concisely, this optimal control problem may be stated as

$$\min_{a, \mathbf{x}(a), b, \mathbf{x}(b), \mathbf{u}} \left[p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \int_a^b L(t, \mathbf{x}, \mathbf{u}, \mu) dt \right] \text{ s.t. } \begin{cases} \dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu), \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}, \end{cases} \quad (3.7)$$

or even more concisely as

$$\min_{a, \mathbf{x}(a), b, \mathbf{u}} J \text{ s.t. } \begin{cases} \dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu), \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}. \end{cases} \quad (3.8)$$

Observe that the optimal control problem encapsulated by (3.7) or (3.8) ignores path inequality constraints such as $\mathbf{D}(t, \mathbf{x}, \mathbf{u}, \mu) \leq \mathbf{0}$, where \mathbf{D} is a $r \times 1$ vector-valued function. Path inequality constraints can be incorporated in (3.7) or (3.8) as soft constraints through penalty functions in the integrand cost function L or the endpoint cost function p . The augmented performance index for this optimal control problem is obtained by adjoining the dynamic, initial, and final constraints to the original performance index (3.3) via Lagrange multipliers:

$$\begin{aligned} \tilde{J} &\equiv J + \boldsymbol{\xi}^\top \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) + \boldsymbol{\nu}^\top \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) + \int_a^b \boldsymbol{\lambda}^\top (\mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) - \dot{\mathbf{x}}) dt \\ &= p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \boldsymbol{\xi}^\top \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) + \boldsymbol{\nu}^\top \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) \\ &\quad + \int_a^b \left[L(t, \mathbf{x}, \mathbf{u}, \mu) + \boldsymbol{\lambda}^\top (\mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) - \dot{\mathbf{x}}) \right] dt \\ &= G(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) + \int_a^b \left[H(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu) - \boldsymbol{\lambda}^\top \dot{\mathbf{x}} \right] dt, \end{aligned} \quad (3.9)$$

where the endpoint function G and the Hamiltonian H are defined by

$$\begin{aligned} G(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) &\equiv p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \boldsymbol{\xi}^\top \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) + \boldsymbol{\nu}^\top \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) \\ H(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu) &\equiv L(t, \mathbf{x}, \mathbf{u}, \mu) + \boldsymbol{\lambda}^\top \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu), \end{aligned} \quad (3.10)$$

and where $\boldsymbol{\xi}$ is a $k_1 \times 1$ constant Lagrange multiplier vector, $\boldsymbol{\nu}$ is a $k_2 \times 1$ constant Lagrange multiplier vector, and $\boldsymbol{\lambda}$ is an $n \times 1$ time-varying Lagrange multiplier vector. In the literature, the time-varying Lagrange multiplier vector used to adjoin the system dynamics to the integrand cost function is often called the adjoint variable or the costate. Henceforth, the time-varying Lagrange multiplier vector is referred to as the costate and the elements in this vector are referred to as the costates.

Note on Normal and Abnormal Extremals There is a slightly more general formulation of the endpoint function and the Hamiltonian where $G(a, \mathbf{x}(a), \xi_0, \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) = \xi_0 p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \boldsymbol{\xi}^\top \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) + \boldsymbol{\nu}^\top \boldsymbol{\psi}(b, \mathbf{x}(b), \mu)$ and $H(t, \mathbf{x}, \lambda_0, \boldsymbol{\lambda}, \mathbf{u}, \mu) = \lambda_0 L(t, \mathbf{x}, \mathbf{u}, \mu) + \boldsymbol{\lambda}^\top \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu)$, where ξ_0 is a constant Lagrange multiplier scalar and λ_0 is a time-varying Lagrange multiplier scalar. But it can be shown (Pontryagin showed this in [39]) that λ_0 must be a nonnegative constant for an optimal solution! If $\lambda_0 > 0$, the extremal is normal, and if $\lambda_0 = 0$, the extremal is abnormal. If $\lambda_0 > 0$, then the Hamiltonian can be normalized so that $\lambda_0 = 1$.

Taking the differential of \tilde{J} yields

$$\begin{aligned}
d\tilde{J} &= G_a da + G_{\mathbf{x}(a)} d\mathbf{x}(a) + G_\xi d\xi + G_b db + G_{\mathbf{x}(b)} d\mathbf{x}(b) + G_\nu d\nu \\
&\quad + \left[\left(H - \boldsymbol{\lambda}^\top \dot{\mathbf{x}} \right) dt \right]_a^b + \int_a^b \left[H_{\mathbf{x}} \delta \mathbf{x} - \boldsymbol{\lambda}^\top \delta \dot{\mathbf{x}} + \left(H_\lambda - \dot{\mathbf{x}}^\top \right) \delta \boldsymbol{\lambda} + H_{\mathbf{u}} \delta \mathbf{u} \right] dt \\
&= (G_a - H|_{t=a}) da + G_{\mathbf{x}(a)} d\mathbf{x}(a) + G_\xi d\xi + (G_b + H|_{t=b}) db + G_{\mathbf{x}(b)} d\mathbf{x}(b) + G_\nu d\nu \\
&\quad - \left[\boldsymbol{\lambda}^\top (\delta \mathbf{x} + \dot{\mathbf{x}} dt) \right]_a^b + \int_a^b \left[\left(H_{\mathbf{x}} + \dot{\boldsymbol{\lambda}}^\top \right) \delta \mathbf{x} + \left(H_\lambda - \dot{\mathbf{x}}^\top \right) \delta \boldsymbol{\lambda} + H_{\mathbf{u}} \delta \mathbf{u} \right] dt \\
&= (G_a - H|_{t=a}) da + G_{\mathbf{x}(a)} d\mathbf{x}(a) + G_\xi d\xi + (G_b + H|_{t=b}) db + G_{\mathbf{x}(b)} d\mathbf{x}(b) + G_\nu d\nu \\
&\quad - \left[\boldsymbol{\lambda}^\top d\mathbf{x} \right]_a^b + \int_a^b \left[\left(H_{\mathbf{x}} + \dot{\boldsymbol{\lambda}}^\top \right) \delta \mathbf{x} + \left(H_\lambda - \dot{\mathbf{x}}^\top \right) \delta \boldsymbol{\lambda} + H_{\mathbf{u}} \delta \mathbf{u} \right] dt \\
&= (G_a - H|_{t=a}) da + \left(G_{\mathbf{x}(a)} + \boldsymbol{\lambda}^\top \Big|_{t=a} \right) d\mathbf{x}(a) + G_\xi d\xi \\
&\quad + (G_b + H|_{t=b}) db + \left(G_{\mathbf{x}(b)} - \boldsymbol{\lambda}^\top \Big|_{t=b} \right) d\mathbf{x}(b) + G_\nu d\nu \\
&\quad + \int_a^b \left[\left(H_{\mathbf{x}} + \dot{\boldsymbol{\lambda}}^\top \right) \delta \mathbf{x} + \left(H_\lambda - \dot{\mathbf{x}}^\top \right) \delta \boldsymbol{\lambda} + H_{\mathbf{u}} \delta \mathbf{u} \right] dt.
\end{aligned} \tag{3.11}$$

In the first equality, Leibnitz's rule (3.2) is used to compute the differential of the integral. In the second equality, integration by parts is used. In the third equality, the formula $d\mathbf{x}(t) = \delta\mathbf{x}(t) + \dot{\mathbf{x}}(t)dt$, or more concisely $d\mathbf{x} = \delta\mathbf{x} + \dot{\mathbf{x}}dt$, is used.

The necessary conditions on \mathbf{x} , $\boldsymbol{\lambda}$, and \mathbf{u} which make $d\tilde{J} = 0$ are the differential-algebraic equations (DAEs) defined for $a \leq t \leq b$

$$\begin{aligned}
\dot{\mathbf{x}} &= H_\lambda^\top(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu) = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) \\
\dot{\boldsymbol{\lambda}} &= -H_{\mathbf{x}}^\top(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu) \\
\mathbf{0} &= H_{\mathbf{u}}^\top(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu),
\end{aligned} \tag{3.12}$$

the left boundary conditions defined at time $t = a$

$$H|_{t=a} = G_a, \quad \boldsymbol{\lambda}|_{t=a} = -G_{\mathbf{x}(a)}^\top, \quad G_\xi^\top = \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \tag{3.13}$$

and the right boundary conditions defined at time $t = b$

$$H|_{t=b} = -G_b, \quad \boldsymbol{\lambda}|_{t=b} = G_{\mathbf{x}(b)}^\top, \quad G_\nu^\top = \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}. \tag{3.14}$$

If the initial time a is prescribed, then the left boundary condition $H|_{t=a} = G_a$ is dropped. If the final time b is prescribed, then the right boundary condition $H|_{t=b} = -G_b$ is dropped. The necessary conditions (3.12), (3.13), and (3.14) constitute a DAE TPBVP.

If $H_{\mathbf{u}\mathbf{u}}$ is nonsingular, then the optimal control problem is said to be regular or nonsingular; otherwise if $H_{\mathbf{u}\mathbf{u}}$ is singular, then the optimal control problem is said to be singular. If $H_{\mathbf{u}\mathbf{u}}$ is nonsingular, then by the implicit function theorem, the condition $H_{\mathbf{u}} = \mathbf{0}$ guarantees the existence of a unique function, say $\boldsymbol{\pi}$, for which $\mathbf{u} = \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu)$. If $H_{\mathbf{u}\mathbf{u}}$ is nonsingular and the condition $H_{\mathbf{u}} = \mathbf{0}$ can be used to construct an explicit function $\boldsymbol{\pi}$ such that $\mathbf{u} = \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu)$, then the Hamiltonian may be re-expressed as a function of

t , \mathbf{x} , $\boldsymbol{\lambda}$, and μ via the regular or reduced Hamiltonian $\hat{H}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) = H(t, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu), \mu)$ and the necessary conditions on \mathbf{x} and $\boldsymbol{\lambda}$ which make $d\tilde{J} = 0$ are the ODEs defined for $a \leq t \leq b$

$$\begin{aligned}\dot{\mathbf{x}} &= \hat{H}_{\boldsymbol{\lambda}}^{\top}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) \\ \dot{\boldsymbol{\lambda}} &= -\hat{H}_{\mathbf{x}}^{\top}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu),\end{aligned}\tag{3.15}$$

the left boundary conditions defined at time $t = a$

$$\hat{H}\Big|_{t=a} = G_a, \quad \boldsymbol{\lambda}\Big|_{t=a} = -G_{\mathbf{x}(a)}^{\top}, \quad G_{\boldsymbol{\xi}}^{\top} = \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0},\tag{3.16}$$

and the right boundary conditions defined at time $t = b$

$$\hat{H}\Big|_{t=b} = -G_b, \quad \boldsymbol{\lambda}\Big|_{t=b} = G_{\mathbf{x}(b)}^{\top}, \quad G_{\boldsymbol{\nu}}^{\top} = \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}.\tag{3.17}$$

If the initial time a is prescribed, then the left boundary condition $\hat{H}\Big|_{t=a} = G_a$ is dropped. If the final time b is prescribed, then the right boundary condition $\hat{H}\Big|_{t=b} = -G_b$ is dropped. The necessary conditions (3.15), (3.16), and (3.17) constitute an ODE TPBVP.

A solution of the DAE TPBVP (3.12), (3.13), and (3.14) or of the ODE TPBVP (3.15), (3.16), and (3.17) is said to be an extremal solution of the optimal control problem (3.8). Note that an extremal solution only satisfies necessary conditions for a minimum of the optimal control problem (3.8), so that an extremal solution is not guaranteed to be a local minimum of (3.8). Since the DAE BVP (3.12), (3.13), and (3.14) and the ODE TPBVP (3.15), (3.16), and (3.17) have small convergence radii, a continuation method (performing continuation in the parameter μ) is often required to numerically solve them starting from a solution to a simpler optimal control problem. The solution to the simpler optimal control problem might be obtained via analytics or a direct method. Appendices C and D describe predictor-corrector continuation methods for solving ODE TPBVP. In Chapters 4 and 5, the continuation parameter μ is used to vary integrand cost function coefficients in L in order to numerically solve the optimal control ODE TPBVPs for Suslov's problem and the rolling ball via continuation.

Because path inequality constraints have been omitted from the optimal control problem (3.8), the control is not restricted to lie in a compact set. Hence, an extremal solution solving (3.12), (3.13), and (3.14) or (3.15), (3.16), and (3.17) does not lie on any boundary. If the control is restricted to lie in a compact set (due to path inequality constraints), then the control of an extremal solution may lie on the boundary of this compact set; if the control is discontinuous, hopping abruptly between points on the boundary, then it is said to be bang-bang.

3.C Implementation Details for Solving the Optimal Control ODE TPBVP

In order to numerically solve the ODE TPBVP (3.15), (3.16), and (3.17), most solvers require that the ODE TPBVP be defined on a fixed time interval and any unknown parameters, such as $\boldsymbol{\xi}$, $\boldsymbol{\nu}$, a , and b , must

often be modeled as dummy constant dependent variables with zero derivatives. To aid convergence, solvers can exploit Jacobians of the ODE system and of the boundary conditions. Thus, (3.15) is redefined on the normalized time interval $[0, 1]$ through the change of independent variable $s = \frac{t-a}{T}$, where $T = b - a$. Note that $t(s) = Ts + a$. Define the normalized state $\tilde{\mathbf{x}}(s) = \mathbf{x}(t(s))$ and normalized costate $\tilde{\boldsymbol{\lambda}}(s) = \boldsymbol{\lambda}(t(s))$. Define the expanded un-normalized ODE TPBVP dependent variable vector

$$\mathbf{z}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \boldsymbol{\lambda}(t) \\ \boldsymbol{\xi} \\ \boldsymbol{\nu} \\ a \\ b \end{bmatrix}. \quad (3.18)$$

Defining $\tilde{\mathbf{z}}(s) = \mathbf{z}(t(s))$, the expanded normalized ODE TPBVP dependent variable vector is

$$\tilde{\mathbf{z}}(s) = \mathbf{z}(t(s)) = \begin{bmatrix} \mathbf{x}(t(s)) \\ \boldsymbol{\lambda}(t(s)) \\ \boldsymbol{\xi} \\ \boldsymbol{\nu} \\ a \\ b \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{x}}(s) \\ \tilde{\boldsymbol{\lambda}}(s) \\ \boldsymbol{\xi} \\ \boldsymbol{\nu} \\ a \\ b \end{bmatrix}. \quad (3.19)$$

By the chain rule, (3.15), and since $\frac{dt(s)}{ds} = T$,

$$\begin{aligned} \dot{\tilde{\mathbf{z}}}(s) &= \frac{d\tilde{\mathbf{z}}(s)}{ds} = \begin{bmatrix} \dot{\tilde{\mathbf{x}}}(s) \\ \dot{\tilde{\boldsymbol{\lambda}}}(s) \\ \mathbf{0}_{(k_1+k_2+2) \times 1} \end{bmatrix} = \frac{d\mathbf{z}(t(s))}{dt} \frac{dt(s)}{ds} = \begin{bmatrix} \frac{d\mathbf{x}(t(s))}{dt} \\ \frac{d\boldsymbol{\lambda}(t(s))}{dt} \\ \mathbf{0}_{(k_1+k_2+2) \times 1} \end{bmatrix} \frac{dt(s)}{ds} \\ &= \begin{bmatrix} \hat{H}_{\boldsymbol{\lambda}}^{\top}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu) \\ -\hat{H}_{\mathbf{x}}^{\top}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu) \\ \mathbf{0}_{(k_1+k_2+2) \times 1} \end{bmatrix} T \\ &= \begin{bmatrix} \hat{H}_{\boldsymbol{\lambda}}^{\top}(t(s), \tilde{\mathbf{x}}(s), \tilde{\boldsymbol{\lambda}}(s), \mu) \\ -\hat{H}_{\mathbf{x}}^{\top}(t(s), \tilde{\mathbf{x}}(s), \tilde{\boldsymbol{\lambda}}(s), \mu) \\ \mathbf{0}_{(k_1+k_2+2) \times 1} \end{bmatrix} T. \end{aligned} \quad (3.20)$$

Define $\tilde{\boldsymbol{\Phi}}(s, \tilde{\mathbf{z}}(s), \mu)$ to be the right-hand side of (3.20), so that

$$\tilde{\boldsymbol{\Phi}}(s, \tilde{\mathbf{z}}(s), \mu) = \begin{bmatrix} \hat{H}_{\boldsymbol{\lambda}}^{\top}(t(s), \tilde{\mathbf{x}}(s), \tilde{\boldsymbol{\lambda}}(s), \mu) \\ -\hat{H}_{\mathbf{x}}^{\top}(t(s), \tilde{\mathbf{x}}(s), \tilde{\boldsymbol{\lambda}}(s), \mu) \\ \mathbf{0}_{(k_1+k_2+2) \times 1} \end{bmatrix} T. \quad (3.21)$$

The Jacobian of $\tilde{\Phi}$ with respect to $\tilde{z}(s)$ is

$$\tilde{\Phi}_{\tilde{z}(s)}(s, \tilde{z}(s), \mu) = \begin{bmatrix} \hat{H}_{\lambda x} T & \hat{H}_{\lambda \lambda} T & \mathbf{0}_{n \times (k_1 + k_2)} & -\hat{H}_{\lambda}^T + \hat{H}_{\lambda t}(1-s)T & \hat{H}_{\lambda}^T + \hat{H}_{\lambda t} s T \\ -\hat{H}_{x x} T & -\hat{H}_{x \lambda} T & \mathbf{0}_{n \times (k_1 + k_2)} & \hat{H}_x^T - \hat{H}_{x t}(1-s)T & -\hat{H}_x^T - \hat{H}_{x t} s T \\ \mathbf{0}_{(k_1 + k_2 + 2) \times n} & \mathbf{0}_{(k_1 + k_2 + 2) \times n} & \mathbf{0}_{(k_1 + k_2 + 2) \times (k_1 + k_2)} & \mathbf{0}_{(k_1 + k_2 + 2) \times 1} & \mathbf{0}_{(k_1 + k_2 + 2) \times 1} \end{bmatrix} \quad (3.22)$$

and the Jacobian of $\tilde{\Phi}$ with respect to μ is

$$\tilde{\Phi}_{\mu}(s, \tilde{z}(s), \mu) = \begin{bmatrix} \hat{H}_{\lambda \mu} T \\ -\hat{H}_{x \mu} T \\ \mathbf{0}_{(k_1 + k_2 + 2) \times 1} \end{bmatrix}. \quad (3.23)$$

In (3.22) and (3.23), shorthand notation is used for conciseness and all first and second derivatives of \hat{H} are evaluated at $(s, \tilde{z}(s), \mu)$. An explanation of the meaning of the shorthand notation used to express all first and second derivatives of \hat{H} is given in Table 3.1.

Shorthand	More Shorthand	Normalized	Un-Normalized
\hat{H}_x^T	$= \hat{H}_x^T _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_x^T(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_x^T(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
\hat{H}_{λ}^T	$= \hat{H}_{\lambda}^T _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{\lambda}^T(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{\lambda}^T(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{x x}$	$= \hat{H}_{x x} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{x x}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{x x}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{x \lambda}$	$= \hat{H}_{x \lambda} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{x \lambda}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{x \lambda}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{x t}$	$= \hat{H}_{x t} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{x t}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{x t}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{x \mu}$	$= \hat{H}_{x \mu} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{x \mu}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{x \mu}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{\lambda \lambda}$	$= \hat{H}_{\lambda \lambda} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{\lambda \lambda}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{\lambda \lambda}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{\lambda x}$	$= \hat{H}_{\lambda x} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{\lambda x}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{\lambda x}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{\lambda t}$	$= \hat{H}_{\lambda t} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{\lambda t}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{\lambda t}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$
$\hat{H}_{\lambda \mu}$	$= \hat{H}_{\lambda \mu} _{(s, \tilde{z}(s), \mu)}$	$= \hat{H}_{\lambda \mu}(t(s), \tilde{x}(s), \tilde{\lambda}(s), \mu)$	$= \hat{H}_{\lambda \mu}(t(s), \mathbf{x}(t(s)), \boldsymbol{\lambda}(t(s)), \mu)$

Table 3.1: Explanation of shorthand notation for first and second derivatives of \hat{H} used in (3.22) and (3.23).

Now the boundary conditions (3.16)-(3.17) are considered. Letting

$$\Upsilon_1(z(a), z(b), \mu) = \begin{bmatrix} \hat{H}(a, \mathbf{x}(a), \boldsymbol{\lambda}(a), \mu) \\ \boldsymbol{\lambda}(a) \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) \\ \hat{H}(b, \mathbf{x}(b), \boldsymbol{\lambda}(b), \mu) \\ \boldsymbol{\lambda}(b) \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) \end{bmatrix}, \quad (3.24)$$

$$\Upsilon_2(z(a), z(b), \mu) = \begin{bmatrix} G_a(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ -G_{\mathbf{x}(a)}^\top(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_1 \times 1} \\ -G_b(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ G_{\mathbf{x}(b)}^\top(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_2 \times 1} \end{bmatrix}, \quad (3.25)$$

and

$$\begin{aligned} \Upsilon(z(a), z(b), \mu) &= \Upsilon_1(z(a), z(b), \mu) - \Upsilon_2(z(a), z(b), \mu) \\ &= \begin{bmatrix} \hat{H}(a, \mathbf{x}(a), \boldsymbol{\lambda}(a), \mu) \\ \boldsymbol{\lambda}(a) \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) \\ \hat{H}(b, \mathbf{x}(b), \boldsymbol{\lambda}(b), \mu) \\ \boldsymbol{\lambda}(b) \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) \end{bmatrix} - \begin{bmatrix} G_a(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ -G_{\mathbf{x}(a)}^\top(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_1 \times 1} \\ -G_b(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ G_{\mathbf{x}(b)}^\top(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_2 \times 1} \end{bmatrix}, \end{aligned} \quad (3.26)$$

the boundary conditions (3.16)-(3.17) in un-normalized dependent variables are

$$\Upsilon(z(a), z(b), \mu) = \mathbf{0}_{2n+k_1+k_2+2}. \quad (3.27)$$

The Jacobians of Υ with respect to $z(a)$, $z(b)$, and μ are

$$\Upsilon_{z(a)}(z(a), z(b), \mu) = \Upsilon_{1,z(a)}(z(a), z(b), \mu) - \Upsilon_{2,z(a)}(z(a), z(b), \mu), \quad (3.28)$$

$$\Upsilon_{z(b)}(z(a), z(b), \mu) = \Upsilon_{1,z(b)}(z(a), z(b), \mu) - \Upsilon_{2,z(b)}(z(a), z(b), \mu), \quad (3.29)$$

and

$$\Upsilon_\mu(z(a), z(b), \mu) = \Upsilon_{1,\mu}(z(a), z(b), \mu) - \Upsilon_{2,\mu}(z(a), z(b), \mu), \quad (3.30)$$

where

$$\Upsilon_{1,z(a)}(z(a), z(b), \mu) = \begin{bmatrix} \hat{H}_{\mathbf{x}(a)} & \hat{H}_{\boldsymbol{\lambda}(a)} & \mathbf{0}_{1 \times k_1} & \mathbf{0}_{1 \times k_2} & \hat{H}_a & 0 \\ \mathbf{0}_{n \times n} & I_{n \times n} & \mathbf{0}_{n \times k_1} & \mathbf{0}_{n \times k_2} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} \\ \boldsymbol{\sigma}_{\mathbf{x}(a)} & \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times k_1} & \mathbf{0}_{k_1 \times k_2} & \boldsymbol{\sigma}_a & \mathbf{0}_{k_1 \times 1} \\ \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times k_1} & \mathbf{0}_{1 \times k_2} & 0 & \hat{H}_b \\ \mathbf{0}_{n \times n} & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times k_1} & \mathbf{0}_{n \times k_2} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times k_1} & \mathbf{0}_{k_2 \times k_2} & \mathbf{0}_{k_2 \times 1} & \boldsymbol{\psi}_b \end{bmatrix}, \quad (3.31)$$

$$\Upsilon_{1,z(b)}(z(a), z(b), \mu) = \begin{bmatrix} \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times n} & \mathbf{0}_{1 \times k_1} & \mathbf{0}_{1 \times k_2} & \hat{H}_a & 0 \\ \mathbf{0}_{n \times n} & \mathbf{0}_{n \times n} & \mathbf{0}_{n \times k_1} & \mathbf{0}_{n \times k_2} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times k_1} & \mathbf{0}_{k_1 \times k_2} & \boldsymbol{\sigma}_a & \mathbf{0}_{k_1 \times 1} \\ \hat{H}_{\mathbf{x}(b)} & \hat{H}_{\boldsymbol{\lambda}(b)} & \mathbf{0}_{1 \times k_1} & \mathbf{0}_{1 \times k_2} & 0 & \hat{H}_b \\ \mathbf{0}_{n \times n} & I_{n \times n} & \mathbf{0}_{n \times k_1} & \mathbf{0}_{n \times k_2} & \mathbf{0}_{n \times 1} & \mathbf{0}_{n \times 1} \\ \boldsymbol{\psi}_{\mathbf{x}(b)} & \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times k_1} & \mathbf{0}_{k_2 \times k_2} & \mathbf{0}_{k_2 \times 1} & \boldsymbol{\psi}_b \end{bmatrix}, \quad (3.32)$$

$$\Upsilon_{1,\mu}(z(a), z(b), \mu) = \begin{bmatrix} \hat{H}_\mu|_a \\ \mathbf{0}_{n \times 1} \\ \boldsymbol{\sigma}_\mu \\ \hat{H}_\mu|_b \\ \mathbf{0}_{n \times 1} \\ \boldsymbol{\psi}_\mu \end{bmatrix}, \quad (3.33)$$

$$\Upsilon_{2,z(a)}(z(a), z(b), \mu) = \begin{bmatrix} G_{a\mathbf{x}(a)} & \mathbf{0}_{1 \times n} & G_{a\xi} & G_{a\nu} & G_{aa} & G_{ab} \\ -G_{\mathbf{x}(a)\mathbf{x}(a)} & \mathbf{0}_{n \times n} & -G_{\mathbf{x}(a)\xi} & -G_{\mathbf{x}(a)\nu} & -G_{\mathbf{x}(a)a} & -G_{\mathbf{x}(a)b} \\ \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times k_1} & \mathbf{0}_{k_1 \times k_2} & \mathbf{0}_{k_1 \times 1} & \mathbf{0}_{k_1 \times 1} \\ -G_{b\mathbf{x}(a)} & \mathbf{0}_{1 \times n} & -G_{b\xi} & -G_{b\nu} & -G_{ba} & -G_{bb} \\ G_{\mathbf{x}(b)\mathbf{x}(a)} & \mathbf{0}_{n \times n} & G_{\mathbf{x}(b)\xi} & G_{\mathbf{x}(b)\nu} & G_{\mathbf{x}(b)a} & G_{\mathbf{x}(b)b} \\ \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times k_1} & \mathbf{0}_{k_2 \times k_2} & \mathbf{0}_{k_2 \times 1} & \mathbf{0}_{k_2 \times 1} \end{bmatrix}, \quad (3.34)$$

$$\Upsilon_{2,z(b)}(z(a), z(b), \mu) = \begin{bmatrix} G_{a\mathbf{x}(b)} & \mathbf{0}_{1 \times n} & G_{a\xi} & G_{a\nu} & G_{aa} & G_{ab} \\ -G_{\mathbf{x}(a)\mathbf{x}(b)} & \mathbf{0}_{n \times n} & -G_{\mathbf{x}(a)\xi} & -G_{\mathbf{x}(a)\nu} & -G_{\mathbf{x}(a)a} & -G_{\mathbf{x}(a)b} \\ \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times n} & \mathbf{0}_{k_1 \times k_1} & \mathbf{0}_{k_1 \times k_2} & \mathbf{0}_{k_1 \times 1} & \mathbf{0}_{k_1 \times 1} \\ -G_{b\mathbf{x}(b)} & \mathbf{0}_{1 \times n} & -G_{b\xi} & -G_{b\nu} & -G_{ba} & -G_{bb} \\ G_{\mathbf{x}(b)\mathbf{x}(b)} & \mathbf{0}_{n \times n} & G_{\mathbf{x}(b)\xi} & G_{\mathbf{x}(b)\nu} & G_{\mathbf{x}(b)a} & G_{\mathbf{x}(b)b} \\ \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times n} & \mathbf{0}_{k_2 \times k_1} & \mathbf{0}_{k_2 \times k_2} & \mathbf{0}_{k_2 \times 1} & \mathbf{0}_{k_2 \times 1} \end{bmatrix}, \quad (3.35)$$

and

$$\Upsilon_{2,\mu}(z(a), z(b), \mu) = \begin{bmatrix} G_{a\mu} \\ -G_{\mathbf{x}(a)\mu} \\ \mathbf{0}_{k_1 \times 1} \\ -G_{b\mu} \\ G_{\mathbf{x}(b)\mu} \\ \mathbf{0}_{k_2 \times 1} \end{bmatrix}. \quad (3.36)$$

In equations (3.31), (3.32), and (3.33), all first derivatives of \hat{H} in row 1 are evaluated at $(a, \mathbf{x}(a), \boldsymbol{\lambda}(a), \mu)$, all first derivatives of $\boldsymbol{\sigma}$ in rows $n+2$ through $n+1+k_1$ are evaluated at $(a, \mathbf{x}(a), \mu)$, all first derivatives of \hat{H} in row $n+2+k_1$ are evaluated at $(b, \mathbf{x}(b), \boldsymbol{\lambda}(b), \mu)$, and all first derivatives of $\boldsymbol{\psi}$ in rows $2n+3+k_1$ through $2n+2+k_1+k_2$ are evaluated at $(b, \mathbf{x}(b), \mu)$. In equations (3.34), (3.35), and (3.36), all second derivatives of G are evaluated at $(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu)$.

To express the boundary conditions in terms of normalized dependent variables, let $\tilde{\Upsilon}_1(\tilde{z}(0), \tilde{z}(1), \mu) = \Upsilon_1(z(a), z(b), \mu)$, $\tilde{\Upsilon}_2(\tilde{z}(0), \tilde{z}(1), \mu) = \Upsilon_2(z(a), z(b), \mu)$, and $\tilde{\Upsilon}(\tilde{z}(0), \tilde{z}(1), \mu) = \Upsilon(z(a), z(b), \mu)$. Thus

$$\tilde{\Upsilon}_1(\tilde{z}(0), \tilde{z}(1), \mu) = \begin{bmatrix} \hat{H}(a, \tilde{\mathbf{x}}(0), \tilde{\boldsymbol{\lambda}}(0), \mu) \\ \tilde{\boldsymbol{\lambda}}(0) \\ \boldsymbol{\sigma}(a, \tilde{\mathbf{x}}(0), \mu) \\ \hat{H}(b, \tilde{\mathbf{x}}(1), \tilde{\boldsymbol{\lambda}}(1), \mu) \\ \tilde{\boldsymbol{\lambda}}(1) \\ \boldsymbol{\psi}(b, \tilde{\mathbf{x}}(1), \mu) \end{bmatrix}, \quad (3.37)$$

$$\tilde{\Upsilon}_2(\tilde{z}(0), \tilde{z}(1), \mu) = \begin{bmatrix} G_a(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ -G_{\mathbf{x}(a)}^\top(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_1 \times 1} \\ -G_b(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ G_{\mathbf{x}(b)}^\top(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_2 \times 1} \end{bmatrix}, \quad (3.38)$$

and

$$\begin{aligned} \tilde{\Upsilon}(\tilde{z}(0), \tilde{z}(1), \mu) &= \tilde{\Upsilon}_1(\tilde{z}(0), \tilde{z}(1), \mu) - \tilde{\Upsilon}_2(\tilde{z}(0), \tilde{z}(1), \mu) \\ &= \begin{bmatrix} \hat{H}(a, \tilde{x}(0), \tilde{\boldsymbol{\lambda}}(0), \mu) \\ \tilde{\boldsymbol{\lambda}}(0) \\ \boldsymbol{\sigma}(a, \tilde{x}(0), \mu) \\ \hat{H}(b, \tilde{x}(1), \tilde{\boldsymbol{\lambda}}(1), \mu) \\ \tilde{\boldsymbol{\lambda}}(1) \\ \psi(b, \tilde{x}(1), \mu) \end{bmatrix} - \begin{bmatrix} G_a(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ -G_{\mathbf{x}(a)}^\top(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_1 \times 1} \\ -G_b(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ G_{\mathbf{x}(b)}^\top(a, \tilde{x}(0), \boldsymbol{\xi}, b, \tilde{x}(1), \boldsymbol{\nu}, \mu) \\ \mathbf{0}_{k_2 \times 1} \end{bmatrix}, \end{aligned} \quad (3.39)$$

and the boundary conditions (3.16)-(3.17) in normalized dependent variables are

$$\tilde{\Upsilon}(\tilde{z}(0), \tilde{z}(1), \mu) = \mathbf{0}_{(2n+k_1+k_2+2) \times 1}. \quad (3.40)$$

The Jacobians of $\tilde{\Upsilon}$ with respect to $\tilde{z}(0)$, $\tilde{z}(1)$, and μ are

$$\tilde{\Upsilon}_{\tilde{z}(0)}(\tilde{z}(0), \tilde{z}(1), \mu) = \tilde{\Upsilon}_{1, \tilde{z}(0)}(\tilde{z}(0), \tilde{z}(1), \mu) - \Upsilon_{2, \tilde{z}(0)}(\tilde{z}(0), \tilde{z}(1), \mu), \quad (3.41)$$

$$\tilde{\Upsilon}_{\tilde{z}(1)}(\tilde{z}(0), \tilde{z}(1), \mu) = \tilde{\Upsilon}_{1, \tilde{z}(1)}(\tilde{z}(0), \tilde{z}(1), \mu) - \tilde{\Upsilon}_{2, \tilde{z}(1)}(\tilde{z}(0), \tilde{z}(1), \mu), \quad (3.42)$$

and

$$\tilde{\Upsilon}_\mu(\tilde{z}(0), \tilde{z}(1), \mu) = \tilde{\Upsilon}_{1, \mu}(\tilde{z}(0), \tilde{z}(1), \mu) - \tilde{\Upsilon}_{2, \mu}(\tilde{z}(0), \tilde{z}(1), \mu), \quad (3.43)$$

where the equality between the Jacobians of $\tilde{\Upsilon}$, $\tilde{\Upsilon}_1$, and $\tilde{\Upsilon}_2$ with respect to $\tilde{z}(0)$, $\tilde{z}(1)$, and μ and the Jacobians of Υ , Υ_1 , and Υ_2 with respect to $z(0)$, $z(1)$, and μ is given in Table 3.2.

Normalized	Un-Normalized
$\tilde{\Upsilon}_{\tilde{z}(0)}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{z(a)}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{1, \tilde{z}(0)}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{1, z(a)}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{2, \tilde{z}(0)}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{2, z(a)}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{\tilde{z}(1)}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{z(b)}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{1, \tilde{z}(1)}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{1, z(b)}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{2, \tilde{z}(1)}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{2, z(b)}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_\mu(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_\mu(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{1, \mu}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{1, \mu}(z(a), z(b), \mu)$
$\tilde{\Upsilon}_{2, \mu}(\tilde{z}(0), \tilde{z}(1), \mu)$	$= \Upsilon_{2, \mu}(z(a), z(b), \mu)$

Table 3.2: Equality between Jacobians of boundary condition functions in normalized and un-normalized coordinates.

Special care must be taken when implementing the Jacobians (3.41) and (3.42). Since the unknown constants ξ , ν , a , and b appear at the end of both $\tilde{z}(0)$ and $\tilde{z}(1)$, the unknown constants from only one of $\tilde{z}(0)$ and $\tilde{z}(1)$ are actually used to construct each term in $\tilde{\mathbf{Y}}$ involving ξ , ν , a , and b . The trailing columns in (3.41) are actually the Jacobian of $\tilde{\mathbf{Y}}$ with respect to ξ , ν , a , and b in $\tilde{z}(0)$, while the trailing columns in (3.42) are actually the Jacobian of $\tilde{\mathbf{Y}}$ with respect to ξ , ν , a , and b in $\tilde{z}(1)$. Thus, the trailing columns in (3.41) and (3.42) corresponding to the Jacobian of $\tilde{\mathbf{Y}}$ with respect to ξ , ν , a , and b should not coincide in a software implementation. For example, if the unknown constants are extracted from $\tilde{z}(0)$ to construct $\tilde{\mathbf{Y}}$, $\tilde{\mathbf{Y}}_{\tilde{z}(0)}$ is as shown in (3.41) while the trailing columns in (3.42) corresponding to the Jacobian of $\tilde{\mathbf{Y}}$ with respect to the unknown constants in $\tilde{z}(1)$ should be all zeros. Alternatively, if the unknown constants are extracted from $\tilde{z}(1)$ to construct $\tilde{\mathbf{Y}}$, $\tilde{\mathbf{Y}}_{\tilde{z}(1)}$ is as shown in (3.42) while the trailing columns in (3.41) corresponding to the Jacobian of $\tilde{\mathbf{Y}}$ with respect to the unknown constants in $\tilde{z}(0)$ should be all zeros.

In equations (3.18), (3.19), (3.20), (3.21), and (3.23), the second to last row is needed only if the initial time a is free and the last row is needed only if the final time b is free. In equation (3.22), the second to last row and column are needed only if the initial time a is free and the last row and column are needed only if the final time b is free.

In equations (3.24), (3.25), (3.26), (3.27), (3.30), (3.33), (3.36), (3.37), (3.38), (3.39), (3.40), and (3.43) the first row is needed only if the initial time a is free and row $n + k_1 + 2$ is needed only if the final time b is free. In equations (3.28), (3.29), (3.31), (3.32), (3.34), (3.35), (3.41), and (3.42) the first row and second to last column are needed only if the initial time a is free and row $n + k_1 + 2$ and the last column are needed only if the final time b is free.

In order to numerically solve the ODE TPBVP (3.15), (3.16), and (3.17) without continuation or with a monotonic continuation solver (such as `acdc` or `acdcc`), the solver should be provided (3.21), (3.22), (3.39), (3.41), and (3.42). In order to numerically solve the ODE TPBVP (3.15), (3.16), and (3.17) with a non-monotonic continuation solver (such as the predictor-corrector methods discussed in Appendices C and D), the solver should be provided (3.21), (3.22), (3.23), (3.39), (3.41), (3.42), and (3.43). While (3.21), (3.22), (3.23), (3.39), (3.41), (3.42), and (3.43) are complicated, they are readily constructed numerically in Chapters 4 and 5 to simulate the optimal control of Suslov's problem and the rolling ball through automatic differentiation of the regular Hamiltonian \hat{H} and the endpoint function G . There are many free automatic differentiation toolboxes available. Specifically, the MATLAB automatic differentiation toolbox `ADiGator` [40, 41] is utilized in this research to construct the equations numerically. Moreover, these equations are constructed numerically very efficiently in MATLAB by exploiting vectorization; the non-vectorized version of these equations execute too slowly in MATLAB to complete timely simulations. The use of vectorized automatic differentiation to realize these equations in MATLAB is noteworthy, because it is tedious to manually derive the non-vectorized version of these equations and terribly difficult to manually derive the vectorized version of these equations.

Chapter 4

Suslov's Problem

This chapter investigates the optimal control of Suslov's problem. By applying Euler-Poincaré's method, Lagrange-d'Alembert's principle, and Pontryagin's minimum principle, Section 4.A derives the uncontrolled and controlled equations of motion for Suslov's problem for an arbitrary group, while Section 4.B derives the uncontrolled and controlled equations of motion for Suslov's problem for $SO(3)$. Subsection 4.B.2 shows that Suslov's problem for $SO(3)$ is controllable. In Section 4.C, the controlled equations of motion for Suslov's problem for $SO(3)$ are solved numerically via monotonic continuation, starting from an analytical solution to a singular optimal control problem. The numerical solution of the controlled equations of motion for Suslov's problem for $SO(3)$ is aided by the numerical construction of their Jacobians via automatic differentiation.

4.A Suslov's Optimal Control Problem for an Arbitrary Group

4.A.1 Derivation of Suslov's Uncontrolled Equations of Motion

Suppose G is a Lie group having all appropriate properties for the application of Euler-Poincaré's method [42, 16]. As discussed in Section 2.A, if the Lagrangian $L = L(g, \dot{g})$ is left G -invariant, then the problem can be reduced to the consideration of the symmetry-reduced Lagrangian $\ell(\Omega)$ with $\Omega = g^{-1}\dot{g}$. Here, we concentrate on the left-invariant Lagrangians as being pertinent to the dynamics of a rigid body. A parallel theory of right-invariant Lagrangians can be developed as well in a completely equivalent fashion [16]. We also assume that there is a suitable pairing between the Lie algebra \mathfrak{g} and its dual \mathfrak{g}^* , which leads to the coadjoint operator defined previously in (2.19) and repeated below:

$$\langle \text{ad}_a^* \alpha, b \rangle := \langle \alpha, \text{ad}_a b \rangle \quad \forall a, b \in \mathfrak{g}, \alpha \in \mathfrak{g}^*.$$

Then, the equations of motion are obtained by Euler-Poincaré's method

$$\delta \int_a^b \ell(\Omega) dt = 0 \tag{4.1}$$

with the variations $\delta\Omega$ satisfying

$$\delta\Omega = \dot{\eta} + \text{ad}_\Omega \eta, \quad (4.2)$$

where $\eta(t)$ is an arbitrary \mathfrak{g} -valued function satisfying $\eta(a) = \eta(b) = 0$. Then, the equations of motion are the *Euler-Poincaré* equations of motion

$$\dot{\Pi} - \text{ad}_\Omega^* \Pi = 0, \quad \Pi := \frac{\delta\ell}{\delta\Omega}. \quad (4.3)$$

Let $\xi(t) \in \mathfrak{g}^*$, with $\xi(t) \neq 0 \quad \forall t$ and introduce the constraint

$$\langle \xi, \Omega \rangle = \gamma(\xi, t). \quad (4.4)$$

Due to the constraint (4.4), Lagrange-d'Alembert's principle states that the variations $\eta \in \mathfrak{g}$ have to satisfy

$$\langle \xi, \eta \rangle = 0. \quad (4.5)$$

Using (4.5), Suslov's uncontrolled equations of motion are obtained:

$$\dot{\Pi} - \text{ad}_\Omega^* \Pi = \lambda \xi, \quad \Pi := \frac{\delta\ell}{\delta\Omega}, \quad (4.6)$$

where λ is the Lagrange multiplier enforcing (4.5). In order to explicitly solve (4.6) for λ , we will need to further assume a linear connection between the angular momentum Π and the angular velocity Ω . Thus, we assume that $\Pi = \mathbb{I}\Omega$, where $\mathbb{I} : \mathfrak{g} \rightarrow \mathfrak{g}^*$ is an invertible linear operator with an adjoint $\mathbb{I}^* : \mathfrak{g}^* \rightarrow \mathfrak{g}$; \mathbb{I} has the physical meaning of the inertia operator when the Lie group G under consideration is the rotation group $SO(3)$. Under this assumption, we pair both sides of (4.6) with $\mathbb{I}^{-1*}\xi$ and obtain the following expression for the Lagrange multiplier λ :

$$\lambda = \frac{\frac{d}{dt} \langle \Pi, \mathbb{I}^{-1*}\xi \rangle - \left\langle \Pi, \frac{d}{dt} [\mathbb{I}^{-1*}\xi] + \text{ad}_\Omega \mathbb{I}^{-1*}\xi \right\rangle}{\langle \xi, \mathbb{I}^{-1*}\xi \rangle} \quad (4.7)$$

One motivation for this particular pairing is that the denominator in (4.7) is non-zero for non-zero ξ , enabling an explicit solution for λ . Making use of the constraint (4.4) in the formula (4.7) for λ by using the assumption $\Pi = \mathbb{I}\Omega$, $\langle \Pi, \mathbb{I}^{-1*}\xi \rangle = \langle \mathbb{I}\Omega, \mathbb{I}^{-1*}\xi \rangle = \langle \Omega, \xi \rangle = \gamma(\xi, t)$, so that (4.7) becomes

$$\lambda(\Omega, \xi) = \frac{\frac{d}{dt} [\gamma(\xi, t)] - \left\langle \mathbb{I}\Omega, \frac{d}{dt} [\mathbb{I}^{-1*}\xi] + \text{ad}_\Omega \mathbb{I}^{-1*}\xi \right\rangle}{\langle \xi, \mathbb{I}^{-1*}\xi \rangle}. \quad (4.8)$$

If we moreover assume that $\gamma(\xi, t)$ is a constant, e.g. $\gamma(\xi, t) = 0$ as is in the standard formulation of Suslov's problem, and $\mathbb{I} : \mathfrak{g} \rightarrow \mathfrak{g}^*$ is a time-independent, invertible linear operator that is also self-adjoint (i.e. $\mathbb{I} = \mathbb{I}^*$), then (4.8) simplifies to

$$\lambda(\Omega, \xi) = - \frac{\langle \mathbb{I}\Omega, \text{ad}_\Omega \mathbb{I}^{-1}\xi \rangle + \langle \Omega, \dot{\xi} \rangle}{\langle \xi, \mathbb{I}^{-1}\xi \rangle}, \quad (4.9)$$

the kinetic energy is

$$T(t) = \frac{1}{2} \langle \Omega, \Pi \rangle = \frac{1}{2} \langle \Omega, \mathbb{I}\Omega \rangle, \quad (4.10)$$

the time derivative of the kinetic energy is

$$\begin{aligned} \dot{T}(t) &= \frac{1}{2} \left[\langle \dot{\Omega}, \mathbb{I}\Omega \rangle + \langle \Omega, \mathbb{I}\dot{\Omega} \rangle \right] = \left\langle \Omega, \frac{1}{2} [\mathbb{I} + \mathbb{I}^*] \dot{\Omega} \right\rangle = \langle \Omega, \mathbb{I}\dot{\Omega} \rangle = \langle \Omega, \dot{\Pi} \rangle \\ &= \langle \Omega, \text{ad}_{\Omega}^* \Pi + \lambda \xi \rangle = \langle \text{ad}_{\Omega} \Omega, \Pi \rangle + \lambda \langle \Omega, \xi \rangle = \lambda \gamma(\xi, t), \end{aligned} \quad (4.11)$$

and kinetic energy is conserved if $\gamma(\xi, t) = 0$.

4.A.2 Derivation of Suslov's Controlled Equations of Motion

Consider the problem (4.6) and assume that $\Pi = \mathbb{I}\Omega$ so that the explicit equation for the Lagrange multiplier (4.8) holds. We now turn to the central question of this chapter, namely, optimal control of the system by varying the nullifier (or annihilator) $\xi(t)$. The optimal control problem is defined as follows. Consider a fixed initial time a , a fixed or free final time $b > a$, the integrand cost function $C(\Omega, \dot{\Omega}, \xi, \dot{\xi}, t)$, and the following optimal control problem

$$\min_{\xi(t), b} \int_a^b C(\Omega, \dot{\Omega}, \xi, \dot{\xi}, t) dt \quad \text{subject to } \Omega(t), \xi(t) \text{ satisfying (4.6) and (4.8)} \quad (4.12)$$

and subject to the initial and final conditions $\Omega(a) = \Omega_a$ and $\Omega(b) = \Omega_b$. Observe that this optimal control problem ignores path inequality constraints such as $\mathbf{D}(\Omega, \dot{\Omega}, \xi, \dot{\xi}, t) \leq \mathbf{0}$, where \mathbf{D} is a $r \times 1$ vector-valued function. Path inequality constraints can be incorporated in (4.12) as soft constraints through penalty functions in the integrand cost function C .

To solve this optimal control problem, construct the augmented performance index

$$\begin{aligned} S &= \langle \rho, \Omega(a) - \Omega_a \rangle + \langle \nu, \Omega(b) - \Omega_b \rangle + \int_a^b [C + \langle \kappa, (\mathbb{I}\Omega) \cdot - \text{ad}_{\Omega}^* \mathbb{I}\Omega - \lambda \xi \rangle] dt \\ &= \langle \rho, \Omega(a) - \Omega_a \rangle + \langle \nu, \Omega(b) - \Omega_b \rangle + \langle \kappa, \mathbb{I}\Omega \rangle_a^b + \int_a^b [C - \langle \dot{\kappa} + \text{ad}_{\Omega} \kappa, \mathbb{I}\Omega \rangle - \lambda \langle \kappa, \xi \rangle] dt, \end{aligned} \quad (4.13)$$

where the additional unknowns are a \mathfrak{g} -valued function of time $\kappa(t)$ enforcing the uncontrolled equations of motion and the constants $\rho, \nu \in \mathfrak{g}^*$ enforcing the initial and final conditions.

Remark 4.1 (On the nature of the pairing in (4.13)). For simplicity of calculation and notation, we assume that the pairing in (4.13) between vectors in \mathfrak{g} and \mathfrak{g}^* is the same as the one used in the derivation of Suslov's problem in Subsection 4.A.1. In principle, one could use a different pairing which would necessitate a different notation for the ad operator. We believe that while such generalization is rather straightforward, it introduces a cumbersome and non-intuitive notation. For the case when $G = SO(3)$ considered later in Section 4.B, we will take the simplest possible pairing, the scalar product of vectors in \mathbb{R}^3 . In that case, the ad and ad^* operators are simply the vector cross product with an appropriate sign.

Pontryagin's minimum principle gives necessary conditions that a minimum solution of (4.12) must satisfy,

if it exists. These necessary conditions are obtained by equating the differential of S to 0, resulting in appropriately coupled equations for the state and control variables. While this calculation is well-established [43, 38], we present it here for completeness of the exposition as it is relevant to our further discussion.

Following [43], we denote all variations of S coming from the time-dependent variables κ , Ω , and ξ as δS and write $\delta S = \delta_\kappa S + \delta_\Omega S + \delta_\xi S$. By using partial differentiation, the variation of S with respect to each time-independent variable ρ , ν , and b is $\left\langle \frac{\partial S}{\partial \rho}, d\rho \right\rangle$, $\left\langle \frac{\partial S}{\partial \nu}, d\nu \right\rangle$, and $\frac{\partial S}{\partial b} db$, respectively. Thus, the differential of S is given by

$$\begin{aligned} dS &= \delta S + \left\langle \frac{\partial S}{\partial \rho}, d\rho \right\rangle + \left\langle \frac{\partial S}{\partial \nu}, d\nu \right\rangle + \frac{\partial S}{\partial b} db \\ &= \delta_\kappa S + \delta_\Omega S + \delta_\xi S + \left\langle \frac{\partial S}{\partial \rho}, d\rho \right\rangle + \left\langle \frac{\partial S}{\partial \nu}, d\nu \right\rangle + \frac{\partial S}{\partial b} db. \end{aligned} \quad (4.14)$$

Each term in dS is computed below. It is important to present this calculation in some detail, in particular, because of the contribution of the boundary conditions. The variation of S with respect to κ is

$$\delta_\kappa S = \int_a^b \langle (\mathbb{I}\Omega)^\cdot - \text{ad}_\Omega^* \mathbb{I}\Omega - \lambda \xi, \delta \kappa \rangle dt. \quad (4.15)$$

Since $\delta\Omega(a) = 0$, $d\Omega(b) = \delta\Omega(b) + \dot{\Omega}(b)db$, and

$$\begin{aligned} \delta_\Omega \langle \kappa, \text{ad}_\Omega^* \mathbb{I}\Omega \rangle &= \langle \kappa, \text{ad}_{\delta\Omega}^* \mathbb{I}\Omega \rangle + \langle \kappa, \text{ad}_\Omega^* \mathbb{I}\delta\Omega \rangle \\ &= \langle \text{ad}_{\delta\Omega} \kappa, \mathbb{I}\Omega \rangle + \langle \text{ad}_\Omega \kappa, \mathbb{I}\delta\Omega \rangle \\ &= \langle -\text{ad}_\kappa \delta\Omega, \mathbb{I}\Omega \rangle + \langle \mathbb{I}^* \text{ad}_\Omega \kappa, \delta\Omega \rangle \\ &= \langle -\text{ad}_\kappa^* \mathbb{I}\Omega + \mathbb{I}^* \text{ad}_\Omega \kappa, \delta\Omega \rangle, \end{aligned} \quad (4.16)$$

the variation of S with respect to Ω is

$$\begin{aligned} \delta_\Omega S &= \langle \rho, \delta\Omega(a) \rangle + \langle \nu, \delta\Omega(b) \rangle + \langle \kappa, \mathbb{I}\delta\Omega \rangle \Big|_a^b + \left\langle \frac{\partial C}{\partial \dot{\Omega}}, \delta\Omega \right\rangle \Big|_a^b \\ &\quad + \int_a^b \left\langle \frac{\partial C}{\partial \Omega} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\Omega}} - \mathbb{I}^* (\dot{\kappa} + \text{ad}_\Omega \kappa) + \text{ad}_\kappa^* \mathbb{I}\Omega - \frac{\partial \lambda}{\partial \Omega} \langle \kappa, \xi \rangle, \delta\Omega \right\rangle dt \\ &= \left\langle \nu + \mathbb{I}^* \kappa + \frac{\partial C}{\partial \dot{\Omega}}, \delta\Omega \right\rangle \Big|_{t=b} + \int_a^b \left\langle \frac{\partial C}{\partial \Omega} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\Omega}} - \mathbb{I}^* (\dot{\kappa} + \text{ad}_\Omega \kappa) + \text{ad}_\kappa^* \mathbb{I}\Omega - \frac{\partial \lambda}{\partial \Omega} \langle \kappa, \xi \rangle, \delta\Omega \right\rangle dt \quad (4.17) \\ &= \left\langle \nu + \mathbb{I}^* \kappa + \frac{\partial C}{\partial \dot{\Omega}}, d\dot{\Omega} \right\rangle \Big|_{t=b} - \left\langle \nu + \mathbb{I}^* \kappa + \frac{\partial C}{\partial \dot{\Omega}}, \dot{\Omega} \right\rangle \Big|_{t=b} db \\ &\quad + \int_a^b \left\langle \frac{\partial C}{\partial \Omega} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\Omega}} - \mathbb{I}^* (\dot{\kappa} + \text{ad}_\Omega \kappa) + \text{ad}_\kappa^* \mathbb{I}\Omega - \frac{\partial \lambda}{\partial \Omega} \langle \kappa, \xi \rangle, \delta\Omega \right\rangle dt. \end{aligned}$$

Since $d\xi(b) = \delta\xi(b) + \dot{\xi}(b)db$, the variation of S with respect to ξ is

$$\begin{aligned}\delta_\xi S &= \left\langle \frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}}, \delta \xi \right\rangle \Big|_a^b + \int_a^b \left\langle -\frac{d}{dt} \left(\frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}} \right) + \left(\frac{\partial C}{\partial \xi} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \xi} \right) - \lambda \kappa, \delta \xi \right\rangle dt \\ &= \left\langle \frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}}, d\xi \right\rangle \Big|_{t=b} - \left\langle \frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}}, \dot{\xi} \right\rangle \Big|_{t=b} db - \left\langle \frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}}, \delta \xi \right\rangle \Big|_{t=a} \\ &\quad + \int_a^b \left\langle -\frac{d}{dt} \left(\frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}} \right) + \left(\frac{\partial C}{\partial \xi} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \xi} \right) - \lambda \kappa, \delta \xi \right\rangle dt.\end{aligned}\tag{4.18}$$

The remaining terms in dS , due to variations of S with respect to the time-independent variables, are

$$\left\langle \frac{\partial S}{\partial \rho}, d\rho \right\rangle = \langle \Omega(a) - \Omega_a, d\rho \rangle,\tag{4.19}$$

$$\left\langle \frac{\partial S}{\partial \nu}, d\nu \right\rangle = \langle \Omega(b) - \Omega_b, d\nu \rangle,\tag{4.20}$$

and

$$\frac{\partial S}{\partial b} db = \left[\langle \nu, \dot{\Omega} \rangle + C + \langle \kappa, (\mathbb{I}\Omega)^\cdot - \text{ad}_\Omega^* \mathbb{I}\Omega - \lambda \xi \rangle \right]_{t=b} db.\tag{4.21}$$

Adding all the terms in dS together and demanding that $dS = 0$ for all $\delta\kappa$, $\delta\Omega$, $\delta\xi$, $d\Omega(b)$, $d\xi(b)$, $d\rho$, $d\nu$, and db (note here that $\delta\kappa$, $\delta\Omega$, and $\delta\xi$ are variations defined for $a \leq t \leq b$) gives the two-point boundary value problem defined by the following equations of motion on $a \leq t \leq b$

$$\delta\kappa: \quad (\mathbb{I}\Omega)^\cdot - \text{ad}_\Omega^* \mathbb{I}\Omega - \lambda \xi = 0\tag{4.22}$$

$$\delta\Omega: \quad \frac{\partial C}{\partial \Omega} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\Omega}} - \mathbb{I}^* (\dot{\kappa} + \text{ad}_\Omega \kappa) + \text{ad}_\kappa^* \mathbb{I}\Omega - \frac{\partial \lambda}{\partial \Omega} \langle \kappa, \xi \rangle = 0\tag{4.23}$$

$$\delta\xi: \quad -\frac{d}{dt} \left(\frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}} \right) + \left(\frac{\partial C}{\partial \xi} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \xi} \right) - \lambda \kappa = 0\tag{4.24}$$

the left boundary conditions at $t = a$

$$d\rho: \quad \Omega(a) = \Omega_a\tag{4.25}$$

$$\delta\xi(a): \quad \left[\frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}} \right]_{t=a} = 0\tag{4.26}$$

and the right boundary conditions at $t = b$

$$d\nu: \quad \Omega(b) = \Omega_b\tag{4.27}$$

$$d\xi(b): \quad \left[\frac{\partial C}{\partial \dot{\xi}} - \langle \kappa, \xi \rangle \frac{\partial \lambda}{\partial \dot{\xi}} \right]_{t=b} = 0\tag{4.28}$$

$$db: \quad \left[C - \left\langle \frac{\partial C}{\partial \Omega}, \dot{\Omega} \right\rangle - \left\langle \kappa, -\dot{\mathbb{I}}\Omega + \text{ad}_\Omega^* \mathbb{I}\Omega + \lambda \xi \right\rangle \right]_{t=b} = 0\tag{4.29}$$

where λ is given by (4.8) and the final right boundary condition (4.29) is only needed if the final time b is free. Equations (4.22), (4.23), and (4.24) together with the left boundary conditions (4.25)-(4.26) and the right boundary conditions (4.27)-(4.28) and, if needed, (4.29), constitute the controlled equations of motion for Suslov's problem using change in the nonholonomic constraint direction as the control.

4.B Suslov's Optimal Control Problem for Rigid Body Motion

4.B.1 Derivation of Suslov's Uncontrolled Equations of Motion

Having discussed the formulation of Suslov's problem in the general case for an arbitrary group, let us now turn our attention to the case of the particular Lie group $G = SO(3)$, which represents Suslov's problem in its original formulation and where the unreduced Lagrangian is $L = L(\Lambda, \dot{\Lambda})$, with $\Lambda \in SO(3)$. Suslov's problem studies the behavior of the body angular velocity $\boldsymbol{\Omega} \equiv \left[\Lambda^{-1} \dot{\Lambda} \right]^\vee \in \mathbb{R}^3$ subject to the nonholonomic constraint

$$\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0 \quad (4.30)$$

for some prescribed, possibly time-varying vector $\boldsymbol{\xi} \in \mathbb{R}^3$ *expressed in the body frame*. Physically, such a system corresponds to a rigid body rotating about a fixed point, with the rotation required to be normal to the prescribed vector $\boldsymbol{\xi}(t) \in \mathbb{R}^3$. The fact that the vector $\boldsymbol{\xi}$ identifying the nonholonomic constraint is defined in the body frame makes direct physical interpretation and realization of Suslov's problem somewhat challenging. Still, Suslov's problem is perhaps one of the simplest and, at the same time, most insightful and pedagogical problems in the field of nonholonomic mechanics, and has attracted considerable attention in the literature. The original formulation of this problem is due to Suslov in 1902 [7] (still only available in Russian), where he assumed that $\boldsymbol{\xi}$ was constant. This chapter considers the more general case where $\boldsymbol{\xi}$ varies with time. In order to match the standard state-space notation in control theory, the state-space control is assumed to be $\mathbf{u} = \dot{\boldsymbol{\xi}}$. We shall also note that the control-theoretical treatment of unconstrained rigid body motion from the geometric point of view is discussed in detail in [44], Chapters 19 (for general compact Lie groups) and 22.

For conciseness, the time-dependence of $\boldsymbol{\xi}$ is often suppressed in what follows. We shall note that there is a more general formulation of Suslov's problem when $G = SO(3)$ which includes a potential energy in the Lagrangian,

$$\ell(\boldsymbol{\Omega}, \boldsymbol{\Gamma}) = \frac{1}{2} \langle \mathbb{I} \boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle - U(\boldsymbol{\Gamma}), \quad \boldsymbol{\Gamma} = \Lambda^{-1} \mathbf{e}_3. \quad (4.31)$$

Depending on the type of potential energy, there are up to 3 additional integrals of motion. For a review of Suslov's problem and a summary of results in this area, the reader is referred to an article by Kozlov [11].

Let us choose a body frame coordinate system with an orthonormal basis $(\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3)$ in which the rigid body's inertia matrix \mathbb{I} is diagonal (i.e. $\mathbb{I} = \text{diag}(\mathbb{I}_1, \mathbb{I}_2, \mathbb{I}_3)$) and suppose henceforth that all body frame tensors are expressed with respect to this particular choice of coordinate system. Let $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ denote the orthonormal basis for the spatial frame coordinate system and denote the transformation from the body to spatial frame coordinate systems by the rotation matrix $\Lambda(t) \in SO(3)$. The rigid body's symmetry-reduced Lagrangian is its kinetic energy: $l = \frac{1}{2} \langle \mathbb{I} \boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle$. The action integral is

$$S(\boldsymbol{\Omega}) = \int_a^b l dt = \int_a^b \frac{1}{2} \langle \mathbb{I} \boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle dt. \quad (4.32)$$

The variation of the action integral with respect to all variations Σ such that $\Sigma(a) = \Sigma(b) = \mathbf{0}$ is

$$\begin{aligned} \delta S(\Omega) &= \delta \int_a^b l dt = \int_a^b \delta l dt = \int_a^b \langle \mathbb{I}\Omega, \delta\Omega \rangle dt = \int_a^b \langle \mathbb{I}\Omega, \dot{\Sigma} + \Omega \times \Sigma \rangle dt \\ &= - \int_a^b \left\langle \left(\frac{d}{dt} + \Omega \times \right) \mathbb{I}\Omega, \Sigma \right\rangle dt + \langle \mathbb{I}\Omega, \Sigma \rangle \Big|_a^b \\ &= - \int_a^b \left\langle \left(\frac{d}{dt} + \Omega \times \right) \mathbb{I}\Omega, \Sigma \right\rangle dt, \end{aligned} \quad (4.33)$$

where $\Sigma \equiv [\Lambda^{-1} \delta\Lambda]^\vee \in \mathfrak{so}(3)$, by using $\delta\Omega = \dot{\Sigma} + \Omega \times \Sigma$, integrating by parts, and applying the vanishing endpoint conditions $\Sigma(a) = [\Lambda^{-1}(a) \delta\Lambda(a)]^\vee = \mathbf{0}$ and $\Sigma(b) = [\Lambda^{-1}(b) \delta\Lambda(b)]^\vee = \mathbf{0}$. Applying Lagrange-d'Alembert's principle to the nonholonomic constraint (4.30) yields the constraint on variations required to derive the equations of motion:

$$\langle [\Lambda^{-1} \delta\Lambda]^\vee, \xi \rangle = \langle \Sigma, \xi \rangle = 0.$$

The principle of stationary action states that Ω must satisfy the ordinary differential equations arising from $\delta S(\Omega) = 0$ for all variations Σ such that $\Sigma(a) = \Sigma(b) = \mathbf{0}$ and $\langle \Sigma, \xi \rangle = 0$, in addition to satisfying the constraint $\langle \Omega, \xi \rangle = 0$. The first part of this statement implies that Ω must satisfy

$$\begin{aligned} \delta S(\Omega) + \int_a^b \lambda \langle \Sigma, \xi \rangle dt &= \int_a^b - \left\langle \left(\frac{d}{dt} + \Omega \times \right) \mathbb{I}\Omega, \Sigma \right\rangle dt + \int_a^b \lambda \langle \Sigma, \xi \rangle dt \\ &= \int_a^b \left[- \left\langle \left(\frac{d}{dt} + \Omega \times \right) \mathbb{I}\Omega, \Sigma \right\rangle + \lambda \langle \Sigma, \xi \rangle \right] dt \\ &= \int_a^b \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \mathbb{I}\Omega + \lambda \xi, \Sigma \right\rangle dt \\ &= 0 \end{aligned} \quad (4.34)$$

for all variations Σ such that $\Sigma(a) = \Sigma(b) = \mathbf{0}$ and for some time-varying Lagrange multiplier λ . Consequently, Ω must satisfy the system of ordinary differential equations

$$- \left(\frac{d}{dt} + \Omega \times \right) \mathbb{I}\Omega + \lambda \xi = 0, \quad (4.35)$$

subject to

$$\langle \Omega, \xi \rangle = 0.$$

Equation (4.35) can be expressed as

$$\mathbb{I}\dot{\Omega} = (\mathbb{I}\Omega) \times \Omega + \lambda \xi. \quad (4.36)$$

The next step is to determine the Lagrange multiplier λ . Both sides of (4.36) are dotted with $\mathbb{I}^{-1}\xi$ to give

$$\langle \mathbb{I}\dot{\Omega}, \mathbb{I}^{-1}\xi \rangle = \langle (\mathbb{I}\Omega) \times \Omega, \mathbb{I}^{-1}\xi \rangle + \lambda \langle \xi, \mathbb{I}^{-1}\xi \rangle, \quad (4.37)$$

from which λ may easily be solved for:

$$\lambda = \frac{\langle \mathbb{I}\dot{\Omega}, \mathbb{I}^{-1}\xi \rangle - \langle (\mathbb{I}\Omega) \times \Omega, \mathbb{I}^{-1}\xi \rangle}{\langle \xi, \mathbb{I}^{-1}\xi \rangle}. \quad (4.38)$$

But wait. This is not quite the formula needed for the Lagrange multiplier, because the formula must somehow incorporate the constraint $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$. By using the product rule and the constraint $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$,

$$\langle \mathbb{I}\dot{\boldsymbol{\Omega}}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle = \langle \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi} \rangle = \frac{d}{dt} \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle - \langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle = -\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle. \quad (4.39)$$

Note that this relation holds even if the constraint were $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = c$, for any constant c . Using this relation, the equation for λ can now be re-written as

$$\lambda = -\frac{\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}, \quad (4.40)$$

thereby incorporating the constraint equation. In order to make λ well-defined in (4.40), note that it is implicitly assumed that $\boldsymbol{\xi} \neq 0$ (i.e. $\boldsymbol{\xi}(t) \neq 0 \quad \forall t$). As is easy to verify, equations (4.36) and (4.40) are a particular case of the equations of motion (4.6) and the Lagrange multiplier (4.9). Also, equations (4.36) and (4.40) generalize the well-known equations of motion for Suslov's problem [30] to the case of time-varying $\boldsymbol{\xi}(t)$. When this formula for λ is substituted into (4.36), the equations of motion become

$$\mathbb{I}\dot{\boldsymbol{\Omega}} = (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} - \frac{\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \boldsymbol{\xi}, \quad (4.41)$$

which is equivalent to

$$\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \left[\mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} = 0. \quad (4.42)$$

As noted during the construction of λ , these equations of motion only guarantee that $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = c$, for a possibly non-zero constant c . To ensure that $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$ for all time, $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$ must be satisfied as an initial condition at time $t = a$. For conciseness the expression appearing on the left hand side of (4.42) is denoted by \mathbf{q} , i.e.

$$\mathbf{q}(\boldsymbol{\Omega}, \boldsymbol{\xi}) := \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \left[\mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} = \mathbf{0}. \quad (4.43)$$

We would like to state several useful observations about the nature of the dynamics in the free Suslov's problem, i.e. the results that are valid for arbitrary $\boldsymbol{\xi}(t)$, before proceeding to the optimal control problem.

On the nature of constraint preservation Suppose that $\boldsymbol{\Omega}(t)$ is a solution to (4.43) (equivalently (4.36)), for a given $\boldsymbol{\xi}(t)$ with λ given by (4.40). We can rewrite the equation for the Lagrange multiplier as

$$\begin{aligned} \lambda &= -\frac{\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \\ &= -\frac{\frac{d}{dt} \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle - \langle \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \\ &= -\frac{1}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{d}{dt} \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle + \frac{\langle \mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}. \end{aligned} \quad (4.44)$$

On the other hand, multiplying both sides of (4.36) by $\mathbb{I}^{-1}\boldsymbol{\xi}$ and solving for λ gives

$$\lambda = \frac{\langle \mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle}. \quad (4.45)$$

Thus, from (4.44) and (4.45) it follows that the equations of motion (4.43) with λ given by (4.40) lead to $\frac{d}{dt} \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$, so that $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = c$, a constant that is not necessarily equal to 0. In other words, the equations (4.43), (4.40) need an additional condition determining the value of $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$. Therefore, a solution $(\boldsymbol{\Omega}, \boldsymbol{\xi})$ to Suslov's problem requires that $\mathbf{q}(\boldsymbol{\Omega}, \boldsymbol{\xi}) = \mathbf{0}$ and $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$, where $t = a$ is the initial time.

On the invariance of solutions with respect to scaling of $\boldsymbol{\xi}$ In the classical formulation of Suslov's problem, it is usually assumed that $|\boldsymbol{\xi}| = 1$. When $\boldsymbol{\xi}(t)$ is allowed to change, the normalization of $\boldsymbol{\xi}$ becomes an issue that needs to be clarified. Indeed, suppose that $\boldsymbol{\Omega}(t)$ is a solution to (4.43) for a given $\boldsymbol{\xi}(t)$, so that $\mathbf{q}(\boldsymbol{\Omega}, \boldsymbol{\xi}) = \mathbf{0}$ and further assume that $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$. Next, consider a smooth, scalar-valued function $\pi(t)$ with $\pi(t) \neq 0$ on the interval $t \in [a, b]$, and consider the pair $(\boldsymbol{\Omega}, \pi\boldsymbol{\xi})$. Then

$$\begin{aligned} \mathbf{q}(\boldsymbol{\Omega}, \pi\boldsymbol{\xi}) &= \langle \pi\boldsymbol{\xi}, \mathbb{I}^{-1}(\pi\boldsymbol{\xi}) \rangle \left[\mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\langle \boldsymbol{\Omega}, (\pi\dot{\boldsymbol{\xi}}) \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}(\pi\boldsymbol{\xi}) \rangle \right] \pi\boldsymbol{\xi} \\ &= \pi^2 \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \left[\mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\langle \boldsymbol{\Omega}, \dot{\pi}\boldsymbol{\xi} + \pi\dot{\boldsymbol{\xi}} \rangle + \pi \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \right] \pi\boldsymbol{\xi} \\ &= \pi^2 \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \left[\mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\dot{\pi} \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle + \pi \langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \pi \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \right] \pi\boldsymbol{\xi} \\ &= \pi^2 \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \left[\mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\pi \langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \pi \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \right] \pi\boldsymbol{\xi} \\ &= \pi^2 \mathbf{q}(\boldsymbol{\Omega}, \boldsymbol{\xi}) = \mathbf{0}. \end{aligned} \quad (4.46)$$

Hence, a solution $\boldsymbol{\Omega}(t)$ to (4.43) with $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = c = 0$ does not depend on the magnitude of $\boldsymbol{\xi}(t)$. As it turns out, this creates a degeneracy in the optimal control problem that has to be treated with care.

Energy conservation Multiplying both sides of (4.36) by $\boldsymbol{\Omega}$, gives the time derivative of kinetic energy:

$$\dot{T}(t) = \frac{d}{dt} \left\{ \frac{1}{2} \langle \mathbb{I}\boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle \right\} = \langle \mathbb{I}\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}} \rangle = \lambda \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = \lambda c, \quad (4.47)$$

where we have denoted $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = c = \text{const.}$ Thus, if $c = 0$ (as is the case for Suslov's problem), kinetic energy is conserved:

$$T(t) = \frac{1}{2} \langle \mathbb{I}\boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle = \frac{1}{2} \sum_{i=1}^3 \mathbb{I}_i \Omega_i^2 = e_S, \quad (4.48)$$

for some positive constant e_S , and $\boldsymbol{\Omega}$ lies on the surface of an ellipsoid which we will denote by E . The constant kinetic energy ellipsoid determined by the rigid body's inertia matrix \mathbb{I} and initial body angular velocity $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a$ on which $\boldsymbol{\Omega}$ lies is denoted by

$$E = E(\mathbb{I}, \boldsymbol{\Omega}_a) = \{ \mathbf{v} \in \mathbb{R}^3 : \langle \mathbf{v}, \mathbb{I}\mathbf{v} \rangle = \langle \boldsymbol{\Omega}_a, \mathbb{I}\boldsymbol{\Omega}_a \rangle \}. \quad (4.49)$$

Integrating (4.47) with respect to time from a to b gives the change in kinetic energy:

$$T(b) - T(a) = c \int_a^b \lambda dt. \quad (4.50)$$

Thus, $\Omega(a)$ and $\Omega(b)$ lie on the surface of the same ellipsoid iff $c = 0$ or $\int_a^b \lambda dt = 0$.

If $c = 0$, as is the case for Suslov's problem, the conservation of kinetic energy holds for all choices of ξ , constant or time-dependent. We shall note that if the vector ξ is constant in time, and is an eigenvector of the inertia matrix \mathbb{I} , then there is an additional integral $\frac{1}{2} \langle \mathbb{I}\Omega, \mathbb{I}\Omega \rangle$. However, for $\xi(t)$ varying in time, which is the case studied here, such an integral does not apply.

4.B.2 Controllability and Accessibility of Suslov's Uncontrolled Equations of Motion

We shall now turn our attention to the problem of controlling Suslov's problem by changing the vector $\xi(t)$ in time. Before posing the optimal control problem, let us first consider the general question of controllability and accessibility using the Lie group approach to controllability as derived in [45], [46], and [47]. Since for the constraint $\langle \Omega, \xi \rangle = 0$ all trajectories must lie on the energy ellipsoid (4.49), both the initial and final point of the trajectory must lie on the ellipsoid corresponding to the same energy. We shall therefore assume that the initial and final points, as well as the trajectory itself, lie on the ellipsoid (4.49). Before we proceed, let us remind the reader of the relevant definitions and theorems concerning controllability and accessibility, following [30].

Definition 4.2. *An affine nonlinear control system is a differential equation having the form*

$$\dot{x} = f(x) + \sum_{i=1}^k g_i(x)u_i, \quad (4.51)$$

where M is a smooth n -dimensional manifold, $x \in M$, $u = (u_1, \dots, u_k)$ is a time-dependent, vector-valued map from \mathbb{R} to a constraint set $\Phi \subset \mathbb{R}^k$, and f and g_i , $i = 1, \dots, k$, are smooth vector fields on M . The manifold M is said to be the state-space of the system, u is said to be the control, f is said to be the drift vector field, and g_i , $i = 1, \dots, k$, are said to be the control vector fields. u is assumed to be piecewise smooth or piecewise analytic, and such a u is said to be admissible. If $f \equiv 0$, the system (4.51) is said to be driftless; otherwise, the system (4.51) is said to have drift.

Definition 4.3. *Let a be a fixed initial time. The system (4.51) is said to be controllable if for any pair of states $x_a, x_b \in M$ there exists a final time $b \geq a$ and an admissible control u defined on the time interval $[a, b]$ such that there is a trajectory of (4.51) with $x(a) = x_a$ and $x(b) = x_b$.*

Definition 4.4. *Given $x_a \in M$ and a time $t \geq a$, $R(x_a, t)$ is defined to be the set of all $y \in M$ for which there exists an admissible control u defined on the time interval $[a, t]$ such that there is a trajectory of (4.51) with $x(a) = x_a$ and $x(t) = y$. The reachable set from x_a at time $b \geq a$ is defined to be*

$$R_b(x_a) = \bigcup_{a \leq t \leq b} R(x_a, t). \quad (4.52)$$

Definition 4.5. The accessibility algebra \mathcal{C} of the system (4.51) is the smallest Lie algebra of vector fields on M that contains the vector fields f and g_i , $i = 1, \dots, k$; that is, $\mathcal{C} = \text{Lie}\{\mathbf{f}, \mathbf{g}_1, \dots, \mathbf{g}_k\}$ is the span of all possible Lie brackets of f and g_i , $i = 1, \dots, k$.

Definition 4.6. The accessibility distribution C of the system (4.51) is the distribution generated by the vector fields in \mathcal{C} ; that is, given $x_a \in M$, $C(x_a) = \text{Lie}_{x_a}\{\mathbf{f}, \mathbf{g}_1, \dots, \mathbf{g}_k\}$ is the span of the vector fields X in \mathcal{C} at x_a .

Definition 4.7. The system (4.51) is said to be accessible from $x_a \in M$ if for every $b > a$, $R_b(x_a)$ contains a nonempty open set.

Theorem 4.8. If $\dim C(x_a) = n$ for some $x_a \in M$, then the system (4.51) is accessible from x_a .

Theorem 4.9. Suppose the system (4.51) is analytic. If $\dim C(x_a) = n \quad \forall x_a \in M$ and $f = 0$, then the system (4.51) is controllable.

To apply the theory of controllability and accessibility to Suslov's problem, we first need to rewrite the equations of motion for Suslov's problem in the "affine nonlinear control" form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \sum_{i=1}^3 \mathbf{g}_i(\mathbf{x})u_i, \quad (4.53)$$

where \mathbf{x} is the state variable and u_i are the controls. We denote the state of the system by $\mathbf{x} \equiv \begin{bmatrix} \Omega \\ \xi \end{bmatrix}$ and the control by $\mathbf{u} \equiv \dot{\xi}$. Thus, the individual components of the state and control are $x_1 = \Omega_1$, $x_2 = \Omega_2$, $x_3 = \Omega_3$, $x_4 = \xi_1$, $x_5 = \xi_2$, $x_6 = \xi_3$, $u_1 = \dot{\xi}_1$, $u_2 = \dot{\xi}_2$, and $u_3 = \dot{\xi}_3$. The equations of motion (4.43) can be expressed as

$$\dot{\Omega} = \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1}\xi \rangle} \left\{ \langle \xi, \mathbb{I}^{-1}\xi \rangle (\mathbb{I}\Omega) \times \Omega - \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I}\Omega) \times \Omega, \mathbb{I}^{-1}\xi \rangle \right] \xi \right\}, \quad \dot{\xi} = \mathbf{u}. \quad (4.54)$$

To correlate (4.54) with (4.53), the functions \mathbf{f} and \mathbf{g} in (4.53) are defined as

$$\mathbf{f}(\mathbf{x}) \equiv \begin{bmatrix} \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1}\xi \rangle} \left\{ \langle \xi, \mathbb{I}^{-1}\xi \rangle (\mathbb{I}\Omega) \times \Omega - \langle (\mathbb{I}\Omega) \times \Omega, \mathbb{I}^{-1}\xi \rangle \xi \right\} \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (4.55)$$

and

$$\mathbf{g}_i(\mathbf{x}) \equiv \begin{bmatrix} -\frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1}\xi \rangle} \Omega_i \xi \\ \mathbf{e}_i \end{bmatrix} \quad \text{for } 1 \leq i \leq 3. \quad (4.56)$$

Here, $\mathbf{f}(\mathbf{x})$ is the drift vector field and $\mathbf{g}_i(\mathbf{x})$, $1 \leq i \leq 3$, are the control vector fields; $\mathbf{0}_{3 \times 1} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ denotes the 3×1 column vector of zeros and \mathbf{e}_i , $i = 1, 2, 3$, denote the standard orthonormal basis vectors for \mathbb{R}^3 . An alternative way to express each control vector field \mathbf{g}_i , $1 \leq i \leq 3$, is through the differential-geometric notation

$$\mathbf{g}_i = \frac{-\Omega_i \xi_m}{d_m \langle \xi, \mathbb{I}^{-1}\xi \rangle} \frac{\partial}{\partial \Omega_m} + \frac{\partial}{\partial \xi_i}. \quad (4.57)$$

As noted in the previous section, the first three components, Ω , of the state \mathbf{x} solving (4.53) must lie on the

ellipsoid E given in (4.49), under the assumption that

$$\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0 \quad (4.58)$$

for some time a . As shown in the previous section, (4.58) implies that a solution of (4.53) satisfies $\langle \boldsymbol{\Omega}(t), \boldsymbol{\xi}(t) \rangle = 0$ for all t . Also, it is assumed that $\boldsymbol{\xi} \neq 0$ (i.e. $\boldsymbol{\xi}(t) \neq 0 \quad \forall t$). Hence, the state-space manifold is $M = \{\mathbf{x} \in \mathbb{R}^6 \mid \frac{1}{2} \langle \mathbb{I}\boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle = e_S, \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0, \boldsymbol{\xi} \neq 0\}$. Let $K = \mathbb{R}^6 \setminus \mathbf{0}$, a 6-dimensional submanifold of \mathbb{R}^6 . Note that $M = \Phi^{-1}(\mathbf{0}_{2 \times 1})$, where $\Phi : K \rightarrow \mathbb{R}^2$ is defined by

$$\Phi(\mathbf{x}) = \begin{bmatrix} \frac{1}{2} \langle \mathbb{I}\boldsymbol{\Omega}, \boldsymbol{\Omega} \rangle - e_S \\ \langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle \end{bmatrix}. \quad (4.59)$$

The derivative of Φ at $\mathbf{x} \in K$, $(\Phi_*)_{\mathbf{x}} : T_{\mathbf{x}}K \rightarrow T_{\Phi(\mathbf{x})}\mathbb{R}^2$, is

$$(\Phi_*)_{\mathbf{x}} = \begin{bmatrix} d_1\Omega_1 & d_2\Omega_2 & d_3\Omega_3 & 0 & 0 & 0 \\ \xi_1 & \xi_2 & \xi_3 & \Omega_1 & \Omega_2 & \Omega_3 \end{bmatrix} = \begin{bmatrix} \mathbb{I}\boldsymbol{\Omega} & \boldsymbol{\xi} \\ \mathbf{0}_{3 \times 1} & \boldsymbol{\Omega} \end{bmatrix}^T. \quad (4.60)$$

Since $(\Phi_*)_{\mathbf{x}}$ has rank 2 for each $\mathbf{x} \in K$, Φ is by definition a submersion and $M = \Phi^{-1}(\mathbf{0}_{2 \times 1})$ is a closed embedded submanifold of K of dimension 4 by Corollary 8.9 of [48]. Being an embedded submanifold of K , M is also an immersed submanifold of K [48].

The tangent space to M at $\mathbf{x} \in M$ is

$$T_{\mathbf{x}}M = \{\mathbf{v} \in T_{\mathbf{x}}K = \mathbb{R}^6 \mid (\Phi_*)_{\mathbf{x}}(\mathbf{v}) = \mathbf{0}_{2 \times 1}\}. \quad (4.61)$$

Using (4.55), (4.56), and (4.60), it is easy to check that $(\Phi_*)_{\mathbf{x}}(\mathbf{f}(\mathbf{x})) = \mathbf{0}_{2 \times 1}$ and $(\Phi_*)_{\mathbf{x}}(\mathbf{g}_i(\mathbf{x})) = \mathbf{0}_{2 \times 1}$ for $1 \leq i \leq 3$. Hence, $\mathbf{f}(\mathbf{x}) \in T_{\mathbf{x}}M$ and $\mathbf{g}_i(\mathbf{x}) \in T_{\mathbf{x}}M$ for $1 \leq i \leq 3$ by Lemma 8.15 of [48]. So $\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2$, and \mathbf{g}_3 are smooth vector fields on K which are also tangent to M . Since M is an immersed submanifold of K , $[\mathbf{X}, \mathbf{Y}]$ is tangent to M if \mathbf{X} and \mathbf{Y} are smooth vector fields on K that are tangent to M , by Corollary 8.28 of [48]. Hence, $\text{Lie}_{\mathbf{x}}\{\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\} \subset T_{\mathbf{x}}M$ and therefore $\text{rank Lie}_{\mathbf{x}}\{\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\} \leq \dim T_{\mathbf{x}}M = 4$.

For $1 \leq i, j \leq 3$ and $i \neq j$, the Lie bracket of the control vector field \mathbf{g}_i with the control vector field \mathbf{g}_j is computed as

$$\begin{aligned} [\mathbf{g}_i, \mathbf{g}_j] &= \left[\frac{-\Omega_i \xi_m}{d_m \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{\partial}{\partial \Omega_m} + \frac{\partial}{\partial \xi_i}, \frac{-\Omega_j \xi_l}{d_l \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{\partial}{\partial \Omega_l} + \frac{\partial}{\partial \xi_j} \right] \\ &= \frac{\Omega_i \xi_m \xi_l \delta_{mj}}{d_m d_l \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle^2} \frac{\partial}{\partial \Omega_l} - \frac{\Omega_j}{d_l} \left\{ \frac{\partial}{\partial \xi_i} \left(\frac{\xi_l}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \right) \right\} \frac{\partial}{\partial \Omega_l} \\ &\quad - \frac{\Omega_j \xi_l \xi_m \delta_{il}}{d_l d_m \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle^2} \frac{\partial}{\partial \Omega_m} + \frac{\Omega_i}{d_m} \left\{ \frac{\partial}{\partial \xi_j} \left(\frac{\xi_m}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \right) \right\} \frac{\partial}{\partial \Omega_m} \\ &= \frac{\Omega_i \frac{\xi_j}{d_j} - \Omega_j \frac{\xi_i}{d_i}}{d_l \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle^2} \xi_l \frac{\partial}{\partial \Omega_l} - \frac{\Omega_j \delta_{il}}{d_l \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{\partial}{\partial \Omega_l} + \frac{2\Omega_j \xi_i \xi_l}{d_i d_l \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle^2} \frac{\partial}{\partial \Omega_l} \\ &\quad + \frac{\Omega_i \delta_{jm}}{d_m \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{\partial}{\partial \Omega_m} - \frac{2\Omega_i \xi_j \xi_m}{d_j d_m \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle^2} \frac{\partial}{\partial \Omega_m} \\ &= \frac{\Omega_j \frac{\xi_i}{d_i} - \Omega_i \frac{\xi_j}{d_j}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle^2} \xi_l \frac{\partial}{d_l \partial \Omega_l} + \frac{\Omega_i}{d_j \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{\partial}{\partial \Omega_j} - \frac{\Omega_j}{d_i \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle} \frac{\partial}{\partial \Omega_i}, \end{aligned} \quad (4.62)$$

recalling that $\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle = \sum_{i=1}^3 \frac{\xi_i^2}{d_i}$.

Next, to prove controllability and compute the appropriate prolongation, consider the 6×6 matrix comprised of the columns of the vector fields $\mathbf{g}_i(\mathbf{x})$ and their commutators $[\mathbf{g}_i(\mathbf{x}), \mathbf{g}_j(\mathbf{x})]$, projected to the basis of the space $(\partial_{\boldsymbol{\Omega}}, \partial_{\boldsymbol{\xi}})$:

$$V = [\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3, [\mathbf{g}_1, \mathbf{g}_2], [\mathbf{g}_1, \mathbf{g}_3], [\mathbf{g}_2, \mathbf{g}_3]] = \begin{bmatrix} -\frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} \boldsymbol{\xi} \otimes \boldsymbol{\Omega}^\top & \frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} A \\ I_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}, \quad (4.63)$$

where we have defined

$$A = \begin{bmatrix} -\Omega_2 & -\Omega_3 & 0 \\ \Omega_1 & 0 & -\Omega_3 \\ 0 & \Omega_1 & \Omega_2 \end{bmatrix} + \frac{1}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} \boldsymbol{\xi} \otimes [(\boldsymbol{\Omega} \times \mathbb{I}^{-1} \boldsymbol{\xi})^\top D], \quad D = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}. \quad (4.64)$$

In (4.63), $I_{3 \times 3}$ denotes the 3×3 identity matrix and $\mathbf{0}_{3 \times 3}$ denotes the 3×3 zero matrix. Since $V \subset \text{Lie}_{\mathbf{x}} \{\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$, $\text{rank } V \leq \text{rank Lie}_{\mathbf{x}} \{\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\} \leq \dim T_{\mathbf{x}}M = 4$. It will be shown that $\text{rank } V = 4$, so that $\text{rank Lie}_{\mathbf{x}} \{\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\} = \dim T_{\mathbf{x}}M = 4$.

Since the bottom 3 rows of the first 3 columns of V are $I_{3 \times 3}$, the first 3 columns of V are linearly independent. Note that since \mathbb{I}^{-1} is a diagonal matrix with positive diagonal entries, $\text{rank} \frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} A = \text{rank } A$. If $\text{rank} \frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} A = \text{rank } A > 0$, each of the last 3 columns of V , if non-zero, is linearly independent of the first 3 columns of V since the bottom 3 rows of the first 3 columns are $I_{3 \times 3}$ and the bottom 3 rows of the last 3 columns are $\mathbf{0}_{3 \times 3}$. Hence, $\text{rank } V = 3 + \text{rank } A$. Since $\text{rank } V \leq 4$, $\text{rank } A$ is 0 or 1.

The first matrix in the sum composing A in (4.64) has rank 2, since $\boldsymbol{\Omega} \neq \mathbf{0}$ (i.e. at least one component of $\boldsymbol{\Omega}$ is non-zero). The 3 columns of the first matrix in (4.64) are each orthogonal to $\boldsymbol{\Omega}$ and have rank 2; hence, the columns of the first matrix in (4.64) span the 2-dimensional plane in \mathbb{R}^3 orthogonal to $\boldsymbol{\Omega}$. Since $\langle \boldsymbol{\Omega}, \boldsymbol{\xi} \rangle = 0$, $\boldsymbol{\xi}$ lies in the 2-dimensional plane orthogonal to $\boldsymbol{\Omega}$ and so lies in the span of the columns of the first matrix. Thus, $\boldsymbol{\xi}$ and $\boldsymbol{\Omega} \times \boldsymbol{\xi}$ is an orthogonal basis for the plane in \mathbb{R}^3 orthogonal to $\boldsymbol{\Omega}$. Since the columns of the first matrix span this plane, at least one column, say the j^{th} ($1 \leq j \leq 3$), has a non-zero component parallel to $\boldsymbol{\Omega} \times \boldsymbol{\xi}$. The second matrix in the sum composing A in (4.64) consists of 3 column vectors, each of which is a scalar multiple of $\boldsymbol{\xi}$. Hence, the j^{th} column in A has a non-zero component parallel to $\boldsymbol{\Omega} \times \boldsymbol{\xi}$. Thus, A has rank 1, V has rank 4, and $\text{rank Lie}_{\mathbf{x}} \{\mathbf{f}, \mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\} = \dim T_{\mathbf{x}}M = 4$. By Theorems 4.8 and 4.9, this implies that (4.53) is controllable or accessible, depending on whether \mathbf{f} is non-zero. Thus, we have proved

Theorem 4.10 (On the controllability and accessibility of Suslov's problem). *Suppose we have Suslov's problem $\mathbf{q}(\boldsymbol{\Omega}, \boldsymbol{\xi}) = \mathbf{0}$ with the control variable $\dot{\boldsymbol{\xi}}(t)$. Then,*

1. *If $\mathbb{I} = cI_{3 \times 3}$ for a positive constant c , then $\boldsymbol{\Omega}$ lies on a sphere of radius c , $\mathbf{f} = \mathbf{0}$ for all points in M , and (4.53) is driftless and controllable.*
2. *If $\mathbb{I} \neq cI_{3 \times 3}$ for all positive constants c (i.e. at least two of the diagonal entries of \mathbb{I} are unequal), then $\boldsymbol{\Omega}$ lies on a non-spherical ellipsoid, $\mathbf{f} \neq \mathbf{0}$ at most points in M , and (4.53) has drift and is accessible.*

4.B.3 Suslov's Optimal Control Problem

Let us now turn our attention to the optimal control of Suslov's problem by varying the direction $\boldsymbol{\xi}(t)$. Suppose it is desired to maneuver Suslov's rigid body from a prescribed initial body angular velocity $\boldsymbol{\Omega}_a \in E$ at a prescribed initial time $t = a$ to another prescribed final body angular velocity $\boldsymbol{\Omega}_b \in E$ at a fixed or free final time $t = b$, where $b \geq a$, subject to minimizing some time-dependent integrand cost function C over the duration of the maneuver (such as minimizing the energy of the control vector $\boldsymbol{\xi}$ or minimizing the duration $b - a$ of the maneuver). Note that since a solution to Suslov's problem conserves kinetic energy, it is always assumed that $\boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b \in E$. Thus a time-varying control vector $\boldsymbol{\xi}$ and final time b are sought that generate a time-varying body angular velocity $\boldsymbol{\Omega}$, such that $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E$, $\boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b \in E$, $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$, the uncontrolled equations of motion $\mathbf{q} = \mathbf{0}$ are satisfied for $a \leq t \leq b$, and $\int_a^b C dt$ is minimized.

The natural way to formulate this optimal control problem is:

$$\min_{\boldsymbol{\xi}, b} \int_a^b C dt \quad \text{s.t.} \quad \begin{cases} \mathbf{q} = 0, \\ \boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E, \\ \langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0, \\ \boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b \in E. \end{cases} \quad (4.65)$$

The collection of constraints in (4.65) is actually over-determined. To see this, recall that a solution to $\mathbf{q} = 0$, $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a$, and $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$ sits on the constant kinetic energy ellipsoid E . If $(\boldsymbol{\Omega}, \boldsymbol{\xi})$ satisfies $\mathbf{q} = \mathbf{0}$, $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E$, and $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$, then $\boldsymbol{\Omega}(b) \in E$, a 2-d manifold. Thus, only two rather than three parameters of $\boldsymbol{\Omega}(b)$ need to be prescribed. So the constraint $\boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b$ in (4.65) is overprescribed and can lead to singular Jacobians when trying to solve (4.65) numerically, especially via the indirect method. A numerically more stable formulation of the optimal control problem is:

$$\min_{\boldsymbol{\xi}, b} \int_a^b C dt \quad \text{s.t.} \quad \begin{cases} \mathbf{q} = 0, \\ \boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E, \\ \langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0, \\ \phi(\boldsymbol{\Omega}(b)) = \phi(\boldsymbol{\Omega}_b), \text{ where } \boldsymbol{\Omega}_b \in E, \end{cases} \quad (4.66)$$

and where $\phi : E \rightarrow \mathbb{R}^2$ is some parameterization of the 2-d manifold E . For example, ϕ might map a point on E expressed in Cartesian coordinates to its azimuth and elevation in spherical coordinates. Using the properties of the dynamics, the problem (4.65) can be simplified further to read

$$\min_{\boldsymbol{\xi}, b} \int_a^b C dt \quad \text{s.t.} \quad \begin{cases} \mathbf{q} = 0, \\ \boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E, \\ \boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b \in E, \end{cases} \quad (4.67)$$

which omits the constraint $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$. One can see that (4.65) and (4.67) are equivalent as follows. Suppose $(\boldsymbol{\Omega}, \boldsymbol{\xi})$ satisfies $\mathbf{q} = \mathbf{0}$, $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E$, and $\boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b \in E$. Since $\boldsymbol{\Omega}_a, \boldsymbol{\Omega}_b \in E$ have the same kinetic energy, i.e. $T(a) = \frac{1}{2} \langle \boldsymbol{\Omega}_a, \mathbb{I} \boldsymbol{\Omega}_a \rangle = \frac{1}{2} \langle \boldsymbol{\Omega}_b, \mathbb{I} \boldsymbol{\Omega}_b \rangle = T(b)$, equation (4.50) shows that $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$ or $\int_a^b \lambda dt = 0$. The latter possibility, $\int_a^b \lambda dt = 0$, represents an additional constraint and thus is unlikely to occur. Thus, a solution of (4.67) should be expected to satisfy the omitted constraint $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$. Observe that the optimal control problem encapsulated by (4.67) ignores path inequality constraints such

as $\mathbf{D}(\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}, t) \leq \mathbf{0}$, where \mathbf{D} is a $r \times 1$ vector-valued function. Path inequality constraints can be incorporated in (4.67) as soft constraints through penalty functions in the integrand cost function C .

In what follows, we assume the following form of the integrand cost function C in (4.67):

$$C := C_{\alpha, \beta, \gamma, \eta, \delta} \equiv \frac{\alpha}{4} [|\boldsymbol{\xi}|^2 - 1]^2 + \frac{\beta}{2} |\dot{\boldsymbol{\xi}}|^2 + \frac{\gamma}{2} |\boldsymbol{\Omega} - \boldsymbol{\Omega}_d|^2 + \frac{\eta}{2} |\dot{\boldsymbol{\Omega}}|^2 + \delta, \quad (4.68)$$

where α , β , γ , η , and δ are non-negative constant scalars. Suslov's optimal control problem (4.67) becomes

$$\min_{\boldsymbol{\xi}, b} \int_a^b C_{\alpha, \beta, \gamma, \eta, \delta} dt \quad \text{s.t.} \quad \begin{cases} \mathbf{q} = 0, \\ \boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E, \\ \boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b \in E. \end{cases} \quad (4.69)$$

The first term in (4.68), $\frac{\alpha}{4} [|\boldsymbol{\xi}|^2 - 1]^2$, encourages the control vector $\boldsymbol{\xi}$ to have near unit magnitude. The second term in (4.68), $\frac{\beta}{2} |\dot{\boldsymbol{\xi}}|^2$, encourages the control vector $\boldsymbol{\xi}$ to follow a minimum energy trajectory. The first term in (4.68) is needed because the magnitude of $\boldsymbol{\xi}$ does not affect a solution of $\mathbf{q} = 0$, and in the absence of the first term in (4.68), the second term in (4.68) will try to shrink $\boldsymbol{\xi}$ to $\mathbf{0}$, causing numerical instability. An alternative to including the first term in (4.68) is to revise the formulation of the optimal control problem to include the path constraint $|\boldsymbol{\xi}| = 1$. The third term in (4.68), $\frac{\gamma}{2} |\boldsymbol{\Omega} - \boldsymbol{\Omega}_d|^2$, encourages the body angular velocity $\boldsymbol{\Omega}$ to follow a prescribed, time-varying trajectory $\boldsymbol{\Omega}_d$. The fourth term in (4.68), $\frac{\eta}{2} |\dot{\boldsymbol{\Omega}}|^2$, encourages the body angular velocity vector $\boldsymbol{\Omega}$ to follow a minimum energy trajectory. The final term in (4.68), δ , encourages a minimum time solution.

As in Subsection 4.B.2, using state-space terminology, the state is $\mathbf{x} \equiv \begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\xi} \end{bmatrix}$ and the control is $\mathbf{u} \equiv \dot{\boldsymbol{\xi}}$ for the optimal control problem (4.69). It is always assumed that the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is differentiable, and therefore continuous, or equivalently that $\boldsymbol{\xi}$ is twice differentiable.

4.B.4 Derivation of Suslov's Controlled Equations of Motion

Following the method of [43, 38], to construct a control vector $\boldsymbol{\xi}$ and final time b solving (4.69), the uncontrolled equations of motion are added to the integrand cost function through a time-varying Lagrange multiplier vector and the initial and final constraints are added using constant Lagrange multiplier vectors. A control vector $\boldsymbol{\xi}$ and final time b are sought that minimize the augmented performance index

$$S = \langle \boldsymbol{\rho}, \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \rangle + \langle \boldsymbol{\nu}, \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \rangle + \int_a^b [C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle] dt, \quad (4.70)$$

where $\boldsymbol{\rho}$ and $\boldsymbol{\nu}$ are constant Lagrange multiplier vectors enforcing the initial and final constraints $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a$ and $\boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b$ and $\boldsymbol{\kappa}$ is a time-varying Lagrange multiplier vector (i.e. the costate) enforcing the uncontrolled equations of motion defined by $\mathbf{q} = \mathbf{0}$ as given in (4.43).

The control vector $\boldsymbol{\xi}$ and final time b minimizing S are found by finding conditions for which the differential of S , dS , equals 0. For the purpose of computing the differential of S , it is assumed that the integrand

cost function is of the form $C(\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}, t)$. The differential of S is defined as the first-order change in S with respect to changes in $\boldsymbol{\kappa}$, $\boldsymbol{\Omega}$, $\boldsymbol{\xi}$, $\boldsymbol{\rho}$, $\boldsymbol{\nu}$, and b . After computing the differential of S , the goal of the following calculation is to isolate $\delta\boldsymbol{\kappa}$, $\delta\boldsymbol{\Omega}$, and $\delta\boldsymbol{\xi}$ in the integrand terms, to isolate $d\boldsymbol{\xi}(a) = \delta\boldsymbol{\xi}(a)$ and $d\boldsymbol{\rho}$ in the left boundary terms, and to isolate $d\boldsymbol{\Omega}(b)$, $d\boldsymbol{\xi}(b)$, $d\boldsymbol{\nu}$, and db in the right boundary terms. Once these differentials have been isolated, the controlled equations of motion may be obtained readily upon setting $dS = 0$. Utilizing the calculus of variations summarized in Section 3.A and because $\delta S = \delta_{\boldsymbol{\kappa}}S + \delta_{\boldsymbol{\xi}}S + \delta_{\boldsymbol{\Omega}}S$ and $\dot{S}(b) = \frac{\partial S}{\partial b}(b) = \left[\langle \boldsymbol{\nu}, \dot{\boldsymbol{\Omega}} \rangle + C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle \right]_{t=b}$ by the Fundamental Theorem of Calculus, the differential of S is computed as

$$\begin{aligned}
dS &= \delta S + \left\langle \left(\frac{\partial S}{\partial \boldsymbol{\rho}} \right)^\top, d\boldsymbol{\rho} \right\rangle + \left\langle \left(\frac{\partial S}{\partial \boldsymbol{\nu}} \right)^\top, d\boldsymbol{\nu} \right\rangle + \dot{S}(b)db \\
&= \delta_{\boldsymbol{\kappa}}S + \delta_{\boldsymbol{\xi}}S + \delta_{\boldsymbol{\Omega}}S + \langle \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a, d\boldsymbol{\rho} \rangle + \langle \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b, d\boldsymbol{\nu} \rangle + \left[\langle \boldsymbol{\nu}, \dot{\boldsymbol{\Omega}} \rangle + C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle \right]_{t=b} db \\
&= \int_a^b \langle \mathbf{q}, \delta\boldsymbol{\kappa} \rangle dt + \int_a^b \left[\left\langle \left(\frac{\partial C}{\partial \boldsymbol{\xi}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top, \delta\boldsymbol{\xi} \right\rangle + \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\xi}}\mathbf{q} \rangle \right] dt + \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\xi}} \right)^\top, \delta\boldsymbol{\xi} \right\rangle \Big|_a^b \\
&\quad + \langle \boldsymbol{\rho}, \delta\boldsymbol{\Omega}(a) \rangle + \langle \boldsymbol{\nu}, \delta\boldsymbol{\Omega}(b) \rangle + \int_a^b \left[\left\langle \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \delta\boldsymbol{\Omega} \right\rangle + \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\Omega}}\mathbf{q} \rangle \right] dt \\
&\quad + \left\langle \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \delta\boldsymbol{\Omega} \right\rangle \Big|_a^b + \langle \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a, d\boldsymbol{\rho} \rangle + \langle \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b, d\boldsymbol{\nu} \rangle + \left[\langle \boldsymbol{\nu}, \dot{\boldsymbol{\Omega}} \rangle + C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle \right]_{t=b} db.
\end{aligned} \tag{4.71}$$

Because $\delta\boldsymbol{\Omega}(a) = 0$ (since $\boldsymbol{\Omega}(a)$ is fixed to $\boldsymbol{\Omega}_a$) and $d\boldsymbol{\Omega}(b) = \delta\boldsymbol{\Omega}(b) + \dot{\boldsymbol{\Omega}}(b)db$, the differential of S computed in (4.71) simplifies to

$$\begin{aligned}
dS &= \int_a^b \langle \mathbf{q}, \delta\boldsymbol{\kappa} \rangle dt + \int_a^b \left[\left\langle \left(\frac{\partial C}{\partial \boldsymbol{\xi}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top, \delta\boldsymbol{\xi} \right\rangle + \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\xi}}\mathbf{q} \rangle \right] dt + \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\xi}} \right)^\top, \delta\boldsymbol{\xi} \right\rangle \Big|_a^b \\
&\quad + \int_a^b \left[\left\langle \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \delta\boldsymbol{\Omega} \right\rangle + \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\Omega}}\mathbf{q} \rangle \right] dt + \left\langle \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \delta\boldsymbol{\Omega} \right\rangle \Big|_a^b \\
&\quad + \langle \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a, d\boldsymbol{\rho} \rangle + \langle \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b, d\boldsymbol{\nu} \rangle + \langle \boldsymbol{\nu}, d\boldsymbol{\Omega}(b) \rangle + [C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle]_{t=b} db.
\end{aligned} \tag{4.72}$$

In order to isolate $\delta\boldsymbol{\Omega}$ and $\delta\boldsymbol{\xi}$ in (4.72), the integrals $\int_a^b \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\Omega}}\mathbf{q} \rangle dt$ and $\int_a^b \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\xi}}\mathbf{q} \rangle dt$ appearing in (4.72) will be computed. Since

$$\delta_{\boldsymbol{\Omega}}\mathbf{q} = \langle \boldsymbol{\xi}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \left[\mathbb{I}\delta\dot{\boldsymbol{\Omega}} - (\mathbb{I}\delta\boldsymbol{\Omega}) \times \boldsymbol{\Omega} - (\mathbb{I}\boldsymbol{\Omega}) \times \delta\boldsymbol{\Omega} \right] + \left[\langle \delta\boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\delta\boldsymbol{\Omega}) \times \boldsymbol{\Omega} + (\mathbb{I}\boldsymbol{\Omega}) \times \delta\boldsymbol{\Omega}, \mathbb{I}^{-1}\boldsymbol{\xi} \rangle \right] \boldsymbol{\xi}, \tag{4.73}$$

it follows that

$$\begin{aligned}
\langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\xi}} \mathbf{q} \rangle &= \left\langle \boldsymbol{\kappa}, \mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right\rangle \langle 2\mathbb{I}^{-1} \boldsymbol{\xi}, \delta \boldsymbol{\xi} \rangle + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \left[\left\langle \boldsymbol{\Omega}, \delta \dot{\boldsymbol{\xi}} \right\rangle + \langle \mathbb{I}^{-1} ((\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}), \delta \boldsymbol{\xi} \rangle \right] \\
&\quad + \left[\left\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \right\rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \langle \boldsymbol{\kappa}, \delta \boldsymbol{\xi} \rangle \\
&= \left\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \right\rangle \boldsymbol{\Omega}, \delta \dot{\boldsymbol{\xi}} \rangle \\
&\quad + \left\langle 2 \left\langle \boldsymbol{\kappa}, \mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right\rangle \mathbb{I}^{-1} \boldsymbol{\xi} + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}) \right. \\
&\quad \left. + \left[\left\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \right\rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa}, \delta \boldsymbol{\xi} \right\rangle.
\end{aligned} \tag{4.77}$$

Integrating (4.77) and using integration by parts to eliminate $\delta \dot{\boldsymbol{\xi}}$ yields

$$\begin{aligned}
\int_a^b \langle \boldsymbol{\kappa}, \delta_{\boldsymbol{\xi}} \mathbf{q} \rangle dt &= \int_a^b \left\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \right\rangle \boldsymbol{\Omega}, \delta \dot{\boldsymbol{\xi}} \rangle dt \\
&\quad + \int_a^b \left\langle 2 \left\langle \boldsymbol{\kappa}, \mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right\rangle \mathbb{I}^{-1} \boldsymbol{\xi} + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}) \right. \\
&\quad \left. + \left[\left\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \right\rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa}, \delta \boldsymbol{\xi} \right\rangle dt \\
&= - \int_a^b \left\langle \frac{d}{dt} (\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega}), \delta \boldsymbol{\xi} \right\rangle dt + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega}, \delta \boldsymbol{\xi} \rangle \Big|_a^b \\
&\quad + \int_a^b \left\langle 2 \left\langle \boldsymbol{\kappa}, \mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right\rangle \mathbb{I}^{-1} \boldsymbol{\xi} + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}) \right. \\
&\quad \left. + \left[\left\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \right\rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa}, \delta \boldsymbol{\xi} \right\rangle dt \\
&= \int_a^b \left\langle - \frac{d}{dt} (\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega}) + 2 \left\langle \boldsymbol{\kappa}, \mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right\rangle \mathbb{I}^{-1} \boldsymbol{\xi} \right. \\
&\quad \left. + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}) + \left[\left\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \right\rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa}, \delta \boldsymbol{\xi} \right\rangle dt \\
&\quad + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega}, \delta \boldsymbol{\xi} \rangle \Big|_a^b.
\end{aligned} \tag{4.78}$$

Using (4.75) and (4.78), the differential of S computed in (4.72) simplifies to

$$\begin{aligned}
dS &= \int_a^b \langle \mathbf{q}, \delta \boldsymbol{\kappa} \rangle dt + \int_a^b \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\xi}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top - \frac{d}{dt} (\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega}) + 2 \left\langle \boldsymbol{\kappa}, \mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right\rangle \mathbb{I}^{-1} \boldsymbol{\xi} \right. \\
&\quad \left. + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}) + \left[\left\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \right\rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa}, \delta \boldsymbol{\xi} \right\rangle dt \\
&\quad + \left\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \right\rangle \boldsymbol{\Omega} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top, \delta \boldsymbol{\xi} \right\rangle \Big|_a^b \\
&\quad + \int_a^b \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top - \frac{d}{dt} (\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa}) - \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle [\mathbb{I} (\boldsymbol{\Omega} \times \boldsymbol{\kappa}) + \boldsymbol{\kappa} \times (\mathbb{I} \boldsymbol{\Omega})] \right. \\
&\quad \left. + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \right\rangle \left[\dot{\boldsymbol{\xi}} + \mathbb{I} (\boldsymbol{\Omega} \times (\mathbb{I}^{-1} \boldsymbol{\xi})) + (\mathbb{I}^{-1} \boldsymbol{\xi}) \times (\mathbb{I} \boldsymbol{\Omega}) \right], \delta \boldsymbol{\Omega} \right\rangle dt + \left\langle \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \delta \boldsymbol{\Omega} \right\rangle \Big|_a^b \\
&\quad + \langle \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a, d\boldsymbol{\rho} \rangle + \langle \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b, d\boldsymbol{\nu} \rangle + \langle \boldsymbol{\nu}, d\boldsymbol{\Omega}(b) \rangle + [C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle]_{t=b} db.
\end{aligned} \tag{4.79}$$

Finally, because $\delta\Omega(a) = 0$ (since $\Omega(a)$ is fixed to Ω_a), $d\Omega(b) = \delta\Omega(b) + \dot{\Omega}(b)db$, and $d\xi(b) = \delta\xi(b) + \dot{\xi}(b)db$, the differential of S computed in (4.79) simplifies to

$$\begin{aligned}
dS = & \int_a^b \langle \mathbf{q}, \delta\boldsymbol{\kappa} \rangle dt + \int_a^b \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\xi}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top - \frac{d}{dt} \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega} + 2 \langle \boldsymbol{\kappa}, \mathbb{I}\dot{\boldsymbol{\Omega}} - (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \rangle \mathbb{I}^{-1} \boldsymbol{\xi} \right. \\
& + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}) + \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa}, \delta\boldsymbol{\xi} \left. \right\rangle dt \\
& + \left\langle \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top, d\boldsymbol{\xi} \right\rangle \Big|_{t=b} - \left\langle \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top, \delta\boldsymbol{\xi} \right\rangle \Big|_{t=a} \\
& + \int_a^b \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top - \frac{d}{dt} \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I}\boldsymbol{\kappa} - \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle [\mathbb{I}(\boldsymbol{\Omega} \times \boldsymbol{\kappa}) + \boldsymbol{\kappa} \times (\mathbb{I}\boldsymbol{\Omega})] \right. \\
& + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \left[\dot{\boldsymbol{\xi}} + \mathbb{I}(\boldsymbol{\Omega} \times (\mathbb{I}^{-1} \boldsymbol{\xi})) + (\mathbb{I}^{-1} \boldsymbol{\xi}) \times (\mathbb{I}\boldsymbol{\Omega}) \right], \delta\boldsymbol{\Omega} \left. \right\rangle dt \\
& + \langle \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a, d\rho \rangle + \langle \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b, d\nu \rangle + \left\langle \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I}\boldsymbol{\kappa} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top + \boldsymbol{\nu}, d\boldsymbol{\Omega} \right\rangle \Big|_{t=b} \\
& + \left[C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle - \left\langle \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I}\boldsymbol{\kappa} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \dot{\boldsymbol{\Omega}} \right\rangle - \left\langle \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top, \dot{\boldsymbol{\xi}} \right\rangle \right] \Big|_{t=b} db.
\end{aligned} \tag{4.80}$$

Demanding that $dS = 0$ in (4.80) yields the conditions that must be satisfied to find the control vector $\boldsymbol{\xi}$ and the final time b . The controlled equations of motion are

$$\begin{aligned}
\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I}\dot{\boldsymbol{\Omega}} = & \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \\
& - \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} \\
\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I}\dot{\boldsymbol{\kappa}} = & \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top \\
& - \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle [\mathbb{I}(\boldsymbol{\Omega} \times \boldsymbol{\kappa}) + \boldsymbol{\kappa} \times (\mathbb{I}\boldsymbol{\Omega})] \\
& + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \left[\dot{\boldsymbol{\xi}} + \mathbb{I}(\boldsymbol{\Omega} \times (\mathbb{I}^{-1} \boldsymbol{\xi})) + (\mathbb{I}^{-1} \boldsymbol{\xi}) \times (\mathbb{I}\boldsymbol{\Omega}) \right] \\
& - 2 \langle \dot{\boldsymbol{\xi}}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I}\boldsymbol{\kappa} \\
\left[\langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I} - 2\mathbb{I}^{-1} \boldsymbol{\xi} (\mathbb{I}\boldsymbol{\kappa})^\top \right] \dot{\boldsymbol{\Omega}} + \left(\frac{d}{dt} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top + \boldsymbol{\Omega} \boldsymbol{\xi}^\top \dot{\boldsymbol{\kappa}} = & \left(\frac{\partial C}{\partial \boldsymbol{\xi}} \right)^\top - 2 \langle \boldsymbol{\kappa}, (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega} \rangle \mathbb{I}^{-1} \boldsymbol{\xi} \\
& + \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \mathbb{I}^{-1} ((\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}) \\
& + \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I}\boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\kappa} - \langle \boldsymbol{\kappa}, \dot{\boldsymbol{\xi}} \rangle \boldsymbol{\Omega},
\end{aligned} \tag{4.81}$$

which simplify to

$$\begin{aligned}
\dot{\Omega} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \langle \xi, \mathbb{I}^{-1} \xi \rangle (\mathbb{I} \Omega) \times \Omega - \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \xi \right\} \\
\dot{\kappa} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \left(\frac{\partial C}{\partial \Omega} - \frac{d}{dt} \frac{\partial C}{\partial \dot{\Omega}} \right)^\top - \langle \xi, \mathbb{I}^{-1} \xi \rangle [\mathbb{I} (\Omega \times \kappa) + \kappa \times (\mathbb{I} \Omega)] \right. \\
&\quad \left. + \langle \kappa, \xi \rangle \left[\dot{\xi} + \mathbb{I} (\Omega \times (\mathbb{I}^{-1} \xi)) + (\mathbb{I}^{-1} \xi) \times (\mathbb{I} \Omega) \right] - 2 \langle \dot{\xi}, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa \right\} \\
\left(\frac{d}{dt} \frac{\partial C}{\partial \dot{\xi}} \right)^\top &= \left(\frac{\partial C}{\partial \xi} \right)^\top - 2 \langle \kappa, (\mathbb{I} \Omega) \times \Omega \rangle \mathbb{I}^{-1} \xi + \langle \kappa, \xi \rangle \mathbb{I}^{-1} ((\mathbb{I} \Omega) \times \Omega) \\
&\quad + \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \kappa - \langle \kappa, \dot{\xi} \rangle \Omega \\
&\quad - \left[\langle \kappa, \xi \rangle I - 2 \mathbb{I}^{-1} \xi (\mathbb{I} \kappa)^\top \right] \dot{\Omega} - \Omega \xi^\top \dot{\kappa},
\end{aligned} \tag{4.82}$$

for $a \leq t \leq b$, the left boundary conditions

$$\begin{aligned}
\Omega(a) - \Omega_a &= 0 \\
\left[\langle \kappa, \xi \rangle \Omega + \left(\frac{\partial C}{\partial \dot{\xi}} \right)^\top \right]_{t=a} &= 0,
\end{aligned} \tag{4.83}$$

and the right boundary conditions

$$\begin{aligned}
\Omega(b) - \Omega_b &= 0 \\
\left[\langle \kappa, \xi \rangle \Omega + \left(\frac{\partial C}{\partial \dot{\xi}} \right)^\top \right]_{t=b} &= 0 \\
\left[C + \langle \kappa, \mathbf{q} \rangle - \left\langle \langle \xi, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa + \left(\frac{\partial C}{\partial \dot{\Omega}} \right)^\top, \dot{\Omega} \right\rangle - \left\langle \langle \kappa, \xi \rangle \Omega + \left(\frac{\partial C}{\partial \dot{\xi}} \right)^\top, \dot{\xi} \right\rangle \right]_{t=b} &= 0.
\end{aligned} \tag{4.84}$$

Using the first equation in (4.82), which is equivalent to (4.43), and the second equation in (4.84), the third equation in (4.84), corresponding to free final time, can be simplified, so that the right boundary conditions simplify to

$$\begin{aligned}
\Omega(b) - \Omega_b &= 0 \\
\left[\langle \kappa, \xi \rangle \Omega + \left(\frac{\partial C}{\partial \dot{\xi}} \right)^\top \right]_{t=b} &= 0 \\
\left[C - \left\langle \langle \xi, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa + \left(\frac{\partial C}{\partial \dot{\Omega}} \right)^\top, \dot{\Omega} \right\rangle \right]_{t=b} &= 0.
\end{aligned} \tag{4.85}$$

Equations (4.82), (4.83), and (4.85) form an ODE TPBVP. Observe that the unknowns in this ODE TPBVP are κ , Ω , ξ , and b , while the constant Lagrange multiplier vectors ρ and ν are irrelevant.

This application of Pontryagin's minimum principle differs slightly from the classical treatment of optimal control theory reviewed in Chapter 3. Let us connect this derivation to that chapter. In the classical

application, the Hamiltonian involves 6 costates $\boldsymbol{\pi} \in \mathbb{R}^6$ and is given by

$$H = L(\boldsymbol{\Omega}, \boldsymbol{\xi}, \mathbf{u}, t) + \left\langle \boldsymbol{\pi}, \left[\frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} \left\{ \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} - \left[\langle \boldsymbol{\Omega}, \mathbf{u} \rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} \right\} \right] \right\rangle, \quad (4.86)$$

whereas in the derivation above, the Hamiltonian involves only 3 costates $\boldsymbol{\kappa} \in \mathbb{R}^3$ and is given by

$$H_r = C(\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}, t) - \left\langle \boldsymbol{\kappa}, \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} - \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} \right\rangle, \quad (4.87)$$

with $L(\boldsymbol{\Omega}, \boldsymbol{\xi}, \mathbf{u}, t) = C(\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}, t)$, since $\dot{\boldsymbol{\Omega}}$ is a function of $\boldsymbol{\Omega}$, $\boldsymbol{\xi}$, and $\dot{\boldsymbol{\xi}}$ and since $\mathbf{u} = \dot{\boldsymbol{\xi}}$.

Appendix B shows that the classical costates $\boldsymbol{\pi}$ can be obtained from the reduced costates $\boldsymbol{\kappa}$, derived here, via

$$\boldsymbol{\pi} = - \begin{bmatrix} \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top \\ \langle \boldsymbol{\kappa}, \boldsymbol{\xi} \rangle \boldsymbol{\Omega} + \left(\frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \right)^\top \end{bmatrix}. \quad (4.88)$$

Now consider the particular integrand cost function (4.68) corresponding to the optimal control problem (4.69). For this integrand cost function, the partial derivative of the Hamiltonian (4.86) with respect to the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is

$$\begin{aligned} H_{\mathbf{u}} = H_{\dot{\boldsymbol{\xi}}} &= \frac{\partial L}{\partial \mathbf{u}} + \boldsymbol{\pi}_d^\top \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \mathbf{u}} + \boldsymbol{\pi}_e^\top = \frac{\partial C_{\alpha, \beta, \gamma, \eta, \delta}}{\partial \boldsymbol{\Omega}} \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} + \frac{\partial C_{\alpha, \beta, \gamma, \eta, \delta}}{\partial \dot{\boldsymbol{\xi}}} + \boldsymbol{\pi}_d^\top \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} + \boldsymbol{\pi}_e^\top \\ &= \eta \dot{\boldsymbol{\Omega}}^\top \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} + \beta \dot{\boldsymbol{\xi}}^\top + \boldsymbol{\pi}_d^\top \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} + \boldsymbol{\pi}_e^\top, \end{aligned} \quad (4.89)$$

where we have defined for brevity

$$\boldsymbol{\pi}_d \equiv \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \end{bmatrix} \quad \text{and} \quad \boldsymbol{\pi}_e \equiv \begin{bmatrix} \pi_4 \\ \pi_5 \\ \pi_6 \end{bmatrix} \quad \text{and where} \quad \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} = \frac{\mathbb{I}^{-1} \boldsymbol{\xi} \boldsymbol{\Omega}^\top}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle}. \quad (4.90)$$

The second partial derivative of the Hamiltonian (4.86) with respect to the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is

$$H_{\mathbf{u}\mathbf{u}} = H_{\dot{\boldsymbol{\xi}}\dot{\boldsymbol{\xi}}} = \eta \left(\frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} \right)^\top \frac{\partial \dot{\boldsymbol{\Omega}}}{\partial \dot{\boldsymbol{\xi}}} + \beta I = \frac{\eta}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle^2} \left[\mathbb{I}^{-1} \boldsymbol{\xi} \boldsymbol{\Omega}^\top \right]^\top \left[\mathbb{I}^{-1} \boldsymbol{\xi} \boldsymbol{\Omega}^\top \right] + \beta I = \tilde{c} \boldsymbol{\Omega} \boldsymbol{\Omega}^\top + \beta I, \quad (4.91)$$

where $\tilde{c} = \frac{\eta}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle^2} \boldsymbol{\xi}^\top \mathbb{I}^{-2} \boldsymbol{\xi}$ is a nonnegative scalar. Recall that it is assumed that $\beta \geq 0$. If $\beta = 0$, then $H_{\mathbf{u}\mathbf{u}} = \tilde{c} \boldsymbol{\Omega} \boldsymbol{\Omega}^\top$ is singular since $\boldsymbol{\Omega} \boldsymbol{\Omega}^\top$ is a rank 1 matrix. Hence, if $H_{\mathbf{u}\mathbf{u}}$ is nonsingular, then $\beta > 0$. Now suppose that $\beta > 0$. Part of the Sherman-Morrison formula [49] says that given an invertible matrix $A \in \mathbb{R}^{n \times n}$ and $\mathbf{w}, \mathbf{v} \in \mathbb{R}^{n \times 1}$, $A + \mathbf{w} \mathbf{v}^\top$ is invertible if and only if $1 + \mathbf{v}^\top A^{-1} \mathbf{w} \neq 0$. Letting $A = \beta I$ and $\mathbf{w} = \mathbf{v} = \sqrt{\tilde{c}} \boldsymbol{\Omega}$, the Sherman-Morrison formula guarantees that $H_{\mathbf{u}\mathbf{u}}$ is nonsingular if $1 + \frac{\tilde{c}}{\beta} \boldsymbol{\Omega}^\top \boldsymbol{\Omega} \neq 0$. But $\frac{\tilde{c}}{\beta} \boldsymbol{\Omega}^\top \boldsymbol{\Omega} \geq 0$, so $1 + \frac{\tilde{c}}{\beta} \boldsymbol{\Omega}^\top \boldsymbol{\Omega} \geq 1$ and $H_{\mathbf{u}\mathbf{u}}$ is nonsingular. Therefore, $H_{\mathbf{u}\mathbf{u}}$ is nonsingular if and only if $\beta > 0$. Thus, the optimal control problem (4.69) is nonsingular if and only if $\beta > 0$. Since singular optimal control problems require careful analysis and solution methods, it is assumed for the remainder of this chapter, except in Subsection 4.C.1, that $\beta > 0$. As explained in the paragraph after (4.68), $\beta > 0$ requires that

$\alpha > 0$. So for the remainder of this chapter, except in Subsection 4.C.1, it is assumed that $\beta > 0$ and $\alpha > 0$ when considering the optimal control problem (4.69).

For the particular integrand cost function (4.68), with $\alpha > 0$, $\beta > 0$, $\gamma \geq 0$, $\eta \geq 0$, and $\delta \geq 0$, the controlled equations of motion (4.82) defined on $a \leq t \leq b$ become

$$\begin{aligned}
\dot{\Omega} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \langle \xi, \mathbb{I}^{-1} \xi \rangle (\mathbb{I} \Omega) \times \Omega - \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \xi \right\} \\
\dot{\kappa} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \gamma (\Omega - \Omega_d) - \eta \ddot{\Omega} - \langle \xi, \mathbb{I}^{-1} \xi \rangle [\mathbb{I} (\Omega \times \kappa) + \kappa \times (\mathbb{I} \Omega)] \right. \\
&\quad \left. + \langle \kappa, \xi \rangle \left[\dot{\xi} + \mathbb{I} (\Omega \times (\mathbb{I}^{-1} \xi)) + (\mathbb{I}^{-1} \xi) \times (\mathbb{I} \Omega) \right] - 2 \langle \dot{\xi}, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa \right\} \\
\ddot{\xi} &= \frac{1}{\beta} \left\{ \alpha (|\xi|^2 - 1) \xi - 2 \langle \kappa, (\mathbb{I} \Omega) \times \Omega \rangle \mathbb{I}^{-1} \xi + \langle \kappa, \xi \rangle \mathbb{I}^{-1} ((\mathbb{I} \Omega) \times \Omega) \right. \\
&\quad \left. + \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \kappa - \langle \kappa, \dot{\xi} \rangle \Omega \right. \\
&\quad \left. - \left[\langle \kappa, \xi \rangle I - 2 \mathbb{I}^{-1} \xi (\mathbb{I} \kappa)^T \right] \dot{\Omega} - \Omega \langle \xi, \dot{\kappa} \rangle \right\},
\end{aligned} \tag{4.92}$$

the left boundary conditions (4.83) become

$$\begin{aligned}
\Omega(a) - \Omega_a &= 0 \\
\left[\langle \kappa, \xi \rangle \Omega + \beta \dot{\xi} \right]_{t=a} &= 0,
\end{aligned} \tag{4.93}$$

and the right boundary conditions (4.85) become

$$\begin{aligned}
\Omega(b) - \Omega_b &= 0 \\
\left[\langle \kappa, \xi \rangle \Omega + \beta \dot{\xi} \right]_{t=b} &= 0 \\
\left[\frac{\alpha}{4} \left[|\xi|^2 - 1 \right]^2 + \frac{\beta}{2} |\dot{\xi}|^2 + \frac{\gamma}{2} |\Omega - \Omega_d|^2 + \delta - \frac{\eta}{2} |\dot{\Omega}|^2 - \left\langle \langle \xi, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa, \dot{\Omega} \right\rangle \right]_{t=b} &= 0.
\end{aligned} \tag{4.94}$$

(4.92) is an implicit system of ODEs since $\dot{\kappa}$ depends on $\ddot{\Omega}$, which in turn depends on $\ddot{\xi}$, while $\ddot{\xi}$ depends on $\dot{\kappa}$. While one can in principle proceed to solve these equations as an implicit system of ODEs, an explicit expression for the highest derivatives can be found which reveals possible singularities in the system. (4.92) may be expressed as an explicit system of ODEs via a few algebraic manipulations. Since

$$\begin{aligned}
\ddot{\Omega} &= \mathbb{I}^{-1} \left\{ \left(\mathbb{I} \dot{\Omega} \right) \times \Omega + (\mathbb{I} \Omega) \times \dot{\Omega} - \left[\frac{\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \right] \dot{\xi} \right. \\
&\quad \left. - \left[\frac{\langle \xi, \mathbb{I}^{-1} \xi \rangle \left[\dot{\mathbf{n}}_1 + \langle \Omega, \ddot{\xi} \rangle \right] - 2 \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \langle \dot{\xi}, \mathbb{I}^{-1} \xi \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \right] \xi \right\},
\end{aligned} \tag{4.95}$$

where

$$\dot{\mathbf{n}}_1 = \langle \dot{\Omega}, \dot{\xi} \rangle + \langle (\mathbb{I} \dot{\Omega}) \times \Omega + (\mathbb{I} \Omega) \times \dot{\Omega}, \mathbb{I}^{-1} \xi \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \dot{\xi} \rangle, \tag{4.96}$$

$\dot{\kappa}$ can be rewritten as

$$\begin{aligned}
\dot{\kappa} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \gamma (\Omega - \Omega_d) - \eta \dot{\Omega} - \langle \xi, \mathbb{I}^{-1} \xi \rangle [\mathbb{I} (\Omega \times \kappa) + \kappa \times (\mathbb{I} \Omega)] \right. \\
&\quad \left. + \langle \kappa, \xi \rangle \left[\dot{\xi} + \mathbb{I} (\Omega \times (\mathbb{I}^{-1} \xi)) + (\mathbb{I}^{-1} \xi) \times (\mathbb{I} \Omega) \right] - 2 \langle \dot{\xi}, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa \right\} \\
&= \mathbf{g} + \frac{\eta \langle \Omega, \ddot{\xi} \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \mathbb{I}^{-2} \xi,
\end{aligned} \tag{4.97}$$

where

$$\begin{aligned}
\mathbf{g} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \gamma (\Omega - \Omega_d) \right. \\
&\quad \left. - \eta \mathbb{I}^{-1} \left\{ \left(\mathbb{I} \dot{\Omega} \right) \times \Omega + (\mathbb{I} \Omega) \times \dot{\Omega} - \left[\frac{\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \right] \dot{\xi} \right. \right. \\
&\quad \left. \left. - \left[\frac{\langle \xi, \mathbb{I}^{-1} \xi \rangle \dot{\mathbf{n}}_1 - 2 \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \langle \dot{\xi}, \mathbb{I}^{-1} \xi \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \right] \xi \right\} \\
&\quad - \langle \xi, \mathbb{I}^{-1} \xi \rangle [\mathbb{I} (\Omega \times \kappa) + \kappa \times (\mathbb{I} \Omega)] \\
&\quad + \langle \kappa, \xi \rangle \left[\dot{\xi} + \mathbb{I} (\Omega \times (\mathbb{I}^{-1} \xi)) + (\mathbb{I}^{-1} \xi) \times (\mathbb{I} \Omega) \right] - 2 \langle \dot{\xi}, \mathbb{I}^{-1} \xi \rangle \mathbb{I} \kappa \}.
\end{aligned} \tag{4.98}$$

Using this formula for $\dot{\kappa}$, $\ddot{\xi}$ can be rewritten as

$$\begin{aligned}
\ddot{\xi} &= \frac{1}{\beta} \left\{ \alpha \left(|\xi|^2 - 1 \right) \xi - 2 \langle \kappa, (\mathbb{I} \Omega) \times \Omega \rangle \mathbb{I}^{-1} \xi + \langle \kappa, \xi \rangle \mathbb{I}^{-1} ((\mathbb{I} \Omega) \times \Omega) \right. \\
&\quad \left. + \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \kappa - \langle \kappa, \dot{\xi} \rangle \Omega \right. \\
&\quad \left. - \left[\langle \kappa, \xi \rangle I - 2 \mathbb{I}^{-1} \xi (\mathbb{I} \kappa)^{\top} \right] \dot{\Omega} - \Omega \langle \xi, \dot{\kappa} \rangle \right\} \\
&= \frac{1}{\beta} \{ \mathbf{h} - \Omega \langle \xi, \dot{\kappa} \rangle \} \\
&= \frac{1}{\beta} \left\{ \mathbf{h} - \Omega \left\langle \xi, \mathbf{g} + \frac{\eta \langle \Omega, \ddot{\xi} \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \mathbb{I}^{-2} \xi \right\rangle \right\} \\
&= \frac{1}{\beta} \{ \mathbf{h} - \Omega \langle \xi, \mathbf{g} \rangle \} - \frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \Omega \Omega^{\top} \ddot{\xi},
\end{aligned} \tag{4.99}$$

where

$$\begin{aligned}
\mathbf{h} &= \alpha \left(|\xi|^2 - 1 \right) \xi - 2 \langle \kappa, (\mathbb{I} \Omega) \times \Omega \rangle \mathbb{I}^{-1} \xi + \langle \kappa, \xi \rangle \mathbb{I}^{-1} ((\mathbb{I} \Omega) \times \Omega) \\
&\quad + \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \kappa - \langle \kappa, \dot{\xi} \rangle \Omega \\
&\quad - \left[\langle \kappa, \xi \rangle I - 2 \mathbb{I}^{-1} \xi (\mathbb{I} \kappa)^{\top} \right] \dot{\Omega}.
\end{aligned} \tag{4.100}$$

Thus,

$$\left[I + \frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \Omega \Omega^{\top} \right] \ddot{\xi} = \frac{1}{\beta} \{ \mathbf{h} - \Omega \langle \xi, \mathbf{g} \rangle \}. \tag{4.101}$$

Now $\ddot{\xi}$ can be solved for in terms of Ω , $\dot{\Omega}$ ($\dot{\Omega}$ is actually a function of Ω , ξ , and $\dot{\xi}$), ξ , $\dot{\xi}$, and κ via

$$\begin{aligned}
\ddot{\xi} &= \frac{1}{\beta} \left[I + \frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \Omega \Omega^\top \right]^{-1} \{ \mathbf{h} - \Omega \langle \xi, \mathbf{g} \rangle \} \\
&= \frac{1}{\beta} \left[I - \frac{\frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \Omega \Omega^\top}{1 + \frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle \langle \Omega, \Omega \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2}} \right] \{ \mathbf{h} - \Omega \langle \xi, \mathbf{g} \rangle \} \\
&= \frac{1}{\beta} \left[I - \frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2 + \eta \langle \xi, \mathbb{I}^{-2} \xi \rangle \langle \Omega, \Omega \rangle} \Omega \Omega^\top \right] \{ \mathbf{h} - \Omega \langle \xi, \mathbf{g} \rangle \}.
\end{aligned} \tag{4.102}$$

To compute the ODEs explicitly, $\dot{\Omega}$, \mathbf{g} , \mathbf{h} , $\ddot{\xi}$, and $\dot{\kappa}$ must be computed in that order, given Ω , ξ , $\dot{\xi}$, and κ . The ODE system (4.92) can be rewritten

$$\begin{aligned}
\dot{\Omega} &= \frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \langle \xi, \mathbb{I}^{-1} \xi \rangle (\mathbb{I} \Omega) \times \Omega - \left[\langle \Omega, \dot{\xi} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \xi \right\} \\
\ddot{\xi} &= \frac{1}{\beta} \left[I - \frac{\eta \langle \xi, \mathbb{I}^{-2} \xi \rangle}{\beta \langle \xi, \mathbb{I}^{-1} \xi \rangle^2 + \eta \langle \xi, \mathbb{I}^{-2} \xi \rangle \langle \Omega, \Omega \rangle} \Omega \Omega^\top \right] \{ \mathbf{h} - \Omega \langle \xi, \mathbf{g} \rangle \} \\
\dot{\kappa} &= \mathbf{g} + \frac{\eta \langle \Omega, \ddot{\xi} \rangle}{\langle \xi, \mathbb{I}^{-1} \xi \rangle^2} \mathbb{I}^{-2} \xi.
\end{aligned} \tag{4.103}$$

The ODEs (4.103) and the left and right boundary conditions (4.93)-(4.94) define an ODE TPBVP for the solution to Suslov's optimal control problem (4.69) using the integrand cost function (4.68). We shall also notice that while casting the optimal control problem as an explicit system of ODEs such as (4.103) brings it to the standard form amenable to numerical solution, it loses the geometric background of the optimal control problem derived earlier in Subsection 4.A.2.

Remark 4.11 (On optimal solutions with switching structure and bang-bang control). *It is worth noting that we allow the control $\dot{\xi}$ to be unbounded so that it may take arbitrary values in \mathbb{R}^3 . In addition, note that at the end of the previous subsection, the control $\dot{\xi}$ is assumed to be differentiable and therefore continuous. However, if we were to set up a restriction on the control such as $|\dot{\xi}| \leq M$ for a fixed $|\xi|$, say $|\xi| = 1$, and permit $\dot{\xi}$ to be piecewise continuous, then the solutions to the optimal control problems tend to lead to bang-bang control obtained by piecing together solutions with $|\dot{\xi}| = M$. The constraint $|\xi| = 1$ is equivalent to the constraint $\langle \xi, \dot{\xi} \rangle = 0$ with the initial condition $|\xi(a)| = 1$. The constraint $|\dot{\xi}| \leq M$ is equivalent to the constraint $|\dot{\xi}|^2 - M^2 - \theta^2 = 0$, where θ is a so-called slack variable. To incorporate these constraints, the Hamiltonian given in (4.86) must be amended to*

$$\begin{aligned}
H &= L(\Omega, \xi, \mathbf{u}, t) \\
&+ \left\langle \pi, \left[\frac{\mathbb{I}^{-1}}{\langle \xi, \mathbb{I}^{-1} \xi \rangle} \left\{ \langle \xi, \mathbb{I}^{-1} \xi \rangle (\mathbb{I} \Omega) \times \Omega - \left[\langle \Omega, \mathbf{u} \rangle + \langle (\mathbb{I} \Omega) \times \Omega, \mathbb{I}^{-1} \xi \rangle \right] \xi \right\} \right] \right\rangle \\
&+ \left\langle \mu, \left| \dot{\xi} \right|^2 - M^2 - \theta^2 \right\rangle,
\end{aligned} \tag{4.104}$$

where $\boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} \in \mathbb{R}^2$ are new costates enforcing the new constraints and the control now consists of $\mathbf{u} = \dot{\boldsymbol{\xi}}$ and θ . A solution that minimizes the optimal control problem with Hamiltonian (4.104) is determined from the necessary optimality conditions $H_{\mathbf{u}} = \mathbf{0}$ and $H_{\theta} = -2\mu_2\theta = 0$. The latter condition implies that $\mu_2 = 0$ or $\theta = 0$. If $\mu_2 = 0$, the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is determined from $H_{\mathbf{u}} = \mathbf{0}$ and θ is determined from $\theta^2 = |\dot{\boldsymbol{\xi}}|^2 - M$. If $\theta = 0$, the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is determined from $|\dot{\boldsymbol{\xi}}|^2 = M$ and μ_2 is determined from $H_{\mathbf{u}} = \mathbf{0}$. The difficulty is determining the intervals on which $\mu_2 = 0$ or $\theta = 0$; this is the so-called optimal switching structure. In this thesis, this difficulty is avoided by assuming that the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ is unbounded and differentiable rather than bounded and piecewise continuous. Instead of bounding the control $\mathbf{u} = \dot{\boldsymbol{\xi}}$ through hard constraints, large magnitude controls are penalized by the term $\frac{\beta}{2} |\dot{\boldsymbol{\xi}}|^2$ in the cost function (4.68).

4.C Numerical Solution of Suslov's Optimal Control Problem

4.C.1 Analytical Solution of a Singular Version of Suslov's Optimal Control Problem

In what follows, we shall focus on the numerical solution of the optimal control problem (4.69) by solving (4.103), (4.93), and (4.94), with $\alpha > 0$, $\beta > 0$, $\gamma \geq 0$, $\eta \geq 0$, and $\delta \geq 0$. As these equations represent a nonlinear ODE TPBVP, having a good initial approximate solution is crucial for the convergence of numerical methods. Because of the complexity of the problem, the numerical methods show no convergence to the solution unless the case considered is excessively simple. Instead, we employ the continuation procedure, namely, we solve a problem with the values of the parameters chosen in such a way that an analytical solution of (4.69) can be found. Starting from this analytical solution, we seek a continuation of the solution to the desired values of the parameters. As it turns out, this procedure enables the computation of rather complex trajectories as illustrated by the numerical examples in Subsection 4.C.3.

To begin, let us consider a simplification of the optimal control problem (4.69). Suppose the final time is fixed to $b = b_p$, $\beta = 0$, $\eta = 0$, and $\delta = 0$. In addition, suppose $\boldsymbol{\Omega}_d$ is replaced by $\boldsymbol{\Omega}_p$, where $\boldsymbol{\Omega}_p$ satisfies the following properties:

Property 4.12. $\boldsymbol{\Omega}_p$ is a differentiable function such that $\boldsymbol{\Omega}_p(a) = \boldsymbol{\Omega}_a$ and $\boldsymbol{\Omega}_p(b_p) = \boldsymbol{\Omega}_b$.

Property 4.13. $\boldsymbol{\Omega}_p$ lies on the constant kinetic energy manifold E , i.e. $\langle \mathbb{I}\boldsymbol{\Omega}_p, \dot{\boldsymbol{\Omega}}_p \rangle = 0$ iff $\langle \mathbb{I}\boldsymbol{\Omega}_p, \boldsymbol{\Omega}_p \rangle = \langle \mathbb{I}\boldsymbol{\Omega}_a, \boldsymbol{\Omega}_a \rangle$.

Property 4.14. $\boldsymbol{\Omega}_p$ does not satisfy Euler's equations at any time, i.e. $\mathbb{I}\dot{\boldsymbol{\Omega}}_p(t) - [\mathbb{I}\boldsymbol{\Omega}_p(t)] \times \boldsymbol{\Omega}_p(t) \neq \mathbf{0} \quad \forall t \in [a, b_p]$.

Under these assumptions, (4.69) simplifies to

$$\min_{\boldsymbol{\xi}} \int_a^{b_p} \left[\frac{\alpha}{4} \left[|\boldsymbol{\xi}|^2 - 1 \right]^2 + \frac{\gamma}{2} |\boldsymbol{\Omega} - \boldsymbol{\Omega}_p|^2 \right] dt \quad \text{s.t.} \quad \begin{cases} \mathbf{q} = 0, \\ \boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a \in E, \\ \boldsymbol{\Omega}(b_p) = \boldsymbol{\Omega}_b \in E. \end{cases} \quad (4.105)$$

As discussed immediately after (4.91), (4.105) is a singular optimal control problem since $\beta = 0$. If there exists ξ_p such that $|\xi_p| = 1$ and $\mathbf{q}(\Omega_p, \xi_p) = \mathbf{0}$, then ξ_p is a solution to the singular optimal control problem (4.105) provided Property (4.12) is satisfied. To wit, for such a ξ_p and given Property (4.12), take $\Omega = \Omega_p$ and $\xi = \xi_p$. Then $\mathbf{q}(\Omega, \xi) = \mathbf{q}(\Omega_p, \xi_p) = \mathbf{0}$, $\Omega(a) = \Omega_p(a) = \Omega_a$, $\Omega(b_p) = \Omega_p(b_p) = \Omega_b$, and $\int_a^{b_p} \left[\frac{\alpha}{4} [|\xi|^2 - 1]^2 + \frac{\gamma}{2} |\Omega - \Omega_p|^2 \right] dt = 0$.

Now to construct such a ξ_p , assume Ω_p satisfies Properties (4.12)-(4.14). To motivate the construction of ξ_p , also assume that $\hat{\xi}$ exists for which $\mathbf{q}(\Omega_p, \hat{\xi}) = \mathbf{0}$, $\hat{\xi}(t) \neq \mathbf{0} \quad \forall t \in [a, b_p]$, and $\langle \Omega_p, \hat{\xi} \rangle = c = 0$. Since $\langle \Omega_p, \hat{\xi} \rangle = c = 0$, $\mathbf{q}(\Omega_p, \pi \hat{\xi}) = \mathbf{0}$ for any rescaling π of $\hat{\xi}$. Letting $\tilde{\xi} \equiv \lambda(\Omega_p, \hat{\xi}) \hat{\xi} = \mathbb{I}\dot{\Omega}_p - (\mathbb{I}\Omega_p) \times \Omega_p$, $\mathbf{q}(\Omega_p, \tilde{\xi}) = \mathbf{q}(\Omega_p, \lambda(\Omega_p, \hat{\xi}) \hat{\xi}) = \mathbf{0}$. Next, by Property (4.14) (i.e. $\mathbb{I}\dot{\Omega}_p(t) - [\mathbb{I}\Omega_p(t)] \times \Omega_p(t) \neq \mathbf{0} \quad \forall t \in [a, b_p]$), normalize $\tilde{\xi}$ to produce a unit magnitude control vector ξ_p :

$$\xi_p \equiv \frac{\tilde{\xi}}{|\tilde{\xi}|} = \frac{\mathbb{I}\dot{\Omega}_p - (\mathbb{I}\Omega_p) \times \Omega_p}{|\mathbb{I}\dot{\Omega}_p - (\mathbb{I}\Omega_p) \times \Omega_p|}. \quad (4.106)$$

Again due to scale invariance of the control vector, $\mathbf{q}(\Omega_p, \xi_p) = \mathbf{0}$.

One can note that this derivation of ξ_p possessing the special properties $\mathbf{q}(\Omega_p, \xi_p) = \mathbf{0}$ and $|\xi_p| = 1$ relied on the existence of some $\hat{\xi}$ for which $\mathbf{q}(\Omega_p, \hat{\xi}) = \mathbf{0}$, $\hat{\xi}(t) \neq \mathbf{0} \quad \forall t \in [a, b_p]$, and $\langle \Omega_p, \hat{\xi} \rangle = c = 0$. Given ξ_p defined by (4.106) and by Property (4.13) (i.e. $\langle \mathbb{I}\Omega_p, \dot{\Omega}_p \rangle = 0$), it is trivial to check that $\langle \Omega_p, \xi_p \rangle = 0$, so that indeed $\mathbf{q}(\Omega_p, \xi_p) = \mathbf{0}$ with

$$\lambda(\Omega_p, \xi_p) \equiv -\frac{\langle \Omega_p, \dot{\xi}_p \rangle + \langle (\mathbb{I}\Omega_p) \times \Omega_p, \mathbb{I}^{-1}\xi_p \rangle}{\langle \xi_p, \mathbb{I}^{-1}\xi_p \rangle} = \frac{\langle \mathbb{I}\dot{\Omega}_p - (\mathbb{I}\Omega_p) \times \Omega_p, \mathbb{I}^{-1}\xi_p \rangle}{\langle \xi_p, \mathbb{I}^{-1}\xi_p \rangle} = |\mathbb{I}\dot{\Omega}_p - (\mathbb{I}\Omega_p) \times \Omega_p|.$$

Thus ξ_p defined by (4.106) is a solution of the singular optimal control problem (4.105). Moreover, ξ_p has the desirable property $\langle \Omega_p, \xi_p \rangle = 0$. The costate $\kappa = \mathbf{0}$ satisfies the ODE TPBVP (4.103), (4.93)-(4.94) corresponding to the analytic solution pair (Ω_p, ξ_p) .

4.C.2 Numerical Solution of Suslov's Optimal Control Problem via Continuation

Starting from the analytic solution pair (Ω_p, ξ_p) solving (4.105), the full optimal control problem can then be solved by continuation in γ , β , η , and δ using the following algorithm. We refer the reader to [50] as a comprehensive reference on numerical continuation methods, as well as our discussion in Appendix A. Consider the continuation integrand cost function $C_{\alpha, \beta, \gamma, \eta, \delta}$, where β , γ , η , and δ are variables. If $\gamma = 0$, choose β_m such that $0 < \beta_m \ll \min\{\alpha, \beta, 1\}$; otherwise if $\gamma > 0$, choose β_m such that $0 < \beta_m \ll \min\{\alpha, \beta, \gamma\}$. If the final time b is fixed, choose $b_p = b$; otherwise, if the final time is free, choose b_p as explained below.

If $\gamma = 0$, choose Ω_p to be some nominal function satisfying Properties (4.12)-(4.14), such as the projection of the line segment connecting Ω_a to Ω_b onto E and let b_p be the time such that $\Omega_p(b_p) = \Omega_b$. For fixed final

time b_p , solve (4.69) with integrand cost function $C_{\alpha,\beta_m,\gamma_c,0,0}$ by continuation in γ_c , starting from $\gamma_c = 1$ with the initial solution guess $(\boldsymbol{\Omega}_p, \boldsymbol{\xi}_p)$ and ending at $\gamma_c = \gamma = 0$ with the final solution pair $(\boldsymbol{\Omega}_1, \boldsymbol{\xi}_1)$.

If $\gamma > 0$ and $\boldsymbol{\Omega}_d$ doesn't satisfy Properties (4.12)-(4.14), choose $\boldsymbol{\Omega}_p$ to be some function "near" $\boldsymbol{\Omega}_d$ that does satisfy Properties (4.12)-(4.14) and let b_p be the time such that $\boldsymbol{\Omega}_p(b_p) = \boldsymbol{\Omega}_b$. For fixed final time b_p , solve (4.69) with integrand cost function $C_{\alpha,\beta_m,\gamma_c,0,0}$ by continuation in γ_c , starting from $\gamma_c = 0$ with the initial solution guess $(\boldsymbol{\Omega}_p, \boldsymbol{\xi}_p)$ and ending at $\gamma_c = \gamma$ with the final solution pair $(\boldsymbol{\Omega}_1, \boldsymbol{\xi}_1)$.

If $\gamma > 0$ and $\boldsymbol{\Omega}_d$ satisfies Properties (4.12)-(4.14), choose $\boldsymbol{\Omega}_p = \boldsymbol{\Omega}_d$, let b_p be the time such that $\boldsymbol{\Omega}_d(b_p) = \boldsymbol{\Omega}_b$, and construct the solution pair $(\boldsymbol{\Omega}_1, \boldsymbol{\xi}_1)$ with $\boldsymbol{\Omega}_1 = \boldsymbol{\Omega}_p$ and $\boldsymbol{\xi}_1 = \boldsymbol{\xi}_p$.

For fixed final time b_p , solve (4.69) with integrand cost function $C_{\alpha,\beta_c,\gamma,0,0}$ by continuation in β_c , starting from $\beta_c = \beta_m$ with the initial solution guess $(\boldsymbol{\Omega}_1, \boldsymbol{\xi}_1)$ and ending at $\beta_c = \beta$ with the final solution pair $(\boldsymbol{\Omega}_2, \boldsymbol{\xi}_2)$. Next, for fixed final time b_p , solve (4.69) with integrand cost function $C_{\alpha,\beta,\gamma,\eta_c,0}$ by continuation in η_c , starting from $\eta_c = 0$ with the initial solution guess $(\boldsymbol{\Omega}_2, \boldsymbol{\xi}_2)$ and ending at $\eta_c = \eta$ with the final solution pair $(\boldsymbol{\Omega}_3, \boldsymbol{\xi}_3)$. If the final time is fixed, then this is the final solution. If the final time is free, solve (4.69) with integrand cost function $C_{\alpha,\beta,\gamma,\eta,\delta_c}$, letting the final time vary, by continuation in δ_c , starting from

$$\delta_c = - \left[\frac{\alpha}{4} \left[|\boldsymbol{\xi}|^2 - 1 \right]^2 + \frac{\beta}{2} \left| \dot{\boldsymbol{\xi}} \right|^2 + \frac{\gamma}{2} |\boldsymbol{\Omega} - \boldsymbol{\Omega}_d|^2 - \frac{\eta}{2} \left| \dot{\boldsymbol{\Omega}} \right|^2 - \left\langle \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa}, \dot{\boldsymbol{\Omega}} \right\rangle \right]_{t=b} \quad (4.107)$$

with the initial solution guess $(\boldsymbol{\Omega}_3, \boldsymbol{\xi}_3, b_p)$ and ending at $\delta_c = \delta$ with final solution triple $(\boldsymbol{\Omega}_4, \boldsymbol{\xi}_4, b_4)$. If the final time is free, then this is the final solution.

4.C.3 Numerical Solution of Suslov's Optimal Control Problem via the Indirect Method and Continuation

Suslov's optimal control problem was solved numerically using the following inputs and setup. The rigid body's inertia matrix is

$$\mathbb{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}. \quad (4.108)$$

The initial time is $a = 0$ and the final time b is free. The initial and final body angular velocities are $\boldsymbol{\Omega}_a = \boldsymbol{\phi}(a)/2 = [5, 0, 0]^\top$ and $\boldsymbol{\Omega}_b = \boldsymbol{\phi}_\parallel(b_d) \approx [2.7541, -2.3109, -1.4983]^\top$, respectively, where $\boldsymbol{\phi}$ and $\boldsymbol{\phi}_\parallel$ are defined below in (4.109)-(4.111) and $b_d = 10$.

The desired body angular velocity $\boldsymbol{\Omega}_d$ (see Figure 4.1) is approximately the projection of a spiral onto the constant kinetic energy ellipsoid E determined by the rigid body's inertia matrix \mathbb{I} and initial body angular velocity $\boldsymbol{\Omega}_a$ and defined in (4.49). Concretely, we aim to track a spiral-like trajectory $\boldsymbol{\Omega}_d$ on the constant kinetic energy ellipsoid E :

$$\boldsymbol{\phi}(t) = [10, t \cos t, t \sin t]^\top, \quad (4.109)$$

$$\mathbf{v}_\parallel = \sqrt{\frac{\langle \boldsymbol{\Omega}_a, \mathbb{I} \boldsymbol{\Omega}_a \rangle}{\langle \mathbf{v}, \mathbb{I} \mathbf{v} \rangle}} \mathbf{v} \quad \text{for } \mathbf{v} \in \mathbb{R}^3 \setminus \mathbf{0}, \quad (4.110)$$

$$\phi_{\parallel}(t) = [\phi(t)]_{\parallel}, \quad (4.111)$$

$$\sigma(t) = \frac{1}{2} \left[1 + \tanh \left(\frac{t}{.01} \right) \right], \quad (4.112)$$

$$s(t) = \sigma(t - b_d), \quad (4.113)$$

$$\Omega_d(t) = \phi_{\parallel}(t) (1 - s(t)) + \Omega_b s(t). \quad (4.114)$$

The setup for Ω_d is to be understood as follows. The graph of ϕ (4.109) defines a spiral in the plane $x = 10$. Given a nonzero vector $\mathbf{v} \in \mathbb{R}^3$, the parallel projection operator \parallel (4.110) constructs the vector \mathbf{v}_{\parallel} that lies at the intersection between the ray $R_{\mathbf{v}} = \{t\mathbf{v} : t > 0\}$ and the ellipsoid E . The spiral ϕ_{\parallel} defined by (4.111) is the projection of the spiral ϕ onto the ellipsoid E , which begins at Ω_a at time a , and terminates at Ω_b at time b_d . Also, σ (5.98) is a sigmoid function, i.e. a smooth approximation of the unit step function, and s (4.113) is the time translation of σ to time b_d . Ω_d (4.114) utilizes the translated sigmoid function s to compute a weighted average of the projected spiral ϕ_{\parallel} and Ω_b so that Ω_d follows the projected spiral ϕ_{\parallel} for $0 \leq t < b_d$, holds steady at Ω_b for $t > b_d$, and smoothly transitions between ϕ_{\parallel} and Ω_b at time b_d . The coefficients of the integrand cost function (4.68) are chosen to be $\alpha = 1$, $\beta = .1$, $\gamma = 1$, $\eta = 1$ or $.01$, and $\delta = .2$.

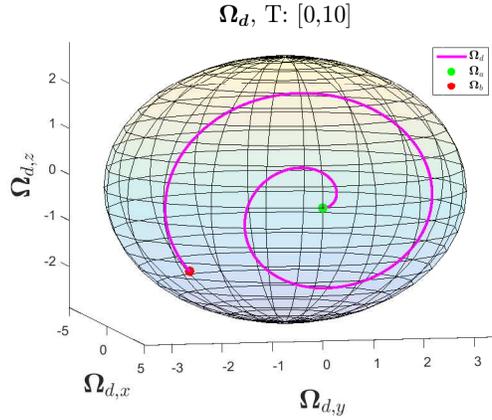


Figure 4.1: The desired body angular velocity is approximately the projection of a spiral onto the constant kinetic energy ellipsoid.

The optimal control problem (4.69) was solved numerically via the indirect method, i.e. by numerically solving the ODE TPBVP (4.103), (4.93)-(4.94) through continuation in β , η , and δ starting from the analytic solution to the singular optimal control problem (4.105), as outlined in Section 4.C.2. Because most ODE BVP solvers only solve problems defined on a fixed time interval, the ODE TPBVP (4.103), (4.93)-(4.94) was reformulated on the normalized time interval $[0, 1]$ through a change of variables by defining $T \equiv b - a$ and by defining normalized time $s \equiv \frac{t-a}{T}$; if the final time b is fixed, then T is a known constant, whereas if the final time b is free, then T is an unknown parameter that must be solved for in the ODE TPBVP. The collocation automatic continuation solver `acdc` from the MATLAB package `bvptwp` was used to solve the ODE TPBVP by performing continuation in β , η , and δ , with the relative error tolerance set to $1e-8$. The result of `acdc` was then passed through the MATLAB collocation solver `sbvp` using Gauss (rather than equidistant) collocation points with the absolute and relative error tolerances set to $1e-8$. `sbvp` was used to clean up the solution provided by `acdc` because collocation exhibits superconvergence when solving regular (as opposed to singular) ODE TPBVP using Gauss collocation points. To make `acdc` and `sbvp`

execute efficiently, the ODEs were implemented in MATLAB in vectorized fashion. For accuracy and efficiency, the MATLAB software ADiGator was used to supply vectorized, automatic ODE Jacobians to `acdc` and `sbvp`. For accuracy, the MATLAB Symbolic Math Toolbox was used to supply symbolically-computed BC Jacobians to `acdc` and `sbvp`. ADiGator constructs Jacobians through automatic differentiation, while the MATLAB Symbolic Math Toolbox constructs Jacobians through symbolic differentiation.

Figures 4.2 and 4.3 show the results for $\eta = 1$ and $\eta = .01$, respectively. The optimal final time is $b = 11.36$ for $\eta = 1$ and is $b = 9.84$ for $\eta = .01$. Figures 4.2a and 4.3a show the optimal body angular velocity $\mathbf{\Omega}$, the desired body angular velocity $\mathbf{\Omega}_d$, and the projection $\mathbf{\xi}_{\parallel}$ of the control vector $\mathbf{\xi}$ onto the ellipsoid E . Recall that γ , through the integrand cost function term $\frac{\gamma}{2} |\mathbf{\Omega} - \mathbf{\Omega}_d|^2$, influences how closely the optimal body angular velocity $\mathbf{\Omega}$ tracks the desired body angular velocity $\mathbf{\Omega}_d$, while η , through the integrand cost function term $\frac{\eta}{2} |\dot{\mathbf{\Omega}}|^2$, influences how closely the optimal body angular velocity $\mathbf{\Omega}$ tracks a minimum energy trajectory. For $\gamma = 1$, $\frac{\gamma}{\eta} = 1$ when $\eta = 1$ and $\frac{\gamma}{\eta} = 100$ when $\eta = .01$. As expected, comparing Figures 4.2a and 4.3a, the optimal body angular velocity $\mathbf{\Omega}$ tracks the desired body angular velocity $\mathbf{\Omega}_d$ much more accurately for $\eta = .01$ compared to $\eta = 1$. Figures 4.2b and 4.3b demonstrate that the numerical solutions preserve the nonholonomic orthogonality constraint $\langle \mathbf{\Omega}, \mathbf{\xi} \rangle = 0$ to machine precision. Figures 4.2c and 4.3c show that the magnitude $|\mathbf{\xi}|$ of the control vector $\mathbf{\xi}$ remains close to 1, as encouraged by the integrand cost function term $\frac{\alpha}{4} [|\mathbf{\xi}|^2 - 1]^2$. Figures 4.2d and 4.3d show the costates $\mathbf{\kappa}$. In Figures 4.2a, 4.3a, 4.2d, and 4.3d, a green marker indicates the beginning of a trajectory, while a red marker indicates the end of trajectory. In Figure 4.3a, the yellow marker on the desired body angular velocity indicates $\mathbf{\Omega}_d(b)$, where $b = 9.84$ is the optimal final time for $\eta = .01$.

To investigate the stability of the controlled system, we have perturbed the control $\dot{\mathbf{\xi}}$ obtained from solving the optimal control ODE TPBVP and observed that the perturbed solution $\mathbf{\Omega}$ obtained by solving the uncontrolled equations of motion (4.43) as an ODE IVP using this perturbed control is similar to the anticipated $\mathbf{\Omega}$ corresponding to the solution of the optimal control ODE TPBVP and the unperturbed control. While more studies of stability are needed, this is an indication that the controlled system we studied is stable, at least in terms of the state variables $\mathbf{\Omega}$ and $\mathbf{\xi}$ under perturbations of the control $\dot{\mathbf{\xi}}$. More studies of the stability of the controlled system will be undertaken in the future.

Verification of a local minimum solution It is also desirable to verify that the numerical solutions obtained by our continuation indirect method do indeed provide a local minimum of the optimal control problem. Chapter 21 in reference [44] and also reference [51] provide sufficient conditions for a solution satisfying Pontryagin’s minimum principle to be a local minimum, however the details are quite technical and may be investigated in future work. These sufficient conditions must be checked numerically rather than analytically. COTCOT and HamPath, also mentioned in Appendix A, are numerical software packages which do check these sufficient conditions numerically.

Due to the technicality of the sufficient conditions discussed in [44, 51], we have resorted to a different numerical justification. More precisely, to validate that the solutions obtained by our optimal control procedure, or the so-called *indirect* method solutions, indeed correspond to local minima, we have fed the solutions obtained by our method into several different MATLAB direct method solvers as initial solution guesses. We provide a survey of the current state of direct method solvers for optimal control problems in Appendix A.

Note that the indirect method only produces a solution that meets the necessary conditions for a local minimum to (4.69), while the direct method solution meets the necessary and sufficient conditions for a local minimum to a finite-dimensional approximation of (4.69). Thus, it may be concluded that an indirect method solution is indeed a local minimum solution of (4.69) if the direct method solution is close to the indirect method solution. The indirect method solutions were validated against the MATLAB direct method solvers GPOPS-II and FALCON.m. GPOPS-II uses pseudospectral collocation techniques, uses the IPOPT NLP solver, uses *hp*-adaptive mesh refinement, and can use ADiGator to supply vectorized, automatic Jacobians and Hessians. FALCON.m uses trapezoidal or backward Euler local collocation techniques, uses the IPOPT NLP solver, and can use the MATLAB Symbolic Math Toolbox to supply symbolically-computed Jacobians and Hessians. Both direct method solvers we have tried converged to a solution close to that provided by the indirect method, which is to be expected since the direct method solvers are only solving a finite-dimensional approximation of (4.69). Thus, we are confident that the solutions we have found in this section indeed correspond to local minima of the optimal control problems.

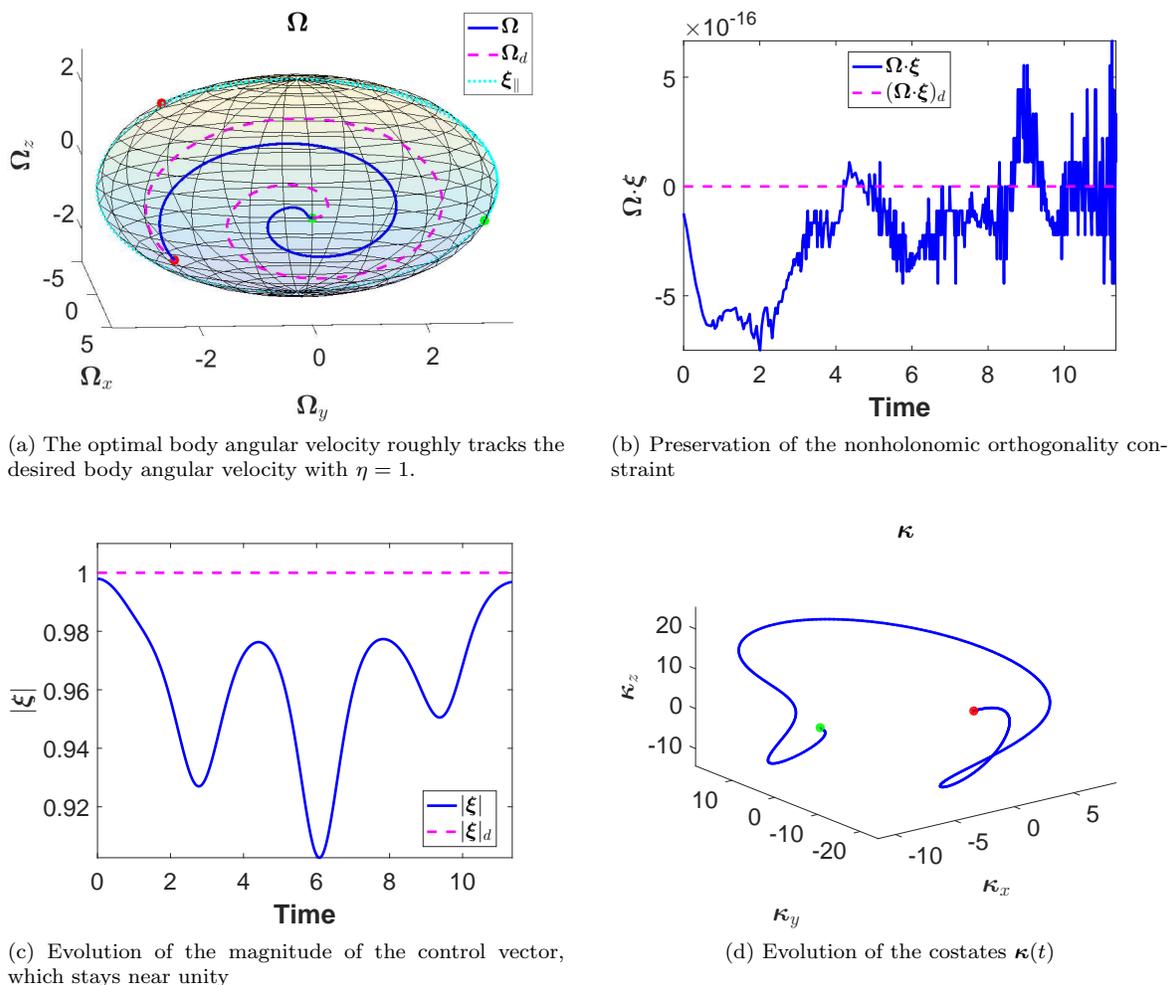
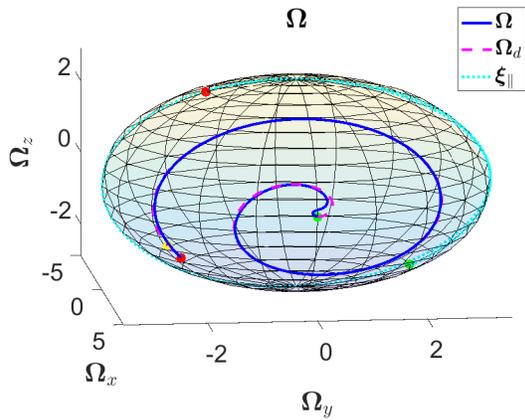
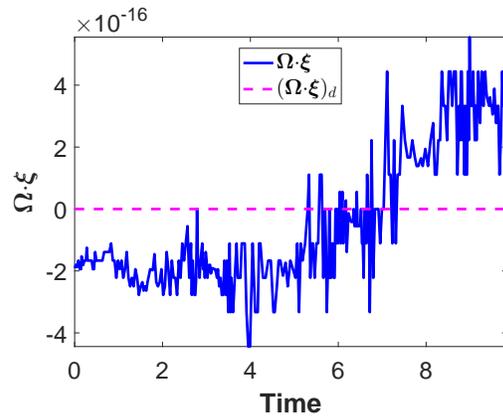


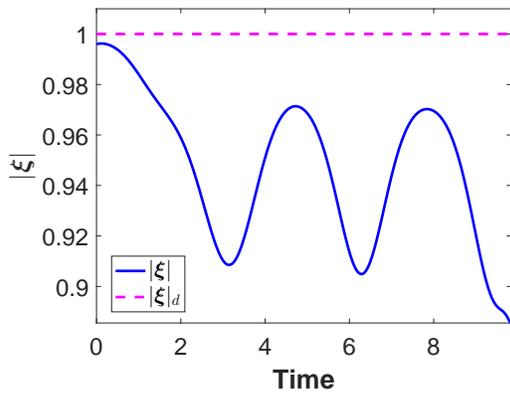
Figure 4.2: Numerical solution of the optimal control problem for $\alpha = 1$, $\beta = .1$, $\gamma = 1$, $\eta = 1$, $\delta = .2$, and free final time. The optimal final time is $b = 11.36$.



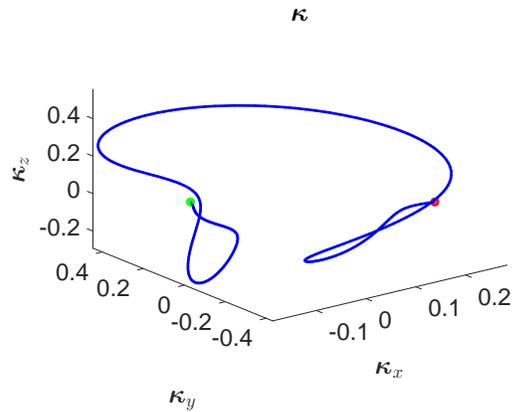
(a) The optimal body angular velocity accurately tracks the desired body angular velocity with $\eta = .01$.



(b) Preservation of the nonholonomic orthogonality constraint



(c) Evolution of the magnitude of the control vector, which stays near unity



(d) Evolution of the costates $\kappa(t)$

Figure 4.3: Numerical solution of the optimal control problem for $\alpha = 1$, $\beta = .1$, $\gamma = 1$, $\eta = .01$, $\delta = .2$, and free final time. The optimal final time is $b = 9.84$.

Chapter 5

The Rolling Ball

This chapter investigates the optimal control of the rolling ball. Section 5.A discusses the specific type of rolling ball considered, presents natural questions about this rolling ball that motivate this chapter, and defines coordinates systems and notation used to describe this rolling ball. By applying Euler-Poincaré’s method and Lagrange-d’Alembert’s principle, Section 5.B derives the uncontrolled equations of motion for the rolling ball. Section 5.C formulates the regular Hamiltonian and endpoint function required to construct the controlled equations of motion (and their Jacobians) for the rolling ball according to the formulas derived in Section 3.C. In Section 5.D, the controlled equations of motion (and their Jacobians) for the rolling ball are constructed numerically via automatic differentiation, after which the controlled equations of motion are solved numerically via a predictor-corrector continuation method, starting from an initial solution provided by a direct method.

5.A Mechanical System and Motivation

Consider a rigid ball of radius r containing some static internal structure as well as $n \in \mathbb{N}^0$ point masses. This ball rolls without slipping on a flat surface in the presence of a uniform gravitational field. For $1 \leq i \leq n$, the i^{th} point mass may move within the ball along a trajectory ξ_i , expressed with respect to the ball’s frame of reference; since the motion of the i^{th} point mass along its trajectory ξ_i may actuate motion of the ball, the i^{th} point mass is hereafter called the i^{th} control mass. Refer to Figure 5.2 for an illustration. The trajectory ξ_i may be constrained in some way, such as being required to move along a 1-d control rail (like a circular hoop), across a 2-d control surface (like a sphere), or within a 3-d control region (like a ball) fixed within the ball. The ball with its static internal structure has mass m_0 and the i^{th} control mass has mass m_i for $1 \leq i \leq n$. Let $M = \sum_{i=0}^n m_i$ denote the mass of the total system. The total mechanical system consisting of the ball with its static internal structure and the n control masses is referred to as the ball or the rolling ball, the ball with its static internal structure but without the n control masses may also be referred to as m_0 , and the i^{th} control mass may also be referred to as m_i for $1 \leq i \leq n$.

It is natural to ask the following questions for this mechanical system:

1. How does the ball move if the n control masses are held fixed in place?
2. Given some prescribed motion of the n control masses, how does the ball move along the flat surface?
3. Suppose that it is desired to move the ball in a prescribed manner, such as moving the ball's geometric center along a prescribed trajectory parallel to the flat surface or performing obstacle avoidance. How might the n control masses be moved to realize such a motion? Figure 5.1 illustrates this problem for 2 control masses.

The aim of this chapter is to answer these questions. The answer to the 2nd question also answers the 1st, by insisting that the prescribed motion for each control mass be that of holding it fixed within the ball. The 3rd question is the inverse of the 2nd. Chaplygin answered the 1st question analytically for two special cases in his seminal 1897 and 1903 papers [17, 18]. As far as the author knows, the 2nd and 3rd questions have not been answered previously. The 1st and 2nd questions are answered in Section 5.B, while the 3rd question is answered in Section 5.C.

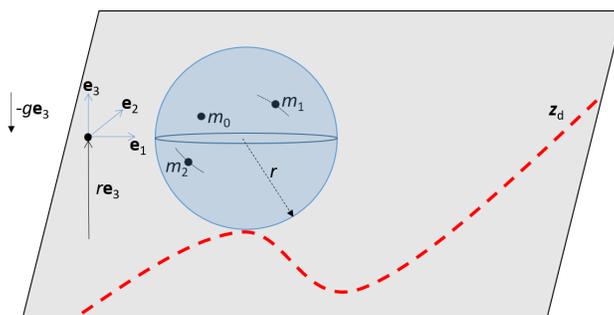


Figure 5.1: A ball of radius r and mass m_0 rolls without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g . The ball's center of mass is denoted by m_0 . In addition, the ball contains 2 internal point masses, m_1 and m_2 , that may move within the ball. How must m_1 and m_2 be moved to induce the ball to follow the prescribed trajectory \mathbf{z}_d ?

5.A.1 Coordinate Systems and Notation

Two coordinate systems, or frames of reference, will be used to describe the motion of the rolling ball, an inertial spatial coordinate system and a body coordinate system in which each particle within the ball is always fixed. For conciseness, the spatial coordinate system will be referred to as the spatial frame and the body coordinate system will be referred to as the body frame. These two frames are depicted in Figure 5.2. The spatial frame has orthonormal axes \mathbf{e}_1 , \mathbf{e}_2 , \mathbf{e}_3 , such that the \mathbf{e}_1 - \mathbf{e}_2 plane is parallel to the flat surface and passes through the ball's geometric center (i.e. the \mathbf{e}_1 - \mathbf{e}_2 plane is a height r above the flat surface), such that \mathbf{e}_3 is vertical (i.e. \mathbf{e}_3 is perpendicular to the flat surface) and points “upward” and away from the flat surface, and such that $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ forms a right-handed coordinate system. For simplicity, the spatial frame axes are chosen to be

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (5.1)$$

The acceleration due to gravity in the uniform gravitational field is $\mathbf{g} = -g\mathbf{e}_3 = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^\top$ in the spatial frame.

The body frame's origin is chosen to coincide with the position of m_0 's center of mass. The body frame has orthonormal axes \mathbf{E}_1 , \mathbf{E}_2 , and \mathbf{E}_3 , chosen to coincide with m_0 's principal axes, in which m_0 's inertia tensor \mathbb{I}_0 is diagonal, with corresponding principle moments of inertia d_1 , d_2 , and d_3 . That is, in this body frame the inertia tensor is the diagonal matrix

$$\mathbb{I}_0 = \begin{bmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{bmatrix}. \quad (5.2)$$

Moreover, \mathbf{E}_1 , \mathbf{E}_2 , and \mathbf{E}_3 are chosen so that $(\mathbf{E}_1, \mathbf{E}_2, \mathbf{E}_3)$ forms a right-handed coordinate system. For simplicity, the body frame axes are chosen to be

$$\mathbf{E}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{E}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{E}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (5.3)$$

In the spatial frame, the body frame is the moving frame $(\Lambda(t)\mathbf{E}_1, \Lambda(t)\mathbf{E}_2, \Lambda(t)\mathbf{E}_3)$, where $\Lambda(t) \in SO(3)$ defines the orientation (or attitude) of the ball at time t relative to its reference configuration, for example at some initial time. For $1 \leq i \leq n$, it is assumed that $\boldsymbol{\xi}_i(t)$, the position of m_i 's center of mass, is expressed with respect to the body frame. Since m_0 's center of mass is always $\mathbf{0} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$ in the body frame (by choice of that frame's origin), let $\boldsymbol{\xi}_0 \equiv \mathbf{0}$; with this definition, m_i 's center of mass is located at $\boldsymbol{\xi}_i(t)$ for all $0 \leq i \leq n$.

Let $\mathbf{z}_i(t)$ denote the position of m_i 's center of mass in the spatial frame. Let $\boldsymbol{\chi}_i(t)$ denote the body frame vector from the ball's geometric center to m_i 's center of mass. Then for m_0 , $\boldsymbol{\chi}_0$ is the constant vector from the ball's geometric center to m_0 's center of mass. Note that the position of m_i 's center of mass in the body frame is $\boldsymbol{\xi}_i(t) = \boldsymbol{\chi}_i(t) - \boldsymbol{\chi}_0$ and in the spatial frame is $\mathbf{z}_i(t) = \mathbf{z}_0(t) + \Lambda(t)\boldsymbol{\xi}_i(t) = \mathbf{z}_0(t) + \Lambda(t)[\boldsymbol{\chi}_i(t) - \boldsymbol{\chi}_0]$. In general, a particle with position $\mathbf{w}(t)$ in the body frame has position $\mathbf{z}(t) = \mathbf{z}_0(t) + \Lambda(t)\mathbf{w}(t)$ in the spatial frame and has position $\mathbf{w}(t) + \boldsymbol{\chi}_0$ in the body frame translated to the ball's geometric center.

For conciseness, the ball's geometric center is often denoted GC and m_0 's center of mass is often denoted CM. The GC is located at $\mathbf{z}_0(t) - \Lambda(t)\boldsymbol{\chi}_0$ in the spatial frame, at $-\boldsymbol{\chi}_0$ in the body frame, and at $\mathbf{0}$ in the body frame translated to the GC. The CM is located at $\mathbf{z}_0(t)$ in the spatial frame, at $\mathbf{0}$ in the body frame, and at $\boldsymbol{\chi}_0$ in the body frame translated to the GC.

For conciseness, the explicit time dependence of variables is often dropped. That is, the orientation of the ball at time t is denoted simply Λ rather than $\Lambda(t)$, the position of m_i 's center of mass in the spatial frame at time t is denoted \mathbf{z}_i rather than $\mathbf{z}_i(t)$, the position of m_i 's center of mass in the body frame at time t is denoted $\boldsymbol{\xi}_i$ rather than $\boldsymbol{\xi}_i(t)$, and the position of m_i 's center of mass in the body frame translated to the GC at time t is denoted $\boldsymbol{\chi}_i$ rather than $\boldsymbol{\chi}_i(t)$.

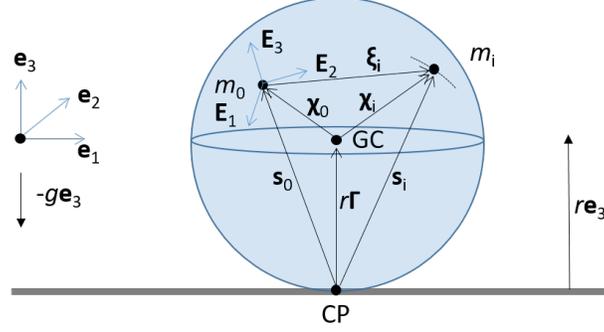


Figure 5.2: A ball of radius r and mass m_0 rolls without slipping on a flat surface in the presence of a uniform gravitational field of magnitude g . The ball's geometric center, center of mass, and contact point with the flat surface are denoted by GC , m_0 , and CP , respectively. The ball's motion is actuated by n point masses, each of mass m_i , $1 \leq i \leq n$, that move inside the ball. The spatial frame has origin located at height r above the flat surface and orthonormal axes \mathbf{e}_1 , \mathbf{e}_2 , and \mathbf{e}_3 . The body frame has origin located at the ball's center of mass (denoted by m_0) and orthonormal axes \mathbf{E}_1 , \mathbf{E}_2 , and \mathbf{E}_3 . All vectors inside the ball are expressed with respect to the body frame, while all vectors outside the ball are expressed with respect to the spatial frame.

5.B Uncontrolled Equations of Motion

The uncontrolled equations of motion for the rolling ball actuated by internal point masses, as depicted in Figure 5.2, are derived. Next, as special cases, the uncontrolled equations of motion for a rolling ball with static internal structure and the uncontrolled equations of motion for a rolling ball actuated by internal point masses that move along rails fixed within the ball are derived. Finally, as an even more special case, the uncontrolled equations of motion for a rolling disk actuated by internal point masses that move along rails fixed within the disk are derived.

5.B.1 Kinetic Energy, Potential Energy, and Lagrangian

As a first step to deriving the uncontrolled equations of motion for the rolling ball, the ball's kinetic and potential energies must be constructed, from which the ball's Lagrangian is easily constructed.

Kinetic Energy By definition, $\boldsymbol{\Omega} \equiv (\Lambda^{-1}\dot{\Lambda})^\vee$ is the ball's body angular velocity. m_0 's kinetic energy is the sum of its translational and rotational kinetic energy contributions about its center of mass:

$$T_0 = \frac{1}{2}m_0 |\dot{\mathbf{z}}_0|^2 + \frac{1}{2} \langle \boldsymbol{\Omega}, \mathbb{I}_0 \boldsymbol{\Omega} \rangle. \quad (5.4)$$

For $1 \leq i \leq n$, since m_i is a point mass, m_i 's kinetic energy is just its translational kinetic energy:

$$T_i = \frac{1}{2}m_i |\dot{\mathbf{z}}_i|^2. \quad (5.5)$$

But to be consistent with m_0 's kinetic energy formula, for $1 \leq i \leq n$, m_i 's inertia tensor will be defined to be zero so that $\mathbb{I}_i \equiv 0$ and so that m_i 's kinetic energy is

$$T_i = \frac{1}{2} m_i |\dot{\mathbf{z}}_i|^2 + \frac{1}{2} \langle \boldsymbol{\Omega}, \mathbb{I}_i \boldsymbol{\Omega} \rangle. \quad (5.6)$$

Define $\mathbf{Y}_i = \Lambda^{-1} \dot{\mathbf{z}}_i$. Since $|\dot{\mathbf{z}}_i|^2 = |\Lambda^{-1} \dot{\mathbf{z}}_i|^2 = |\mathbf{Y}_i|^2$, m_i 's kinetic energy becomes

$$T_i = \frac{1}{2} m_i |\mathbf{Y}_i|^2 + \frac{1}{2} \langle \boldsymbol{\Omega}, \mathbb{I}_i \boldsymbol{\Omega} \rangle. \quad (5.7)$$

Thus, the ball's kinetic energy is

$$\begin{aligned} T &= \sum_{i=0}^n T_i = \sum_{i=0}^n \left[\frac{1}{2} m_i |\mathbf{Y}_i|^2 + \frac{1}{2} \langle \boldsymbol{\Omega}, \mathbb{I}_i \boldsymbol{\Omega} \rangle \right] = \frac{1}{2} \sum_{i=0}^n m_i |\mathbf{Y}_i|^2 + \frac{1}{2} \sum_{i=0}^n \langle \boldsymbol{\Omega}, \mathbb{I}_i \boldsymbol{\Omega} \rangle \\ &= \frac{1}{2} \sum_{i=0}^n m_i |\mathbf{Y}_i|^2 + \frac{1}{2} \left\langle \boldsymbol{\Omega}, \sum_{i=0}^n \mathbb{I}_i \boldsymbol{\Omega} \right\rangle \\ &= \frac{1}{2} \sum_{i=0}^n m_i |\mathbf{Y}_i|^2 + \frac{1}{2} \langle \boldsymbol{\Omega}, \mathbb{I} \boldsymbol{\Omega} \rangle, \end{aligned} \quad (5.8)$$

where $\mathbb{I} \equiv \sum_{i=0}^n \mathbb{I}_i = \mathbb{I}_0$ and

$$\mathbb{I} = \mathbb{I}_0 = \begin{bmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{bmatrix}. \quad (5.9)$$

Potential Energy The potential energy due to mass m_i is $V_i = m_i g \langle \boldsymbol{\chi}_i, \boldsymbol{\Gamma} \rangle$, where $\boldsymbol{\Gamma} = \Lambda^{-1} \mathbf{e}_3$. Thus, the ball's potential energy is

$$V = \sum_{i=0}^n V_i = \sum_{i=0}^n m_i g \langle \boldsymbol{\chi}_i, \boldsymbol{\Gamma} \rangle = g \left\langle \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \boldsymbol{\Gamma} \right\rangle. \quad (5.10)$$

Lagrangian Since the spatial position of m_i 's center of mass is $\mathbf{z}_i = \mathbf{z}_0 + \Lambda [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0]$, the spatial velocity of m_i 's center of mass is $\dot{\mathbf{z}}_i = \dot{\mathbf{z}}_0 + \dot{\Lambda} [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \Lambda \dot{\boldsymbol{\chi}}_i$. Hence,

$$\begin{aligned} \mathbf{Y}_i &= \Lambda^{-1} \dot{\mathbf{z}}_i = \Lambda^{-1} \left[\dot{\mathbf{z}}_0 + \dot{\Lambda} [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \Lambda \dot{\boldsymbol{\chi}}_i \right] = \Lambda^{-1} \dot{\mathbf{z}}_0 + \Lambda^{-1} \dot{\Lambda} [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i \\ &= \mathbf{Y}_0 + \widehat{\boldsymbol{\Omega}} [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i \\ &= \mathbf{Y}_0 + \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i. \end{aligned} \quad (5.11)$$

The ball's Lagrangian is the difference between its kinetic and potential energies:

$$l = T - V = \frac{1}{2} \sum_{i=0}^n m_i |\mathbf{Y}_i|^2 + \frac{1}{2} \langle \boldsymbol{\Omega}, \mathbb{I} \boldsymbol{\Omega} \rangle - g \left\langle \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \boldsymbol{\Gamma} \right\rangle. \quad (5.12)$$

Since \mathbf{Y}_i can be expressed as a function of \mathbf{Y}_0 and $\boldsymbol{\Omega}$ for $1 \leq i \leq n$, note that the ball's Lagrangian should be expressed as $l(\boldsymbol{\Omega}, \mathbf{Y}_0, \boldsymbol{\Gamma})$, but this functional dependence is suppressed for the sake of conciseness.

5.B.2 Rolling Constraint and Lagrange-d'Alembert's Principle

Having constructed the rolling ball's Lagrangian, the variation of the action integral is now computed, taking into consideration the rolling constraint and Lagrange-d'Alembert's principle.

Rolling Constraint Recall that it is assumed that the ball rolls along the flat surface without slipping. The vector pointing from the contact point (i.e. the point on the flat surface touching the bottom of the ball) to m_0 's center of mass (located at \mathbf{z}_0 in the spatial frame and at $\boldsymbol{\xi}_0 \equiv \mathbf{0}$ in the body frame) is

$$\boldsymbol{\sigma}_0 = r\mathbf{e}_3 + \Lambda\boldsymbol{\chi}_0, \quad (5.13)$$

in the spatial frame and is

$$\mathbf{s}_0 = \Lambda^{-1}\boldsymbol{\sigma}_0 = r\Lambda^{-1}\mathbf{e}_3 + \boldsymbol{\chi}_0 = r\boldsymbol{\Gamma} + \boldsymbol{\chi}_0, \quad (5.14)$$

in the body frame. Differentiating (5.14) with respect to time, using the identity $\dot{\boldsymbol{\Gamma}} = \boldsymbol{\Gamma} \times \boldsymbol{\Omega}$, and using the identity $-r\boldsymbol{\Gamma} = \boldsymbol{\chi}_0 - \mathbf{s}_0$, which follows trivially from (5.14), yields the following useful result:

$$\dot{\mathbf{s}}_0 = r\dot{\boldsymbol{\Gamma}} = r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} = \boldsymbol{\Omega} \times (-r\boldsymbol{\Gamma}) = \boldsymbol{\Omega} \times (\boldsymbol{\chi}_0 - \mathbf{s}_0). \quad (5.15)$$

Another useful result that follows trivially from (5.14) is

$$\mathbf{s}_0 \times \boldsymbol{\Gamma} = (r\boldsymbol{\Gamma} + \boldsymbol{\chi}_0) \times \boldsymbol{\Gamma} = r\boldsymbol{\Gamma} \times \boldsymbol{\Gamma} + \boldsymbol{\chi}_0 \times \boldsymbol{\Gamma} = \boldsymbol{\chi}_0 \times \boldsymbol{\Gamma}. \quad (5.16)$$

The rolling constraint is imposed by stipulating

$$\dot{\mathbf{z}}_0 = \dot{\Lambda}\mathbf{s}_0 = \dot{\Lambda}\Lambda^{-1}\boldsymbol{\sigma}_0 = \widehat{\boldsymbol{\omega}}\boldsymbol{\sigma}_0 = \boldsymbol{\omega} \times \boldsymbol{\sigma}_0, \quad (5.17)$$

where $\widehat{\boldsymbol{\omega}} \equiv \dot{\Lambda}\Lambda^{-1} \in \mathfrak{so}(3)$, or equivalently, by stipulating

$$\mathbf{Y}_0 = \Lambda^{-1}\dot{\mathbf{z}}_0 = \Lambda^{-1}\dot{\Lambda}\mathbf{s}_0 = \widehat{\boldsymbol{\Omega}}\mathbf{s}_0 = \boldsymbol{\Omega} \times \mathbf{s}_0. \quad (5.18)$$

As a consequence of the rolling constraint (5.18),

$$\mathbf{Y}_i = \mathbf{Y}_0 + \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i = \boldsymbol{\Omega} \times \mathbf{s}_0 + \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i = \boldsymbol{\Omega} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i. \quad (5.19)$$

Lagrange-d'Alembert's Principle Since $\boldsymbol{\Psi} \equiv \Lambda^{-1}\delta\mathbf{z}_0$,

$$\begin{aligned} \dot{\boldsymbol{\Psi}} &= [\Lambda^{-1}\delta\mathbf{z}_0]^\cdot = [\Lambda^{-1}]^\cdot \delta\mathbf{z}_0 + \Lambda^{-1}[\delta\mathbf{z}_0]^\cdot = -\Lambda^{-1}\dot{\Lambda}\Lambda^{-1}\delta\mathbf{z}_0 + \Lambda^{-1}\delta\dot{\mathbf{z}}_0 \\ &= -\widehat{\boldsymbol{\Omega}}\boldsymbol{\Psi} + \Lambda^{-1}\delta\dot{\mathbf{z}}_0 \\ &= -\boldsymbol{\Omega} \times \boldsymbol{\Psi} + \Lambda^{-1}\delta\dot{\mathbf{z}}_0. \end{aligned} \quad (5.20)$$

Hence $\Lambda^{-1}\delta\dot{\mathbf{z}}_0 = \dot{\boldsymbol{\Psi}} + \boldsymbol{\Omega} \times \boldsymbol{\Psi}$.

Since $\mathbf{Y}_0 = \Lambda^{-1}\dot{\mathbf{z}}_0$,

$$\begin{aligned}
\delta\mathbf{Y}_0 &= \delta[\Lambda^{-1}\dot{\mathbf{z}}_0] = [\delta(\Lambda^{-1})]\dot{\mathbf{z}}_0 + \Lambda^{-1}\delta\dot{\mathbf{z}}_0 = -\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\mathbf{z}}_0 + \Lambda^{-1}\delta\dot{\mathbf{z}}_0 \\
&= -\widehat{\Sigma}\mathbf{Y}_0 + \Lambda^{-1}\delta\dot{\mathbf{z}}_0 \\
&= -\Sigma \times \mathbf{Y}_0 + \Lambda^{-1}\delta\dot{\mathbf{z}}_0 \\
&= \dot{\Psi} + \Omega \times \Psi - \Sigma \times \mathbf{Y}_0,
\end{aligned} \tag{5.21}$$

where $\widehat{\Sigma} \equiv \Lambda^{-1}\delta\Lambda \in \mathfrak{so}(3)$.

Since $\mathbf{Y}_i = \mathbf{Y}_0 + \Omega \times [\chi_i - \chi_0] + \dot{\chi}_i$ and since the control masses move along prescribed trajectories $\{\chi_i\}_{i=0}^n$ (so that the variation of \mathbf{Y}_i is computed with respect to \mathbf{Y}_0 and Ω , but not with respect to $\{\chi_i\}_{i=0}^n$),

$$\delta\mathbf{Y}_i = \delta\mathbf{Y}_0 + \delta\Omega \times [\chi_i - \chi_0]. \tag{5.22}$$

Since $\widehat{\Sigma} = \Lambda^{-1}\delta\Lambda$,

$$\dot{\widehat{\Sigma}} = [\Lambda^{-1}\delta\Lambda]' = [\Lambda^{-1}]' \delta\Lambda + \Lambda^{-1}[\delta\Lambda]' = -\Lambda^{-1}\dot{\Lambda}\Lambda^{-1}\delta\Lambda + \Lambda^{-1}\delta\dot{\Lambda} = -\widehat{\Omega}\widehat{\Sigma} + \Lambda^{-1}\delta\dot{\Lambda}. \tag{5.23}$$

Hence $\Lambda^{-1}\delta\dot{\Lambda} = \dot{\widehat{\Sigma}} + \widehat{\Omega}\widehat{\Sigma}$.

Since $\widehat{\Omega} = \Lambda^{-1}\dot{\Lambda}$,

$$\begin{aligned}
\delta\widehat{\Omega} &= \delta[\Lambda^{-1}\dot{\Lambda}] = [\delta(\Lambda^{-1})]\dot{\Lambda} + \Lambda^{-1}\delta\dot{\Lambda} = -\Lambda^{-1}\delta\Lambda\Lambda^{-1}\dot{\Lambda} + \Lambda^{-1}\delta\dot{\Lambda} = -\widehat{\Sigma}\widehat{\Omega} + \dot{\widehat{\Sigma}} + \widehat{\Omega}\widehat{\Sigma} \\
&= \dot{\widehat{\Sigma}} + [\Omega \times \Sigma]^\wedge \\
&= [\dot{\Sigma} + \Omega \times \Sigma]^\wedge.
\end{aligned} \tag{5.24}$$

Hence

$$\delta\Omega = \dot{\Sigma} + \Omega \times \Sigma. \tag{5.25}$$

Since $\Gamma = \Lambda^{-1}\mathbf{e}_3$,

$$\delta\Gamma = [\delta(\Lambda^{-1})]\mathbf{e}_3 = -\Lambda^{-1}\delta\Lambda\Lambda^{-1}\mathbf{e}_3 = -\widehat{\Sigma}\Gamma = -\Sigma \times \Gamma = \Gamma \times \Sigma. \tag{5.26}$$

Recall Lagrange-d'Alembert's principle from Section 2.B. Part of Lagrange-d'Alembert's principle stipulates that due to the rolling constraint (5.17), which says $\dot{\mathbf{z}}_0 = \dot{\Lambda}\mathbf{s}_0$, the variations of \mathbf{z}_0 must have the form $\delta\mathbf{z}_0 = \delta\Lambda\mathbf{s}_0$. Hence, the variations $\Psi = \Lambda^{-1}\delta\mathbf{z}_0$ must take on the following form (as a consequence of the rolling constraint (5.17) and Lagrange-d'Alembert's principle):

$$\Psi = \Lambda^{-1}\delta\mathbf{z}_0 = \Lambda^{-1}\delta\Lambda\mathbf{s}_0 = \widehat{\Sigma}\mathbf{s}_0 = \Sigma \times \mathbf{s}_0. \tag{5.27}$$

The uncontrolled equations of motion are derived here and in the next section from Lagrange-d'Alembert's principle. Recalling that the control masses move along prescribed trajectories $\{\chi_i\}_{i=0}^n$, it is important to

keep in mind that the variation of the action integral is computed with respect to $\{\mathbf{Y}_i\}_{i=0}^n$, $\boldsymbol{\Omega}$, and $\boldsymbol{\Gamma}$, but **not** with respect to $\{\boldsymbol{\chi}_i\}_{i=0}^n$. Once the variation of the action integral is computed, tedious calculations are performed to isolate $\boldsymbol{\Sigma}$, after which the variation of the action integral is equated to zero in order to obtain the uncontrolled equations of motion. Key points in the calculations are: 1) the variations $\boldsymbol{\Psi}$ and $\boldsymbol{\Sigma}$ must satisfy $\boldsymbol{\Psi} = \boldsymbol{\Sigma} \times \mathbf{s}_0$ (5.27), which enforces the constraints on the variations demanded by Lagrange-d'Alembert's principle, and 2) the variation $\boldsymbol{\Sigma}$ must also satisfy $\boldsymbol{\Sigma}(a) = \boldsymbol{\Sigma}(b) = \mathbf{0}$, which enforces the vanishing endpoint constraints. To begin the calculations, the variation of the action integral is computed as

$$\delta S = \delta \int_a^b l dt = \int_a^b \delta l dt = \int_a^b \left[\sum_{i=0}^n m_i \langle \mathbf{Y}_i, \delta \mathbf{Y}_i \rangle + \langle \mathbb{I}\boldsymbol{\Omega}, \delta \boldsymbol{\Omega} \rangle - g \left\langle \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \delta \boldsymbol{\Gamma} \right\rangle \right] dt. \quad (5.28)$$

Using the identity $\delta \mathbf{Y}_i = \delta \mathbf{Y}_0 + \delta \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0]$ (5.22), the variation of the action integral obtained in (5.28) becomes

$$\begin{aligned} \delta S &= \int_a^b \left[\sum_{i=0}^n m_i \langle \mathbf{Y}_i, \delta \mathbf{Y}_0 + \delta \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \rangle + \langle \mathbb{I}\boldsymbol{\Omega}, \delta \boldsymbol{\Omega} \rangle - g \left\langle \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \delta \boldsymbol{\Gamma} \right\rangle \right] dt \\ &= \int_a^b \left[\sum_{i=0}^n m_i \langle \mathbf{Y}_i, \delta \mathbf{Y}_0 \rangle + \left\langle \mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i, \delta \boldsymbol{\Omega} \right\rangle - g \left\langle \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \delta \boldsymbol{\Gamma} \right\rangle \right] dt. \end{aligned} \quad (5.29)$$

Using the identities $\delta \mathbf{Y}_0 = \dot{\boldsymbol{\Psi}} + \boldsymbol{\Omega} \times \boldsymbol{\Psi} - \boldsymbol{\Sigma} \times \mathbf{Y}_0$ (5.21), $\delta \boldsymbol{\Omega} = \dot{\boldsymbol{\Sigma}} + \boldsymbol{\Omega} \times \boldsymbol{\Sigma}$ (5.25), and $\delta \boldsymbol{\Gamma} = \boldsymbol{\Gamma} \times \boldsymbol{\Sigma}$ (5.26), the variation of the action integral obtained in (5.29) becomes

$$\begin{aligned} \delta S &= \int_a^b \left[\sum_{i=0}^n m_i \langle \mathbf{Y}_i, \dot{\boldsymbol{\Psi}} + \boldsymbol{\Omega} \times \boldsymbol{\Psi} - \boldsymbol{\Sigma} \times \mathbf{Y}_0 \rangle + \left\langle \mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i, \dot{\boldsymbol{\Sigma}} + \boldsymbol{\Omega} \times \boldsymbol{\Sigma} \right\rangle \right. \\ &\quad \left. + g \left\langle \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \boldsymbol{\Sigma} \times \boldsymbol{\Gamma} \right\rangle \right] dt. \end{aligned} \quad (5.30)$$

Integrating by parts, the variation of the action integral obtained in (5.30) becomes

$$\begin{aligned} \delta S &= \int_a^b \left[\sum_{i=0}^n m_i \left[- \left\langle \left(\frac{d}{dt} + \boldsymbol{\Omega} \times \right) \mathbf{Y}_i, \boldsymbol{\Psi} \right\rangle + \langle \mathbf{Y}_i \times \mathbf{Y}_0, \boldsymbol{\Sigma} \rangle \right] \right. \\ &\quad \left. - \left\langle \left(\frac{d}{dt} + \boldsymbol{\Omega} \times \right) \left[\mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i \right], \boldsymbol{\Sigma} \right\rangle + g \left\langle \boldsymbol{\Gamma} \times \sum_{i=0}^n m_i \boldsymbol{\chi}_i, \boldsymbol{\Sigma} \right\rangle \right] dt \\ &\quad + \sum_{i=0}^n m_i \langle \mathbf{Y}_i, \boldsymbol{\Psi} \rangle \Big|_a^b + \left\langle \mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i, \boldsymbol{\Sigma} \right\rangle \Big|_a^b \\ &= \int_a^b \left[- \sum_{i=0}^n m_i \left\langle \left(\frac{d}{dt} + \boldsymbol{\Omega} \times \right) \mathbf{Y}_i, \boldsymbol{\Psi} \right\rangle \right. \\ &\quad \left. + \left\langle - \left(\frac{d}{dt} + \boldsymbol{\Omega} \times \right) \left[\mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i \right] + \sum_{i=0}^n m_i (\mathbf{Y}_i \times \mathbf{Y}_0 + g \boldsymbol{\Gamma} \times \boldsymbol{\chi}_i), \boldsymbol{\Sigma} \right\rangle \right] dt \\ &\quad + \sum_{i=0}^n m_i \langle \mathbf{Y}_i, \boldsymbol{\Psi} \rangle \Big|_a^b + \left\langle \mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i, \boldsymbol{\Sigma} \right\rangle \Big|_a^b. \end{aligned} \quad (5.31)$$

Next, (5.31) is evaluated on the constraint distribution given by $\boldsymbol{\Psi} = \boldsymbol{\Sigma} \times \mathbf{s}_0$ (5.27), and the boundary terms

in (5.31) are eliminated since Σ is a variation such that $\Sigma(a) = \Sigma(b) = \mathbf{0}$. With these manipulations, the variation of the action integral obtained in (5.31) becomes

$$\begin{aligned}
\delta S &= \int_a^b \left[- \left\langle \left(\frac{d}{dt} + \Omega \times \right) \left[\sum_{i=0}^n m_i \mathbf{Y}_i \right], \Sigma \times \mathbf{s}_0 \right\rangle \right. \\
&\quad \left. + \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I}\Omega + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i \right] + \sum_{i=0}^n m_i (\mathbf{Y}_i \times \mathbf{Y}_0 + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i), \Sigma \right\rangle \right] dt \\
&= \int_a^b \left\langle \left(\left(\frac{d}{dt} + \Omega \times \right) \left[\sum_{i=0}^n m_i \mathbf{Y}_i \right] \right) \times \mathbf{s}_0 - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I}\Omega + \sum_{i=0}^n m_i [\boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i \right] \right. \\
&\quad \left. + \sum_{i=0}^n m_i (\mathbf{Y}_i \times \mathbf{Y}_0 + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i), \Sigma \right\rangle dt \\
&= \int_a^b \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I}\Omega + \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i \right] \right. \\
&\quad \left. + \sum_{i=0}^n m_i \left(\mathbf{Y}_i \times \left(\mathbf{Y}_0 - \left(\frac{d}{dt} + \Omega \times \right) \mathbf{s}_0 \right) + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i \right), \Sigma \right\rangle dt.
\end{aligned} \tag{5.32}$$

Using the identity $\mathbf{Y}_0 = \Omega \times \mathbf{s}_0$ (5.18), the variation of the action integral obtained in (5.32) becomes

$$\delta S = \int_a^b \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I}\Omega + \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \mathbf{Y}_i \right] + \sum_{i=0}^n m_i (\dot{\mathbf{s}}_0 \times \mathbf{Y}_i + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i), \Sigma \right\rangle dt. \tag{5.33}$$

Finally, using the identities $\mathbf{Y}_i = \Omega \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i$ (5.19) and $\dot{\mathbf{s}}_0 = \Omega \times (\boldsymbol{\chi}_0 - \mathbf{s}_0)$ (5.15), the variation of the action integral obtained in (5.33) becomes

$$\begin{aligned}
\delta S &= \int_a^b \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I}\Omega + \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times [\Omega \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \right] \right. \\
&\quad \left. + \sum_{i=0}^n m_i ([(\mathbf{s}_0 - \boldsymbol{\chi}_0) \times \Omega] \times [\Omega \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i), \Sigma \right\rangle dt \\
&= \int_a^b \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I}\Omega + \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times [\Omega \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \right] \right. \\
&\quad \left. + \sum_{i=0}^n m_i ([(\mathbf{s}_0 - \boldsymbol{\chi}_0) \times \Omega] \times [\Omega \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i), \Sigma \right\rangle dt.
\end{aligned} \tag{5.34}$$

Note the order in which the operations were performed: first variations and simplifications were computed in (5.28)-(5.31), followed by evaluation of the result (5.31) on the constraint distribution (5.27) to obtain (5.32); preserving this order is key to the correct application of Lagrange-d'Alembert's principle.

Now suppose a time-varying external force \mathbf{F}_e acts at the ball's geometric center. For example, this force might be due to the wind blowing on the ball when the ball rolls around outdoors. If the ball's geometric center in the spatial frame is \mathbf{z}_{GC} , then the rolling constraint says that $\dot{\mathbf{z}}_{GC} = \dot{\Lambda} \Lambda^{-1} \mathbf{r} \mathbf{e}_3$ and Lagrange-d'Alembert's principle says that $\delta \mathbf{z}_{GC} = \delta \Lambda \Lambda^{-1} \mathbf{r} \mathbf{e}_3$. Application of the external force yields a new variation of the action integral, $\delta S_1 = \delta S + \int_a^b \langle \mathbf{F}_e, \delta \mathbf{z}_{GC} \rangle dt$ using Lagrange-d'Alembert's principle for incorporating external forces into the variational principle. Performing calculations on the new variation of the action

integral to isolate Σ gives:

$$\begin{aligned}
\delta S_1 &= \delta S + \int_a^b \langle \mathbf{F}_e, \delta \mathbf{z}_{GC} \rangle dt = \delta S + \int_a^b \langle \Lambda^{-1} \mathbf{F}_e, \Lambda^{-1} \delta \mathbf{z}_{GC} \rangle dt \\
&= \delta S + \int_a^b \langle \Lambda^{-1} \mathbf{F}_e, \Lambda^{-1} \delta \Lambda \Lambda^{-1} r \mathbf{e}_3 \rangle dt = \delta S + \int_a^b \langle \tilde{\Gamma}, \widehat{\Sigma} r \Gamma \rangle dt \\
&= \delta S + \int_a^b \langle \tilde{\Gamma}, \Sigma \times r \Gamma \rangle dt = \delta S + \int_a^b \langle r \Gamma \times \tilde{\Gamma}, \Sigma \rangle dt \\
&= \int_a^b \left\langle - \left(\frac{d}{dt} + \Omega \times \right) \left[\mathbb{I} \Omega + \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times [\Omega \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \right] \right. \\
&\quad \left. + \sum_{i=0}^n m_i ([(\mathbf{s}_0 - \boldsymbol{\chi}_0) \times \Omega] \times [\Omega \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] + g \Gamma \times \boldsymbol{\chi}_i) + r \Gamma \times \tilde{\Gamma}, \Sigma \right\rangle dt.
\end{aligned} \tag{5.35}$$

In the fourth equality, the definitions $\tilde{\Gamma} \equiv \Lambda^{-1} \mathbf{F}_e$, $\widehat{\Sigma} \equiv \Lambda^{-1} \delta \Lambda$, and $\Gamma \equiv \Lambda^{-1} \mathbf{e}_3$ are used. In the final equality, the simplification of δS calculated in (5.34) is used.

5.B.3 Uncontrolled Equations of Motion for the Rolling Ball

Having computed the variation of the action integral according to Lagrange-d'Alembert's principle, the uncontrolled equations of motion for the rolling ball actuated by internal point masses are obtained now. In addition, the uncontrolled equations of motion for two important special cases, a ball with static internal structure and a ball with 1-d parameterized control rails, are derived.

Uncontrolled Equations of Motion for the Rolling Ball Actuated by Internal Point Masses

Insisting that the variation δS_1 of the action integral in (5.35) is zero for all variations Σ (i.e. completing the application of Lagrange-d'Alembert principle's by letting $0 = \delta S_1$), the following uncontrolled equations

of motion are obtained:

$$\begin{aligned}
\mathbf{0} &= -\left(\frac{d}{dt} + \boldsymbol{\Omega} \times\right) \left[\mathbb{I}\boldsymbol{\Omega} + \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times [\boldsymbol{\Omega} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \right] \\
&\quad + \sum_{i=0}^n m_i \left([(\mathbf{s}_0 - \boldsymbol{\chi}_0) \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] + g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} \right) \\
&= -\mathbb{I}\dot{\boldsymbol{\Omega}} - \sum_{i=0}^n m_i [\dot{\mathbf{s}}_0 + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \\
&\quad - \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \left[\dot{\boldsymbol{\Omega}} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \boldsymbol{\Omega} \times [\dot{\mathbf{s}}_0 + \dot{\boldsymbol{\chi}}_i] + \ddot{\boldsymbol{\chi}}_i \right] \\
&\quad - \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} - \sum_{i=0}^n m_i \boldsymbol{\Omega} \times \{ [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times [\boldsymbol{\Omega} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \} \\
&\quad + \sum_{i=0}^n m_i \left([(\mathbf{s}_0 - \boldsymbol{\chi}_0) \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] + g \sum_{i=0}^n m_i \boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} \right) \\
&= -\mathbb{I}\dot{\boldsymbol{\Omega}} - \sum_{i=0}^n m_i [\boldsymbol{\Omega} \times (\boldsymbol{\chi}_0 - \mathbf{s}_0) + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \\
&\quad - \sum_{i=0}^n m_i [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times \left[\dot{\boldsymbol{\Omega}} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \boldsymbol{\Omega} \times [\boldsymbol{\Omega} \times (\boldsymbol{\chi}_0 - \mathbf{s}_0) + \dot{\boldsymbol{\chi}}_i] + \ddot{\boldsymbol{\chi}}_i \right] \\
&\quad - \sum_{i=0}^n m_i \boldsymbol{\Omega} \times \{ [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] \times [\boldsymbol{\Omega} \times [\mathbf{s}_0 + \boldsymbol{\chi}_i - \boldsymbol{\chi}_0] + \dot{\boldsymbol{\chi}}_i] \} \\
&\quad + \sum_{i=0}^n m_i \left([(\mathbf{s}_0 - \boldsymbol{\chi}_0) \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] - \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} + g \sum_{i=0}^n m_i \boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} \right).
\end{aligned} \tag{5.36}$$

Negating both sides of (5.36), rearranging terms, and using the identity $r\boldsymbol{\Gamma} = \mathbf{s}_0 - \boldsymbol{\chi}_0$, the uncontrolled equations of motion are

$$\begin{aligned}
\mathbf{0} &= \mathbb{I}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} - r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} - g \sum_{i=0}^n m_i \boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + \sum_{i=0}^n m_i [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times [r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i] + \dot{\boldsymbol{\chi}}_i] \\
&\quad + \sum_{i=0}^n m_i [r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i] \times \left\{ \dot{\boldsymbol{\Omega}} \times [r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i] + \boldsymbol{\Omega} \times [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] + \ddot{\boldsymbol{\chi}}_i \right\} \\
&\quad + \sum_{i=0}^n m_i \boldsymbol{\Omega} \times \{ [r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i] \times [\boldsymbol{\Omega} \times [r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i] + \dot{\boldsymbol{\chi}}_i] \} - \sum_{i=0}^n m_i [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i].
\end{aligned} \tag{5.37}$$

By defining $\mathbf{s}_i \equiv r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i$ for $0 \leq i \leq n$ and combining the summations, the uncontrolled equations of motion are

$$\begin{aligned}
\mathbf{0} &= \mathbb{I}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} - r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} + \sum_{i=0}^n m_i \left\{ -g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times \mathbf{s}_i + \dot{\boldsymbol{\chi}}_i] \right. \\
&\quad \left. + \mathbf{s}_i \times \left\{ \dot{\boldsymbol{\Omega}} \times \mathbf{s}_i + \boldsymbol{\Omega} \times [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] + \ddot{\boldsymbol{\chi}}_i \right\} \right. \\
&\quad \left. + \boldsymbol{\Omega} \times \{ \mathbf{s}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i + \dot{\boldsymbol{\chi}}_i] \} - [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] \right\}.
\end{aligned} \tag{5.38}$$

Since $\mathbf{s}_i \equiv r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i$ for $0 \leq i \leq n$,

$$\begin{aligned}
& [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times \mathbf{s}_i + \dot{\boldsymbol{\chi}}_i] \\
& - [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + \dot{\boldsymbol{\chi}}_i] = [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times \mathbf{s}_i + \dot{\boldsymbol{\chi}}_i] \\
& \quad - [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \{\mathbf{s}_i - r\boldsymbol{\Gamma}\} + \dot{\boldsymbol{\chi}}_i] \\
& = [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega} + \dot{\boldsymbol{\chi}}_i] \times [\boldsymbol{\Omega} \times \mathbf{s}_i + \dot{\boldsymbol{\chi}}_i] - [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}] \times [\boldsymbol{\Omega} \times \mathbf{s}_i + \dot{\boldsymbol{\chi}}_i] \\
& = \dot{\boldsymbol{\chi}}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i].
\end{aligned} \tag{5.39}$$

Moreover, by exploiting the Jacobi identity,

$$\dot{\boldsymbol{\chi}}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i] + \mathbf{s}_i \times [\boldsymbol{\Omega} \times \dot{\boldsymbol{\chi}}_i] + \boldsymbol{\Omega} \times [\mathbf{s}_i \times \dot{\boldsymbol{\chi}}_i] = 2\mathbf{s}_i \times [\boldsymbol{\Omega} \times \dot{\boldsymbol{\chi}}_i]. \tag{5.40}$$

By using (5.39) and (5.40) in (5.38), the uncontrolled equations of motion simplify to

$$\begin{aligned}
\mathbf{0} &= \mathbb{I}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} - r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} + \sum_{i=0}^n m_i \left\{ -g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + 2\mathbf{s}_i \times [\boldsymbol{\Omega} \times \dot{\boldsymbol{\chi}}_i] \right. \\
& \quad \left. + \mathbf{s}_i \times \left\{ \dot{\boldsymbol{\Omega}} \times \mathbf{s}_i + \boldsymbol{\Omega} \times [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}] + \ddot{\boldsymbol{\chi}}_i \right\} + \boldsymbol{\Omega} \times \{\mathbf{s}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i]\} \right\}.
\end{aligned} \tag{5.41}$$

Given vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$, $\mathbf{a} \times \{\mathbf{b} \times [\mathbf{a} \times \mathbf{b}]\} = \mathbf{a} \times \{(\mathbf{b} \cdot \mathbf{b})\mathbf{a} - (\mathbf{a} \cdot \mathbf{b})\mathbf{b}\} = -(\mathbf{a} \cdot \mathbf{b})\mathbf{a} \times \mathbf{b}$. This gives the identity

$$\mathbf{a} \times \{\mathbf{b} \times [\mathbf{a} \times \mathbf{b}]\} + \mathbf{b} \times \{\mathbf{a} \times [\mathbf{b} \times \mathbf{a}]\} = -(\mathbf{a} \cdot \mathbf{b})\mathbf{a} \times \mathbf{b} - (\mathbf{a} \cdot \mathbf{b})\mathbf{b} \times \mathbf{a} = 0. \tag{5.42}$$

Since $\mathbf{s}_i \equiv r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i$ for $0 \leq i \leq n$ and using the identity (5.42),

$$\begin{aligned}
\mathbf{s}_i \times \{\boldsymbol{\Omega} \times [r\boldsymbol{\Gamma} \times \boldsymbol{\Omega}]\} + \boldsymbol{\Omega} \times \{\mathbf{s}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i]\} &= \mathbf{s}_i \times \{\boldsymbol{\Omega} \times [(\mathbf{s}_i - \boldsymbol{\chi}_i) \times \boldsymbol{\Omega}]\} + \boldsymbol{\Omega} \times \{\mathbf{s}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i]\} \\
&= \mathbf{s}_i \times \{\boldsymbol{\Omega} \times [\mathbf{s}_i \times \boldsymbol{\Omega}]\} - \mathbf{s}_i \times \{\boldsymbol{\Omega} \times [\boldsymbol{\chi}_i \times \boldsymbol{\Omega}]\} \\
& \quad + \boldsymbol{\Omega} \times \{\mathbf{s}_i \times [\boldsymbol{\Omega} \times \mathbf{s}_i]\} \\
&= -\mathbf{s}_i \times \{\boldsymbol{\Omega} \times [\boldsymbol{\chi}_i \times \boldsymbol{\Omega}]\}.
\end{aligned} \tag{5.43}$$

Using (5.43) in (5.41), the uncontrolled equations of motion simplify to

$$\begin{aligned}
\mathbf{0} &= \mathbb{I}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} - r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} + \sum_{i=0}^n m_i \left\{ -g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i + 2\mathbf{s}_i \times [\boldsymbol{\Omega} \times \dot{\boldsymbol{\chi}}_i] \right. \\
& \quad \left. + \mathbf{s}_i \times \left\{ \dot{\boldsymbol{\Omega}} \times \mathbf{s}_i - \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i \times \boldsymbol{\Omega}] + \ddot{\boldsymbol{\chi}}_i \right\} \right\}.
\end{aligned} \tag{5.44}$$

Since $\mathbf{s}_i \equiv r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i$ for $0 \leq i \leq n$,

$$-g\boldsymbol{\Gamma} \times \boldsymbol{\chi}_i = g\boldsymbol{\chi}_i \times \boldsymbol{\Gamma} = g(\mathbf{s}_i - r\boldsymbol{\Gamma}) \times \boldsymbol{\Gamma} = g\mathbf{s}_i \times \boldsymbol{\Gamma} = \mathbf{s}_i \times (g\boldsymbol{\Gamma}). \tag{5.45}$$

Using (5.45) in (5.44), the uncontrolled equations of motion become

$$\begin{aligned}
\mathbf{0} &= \mathbb{I}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} - r\boldsymbol{\Gamma} \times \tilde{\boldsymbol{\Gamma}} + \sum_{i=0}^n m_i \left\{ \mathbf{s}_i \times (g\boldsymbol{\Gamma}) + 2\mathbf{s}_i \times [\boldsymbol{\Omega} \times \dot{\boldsymbol{\chi}}_i] \right. \\
& \quad \left. + \mathbf{s}_i \times \left\{ \dot{\boldsymbol{\Omega}} \times \mathbf{s}_i - \boldsymbol{\Omega} \times [\boldsymbol{\chi}_i \times \boldsymbol{\Omega}] + \ddot{\boldsymbol{\chi}}_i \right\} \right\}.
\end{aligned} \tag{5.46}$$

Combining terms and eliminating minus signs by re-ordering cross product pairings in (5.46), the uncontrolled equations of motion simplify further to

$$\mathbf{0} = \mathbb{I}\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} + r\tilde{\boldsymbol{\Gamma}} \times \boldsymbol{\Gamma} + \sum_{i=0}^n m_i \mathbf{s}_i \times \left\{ g\boldsymbol{\Gamma} + \dot{\boldsymbol{\Omega}} \times \mathbf{s}_i + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + 2\dot{\boldsymbol{\chi}}_i) + \ddot{\boldsymbol{\chi}}_i \right\}. \quad (5.47)$$

Note that

$$\mathbf{s}_i \times \left\{ \dot{\boldsymbol{\Omega}} \times \mathbf{s}_i \right\} = -\mathbf{s}_i \times \left\{ \mathbf{s}_i \times \dot{\boldsymbol{\Omega}} \right\} = -\mathbf{s}_i \times \left\{ \hat{\mathbf{s}}_i \dot{\boldsymbol{\Omega}} \right\} = -\hat{\mathbf{s}}_i \hat{\mathbf{s}}_i \dot{\boldsymbol{\Omega}} = -\hat{\mathbf{s}}_i^2 \dot{\boldsymbol{\Omega}}, \quad (5.48)$$

where for $\mathbf{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^\top$, $\hat{\mathbf{v}}^2 = \hat{\mathbf{v}}\hat{\mathbf{v}}$ is the symmetric matrix given by

$$\hat{\mathbf{v}}^2 = \begin{bmatrix} -(v_2^2 + v_3^2) & v_1 v_2 & v_1 v_3 \\ v_1 v_2 & -(v_1^2 + v_3^2) & v_2 v_3 \\ v_1 v_3 & v_2 v_3 & -(v_1^2 + v_2^2) \end{bmatrix}. \quad (5.49)$$

Using (5.48) and solving explicitly for $\dot{\boldsymbol{\Omega}}$ in (5.47), the complete uncontrolled equations of motion become

$$\begin{aligned} \dot{\boldsymbol{\Omega}} &= \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \left[\boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} + r\tilde{\boldsymbol{\Gamma}} \times \boldsymbol{\Gamma} + \sum_{i=0}^n m_i \mathbf{s}_i \times \left\{ g\boldsymbol{\Gamma} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\chi}_i + 2\dot{\boldsymbol{\chi}}_i) + \ddot{\boldsymbol{\chi}}_i \right\} \right], \\ \dot{\boldsymbol{\Gamma}} &= \boldsymbol{\Gamma} \times \boldsymbol{\Omega}, \end{aligned} \quad (5.50)$$

subject to the definitions $\mathbf{s}_i \equiv r\boldsymbol{\Gamma} + \boldsymbol{\chi}_i$ for $0 \leq i \leq n$, $\boldsymbol{\Omega} \equiv (\Lambda^{-1}\dot{\Lambda})^\vee$, $\boldsymbol{\Gamma} \equiv \Lambda^{-1}\mathbf{e}_3$, and $\tilde{\boldsymbol{\Gamma}} \equiv \Lambda^{-1}\mathbf{F}_e$. The uncontrolled equations of motion (5.50) for the rolling ball actuated by internal point masses are new and have not appeared previously in the literature, as far as the author knows.

Uncontrolled Equations of Motion for the Rolling Ball with Static Internal Structure A special case of (5.50) gives the uncontrolled equations of motion for a rolling ball with static internal structure. By fixing all the control masses (i.e. making $\boldsymbol{\chi}_i$ constant for all $1 \leq i \leq n$, so that $\dot{\boldsymbol{\chi}}_i = \ddot{\boldsymbol{\chi}}_i = 0$), (5.50) gives the uncontrolled equations of motion for a rolling ball with static internal structure:

$$\begin{aligned} \dot{\boldsymbol{\Omega}} &= \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \left[\boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} + r\tilde{\boldsymbol{\Gamma}} \times \boldsymbol{\Gamma} + \sum_{i=0}^n m_i \mathbf{s}_i \times \left\{ g\boldsymbol{\Gamma} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\chi}_i) \right\} \right], \\ \dot{\boldsymbol{\Gamma}} &= \boldsymbol{\Gamma} \times \boldsymbol{\Omega}. \end{aligned} \quad (5.51)$$

Alternatively, the uncontrolled equations of motion for a rolling ball with static internal structure may be obtained by setting the number of control masses n to 0 in (5.50):

$$\begin{aligned} \dot{\boldsymbol{\Omega}} &= [m_0 \hat{\mathbf{s}}_0^2 - \mathbb{I}]^{-1} \left[\boldsymbol{\Omega} \times \mathbb{I}\boldsymbol{\Omega} + r\tilde{\boldsymbol{\Gamma}} \times \boldsymbol{\Gamma} + m_0 \mathbf{s}_0 \times \left\{ g\boldsymbol{\Gamma} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\chi}_0) \right\} \right], \\ \dot{\boldsymbol{\Gamma}} &= \boldsymbol{\Gamma} \times \boldsymbol{\Omega}. \end{aligned} \quad (5.52)$$

Uncontrolled Equations of Motion for the Rolling Ball Assuming 1-d Parameterizations of the Control Trajectories For $1 \leq i \leq n$, assume now that the trajectory $\boldsymbol{\xi}_i$ of the i^{th} control mass is required to move along a 1-d control rail, like a circular hoop. Moreover, for $1 \leq i \leq n$, assume that the i^{th} control

rail is parameterized by a 1-d parameter θ_i , so that the position ζ_i of the i^{th} control rail, in the body frame translated to the ball's geometric center, as a function of θ_i is $\zeta_i(\theta_i)$. Thus, the trajectory of the i^{th} control mass as a function of time t is $\chi_i(t) \equiv \zeta_i(\theta_i(t))$, $1 \leq i \leq n$. Refer to Figure 5.3 for an illustration. To make notation consistent, define $\zeta_0(\theta_0) \equiv \chi_0$, so that the constant vector $\chi_0 = \chi_0(t) \equiv \zeta_0(\theta_0(t))$ for any scalar-valued, time-varying function $\theta_0(t)$. By the chain rule and using the notation $\dot{\cdot}$ to denote differentiation with respect to time t and ζ'_i to denote differentiation of ζ_i with respect to θ_i , for $0 \leq i \leq n$,

$$\begin{aligned}\chi_i(t) &\equiv \zeta_i(\theta_i(t)) = \zeta_i, \\ \dot{\chi}_i(t) &= \frac{d\zeta_i}{d\theta_i}(\theta_i(t))\dot{\theta}_i(t) = \zeta'_i(\theta_i(t))\dot{\theta}_i(t) = \zeta'_i\dot{\theta}_i = \dot{\theta}_i\zeta'_i, \\ \ddot{\chi}_i(t) &= \frac{d^2\zeta_i}{d\theta_i^2}(\theta_i(t))\dot{\theta}_i^2(t) + \frac{d\zeta_i}{d\theta_i}(\theta_i(t))\ddot{\theta}_i(t) \\ &= \zeta''_i(\theta_i(t))\dot{\theta}_i^2(t) + \zeta'_i(\theta_i(t))\ddot{\theta}_i(t) = \zeta''_i\dot{\theta}_i^2 + \zeta'_i\ddot{\theta}_i = \dot{\theta}_i^2\zeta''_i + \ddot{\theta}_i\zeta'_i.\end{aligned}\tag{5.53}$$

By plugging the formulas for χ_i , $\dot{\chi}_i$, and $\ddot{\chi}_i$ given in (5.53) into (5.50), the uncontrolled equations of motion become

$$\begin{aligned}\dot{\Omega} &= \left[\sum_{i=0}^n m_i \hat{s}_i^2 - \mathbb{I} \right]^{-1} \left[\Omega \times \mathbb{I}\Omega + r\tilde{\Gamma} \times \Gamma \right. \\ &\quad \left. + \sum_{i=0}^n m_i \mathbf{s}_i \times \left\{ g\Gamma + \Omega \times \left(\Omega \times \zeta_i + 2\dot{\theta}_i\zeta'_i \right) + \dot{\theta}_i^2\zeta''_i + \ddot{\theta}_i\zeta'_i \right\} \right], \\ \dot{\Gamma} &= \Gamma \times \Omega,\end{aligned}\tag{5.54}$$

where with this new notation, $\mathbf{s}_i \equiv r\Gamma + \chi_i = r\Gamma + \zeta_i$ for $0 \leq i \leq n$.

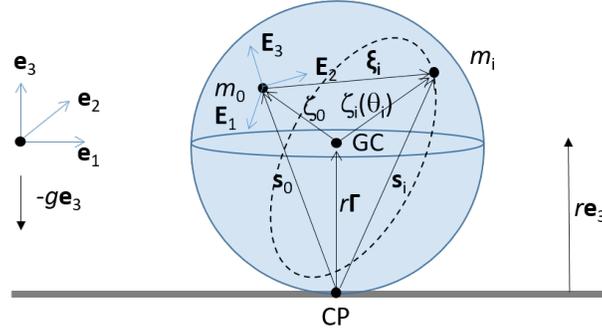


Figure 5.3: Each control mass, denoted by m_i , $1 \leq i \leq n$, moves along a control rail fixed inside the ball depicted here by the dashed hoop. The position of the control rail is denoted by ζ_i and is parameterized by θ_i .

5.B.4 Uncontrolled Equations of Motion for the Rolling Disk

Now suppose that m_0 's inertia is such that one of m_0 's principal axes, say the one labeled \mathbf{E}_2 , is orthogonal to the plane containing the GC and CM. Also assume that all the control masses move along 1-d control rails which lie in the plane containing the GC and CM. Moreover, suppose that the ball is oriented initially so

that the plane containing the GC and CM coincides with the \mathbf{e}_1 - \mathbf{e}_3 plane and that the external force \mathbf{F}_e acts in the \mathbf{e}_1 - \mathbf{e}_3 plane. Then for all time, the ball will remain oriented so that the plane containing the GC and CM coincides with the \mathbf{e}_1 - \mathbf{e}_3 plane and the ball will only move in the \mathbf{e}_1 - \mathbf{e}_3 plane, with the ball's rotation axis always parallel to \mathbf{e}_2 . Note that the dynamics of this system are equivalent to that of the Chaplygin disk [16], equipped with control masses, rolling in the \mathbf{e}_1 - \mathbf{e}_3 plane, and where the Chaplygin disk (minus the control masses) has polar moment of inertia d_2 . Therefore, henceforth, this particular ball with this special inertia, orientation, placement of the control rails, and control masses may be referred to as the disk or the rolling disk. Figure 5.4 depicts the rolling disk.

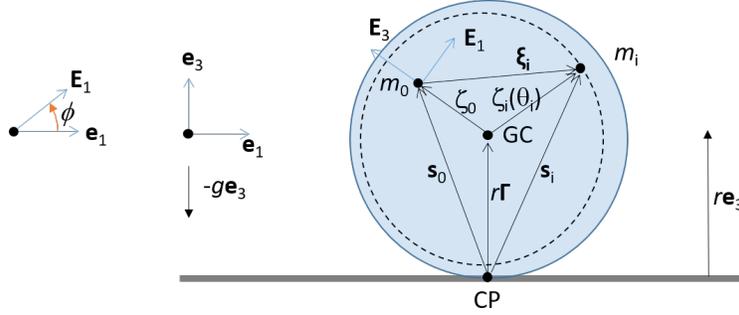


Figure 5.4: A disk of radius r and mass m_0 rolls without slipping in the \mathbf{e}_1 - \mathbf{e}_3 plane. \mathbf{e}_2 and \mathbf{E}_2 are directed into the page and are omitted from the figure. The disk's center of mass is denoted by m_0 . The disk's motion is actuated by n point masses, each of mass m_i , $1 \leq i \leq n$, that move along control rails fixed inside the disk. The control mass depicted here by m_i moves along a circular hoop in the disk that is not centered on the disk's geometric center (GC). The disk's orientation is determined by ϕ , the angle measured counter-clockwise from \mathbf{e}_1 to \mathbf{E}_1 .

Let ϕ denote the angle between \mathbf{e}_1 and \mathbf{E}_1 , measured counter-clockwise from \mathbf{e}_1 to \mathbf{E}_1 . Thus, if $\dot{\phi} > 0$, the disk rolls in the $-\mathbf{e}_1$ direction and $\boldsymbol{\Omega}$ has the same direction as $-\mathbf{e}_2$, and if $\dot{\phi} < 0$, the disk rolls in the \mathbf{e}_1 direction and $\boldsymbol{\Omega}$ has the same direction as \mathbf{e}_2 . Before constructing the equations of motion for the rolling disk using (5.54), some intermediate calculations must be performed.

$$\Lambda = \begin{bmatrix} \cos \phi & 0 & -\sin \phi \\ 0 & 1 & 0 \\ \sin \phi & 0 & \cos \phi \end{bmatrix}, \quad \Lambda^\top = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix}. \quad (5.55)$$

$$\hat{\boldsymbol{\Omega}} = \Lambda^\top \dot{\Lambda} = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \begin{bmatrix} -\sin \phi & 0 & -\cos \phi \\ 0 & 0 & 0 \\ \cos \phi & 0 & -\sin \phi \end{bmatrix} \dot{\phi} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \dot{\phi}. \quad (5.56)$$

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \dot{\phi} = -\dot{\phi} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = -\dot{\phi} \mathbf{e}_2, \quad \mathbb{I} \boldsymbol{\Omega} = -d_2 \dot{\phi} \mathbf{e}_2, \quad \boldsymbol{\Omega} \times \mathbb{I} \boldsymbol{\Omega} = d_2 \dot{\phi}^2 \mathbf{e}_2 \times \mathbf{e}_2 = \mathbf{0}. \quad (5.57)$$

$$\boldsymbol{\Gamma} = \Lambda^\top \mathbf{e}_3 = \begin{bmatrix} \sin \phi \\ 0 \\ \cos \phi \end{bmatrix}, \quad \tilde{\boldsymbol{\Gamma}} = \Lambda^\top \mathbf{F}_e = \begin{bmatrix} \cos \phi F_{e,1} + \sin \phi F_{e,3} \\ 0 \\ -\sin \phi F_{e,1} + \cos \phi F_{e,3} \end{bmatrix}. \quad (5.58)$$

$$\begin{aligned}
r\tilde{\Gamma} \times \Gamma &= r \begin{bmatrix} \cos \phi F_{e,1} + \sin \phi F_{e,3} \\ 0 \\ -\sin \phi F_{e,1} + \cos \phi F_{e,3} \end{bmatrix} \times \begin{bmatrix} \sin \phi \\ 0 \\ \cos \phi \end{bmatrix} \\
&= r \{(-\sin \phi F_{e,1} + \cos \phi F_{e,3}) \sin \phi - (\cos \phi F_{e,1} + \sin \phi F_{e,3}) \cos \phi\} \mathbf{e}_2 \\
&= -r F_{e,1} \mathbf{e}_2.
\end{aligned} \tag{5.59}$$

$$\dot{\Omega} = -\ddot{\phi} \mathbf{e}_2, \quad \dot{\Gamma} = \begin{bmatrix} \cos \phi \\ 0 \\ -\sin \phi \end{bmatrix} \dot{\phi}, \quad \Gamma \times \Omega = -\dot{\phi} \begin{bmatrix} \sin \phi \\ 0 \\ \cos \phi \end{bmatrix} \times \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = -\dot{\phi} \begin{bmatrix} -\cos \phi \\ 0 \\ \sin \phi \end{bmatrix} = \dot{\Gamma}. \tag{5.60}$$

Thus, the second equation, $\dot{\Gamma} = \Gamma \times \Omega$, in (5.54) gives no information about the dynamics and may be ignored for the disk.

$$\zeta_i = \begin{bmatrix} \zeta_{i,1} \\ 0 \\ \zeta_{i,3} \end{bmatrix}, \quad \zeta'_i = \begin{bmatrix} \zeta'_{i,1} \\ 0 \\ \zeta'_{i,3} \end{bmatrix}, \quad \zeta''_i = \begin{bmatrix} \zeta''_{i,1} \\ 0 \\ \zeta''_{i,3} \end{bmatrix}. \tag{5.61}$$

$$\mathbf{s}_i = r\Gamma + \zeta_i = \begin{bmatrix} r \sin \phi + \zeta_{i,1} \\ 0 \\ r \cos \phi + \zeta_{i,3} \end{bmatrix}. \tag{5.62}$$

$$\begin{aligned}
\hat{\mathbf{s}}_i^2 &= \mathbf{s}_i \mathbf{s}_i = \\
&\begin{bmatrix} -(r \cos \phi + \zeta_{i,3})^2 & 0 & (r \sin \phi + \zeta_{i,1})(r \cos \phi + \zeta_{i,3}) \\ 0 & -(r \sin \phi + \zeta_{i,1})^2 - (r \cos \phi + \zeta_{i,3})^2 & 0 \\ (r \sin \phi + \zeta_{i,1})(r \cos \phi + \zeta_{i,3}) & 0 & -(r \sin \phi + \zeta_{i,1})^2 \end{bmatrix}.
\end{aligned} \tag{5.63}$$

$$\Omega \times \zeta_i = -\dot{\phi} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \times \begin{bmatrix} \zeta_{i,1} \\ 0 \\ \zeta_{i,3} \end{bmatrix} = -\dot{\phi} \begin{bmatrix} \zeta_{i,3} \\ 0 \\ -\zeta_{i,1} \end{bmatrix}, \quad \Omega \times \zeta_i + 2\dot{\theta}_i \zeta'_i = \begin{bmatrix} -\dot{\phi} \zeta_{i,3} + 2\dot{\theta}_i \zeta'_{i,1} \\ 0 \\ \dot{\phi} \zeta_{i,1} + 2\dot{\theta}_i \zeta'_{i,3} \end{bmatrix}. \tag{5.64}$$

$$\Omega \times (\Omega \times \zeta_i + 2\dot{\theta}_i \zeta'_i) = -\dot{\phi} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \times \begin{bmatrix} -\dot{\phi} \zeta_{i,3} + 2\dot{\theta}_i \zeta'_{i,1} \\ 0 \\ \dot{\phi} \zeta_{i,1} + 2\dot{\theta}_i \zeta'_{i,3} \end{bmatrix} = -\dot{\phi} \begin{bmatrix} \dot{\phi} \zeta_{i,1} + 2\dot{\theta}_i \zeta'_{i,3} \\ 0 \\ \dot{\phi} \zeta_{i,3} - 2\dot{\theta}_i \zeta'_{i,1} \end{bmatrix}. \tag{5.65}$$

$$g\Gamma + \Omega \times (\Omega \times \zeta_i + 2\dot{\theta}_i \zeta'_i) + \ddot{\theta}_i \zeta''_i + \ddot{\theta}_i \zeta'_i = \begin{bmatrix} g \sin \phi - \dot{\phi} (\dot{\phi} \zeta_{i,1} + 2\dot{\theta}_i \zeta'_{i,3}) + \dot{\theta}_i^2 \zeta''_{i,1} + \ddot{\theta}_i \zeta'_{i,1} \\ 0 \\ g \cos \phi - \dot{\phi} (\dot{\phi} \zeta_{i,3} - 2\dot{\theta}_i \zeta'_{i,1}) + \dot{\theta}_i^2 \zeta''_{i,3} + \ddot{\theta}_i \zeta'_{i,3} \end{bmatrix}. \tag{5.66}$$

$$\mathbf{s}_i \times \left\{ g\Gamma + \Omega \times (\Omega \times \zeta_i + 2\dot{\theta}_i \zeta'_i) + \dot{\theta}_i^2 \zeta''_i + \ddot{\theta}_i \zeta'_i \right\} = K_i \mathbf{e}_2, \tag{5.67}$$

where

$$\begin{aligned}
K_i &= (r \cos \phi + \zeta_{i,3}) \left(g \sin \phi - \dot{\phi} (\dot{\phi} \zeta_{i,1} + 2\dot{\theta}_i \zeta'_{i,3}) + \dot{\theta}_i^2 \zeta''_{i,1} + \ddot{\theta}_i \zeta'_{i,1} \right) \\
&\quad - (r \sin \phi + \zeta_{i,1}) \left(g \cos \phi - \dot{\phi} (\dot{\phi} \zeta_{i,3} - 2\dot{\theta}_i \zeta'_{i,1}) + \dot{\theta}_i^2 \zeta''_{i,3} + \ddot{\theta}_i \zeta'_{i,3} \right) \\
&= (g + r\dot{\phi}^2) (\zeta_{i,3} \sin \phi - \zeta_{i,1} \cos \phi) + (r \cos \phi + \zeta_{i,3}) \left(-2\dot{\phi} \dot{\theta}_i \zeta'_{i,3} + \dot{\theta}_i^2 \zeta''_{i,1} + \ddot{\theta}_i \zeta'_{i,1} \right) \\
&\quad - (r \sin \phi + \zeta_{i,1}) \left(2\dot{\phi} \dot{\theta}_i \zeta'_{i,1} + \dot{\theta}_i^2 \zeta''_{i,3} + \ddot{\theta}_i \zeta'_{i,3} \right).
\end{aligned} \tag{5.68}$$

Plugging (5.57), (5.59), and (5.67) into the first equation in (5.54) gives the equations of motion for the rolling disk as

$$\begin{aligned} -\ddot{\phi}\mathbf{e}_2 &= \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \left[-rF_{e,1}\mathbf{e}_2 + \sum_{i=0}^n m_i K_i \mathbf{e}_2 \right] \\ &= \left(-rF_{e,1} + \sum_{i=0}^n m_i K_i \right) \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \mathbf{e}_2. \end{aligned} \quad (5.69)$$

Note that $\left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \mathbf{e}_2$ is just the middle column of the matrix inverse of $A = \sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I}$. Denote the entries of A by

$$A = \sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}. \quad (5.70)$$

Since \mathbb{I} is diagonal and from (5.63), $a_{12} = a_{21} = a_{23} = a_{32} = 0$, so that

$$A = \sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} = \begin{bmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & 0 \\ a_{31} & 0 & a_{33} \end{bmatrix} \quad (5.71)$$

and the determinant of A simplifies to

$$\begin{aligned} \det A &= a_{11}a_{22}a_{33} + a_{21}a_{32}a_{13} + a_{31}a_{12}a_{23} - a_{11}a_{32}a_{23} - a_{31}a_{22}a_{13} - a_{21}a_{12}a_{33} \\ &= a_{11}a_{22}a_{33} - a_{31}a_{22}a_{13} \\ &= a_{22} (a_{11}a_{33} - a_{31}a_{13}). \end{aligned} \quad (5.72)$$

Using the formula for the inverse of a 3×3 matrix, the middle column of the matrix inverse of $A = \sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I}$ is

$$\begin{aligned} \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \mathbf{e}_2 &= A^{-1} \mathbf{e}_2 = \begin{bmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & 0 \\ a_{31} & 0 & a_{33} \end{bmatrix}^{-1} \mathbf{e}_2 = \frac{1}{\det A} \begin{bmatrix} a_{13}a_{32} - a_{12}a_{33} \\ a_{11}a_{33} - a_{13}a_{31} \\ a_{12}a_{31} - a_{11}a_{32} \end{bmatrix} \\ &= \frac{1}{a_{22} (a_{11}a_{33} - a_{31}a_{13})} \begin{bmatrix} 0 \\ a_{11}a_{33} - a_{13}a_{31} \\ 0 \end{bmatrix} = \frac{1}{a_{22}} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \frac{1}{a_{22}} \mathbf{e}_2. \end{aligned} \quad (5.73)$$

Plugging (5.73) into (5.69), the equations of motion simplify to

$$-\ddot{\phi}\mathbf{e}_2 = \left(-rF_{e,1} + \sum_{i=0}^n m_i K_i \right) \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \mathbf{e}_2 = \frac{1}{a_{22}} \left(-rF_{e,1} + \sum_{i=0}^n m_i K_i \right) \mathbf{e}_2, \quad (5.74)$$

which gives the scalar equation of motion

$$\ddot{\phi} = \frac{-1}{a_{22}} \left(-rF_{e,1} + \sum_{i=0}^n m_i K_i \right). \quad (5.75)$$

From (5.63)

$$a_{22} = \sum_{i=0}^n \left\{ m_i \left[- (r \sin \phi + \zeta_{i,1})^2 - (r \cos \phi + \zeta_{i,3})^2 \right] \right\} - d_2. \quad (5.76)$$

Plugging (5.76) into (5.75) gives the equation of motion for the rolling disk

$$\ddot{\phi} = \frac{-rF_{e,1} + \sum_{i=0}^n m_i K_i}{d_2 + \sum_{i=0}^n m_i \left[(r \sin \phi + \zeta_{i,1})^2 + (r \cos \phi + \zeta_{i,3})^2 \right]} := \kappa \left(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \phi, \dot{\phi}, \ddot{\boldsymbol{\theta}} \right), \quad (5.77)$$

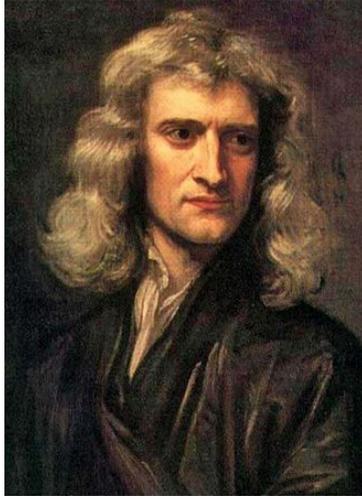
where κ is a function that depends on the control mass parameterized positions ($\boldsymbol{\theta}$), velocities ($\dot{\boldsymbol{\theta}}$), and accelerations ($\ddot{\boldsymbol{\theta}}$) and on the disk's orientation angle (ϕ) and its time-derivative ($\dot{\phi}$).

Verification of the Variational Equations Using Newtonian Mechanics for a Special Case

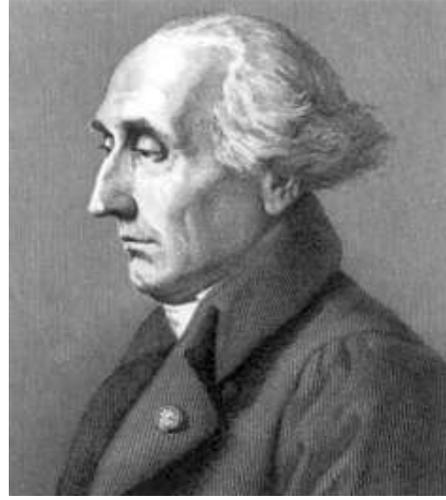
There are two general classical approaches to derive the uncontrolled equations of motion of a mechanical system: Newtonian mechanics and Lagrangian mechanics. Newtonian mechanics was developed by the English mathematician Sir Isaac Newton, depicted in Figure 5.5a, during the latter half of the 16th century. In Newtonian mechanics, the uncontrolled equations of motion are derived by writing down all the forces acting on the system according to Newton's three laws of motion. Lagrangian mechanics was developed by the French mathematician Joseph-Louis Lagrange, depicted in Figure 5.5b, during the latter half of the 17th century. In Lagrangian mechanics, the uncontrolled equations of motion are derived by writing down the system's Lagrangian from the kinetic and potential energies followed by application of a variational principle (e.g. Hamilton's principle, Hamilton-Pontryagin's principle, Lagrange-d'Alembert's principle, etc.) to the Lagrangian. This thesis relies on Lagrangian mechanics as it is much more efficient than Newtonian mechanics. To illustrate the relative computational ease afforded by Lagrangian mechanics compared to Newtonian mechanics, Newtonian mechanics will be used to derive the equations of motion for a simple case of the rolling disk where the CM and GC coincide and where there is a single control mass; in addition, this calculation will serve to validate the equations of motion obtained earlier using Lagrangian mechanics (i.e. Lagrange-d'Alembert's principle with Euler-Poincaré's method).

Consider a disk of mass m_0 and radius r whose CM and GC coincide. The moment of inertia of the disk computed with respect to the CM is d_2 . The disk rolls without slipping along a flat surface in a uniform gravitational field of magnitude g . The disk is actuated by a single control mass of mass m_1 that moves along a circular trajectory of radius r_1 , with $0 < r_1 < r$, centered on the disk's GC. The spatial \mathbf{e}_1 -component of the disk's GC is given by $z(t)$. Since $z(t) = z_a - r(\phi(t) - \phi_a)$, $\dot{z}(t) = -r\dot{\phi}(t)$ and $\ddot{z}(t) = -r\ddot{\phi}(t)$. Since the CM and GC coincide, the body frame coincides with the body frame translated to the GC. The control mass's trajectory in the body frame translated to the GC is

$$\boldsymbol{\zeta}_1(t) = r_1 \begin{bmatrix} \cos \theta_1(t) \\ 0 \\ \sin \theta_1(t) \end{bmatrix} \quad (5.78)$$



(a) Sir Isaac Newton, 1642-1727 [52].



(b) Joseph-Louis Lagrange, 1736-1813 [53].

Figure 5.5: Portraits of Newton and Lagrange, progenitors of classical mechanics.

and in the spatial frame is

$$\begin{aligned}
 \mathbf{z}_1(t) &= \begin{bmatrix} z(t) \\ 0 \\ 0 \end{bmatrix} + \Lambda(t)\zeta_1(t) = \begin{bmatrix} z(t) \\ 0 \\ 0 \end{bmatrix} + r_1 \begin{bmatrix} \cos \phi(t) & 0 & -\sin \phi(t) \\ 0 & 1 & 0 \\ \sin \phi(t) & 0 & \cos \phi(t) \end{bmatrix} \begin{bmatrix} \cos \theta_1(t) \\ 0 \\ \sin \theta_1(t) \end{bmatrix} \\
 &= \begin{bmatrix} z(t) + r_1 \cos(\phi(t) + \theta_1(t)) \\ 0 \\ r_1 \sin(\phi(t) + \theta_1(t)) \end{bmatrix}.
 \end{aligned} \tag{5.79}$$

Observe that the axis of rotation passes through the CM and that the axis of rotation does not change direction. Thus, it is straightforward to determine the dynamics of this system via Newtonian mechanics. Newton's second law says that the sum of all external forces acting on the disk must equal $m_0\ddot{z}\mathbf{e}_1 = -m_0r\ddot{\phi}\mathbf{e}_1$ and that the sum of all external torques acting on the disk about the disk's CM must equal $-d_2\ddot{\phi}\mathbf{e}_2$. The external forces acting on the disk are the force $-m_1\ddot{z}_1 - m_1g\mathbf{e}_3$ exerted by the accelerating control mass, the gravitational force $-m_0g\mathbf{e}_3$ exerted at the CM, a horizontal static frictional force $-f_s\mathbf{e}_1$ exerted by the surface, a normal force $N\mathbf{e}_3$ exerted by the surface, and an external force \mathbf{F}_e exerted at the disk's GC. See Figure 5.6 for the free body diagram depicting all the external forces acting on the disk.

Application of Newton's second law to this system gives the following force and torque balance equations:

$$\begin{aligned}
 \sum F_1 &= -f_s + F_{e,1} - m_1\ddot{z}_{1,1} = m_0\ddot{z} = -m_0r\ddot{\phi} \implies f_s = m_0r\ddot{\phi} + F_{e,1} - m_1\ddot{z}_{1,1} \\
 \sum F_3 &= N + F_{e,3} - m_0g - m_1g - m_1\ddot{z}_{1,3} = 0 \implies N = -F_{e,3} + (m_0 + m_1)g + m_1\ddot{z}_{1,3} \\
 \sum \tau &= rf_s\mathbf{e}_2 + m_1g(z_{1,1} - z)\mathbf{e}_2 - m_1 \left(\mathbf{z}_1 - \begin{bmatrix} z \\ 0 \\ 0 \end{bmatrix} \right) \times \ddot{\mathbf{z}}_1 = -d_2\ddot{\phi}\mathbf{e}_2.
 \end{aligned} \tag{5.80}$$

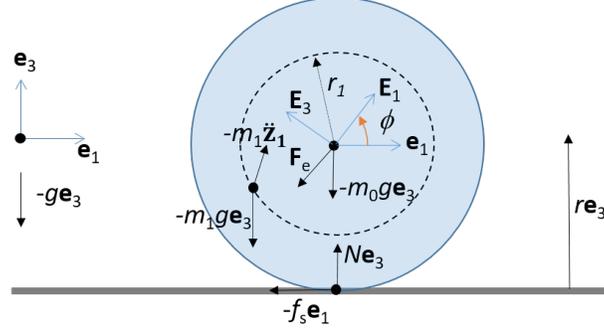


Figure 5.6: A disk actuated by a single control mass.

Plugging the formula for the horizontal static friction force into the torque balance equation yields

$$r \left(m_0 r \ddot{\phi} + F_{e,1} - m_1 \ddot{z}_{1,1} \right) \mathbf{e}_2 + m_1 g (z_{1,1} - z) \mathbf{e}_2 - m_1 \left(\mathbf{z}_1 - \begin{bmatrix} z \\ 0 \\ 0 \end{bmatrix} \right) \times \ddot{\mathbf{z}}_1 + d_2 \ddot{\phi} \mathbf{e}_2 = \mathbf{0}, \quad (5.81)$$

which simplifies to

$$\left\{ d_2 \ddot{\phi} + r \left(m_0 r \ddot{\phi} + F_{e,1} - m_1 \left[\ddot{z} - r_1 \cos(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 - r_1 \sin(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \right] \right) + m_1 g r_1 \cos(\phi + \theta_1) \right\} \mathbf{e}_2 - m_1 r_1 \begin{bmatrix} \cos(\phi + \theta_1) \\ 0 \\ \sin(\phi + \theta_1) \end{bmatrix} \times \begin{bmatrix} \ddot{z} - r_1 \cos(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 - r_1 \sin(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \\ 0 \\ -r_1 \sin(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 + r_1 \cos(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \end{bmatrix} = 0. \quad (5.82)$$

Equation (5.82) is equivalent to

$$\begin{aligned} & d_2 \ddot{\phi} + r \left(m_0 r \ddot{\phi} + F_{e,1} - m_1 \left[-r \ddot{\phi} - r_1 \cos(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 - r_1 \sin(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \right] \right) \\ & + m_1 g r_1 \cos(\phi + \theta_1) \\ & - m_1 r_1 \left\{ \sin(\phi + \theta_1) \left[-r \ddot{\phi} - r_1 \cos(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 - r_1 \sin(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \right] \right. \\ & \left. - \cos(\phi + \theta_1) \left[-r_1 \sin(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 + r_1 \cos(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \right] \right\} = 0, \end{aligned} \quad (5.83)$$

which simplifies to

$$\begin{aligned} & d_2 \ddot{\phi} + r \left(m_0 r \ddot{\phi} + F_{e,1} + m_1 \left[r \ddot{\phi} + r_1 \cos(\phi + \theta_1) (\dot{\phi} + \dot{\theta}_1)^2 + r_1 \sin(\phi + \theta_1) (\ddot{\phi} + \ddot{\theta}_1) \right] \right) \\ & + m_1 g r_1 \cos(\phi + \theta_1) + m_1 r_1 \left\{ r \ddot{\phi} \sin(\phi + \theta_1) + r_1 (\ddot{\phi} + \ddot{\theta}_1) \right\} = 0, \end{aligned} \quad (5.84)$$

which further simplifies to the following equation of motion

$$\ddot{\phi} = - \frac{rF_{e,1} + m_1 r_1 \left[\cos(\phi + \theta_1) \left\{ r \left(\dot{\phi} + \dot{\theta}_1 \right)^2 + g \right\} + \{r_1 + r \sin(\phi + \theta_1)\} \ddot{\theta}_1 \right]}{d_2 + (m_0 + m_1)r^2 + m_1 r_1 [r_1 + 2r \sin(\phi + \theta_1)]}. \quad (5.85)$$

Under all these assumptions for this particular rolling disk, a calculation shows that equation (5.85) coincides with equation (5.77), which was derived earlier by variational methods (i.e. Lagrangian mechanics).

5.C Controlled Equations of Motion

The results of the previous section are now used to derive the controlled equations of motion for the rolling disk and ball. First, using the results of Subsection 5.B.4, the controlled equations of motion for a rolling disk actuated by internal point masses that move along rails fixed within the disk are derived. Next, using the results of Subsection 5.B.3, the controlled equations of motion for a rolling ball actuated by internal point masses that move along rails fixed within the ball are derived.

5.C.1 Controlled Equations of Motion for the Rolling Disk

Before attacking the rolling ball, the controlled equations of motion are first developed for the rolling disk, which is a simpler mechanical system. Let z and \dot{z} denote the spatial \mathbf{e}_1 position and velocity, respectively, of the disk's GC, and recall that $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \dots \ \theta_n]^\top$ denotes the vector of the control mass parameterizations. If the disk's GC is at initial spatial \mathbf{e}_1 position z_a and if the disk's initial orientation is ϕ_a at initial time $t = a$, note that the spatial \mathbf{e}_1 position and velocity of the disk's GC are $z = z_a - r(\phi - \phi_a)$ and $\dot{z} = -r\dot{\phi}$, respectively, due to the sign convention adopted for ϕ in Subsection 5.B.4. The rotation matrix that maps the body to spatial frame at time t is a function of $\phi(t)$ and is given by

$$\tilde{\Lambda}(\phi(t)) = \Lambda(t) = \begin{bmatrix} \cos \phi(t) & 0 & -\sin \phi(t) \\ 0 & 1 & 0 \\ \sin \phi(t) & 0 & \cos \phi(t) \end{bmatrix}. \quad (5.86)$$

Suppose it is desired to roll the disk from some initial configuration at a prescribed or free initial time a to some final configuration at a prescribed or free final time b , without moving the control masses too rapidly along their control rails. In addition, in between the initial and final times, it may be desired that the disk's GC tracks a prescribed spatial \mathbf{e}_1 path z_d or traces out a minimum energy path. Finally, if the initial or final time is free, it may be desired to minimize the duration $b - a$ of the maneuver. How must the control masses be moved in order to accomplish these tasks? This problem can be solved by posing it as an optimal control problem.

Concretely, at the prescribed or free initial time a , the positions of the control mass parameterizations are prescribed to be $\boldsymbol{\theta}(a) = \boldsymbol{\theta}_a$, the velocities of the control mass parameterizations are prescribed to be $\dot{\boldsymbol{\theta}}(a) = \dot{\boldsymbol{\theta}}_a$, the spatial \mathbf{e}_1 position of the disk's GC is prescribed to be $z(a) = z_a - r(\phi(a) - \phi_a) = z_a$ (which

is equivalent to prescribing the disk's orientation to be $\phi(a) = \phi_a$, and the spatial \mathbf{e}_1 velocity of the disk's GC is prescribed to be $\dot{z}(a) = -r\dot{\phi}(a) = \dot{z}_a$ (which is equivalent to prescribing the rate of change of the disk's orientation to be $\dot{\phi}(a) = -\frac{\dot{z}_a}{r}$).

Furthermore, at the prescribed or free final time b , some components (determined by the projection operator $\mathbf{\Pi}$) of the disk's center of mass expressed in the spatial frame translated to the GC are prescribed to be

$$\mathbf{\Pi} \left(\tilde{\Lambda}(\phi(b)) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_i(b)) \right] \right) = \boldsymbol{\Delta}_b, \quad (5.87)$$

the velocities of the control mass parameterizations are prescribed to be $\dot{\boldsymbol{\theta}}(b) = \dot{\boldsymbol{\theta}}_b$, the spatial \mathbf{e}_1 position of the disk's GC is prescribed to be $z(b) = z_a - r(\phi(b) - \phi_a) = z_b$ (which is equivalent to prescribing the disk's orientation to be $\phi(b) = \phi_a - \frac{z_b - z_a}{r}$), and the spatial \mathbf{e}_1 velocity of the disk's GC is prescribed to be $\dot{z}(b) = -r\dot{\phi}(b) = \dot{z}_b$ (which is equivalent to prescribing the rate of change of the disk's orientation to be $\dot{\phi}(b) = -\frac{\dot{z}_b}{r}$).

For example, if it is desired to start and stop the disk at rest, then $\mathbf{\Pi}$ is projection onto the first component, $\boldsymbol{\Delta}_b = \boldsymbol{\Delta}_a = 0$, $\boldsymbol{\theta}_a$ and ϕ_a are such that

$$\mathbf{\Pi} \left(\tilde{\Lambda}(\phi_a) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_{a,i}) \right] \right) = 0, \quad (5.88)$$

$\dot{\boldsymbol{\theta}}_a = \mathbf{0}$, $\dot{z}_a = 0$, $\boldsymbol{\theta}(b)$ and $\phi(b)$ are such that

$$\mathbf{\Pi} \left(\tilde{\Lambda}(\phi(b)) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_i(b)) \right] \right) = 0, \quad (5.89)$$

$\dot{\boldsymbol{\theta}}_b = \mathbf{0}$, and $\dot{z}_b = 0$. With this choice of $\mathbf{\Pi}$, (5.88) and (5.89) mean that the CM in the spatial frame translated to the GC is above or below the GC at the initial and final times.

The system state \mathbf{x} and control \mathbf{u} are

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{\theta} \\ \dot{\boldsymbol{\theta}} \\ \phi \\ \dot{\phi} \end{bmatrix} \quad \text{and} \quad \mathbf{u} = \ddot{\boldsymbol{\theta}}. \quad (5.90)$$

The system dynamics defined for $a \leq t \leq b$ are

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\theta}} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) \equiv \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \mathbf{u} \\ \dot{\phi} \\ \kappa(\mathbf{x}, \mathbf{u}) \end{bmatrix}, \quad (5.91)$$

where $\kappa(\mathbf{x}, \mathbf{u})$ is given by the right-hand side of (5.77), the prescribed initial conditions at time $t = a$ are

$$\boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \begin{bmatrix} \boldsymbol{\theta}(a) - \boldsymbol{\theta}_a \\ \dot{\boldsymbol{\theta}}(a) - \dot{\boldsymbol{\theta}}_a \\ \phi(a) - \phi_a \\ -r\dot{\phi}(a) - \dot{z}_a \end{bmatrix} = \mathbf{0}, \quad (5.92)$$

and the prescribed final conditions at time $t = b$ are

$$\boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \begin{bmatrix} \mathbf{\Pi} \left(\tilde{\Lambda}(\phi(b)) \left[\frac{1}{M} \sum_{i=0}^n m_i \zeta_i(\theta_i(b)) \right] \right) - \mathbf{\Delta}_b \\ \dot{\boldsymbol{\theta}}(b) - \dot{\boldsymbol{\theta}}_b \\ z_a - r(\phi(b) - \phi_a) - z_b \\ -r\dot{\phi}(b) - \dot{z}_b \end{bmatrix} = \mathbf{0}. \quad (5.93)$$

Consider the endpoint and integrand cost functions

$$p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) = \frac{v_a}{2} (a - a_e)^2 + \frac{v_b}{2} (b - b_e)^2 \quad (5.94)$$

and

$$L(t, \mathbf{x}, \mathbf{u}, \mu) = \frac{\alpha}{2} (z_a - r(\phi - \phi_a) - z_d)^2 + \frac{\beta}{2} (-r\dot{\phi})^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \ddot{\theta}_i^2 + \delta, \quad (5.95)$$

for constants a_e and b_e and for fixed nonnegative constants v_a , v_b , α , β , γ_i , $1 \leq i \leq n$, and δ so that the performance index is

$$\begin{aligned} J &= p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \int_a^b L(t, \mathbf{x}, \mathbf{u}, \mu) dt \\ &= \frac{v_a}{2} (a - a_e)^2 + \frac{v_b}{2} (b - b_e)^2 + \int_a^b \left[\frac{\alpha}{2} (z_a - r(\phi - \phi_a) - z_d)^2 + \frac{\beta}{2} (-r\dot{\phi})^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \ddot{\theta}_i^2 + \delta \right] dt. \end{aligned} \quad (5.96)$$

The first summand $\frac{v_a}{2} (a - a_e)^2$ in p encourages the initial time a to be near a_e if the initial time is free, while the second summand $\frac{v_b}{2} (b - b_e)^2$ in p encourages the final time b to be near b_e if the final time is free. The first summand $\frac{\alpha}{2} (z_a - r(\phi - \phi_a) - z_d)^2$ in L encourages the disk's GC to track the desired spatial \mathbf{e}_1 path z_d , the second summand $\frac{\beta}{2} (-r\dot{\phi})^2$ in L encourages the disk's GC to track a minimum energy path, the next n summands $\frac{\gamma_i}{2} \ddot{\theta}_i^2$, $1 \leq i \leq n$, in L limit the magnitude of the acceleration of the i^{th} control mass parameterization, and the final summand δ in L encourages a minimum time maneuver.

For example, the desired spatial \mathbf{e}_1 path might have the form

$$z_d(t) = [z_a w(t) + \tilde{z}_d(t) (1 - w(t))] (1 - y(t)) + z_b y(t), \quad (5.97)$$

where

$$S(t) = \frac{1}{2} \left[1 + \tanh \left(\frac{-t}{\epsilon} \right) \right], \quad (5.98)$$

$$w(t) = S(t - a_e), \quad (5.99)$$

and

$$y(t) = S(-t + b_e). \quad (5.100)$$

z_d (5.97) holds steady at z_a for $t < a_e$, smoothly transitions between z_a and \tilde{z}_d at $t = a_e$, follows \tilde{z}_d for $a_e < t < b_e$, smoothly transitions between \tilde{z}_d and z_b at $t = b_e$, and holds steady at z_b for $b_e < t$. S (5.98) is a time-reversed sigmoid function, i.e. a smooth approximation of the time-reversed unit step function; ϵ in (5.98) is a parameter such as .01 that determines how rapidly S (5.98) transitions from 1 to 0 at time 0. w (5.99) is the time-translation of S (5.98) to time a_e and y (5.100) is the time-translation of the time-reversal of S (5.98) to time b_e . w (5.99) enables z_d (5.97) to smoothly transition between z_a and \tilde{z}_d at $t = a_e$, while y (5.100) enables z_d (5.97) to smoothly transition between \tilde{z}_d and z_b at $t = b_e$. \tilde{z}_d , which appears in (5.97), might be the cubic polynomial

$$\tilde{z}_d(t) = k_1 \left[-\frac{1}{3}t^3 + \frac{1}{2}(a_e + b_e)t - a_e b_e t + k_2 \right] = k_1 [q(t) + k_2], \quad (5.101)$$

where

$$q(t) = -\frac{1}{3}t^3 + \frac{1}{2}(a_e + b_e)t - a_e b_e t, \quad k_1 = \frac{z_b - z_a}{q(b_e) - q(a_e)}, \quad \text{and} \quad k_2 = \frac{z_b}{k_1} - q(b_e). \quad (5.102)$$

\tilde{z}_d has the special properties $\tilde{z}_d(a_e) = z_a$, $\dot{\tilde{z}}_d(a_e) = 0$, $\tilde{z}_d(b_e) = z_b$, and $\dot{\tilde{z}}_d(b_e) = 0$, so that the disk's GC is encouraged to start with zero velocity at \mathbf{e}_1 -coordinate z_a at $t = a_e$ and to stop with zero velocity at \mathbf{e}_1 -coordinate z_b at $t = b_e$. Figure 5.7 illustrates (5.97) using (5.101) with $a_e = 0$, $z_a = 0$, $b_e = 2$, $z_b = 10$, and $\epsilon = .01$.

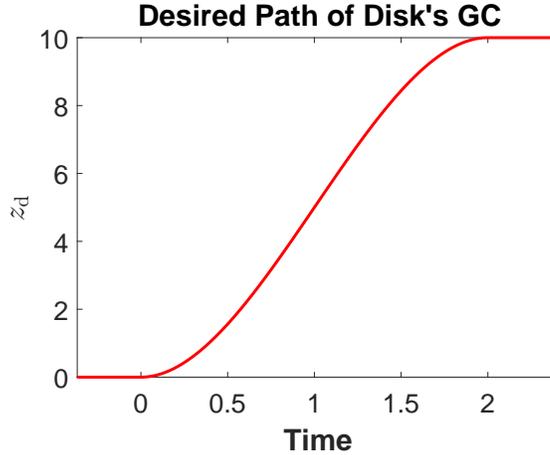


Figure 5.7: Plot of the desired path of the disk's GC. The disk's GC starts from rest at $z = 0$ at time $t = 0$, moves to the right for $0 < t < 2$, and stops at rest at $z = 10$ at time $t = 2$.

The optimal control problem for the rolling disk is

$$\min_{a,b,\mathbf{u}} J \quad \text{s.t.} \quad \begin{cases} \dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu), \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}. \end{cases} \quad (5.103)$$

Observe that the optimal control problem encapsulated by (5.103) ignores path inequality constraints such as

$\mathbf{D}(t, \mathbf{x}, \mathbf{u}, \mu) \leq \mathbf{0}$, where \mathbf{D} is a $r \times 1$ vector-valued function. Path inequality constraints can be incorporated in (5.103) as soft constraints through penalty functions in the integrand cost function L or the endpoint cost function p .

The indirect method, discussed in Section 3.B, is applied now to (5.103) to construct the endpoint function and regular Hamiltonian needed to formulate the ODE TPBVP (3.15), (3.16), and (3.17), which render the controlled equations of motion for the rolling disk.

The endpoint function is

$$\begin{aligned}
G(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) &= p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \boldsymbol{\xi}^\top \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) + \boldsymbol{\nu}^\top \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) \\
&= \frac{v_a}{2} (a - a_e)^2 + \frac{v_b}{2} (b - b_e)^2 + \boldsymbol{\xi}^\top \begin{bmatrix} \boldsymbol{\theta}(a) - \boldsymbol{\theta}_a \\ \dot{\boldsymbol{\theta}}(a) - \dot{\boldsymbol{\theta}}_a \\ \phi(a) - \phi_a \\ -r\dot{\phi}(a) - \dot{z}_a \end{bmatrix} \\
&\quad + \boldsymbol{\nu}^\top \begin{bmatrix} \boldsymbol{\Pi} \left(\tilde{\Lambda}(\phi(b)) \left[\frac{1}{M} \sum_{i=0}^n m_i \zeta_i(\theta_i(b)) \right] \right) - \boldsymbol{\Delta}_b \\ \dot{\boldsymbol{\theta}}(b) - \dot{\boldsymbol{\theta}}_b \\ z_a - r(\phi(b) - \phi_a) - z_b \\ -r\dot{\phi}(b) - \dot{z}_b \end{bmatrix}
\end{aligned} \tag{5.104}$$

and the Hamiltonian is

$$\begin{aligned}
H(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu) &= L(t, \mathbf{x}, \mathbf{u}, \mu) + \boldsymbol{\lambda}^\top \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) \\
&= \frac{\alpha}{2} (z_a - r(\phi - \phi_a) - z_d)^2 + \frac{\beta}{2} (-r\dot{\phi})^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \ddot{\theta}_i^2 + \delta + \boldsymbol{\lambda}^\top \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \mathbf{u} \\ \dot{\phi} \\ \kappa(\mathbf{x}, \mathbf{u}) \end{bmatrix}.
\end{aligned} \tag{5.105}$$

Recall that

$$\kappa(\mathbf{x}, \mathbf{u}) = \frac{-rF_{e,1} + \sum_{i=0}^n m_i K_i}{d_2 + \sum_{i=0}^n m_i \left[(r \sin \phi + \zeta_{i,1})^2 + (r \cos \phi + \zeta_{i,3})^2 \right]}, \tag{5.106}$$

where K_i for $0 \leq i \leq n$ is defined as

$$\begin{aligned}
K_i &= (g + r\dot{\phi}^2) (\zeta_{i,3} \sin \phi - \zeta_{i,1} \cos \phi) + (r \cos \phi + \zeta_{i,3}) \left(-2\dot{\phi}\dot{\theta}_i \zeta'_{i,3} + \dot{\theta}_i^2 \zeta''_{i,1} + \ddot{\theta}_i \zeta'_{i,1} \right) \\
&\quad - (r \sin \phi + \zeta_{i,1}) \left(2\dot{\phi}\dot{\theta}_i \zeta'_{i,1} + \dot{\theta}_i^2 \zeta''_{i,3} + \ddot{\theta}_i \zeta'_{i,3} \right) \\
&= (g + r\dot{\phi}^2) (\zeta_{i,3} \sin \phi - \zeta_{i,1} \cos \phi) + (r \cos \phi + \zeta_{i,3}) \left(-2\dot{\phi}\dot{\theta}_i \zeta'_{i,3} + \dot{\theta}_i^2 \zeta''_{i,1} \right) \\
&\quad - (r \sin \phi + \zeta_{i,1}) \left(2\dot{\phi}\dot{\theta}_i \zeta'_{i,1} + \dot{\theta}_i^2 \zeta''_{i,3} \right) + [(r \cos \phi + \zeta_{i,3}) \zeta'_{i,1} - (r \sin \phi + \zeta_{i,1}) \zeta'_{i,3}] \ddot{\theta}_i.
\end{aligned} \tag{5.107}$$

Differentiating the Hamiltonian (5.105) with respect to the components of the control \mathbf{u} gives

$$H_{u_i} = H_{\dot{\theta}_i} = \gamma_i \ddot{\theta}_i + \lambda_{n+i} + \lambda_{2n+2} \frac{m_i [(r \cos \phi + \zeta_{i,3}) \zeta'_{i,1} - (r \sin \phi + \zeta_{i,1}) \zeta'_{i,3}]}{d_2 + \sum_{i=0}^n m_i \left[(r \sin \phi + \zeta_{i,1})^2 + (r \cos \phi + \zeta_{i,3})^2 \right]}, \tag{5.108}$$

$$H_{u_i u_j} = H_{\ddot{\theta}_i \ddot{\theta}_j} = \gamma_i \delta_{ij}, \quad (5.109)$$

and

$$H_{\mathbf{u}\mathbf{u}} = \text{diag} \left[\gamma_1 \quad \gamma_2 \quad \dots \quad \gamma_n \right]. \quad (5.110)$$

By (5.110), $H_{\mathbf{u}\mathbf{u}} > 0$ iff $\gamma_i > 0$ for all $1 \leq i \leq n$. Consequently, the optimal control problem is regular iff $\gamma_i > 0$ for all $1 \leq i \leq n$. Assume that the optimal control problem is regular, so that $\gamma_i > 0$ for all $1 \leq i \leq n$. $H_{\mathbf{u}} = 0$ iff $H_{u_i} = 0$ for all $1 \leq i \leq n$. From (5.108),

$$H_{u_i} = 0 \iff \ddot{\theta}_i = -\gamma_i^{-1} \left\{ \lambda_{n+i} + \lambda_{2n+2} \frac{m_i [(r \cos \phi + \zeta_{i,3}) \zeta'_{i,1} - (r \sin \phi + \zeta_{i,1}) \zeta'_{i,3}]}{d_2 + \sum_{i=0}^n m_i [(r \sin \phi + \zeta_{i,1})^2 + (r \cos \phi + \zeta_{i,3})^2]} \right\}. \quad (5.111)$$

(5.111) shows that the control $\ddot{\theta}$ may be expressed as a function $\boldsymbol{\pi}$ of \mathbf{x} , $\boldsymbol{\lambda}$, and μ ; to be consistent with the notation of Section 3.B, $\boldsymbol{\pi}$ will also depend on t even though in this particular example it does not. The regular Hamiltonian is

$$\begin{aligned} \hat{H}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) &= H(t, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu), \mu) \\ &= \frac{\alpha}{2} (z_a - r(\phi - \phi_a) - z_d)^2 + \frac{\beta}{2} (-r\dot{\phi})^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \pi_i^2(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) + \delta \\ &\quad + \boldsymbol{\lambda}^\top \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) \\ \dot{\phi} \\ \kappa(\mathbf{x}, \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu)) \end{bmatrix}. \end{aligned} \quad (5.112)$$

One way to solve the optimal control problem (5.103) for the rolling disk is to solve the ODE TPBVP given by (3.15), (3.16), and (3.17) using the endpoint function (5.104) and the regular Hamiltonian (5.112).

5.C.2 Controlled Equations of Motion for the Rolling Ball

Having derived the controlled equations of motion for the rolling disk, the controlled equations of motion are now developed for the rolling ball. Before proceeding, some useful terminology is defined or recalled. Given a vector

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = [v_1 \quad v_2 \quad v_3]^\top \in \mathbb{R}^3, \quad (5.113)$$

the projected vector consisting of the first two components of \mathbf{v} is

$$\mathbf{v}_{12} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = [v_1 \quad v_2]^\top \in \mathbb{R}^2. \quad (5.114)$$

Since a versor is used to parameterize the rolling ball's orientation matrix, quaternions and versors are briefly reviewed here; see Appendix E and the references mentioned there for a more complete review. \mathbb{H} denotes

the set of quaternions, which is isomorphic to \mathbb{R}^4 . A quaternion $\mathbf{p} \in \mathbb{H}$ can be expressed as the column vector

$$\mathbf{p} = \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 \end{bmatrix}^\top = [p_0 ; p_1 ; p_2 ; p_3]. \quad (5.115)$$

Given a column vector $\mathbf{v} \in \mathbb{R}^3$, \mathbf{v}^\sharp is the quaternion $[0 ; \mathbf{v}] \in \mathbb{H}$; that is,

$$\mathbf{v}^\sharp = \begin{bmatrix} 0 \\ \mathbf{v} \end{bmatrix} = [0 ; \mathbf{v}]. \quad (5.116)$$

Given a quaternion $\mathbf{p} \in \mathbb{H}$, $\mathbf{p}^b \in \mathbb{R}^3$ is the column vector such that

$$\mathbf{p} = \begin{bmatrix} p_0 \\ \mathbf{p}^b \end{bmatrix} = [p_0 ; \mathbf{p}^b]. \quad (5.117)$$

Given a column vector $\mathbf{v} \in \mathbb{R}^3$, note that

$$(\mathbf{v}^\sharp)^b = \mathbf{v}. \quad (5.118)$$

However, given a quaternion $\mathbf{p} \in \mathbb{H}$,

$$(\mathbf{p}^b)^\sharp = \mathbf{p} \quad \text{iff} \quad \mathbf{p} = \begin{bmatrix} 0 \\ \mathbf{p}^b \end{bmatrix} = [0 ; \mathbf{p}^b]. \quad (5.119)$$

Given quaternions $\mathbf{p}, \mathbf{q} \in \mathbb{H}$, their product is

$$\mathbf{p}\mathbf{q} = [p_0 ; \mathbf{p}^b] [q_0 ; \mathbf{q}^b] = [p_0q_0 - \mathbf{p}^b \cdot \mathbf{q}^b ; p_0\mathbf{q}^b + q_0\mathbf{p}^b + \mathbf{p}^b \times \mathbf{q}^b] \quad (5.120)$$

and their dot product is

$$\mathbf{p} \cdot \mathbf{q} = [p_0 ; \mathbf{p}^b] \cdot [q_0 ; \mathbf{q}^b] = [p_0 ; p_1 ; p_2 ; p_3] \cdot [q_0 ; q_1 ; q_2 ; q_3] = p_0q_0 + \mathbf{p}^b \cdot \mathbf{q}^b = p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3. \quad (5.121)$$

$\mathcal{S} \subset \mathbb{H}$ denotes the set of unit quaternions, also called versors, which is isomorphic to $\mathbb{S}^3 \subset \mathbb{R}^4$. A versor $\mathbf{q} \in \mathcal{S}$ can be expressed as the column vector

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^\top = [q_0 ; q_1 ; q_2 ; q_3] \quad \text{such that} \quad \mathbf{q} \cdot \mathbf{q} = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1. \quad (5.122)$$

The rolling ball's orientation matrix $\Lambda \in SO(3)$ is parameterized by the versor $\mathbf{q} \in \mathcal{S}$. If $\boldsymbol{\Omega} \in \mathbb{R}^3$ is the rolling ball's body angular velocity, then the time derivative of \mathbf{q} is

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q}\boldsymbol{\Omega}^\sharp. \quad (5.123)$$

If $\mathbf{Y} \in \mathbb{R}^3$ is a body frame vector, then the rotation of \mathbf{Y} by Λ is

$$\Lambda \mathbf{Y} = [\mathbf{q} \mathbf{Y}^\# \mathbf{q}^{-1}]^b. \quad (5.124)$$

If $\mathbf{y} \in \mathbb{R}^3$ is a spatial frame vector, then the rotation of \mathbf{y} by Λ^{-1} is

$$\Lambda^{-1} \mathbf{y} = [\mathbf{q}^{-1} \mathbf{y}^\# \mathbf{q}]^b. \quad (5.125)$$

Since the third coordinate of the location of the ball's GC is always 0, only the first two coordinates of the ball's GC are needed to determine the location of the ball's GC. The first two coordinates of the ball's GC are denoted by \mathbf{z} .

Suppose it is desired to roll the ball from some initial configuration at a prescribed or free initial time a to some final configuration at a prescribed or free final time b , without moving the control masses too rapidly along their control rails. In addition, in between the initial and final times, it may be desired that the ball's GC tracks a prescribed spatial \mathbf{e}_1 - \mathbf{e}_2 path \mathbf{z}_d or traces out a minimum energy path, all while avoiding some obstacles. Finally, if the initial or final time is free, it may be desired to minimize the duration $b - a$ of the maneuver. How must the control masses be moved in order to accomplish these tasks? This problem can be solved by posing it as an optimal control problem.

Concretely, at the prescribed or free initial time a , the positions of the control mass parameterizations are prescribed to be $\boldsymbol{\theta}(a) = \boldsymbol{\theta}_a$, the velocities of the control mass parameterizations are prescribed to be $\dot{\boldsymbol{\theta}}(a) = \dot{\boldsymbol{\theta}}_a$, the orientation of the ball is prescribed to be $\mathbf{q}(a) = \mathbf{q}_a$, the body angular velocity of the ball is prescribed to be $\boldsymbol{\Omega}(a) = \boldsymbol{\Omega}_a$, and the spatial \mathbf{e}_1 - \mathbf{e}_2 position of the ball's GC is prescribed to be $\mathbf{z}(a) = \mathbf{z}_a$.

Furthermore, at the prescribed or free final time b , some components (determined by the projection operator $\mathbf{\Pi}$) of the ball's center of mass expressed in the spatial frame translated to the GC are prescribed to be

$$\mathbf{\Pi} \left(\left[\mathbf{q}(b) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_i(b)) \right]^\# \mathbf{q}(b)^{-1} \right]^b \right) = \boldsymbol{\Delta}_b, \quad (5.126)$$

the velocities of the control mass parameterizations are prescribed to be $\dot{\boldsymbol{\theta}}(b) = \dot{\boldsymbol{\theta}}_b$, the body angular velocity of the ball is prescribed to be $\boldsymbol{\Omega}(b) = \boldsymbol{\Omega}_b$, and the spatial \mathbf{e}_1 - \mathbf{e}_2 position of the ball's GC is prescribed to be $\mathbf{z}(b) = \mathbf{z}_b$.

For example, if it is desired to start and stop the ball at rest, then $\mathbf{\Pi}$ is projection onto the first two components, $\boldsymbol{\Delta}_b = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $\boldsymbol{\theta}_a$ and \mathbf{q}_a are such that

$$\mathbf{\Pi} \left(\left[\mathbf{q}_a \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_{a,i}) \right]^\# \mathbf{q}_a^{-1} \right]^b \right) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad (5.127)$$

$\dot{\boldsymbol{\theta}}_a = \mathbf{0}$, $\boldsymbol{\Omega}_a = \mathbf{0}$, $\boldsymbol{\theta}(b)$ and $\mathbf{q}(b)$ are such that

$$\mathbf{\Pi} \left(\left[\mathbf{q}(b) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\boldsymbol{\theta}_i(b)) \right]^{\sharp} \mathbf{q}(b)^{-1} \right]^b \right) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad (5.128)$$

$\dot{\boldsymbol{\theta}}_b = \mathbf{0}$, and $\boldsymbol{\Omega}_b = \mathbf{0}$. With this choice of $\mathbf{\Pi}$, (5.127) and (5.128) mean that the CM in the spatial frame translated to the GC is above or below the GC at the initial and final times.

The system state \mathbf{x} and control \mathbf{u} are

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{\theta} \\ \dot{\boldsymbol{\theta}} \\ \mathbf{q} \\ \boldsymbol{\Omega} \\ \mathbf{z} \end{bmatrix} \quad \text{and} \quad \mathbf{u} = \ddot{\boldsymbol{\theta}}, \quad (5.129)$$

where $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$, $\ddot{\boldsymbol{\theta}} \in \mathbb{R}^n$, $\mathbf{q} \in \mathcal{S}$, $\boldsymbol{\Omega} \in \mathbb{R}^3$, and $\mathbf{z} \in \mathbb{R}^2$. The system dynamics defined for $a \leq t \leq b$ are

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\theta}} \\ \dot{\mathbf{q}} \\ \dot{\boldsymbol{\Omega}} \\ \dot{\mathbf{z}} \end{bmatrix} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) \equiv \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \mathbf{u} \\ \frac{1}{2} \mathbf{q} \boldsymbol{\Omega}^{\sharp} \\ \boldsymbol{\kappa}(\mathbf{x}, \mathbf{u}) \\ \left(\left[\mathbf{q} \boldsymbol{\Omega}^{\sharp} \mathbf{q}^{-1} \right]^b \times r \mathbf{e}_3 \right)_{12} \end{bmatrix}, \quad (5.130)$$

where $\boldsymbol{\kappa}(\mathbf{x}, \mathbf{u})$ is given by the right-hand side of the formula for $\dot{\boldsymbol{\Omega}}$ in (5.54):

$$\boldsymbol{\kappa}(\mathbf{x}, \mathbf{u}) = \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \left[\boldsymbol{\Omega} \times \mathbb{I} \boldsymbol{\Omega} + r \tilde{\boldsymbol{\Gamma}} \times \boldsymbol{\Gamma} + \sum_{i=0}^n m_i \mathbf{s}_i \times \left\{ g \boldsymbol{\Gamma} + \boldsymbol{\Omega} \times \left(\boldsymbol{\Omega} \times \boldsymbol{\zeta}_i + 2 \dot{\boldsymbol{\theta}}_i \boldsymbol{\zeta}'_i \right) + \dot{\boldsymbol{\theta}}_i^2 \boldsymbol{\zeta}''_i + \ddot{\boldsymbol{\theta}}_i \boldsymbol{\zeta}'_i \right\} \right]. \quad (5.131)$$

Note that in order to construct $\boldsymbol{\kappa}(\mathbf{x}, \mathbf{u})$, $\boldsymbol{\Gamma} = \Lambda^{-1} \mathbf{e}_3$ and $\tilde{\boldsymbol{\Gamma}} = \Lambda^{-1} \mathbf{F}_e$ must be constructed. Given \mathbf{q} , this can be accomplished by first constructing Λ from \mathbf{q} or directly from \mathbf{q} by using the formulas $\boldsymbol{\Gamma} = \Lambda^{-1} \mathbf{e}_3 = \left[\mathbf{q}^{-1} \mathbf{e}_3^{\sharp} \mathbf{q} \right]^b$ and $\tilde{\boldsymbol{\Gamma}} = \Lambda^{-1} \mathbf{F}_e = \left[\mathbf{q}^{-1} \mathbf{F}_e^{\sharp} \mathbf{q} \right]^b$. Likewise, the final formula in (5.130) is $\dot{\mathbf{z}} = (\boldsymbol{\omega} \times r \mathbf{e}_3)_{12}$, where $\boldsymbol{\omega} = \Lambda \boldsymbol{\Omega} = \left[\mathbf{q} \boldsymbol{\Omega}^{\sharp} \mathbf{q}^{-1} \right]^b$. Thus, given \mathbf{q} , $\boldsymbol{\omega}$ can be constructed by first constructing Λ from \mathbf{q} or directly from \mathbf{q} via $\boldsymbol{\omega} = \left[\mathbf{q} \boldsymbol{\Omega}^{\sharp} \mathbf{q}^{-1} \right]^b$. The most computationally efficient method to construct $\boldsymbol{\Gamma} = \Lambda^{-1} \mathbf{e}_3 = \left[\mathbf{q}^{-1} \mathbf{e}_3^{\sharp} \mathbf{q} \right]^b$, $\tilde{\boldsymbol{\Gamma}} = \Lambda^{-1} \mathbf{F}_e = \left[\mathbf{q}^{-1} \mathbf{F}_e^{\sharp} \mathbf{q} \right]^b$, and $\boldsymbol{\omega} = \Lambda \boldsymbol{\Omega} = \left[\mathbf{q} \boldsymbol{\Omega}^{\sharp} \mathbf{q}^{-1} \right]^b$ is to construct Λ from \mathbf{q} and to then multiply $\Lambda^{-1} = \Lambda^{\top}$ against \mathbf{e}_3 and \mathbf{F}_e and to multiply Λ against $\boldsymbol{\Omega}$.

The prescribed initial conditions at time $t = a$ are

$$\boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \begin{bmatrix} \boldsymbol{\theta}(a) - \boldsymbol{\theta}_a \\ \dot{\boldsymbol{\theta}}(a) - \dot{\boldsymbol{\theta}}_a \\ \mathbf{q}(a) - \mathbf{q}_a \\ \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \\ \mathbf{z}(a) - \mathbf{z}_a \end{bmatrix} = \mathbf{0}, \quad (5.132)$$

and the prescribed final conditions at time $t = b$ are

$$\boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \begin{bmatrix} \boldsymbol{\Pi} \left(\left[\mathbf{q}(b) \left[\frac{1}{M} \sum_{i=0}^n m_i \zeta_i(\theta_i(b)) \right]^\# \mathbf{q}(b)^{-1} \right]^b \right) - \boldsymbol{\Delta}_b \\ \dot{\boldsymbol{\theta}}(b) - \dot{\boldsymbol{\theta}}_b \\ \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \\ \mathbf{z}(b) - \mathbf{z}_b \end{bmatrix} = \mathbf{0}. \quad (5.133)$$

Consider the endpoint and integrand cost functions

$$p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) = \frac{v_a}{2} (a - a_e)^2 + \frac{v_b}{2} (b - b_e)^2 \quad (5.134)$$

and

$$L(t, \mathbf{x}, \mathbf{u}, \mu) = \frac{\alpha}{2} |\mathbf{z} - \mathbf{z}_d|^2 + \frac{\beta}{2} \left| \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^b \times r \mathbf{e}_3 \right)_{12} \right|^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \ddot{\theta}_i^2 + \sum_{j=1}^K V_j(\mathbf{z}, \mu) + \delta, \quad (5.135)$$

for constants a_e and b_e and for fixed nonnegative constants v_a , v_b , α , β , γ_i , $1 \leq i \leq n$, and δ so that the performance index is

$$\begin{aligned} J &= p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \int_a^b L(t, \mathbf{x}, \mathbf{u}, \mu) dt \\ &= \frac{v_a}{2} (a - a_e)^2 + \frac{v_b}{2} (b - b_e)^2 \\ &\quad + \int_a^b \left[\frac{\alpha}{2} |\mathbf{z} - \mathbf{z}_d|^2 + \frac{\beta}{2} \left| \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^b \times r \mathbf{e}_3 \right)_{12} \right|^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \ddot{\theta}_i^2 + \sum_{j=1}^K V_j(\mathbf{z}, \mu) + \delta \right] dt. \end{aligned} \quad (5.136)$$

The first summand $\frac{v_a}{2} (a - a_e)^2$ in p encourages the initial time a to be near a_e if the initial time is free, while the second summand $\frac{v_b}{2} (b - b_e)^2$ in p encourages the final time b to be near b_e if the final time is free. The first summand $\frac{\alpha}{2} |\mathbf{z} - \mathbf{z}_d|^2$ in L encourages the ball's GC to track the desired spatial \mathbf{e}_1 - \mathbf{e}_2 path \mathbf{z}_d , the second summand $\frac{\beta}{2} \left| \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^b \times r \mathbf{e}_3 \right)_{12} \right|^2$ in L encourages the ball's GC to track a minimum energy path, the next n summands $\frac{\gamma_i}{2} \ddot{\theta}_i^2$, $1 \leq i \leq n$, in L limit the magnitude of the acceleration of the i^{th} control mass parameterization, the next K summands $V_j(\mathbf{z}, \mu)$, $1 \leq j \leq K$, in L represent obstacles to be avoided, and the final summand δ in L encourages a minimum time maneuver.

For example, the desired spatial \mathbf{e}_1 - \mathbf{e}_2 path might have the form

$$\mathbf{z}_d(t) = [\mathbf{z}_a w(t) + \tilde{\mathbf{z}}_d(t) (1 - w(t))] (1 - y(t)) + \mathbf{z}_b y(t), \quad (5.137)$$

where w , y , and S are given by (5.99), (5.100), and (5.98), respectively. \mathbf{z}_d (5.137) holds steady at \mathbf{z}_a for $t < a_e$, smoothly transitions between \mathbf{z}_a and $\tilde{\mathbf{z}}_d$ at $t = a_e$, follows $\tilde{\mathbf{z}}_d$ for $a_e < t < b_e$, smoothly transitions between $\tilde{\mathbf{z}}_d$ and \mathbf{z}_b at $t = b_e$, and holds steady at \mathbf{z}_b for $b_e < t$. $\tilde{\mathbf{z}}_d$, which appears in (5.137), might be

$$\tilde{\mathbf{z}}_d(t) = \mathbf{k}_1 \left[\left\{ -\frac{1}{3}t^3 + \frac{1}{2}(a_e + b_e)t - a_e b_e t \right\} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \mathbf{k}_2 \right] = \mathbf{k}_1 \left[q(t) \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \mathbf{k}_2 \right], \quad (5.138)$$

where

$$q(t) = -\frac{1}{3}t^3 + \frac{1}{2}(a_e + b_e)t - a_e b_e t, \quad \mathbf{k}_1 = \frac{\mathbf{z}_b - \mathbf{z}_a}{q(b_e) - q(a_e)}, \quad \text{and} \quad \mathbf{k}_2 = \frac{\mathbf{z}_b}{\mathbf{k}_1} - q(b_e) \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \quad (5.139)$$

The multiplication between the vectors \mathbf{k}_1 and $q(t) \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \mathbf{k}_2$ in (5.138) is meant to be performed component-wise. The division $\frac{\mathbf{z}_b}{\mathbf{k}_1}$ used in the construction of \mathbf{k}_2 in (5.139) is meant to be performed component-wise; to avoid division by zero, if a component of \mathbf{k}_1 is zero, then the corresponding component of \mathbf{k}_2 is set to zero. $\tilde{\mathbf{z}}_d$ has the special properties $\tilde{\mathbf{z}}_d(a_e) = \mathbf{z}_a$, $\dot{\tilde{\mathbf{z}}}_d(a_e) = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$, $\tilde{\mathbf{z}}_d(b_e) = \mathbf{z}_b$, and $\dot{\tilde{\mathbf{z}}}_d(b_e) = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$, so that the disk's GC is encouraged to start with zero velocity at \mathbf{z}_a at $t = a_e$ and to stop with zero velocity at \mathbf{z}_b at $t = b_e$. Figure 5.8 illustrates (5.137) using (5.138) with $a_e = 0$, $\mathbf{z}_a = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$, $b_e = .5$, $\mathbf{z}_b = \begin{bmatrix} 1 & 1 \end{bmatrix}^\top$, and $\epsilon = .01$.

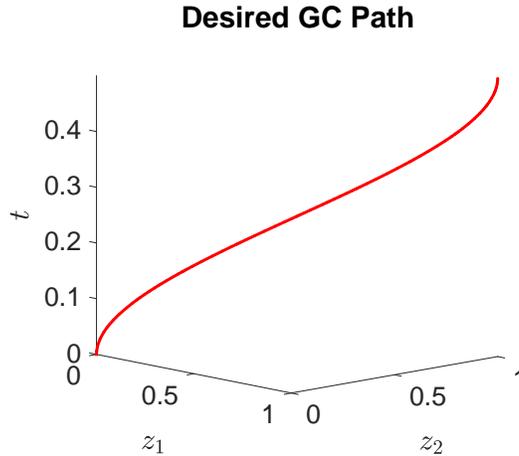


Figure 5.8: Plot of the desired path of the ball's GC. The ball's GC starts from rest at $\mathbf{z} = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$ at time $t = 0$ and stops at rest at $\mathbf{z} = \begin{bmatrix} 1 & 1 \end{bmatrix}^\top$ at time $t = .5$.

Illustratively, for $1 \leq j \leq K$, the j^{th} obstacle of height h_j and radius ρ_j with center at spatial \mathbf{e}_1 - \mathbf{e}_2 position $\mathbf{v}_j = \begin{bmatrix} v_{j,1} & v_{j,2} \end{bmatrix}^\top$ might be modeled via the function

$$V_j(\mathbf{z}, \mu) = h_j S \left(\sqrt{(z_1 - v_{j,1})^2 + (z_2 - v_{j,2})^2} - \rho_j \right), \quad (5.140)$$

where S is given by (5.98). As indicated by (5.140), the radial distance from the ball's GC to the obstacle center should exceed the obstacle radius ρ_j for successful obstacle avoidance. In order to encourage the entire ball to avoid the obstacle, the obstacle radius ρ_j must include the ball's radius r . That is, if the physical radius of the obstacle is ϵ_j , then set $\rho_j = r + \epsilon_j$ to encourage the entire ball to stay away from the obstacle; if $\rho_j = \epsilon_j$, then only the ball's GC is encouraged to stay away from the obstacle.

The ODE formulation of the optimal control problem for the rolling ball is

$$\min_{a,b,\mathbf{u}} J \text{ s.t. } \begin{cases} \dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu), \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}. \end{cases} \quad (5.141)$$

There are also two DAE formulations of the optimal control problem for the rolling ball which explicitly enforce the algebraic versor constraint on \mathbf{q} and which are mathematically equivalent to (5.141). In the first DAE formulation an additional control, \dot{q}_0 , is added to the control \mathbf{u} . The first DAE formulation is

$$\min_{a,b,\mathbf{u}_1} J \text{ s.t. } \begin{cases} \dot{\mathbf{x}} = \mathbf{f}_1(t, \mathbf{x}, \mathbf{u}_1, \mu), \\ h_1(\mathbf{x}) = 1, \\ \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) = \mathbf{0}, \\ \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) = \mathbf{0}, \end{cases} \quad (5.142)$$

where

$$\mathbf{u}_1 = \begin{bmatrix} \ddot{\boldsymbol{\theta}} \\ \dot{q}_0 \end{bmatrix}, \quad \mathbf{f}_1(t, \mathbf{x}, \mathbf{u}_1, \mu) \equiv \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \mathbf{u}_1 \\ \frac{1}{2} \left(\mathbf{q} \boldsymbol{\Omega}^\# \right)^b \\ \boldsymbol{\kappa}(\mathbf{x}, \dot{\boldsymbol{\theta}}) \\ \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^b \times r \mathbf{e}_3 \right)_{12} \end{bmatrix}, \quad \text{and} \quad h_1(\mathbf{x}) = \mathbf{q} \cdot \mathbf{q}. \quad (5.143)$$

In the second DAE formulation, the first component, q_0 , of the versor \mathbf{q} is moved from the state \mathbf{x} to the control \mathbf{u} and an imitator state, \tilde{q}_0 , is used to replace q_0 in \mathbf{x} . $\tilde{q}_{a,0} = q_{a,0}$, so that with perfect integration (i.e. no numerical integration errors), $\tilde{q}_0(t) = q_0(t)$ for $a \leq t \leq b$. Define $\tilde{\mathbf{q}} = \begin{bmatrix} \tilde{q}_0 \\ \mathbf{q}^b \end{bmatrix}$; with perfect integration, $\tilde{\mathbf{q}}(t) = \begin{bmatrix} \tilde{q}_0(t) \\ \mathbf{q}^b(t) \end{bmatrix} = \begin{bmatrix} q_0(t) \\ \mathbf{q}^b(t) \end{bmatrix} = \mathbf{q}(t)$ for $a \leq t \leq b$. \tilde{q}_0 is added to the state since the final conditions require knowledge of q_0 , which is unavailable if it has been moved to the control since the final conditions are not a function of the control. The second DAE formulation is

$$\min_{a,b,\mathbf{u}_2} J \text{ s.t. } \begin{cases} \dot{\mathbf{x}}_2 = \mathbf{f}_2(t, \mathbf{x}_2, \mathbf{u}_2, \mu), \\ h_2(\mathbf{x}_2, \mathbf{u}_2) = 1, \\ \boldsymbol{\sigma}_2(a, \mathbf{x}_2(a), \mu) = \mathbf{0}, \\ \boldsymbol{\psi}_2(b, \mathbf{x}_2(b), \mu) = \mathbf{0}, \end{cases} \quad (5.144)$$

where

$$\mathbf{x}_2 = \begin{bmatrix} \boldsymbol{\theta} \\ \dot{\boldsymbol{\theta}} \\ \tilde{q}_0 \\ \mathbf{q}^b \\ \boldsymbol{\Omega} \\ z \end{bmatrix}, \quad \mathbf{u}_2 = \begin{bmatrix} \ddot{\boldsymbol{\theta}} \\ q_0 \end{bmatrix}, \quad \mathbf{f}_2(t, \mathbf{x}_2, \mathbf{u}_2, \mu) \equiv \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\theta}} \\ \frac{1}{2} \mathbf{q} \boldsymbol{\Omega}^\# \\ \boldsymbol{\kappa}(\mathbf{x}, \ddot{\boldsymbol{\theta}}) \\ \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^b \times r \mathbf{e}_3 \right)_{12} \end{bmatrix}, \quad h_2(\mathbf{x}_2, \mathbf{u}_2) = \mathbf{q} \cdot \mathbf{q}, \quad (5.145)$$

$$\boldsymbol{\sigma}_2(a, \mathbf{x}_2(a), \mu) = \begin{bmatrix} \boldsymbol{\theta}(a) - \boldsymbol{\theta}_a \\ \dot{\boldsymbol{\theta}}(a) - \dot{\boldsymbol{\theta}}_a \\ \tilde{q}(a) - q_a \\ \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \\ z(a) - z_a \end{bmatrix} = \mathbf{0}, \quad (5.146)$$

and

$$\boldsymbol{\psi}_2(b, \mathbf{x}_2(b), \mu) = \begin{bmatrix} \boldsymbol{\Pi} \left(\left[\tilde{q}(b) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_i(b)) \right]^\# \tilde{q}(b)^{-1} \right]^b \right) - \boldsymbol{\Delta}_b \\ \dot{\boldsymbol{\theta}}(b) - \dot{\boldsymbol{\theta}}_b \\ \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \\ z(b) - z_b \end{bmatrix} = \mathbf{0}. \quad (5.147)$$

Even though both DAE formulations (5.142) and (5.144) are mathematically equivalent to the ODE formulation (5.141), the DAE formulations (5.142) and (5.144) tend to be numerically more stable to solve than the ODE formulation (5.141), as explained in Example 6.12 “Reorientation of an Asymmetric Rigid Body” of [54]. While the second DAE formulation (5.144) is computationally more efficient (i.e. faster) than the first (5.142) because it explicitly constructs the control q_0 rather than \dot{q}_0 , the second DAE formulation (5.144) is not as accurate as the first (5.142), because it only constructs an approximation, $\tilde{q}_0(b)$, of $q_0(b)$, which is needed for the final conditions. The direct method was used to solve all three formulations of the optimal control problem for the rolling ball. Because DAE TPBVP solvers are not readily available in MATLAB, the indirect method was only applied to the ODE formulation (5.141).

Observe that the optimal control problem encapsulated by (5.141), (5.142), and (5.144) ignores path inequality constraints such as $\mathbf{D}(t, \mathbf{x}, \mathbf{u}, \mu) \leq \mathbf{0}$, $\mathbf{D}_1(t, \mathbf{x}_1, \mathbf{u}, \mu) \leq \mathbf{0}$, and $\mathbf{D}_2(t, \mathbf{x}_2, \mathbf{u}_2, \mu) \leq \mathbf{0}$, where \mathbf{D} , \mathbf{D}_1 , and \mathbf{D}_2 are $r \times 1$ vector-valued functions. Path inequality constraints can be incorporated in (5.141), (5.142), and (5.144) as soft constraints through penalty functions in the integrand cost function L or the endpoint cost function p .

The indirect method, discussed in Section 3.B, is applied now to (5.141) to construct the endpoint function and regular Hamiltonian needed to formulate the ODE TPBVP (3.15), (3.16), and (3.17), which render the controlled equations of motion for the rolling ball.

The endpoint function is

$$\begin{aligned}
G(a, \mathbf{x}(a), \boldsymbol{\xi}, b, \mathbf{x}(b), \boldsymbol{\nu}, \mu) &= p(a, \mathbf{x}(a), b, \mathbf{x}(b), \mu) + \boldsymbol{\xi}^\top \boldsymbol{\sigma}(a, \mathbf{x}(a), \mu) + \boldsymbol{\nu}^\top \boldsymbol{\psi}(b, \mathbf{x}(b), \mu) \\
&= \frac{v_a}{2} (a - a_e)^2 + \frac{v_b}{2} (b - b_e)^2 + \boldsymbol{\xi}^\top \begin{bmatrix} \boldsymbol{\theta}(a) - \boldsymbol{\theta}_a \\ \dot{\boldsymbol{\theta}}(a) - \dot{\boldsymbol{\theta}}_a \\ \mathbf{q}(a) - \mathbf{q}_a \\ \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \\ \mathbf{z}(a) - \mathbf{z}_a \end{bmatrix} \\
&\quad + \boldsymbol{\nu}^\top \begin{bmatrix} \boldsymbol{\Pi} \left(\left[\mathbf{q}(b) \left[\frac{1}{M} \sum_{i=0}^n m_i \boldsymbol{\zeta}_i(\theta_i(b)) \right]^\# \mathbf{q}(b)^{-1} \right]^\flat \right) - \boldsymbol{\Delta}_b \\ \dot{\boldsymbol{\theta}}(b) - \dot{\boldsymbol{\theta}}_b \\ \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \\ \mathbf{z}(b) - \mathbf{z}_b \end{bmatrix}
\end{aligned} \tag{5.148}$$

and the Hamiltonian is

$$\begin{aligned}
H(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}, \mu) &= L(t, \mathbf{x}, \mathbf{u}, \mu) + \boldsymbol{\lambda}^\top \mathbf{f}(t, \mathbf{x}, \mathbf{u}, \mu) \\
&= \frac{\alpha}{2} |\mathbf{z} - \mathbf{z}_d|^2 + \frac{\beta}{2} \left| \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^\flat \times r \mathbf{e}_3 \right)_{12} \right|^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \dot{\theta}_i^2 + \sum_{j=1}^K V_j(\mathbf{z}, \mu) + \delta \\
&\quad + \boldsymbol{\lambda}^\top \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \mathbf{u} \\ \frac{1}{2} \mathbf{q} \boldsymbol{\Omega}^\# \\ \boldsymbol{\kappa}(\mathbf{x}, \mathbf{u}) \\ \left(\left[\mathbf{q} \boldsymbol{\Omega}^\# \mathbf{q}^{-1} \right]^\flat \times r \mathbf{e}_3 \right)_{12} \end{bmatrix}.
\end{aligned} \tag{5.149}$$

Let $\boldsymbol{\lambda}_\Omega = [\lambda_{2n+5} \quad \lambda_{2n+6} \quad \lambda_{2n+7}]^\top$. Recall that

$$\begin{aligned}
\boldsymbol{\kappa}(\mathbf{x}, \mathbf{u}) &= \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} \left[\boldsymbol{\Omega} \times \mathbb{I} \boldsymbol{\Omega} + r \tilde{\boldsymbol{\Gamma}} \times \boldsymbol{\Gamma} \right. \\
&\quad \left. + \sum_{i=0}^n m_i \mathbf{s}_i \times \left\{ g \boldsymbol{\Gamma} + \boldsymbol{\Omega} \times \left(\boldsymbol{\Omega} \times \boldsymbol{\zeta}_i + 2\dot{\theta}_i \boldsymbol{\zeta}'_i \right) + \dot{\theta}_i^2 \boldsymbol{\zeta}''_i + \ddot{\theta}_i \boldsymbol{\zeta}'_i \right\} \right].
\end{aligned} \tag{5.150}$$

Differentiating the Hamiltonian (5.149) with respect to the components of the control \mathbf{u} gives

$$H_{u_i} = H_{\dot{\theta}_i} = \gamma_i \ddot{\theta}_i + \lambda_{n+i} + \boldsymbol{\lambda}_\Omega^\top \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} [m_i \mathbf{s}_i \times \boldsymbol{\zeta}'_i], \tag{5.151}$$

$$H_{u_i u_j} = H_{\dot{\theta}_i \dot{\theta}_j} = \gamma_i \delta_{ij}, \tag{5.152}$$

and

$$H_{\mathbf{u}\mathbf{u}} = \text{diag} [\gamma_1 \quad \gamma_2 \quad \dots \quad \gamma_n]. \tag{5.153}$$

By (5.153), $H_{\mathbf{u}\mathbf{u}} > 0$ iff $\gamma_i > 0$ for all $1 \leq i \leq n$. Consequently, the optimal control problem is regular iff

$\gamma_i > 0$ for all $1 \leq i \leq n$. Assume that the optimal control problem is regular, so that $\gamma_i > 0$ for all $1 \leq i \leq n$. $H_{\mathbf{u}} = 0$ iff $H_{u_i} = 0$ for all $1 \leq i \leq n$. From (5.151),

$$H_{u_i} = 0 \iff \ddot{\theta}_i = -\gamma_i^{-1} \left\{ \lambda_{n+i} + \boldsymbol{\lambda}_{\Omega}^{\top} \left[\sum_{i=0}^n m_i \hat{\mathbf{s}}_i^2 - \mathbb{I} \right]^{-1} [m_i \mathbf{s}_i \times \boldsymbol{\zeta}'_i] \right\}. \quad (5.154)$$

(5.154) shows that $\ddot{\boldsymbol{\theta}}$ may be expressed as a function $\boldsymbol{\pi}$ of \mathbf{x} , $\boldsymbol{\lambda}$, and μ ; to be consistent with the notation of Section 3.B, $\boldsymbol{\pi}$ will also depend on t even though in this particular example it does not. The regular Hamiltonian is

$$\begin{aligned} \hat{H}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) &= H(t, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu), \mu) \\ &= \frac{\alpha}{2} |\mathbf{z} - \mathbf{z}_d|^2 + \frac{\beta}{2} \left| \left([\mathbf{q}\boldsymbol{\Omega}^{\#}\mathbf{q}^{-1}]^b \times \mathbf{r}\mathbf{e}_3 \right)_{12} \right|^2 + \sum_{i=1}^n \frac{\gamma_i}{2} \pi_i^2(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) + \sum_{j=1}^K V_j(\mathbf{z}, \mu) + \delta \\ &\quad + \boldsymbol{\lambda}^{\top} \begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu) \\ \frac{1}{2} \mathbf{q}\boldsymbol{\Omega}^{\#} \\ \boldsymbol{\kappa}(\mathbf{x}, \boldsymbol{\pi}(t, \mathbf{x}, \boldsymbol{\lambda}, \mu)) \\ \left([\mathbf{q}\boldsymbol{\Omega}^{\#}\mathbf{q}^{-1}]^b \times \mathbf{r}\mathbf{e}_3 \right)_{12} \end{bmatrix}. \end{aligned} \quad (5.155)$$

One way to solve the optimal control problem (5.141) for the rolling ball is to solve the ODE TPBVP given by (3.15), (3.16), and (3.17) using the endpoint function (5.148) and the regular Hamiltonian (5.155). These controlled equations of motion for the rolling ball actuated by internal point masses that move along rails fixed within the ball are new and have not appeared previously in the literature, as far as the author knows. Note that these controlled equations of motion track the ball's orientation (i.e. the mapping from body to spatial frames) and involve many specialized parameters and terms that enable the ball to execute a wide variety of interesting and useful maneuvers. These controlled equations of motion constitute a novel contribution of this thesis.

5.D Numerical Solutions of the Controlled Equations of Motion

In this section, the motion of the rolling disk and ball is simulated by numerically solving the controlled equations of motion (3.15), (3.16), and (3.17) corresponding to the optimal control problems (5.103), for the rolling disk, and (5.141), for the rolling ball. Subsection 5.D.1 simulates the rolling disk, while Subsection 5.D.2 simulates the rolling ball. Because the controlled equations of motion have a very small radius of convergence, a direct method, namely the MATLAB toolbox GPOPS-II [55], is first used to construct a good initial guess. For the rolling disk, the direct method is used to solve the rolling disk optimal control problem (5.103). When using the direct method to solve the rolling ball optimal control problem, one of the DAE formulations (5.142) or (5.144) is solved first. The direct method solution to the DAE formulation is then used as an initial guess to solve the ODE formulation (5.141), which is consistent with the controlled equations of motion for the rolling ball, by the direct method; recall that the controlled equations of motion for

the rolling ball were obtained from the ODE formulation (5.141) of the rolling ball optimal control problem. The MATLAB automatic differentiation toolbox ADiGator [40, 41] is used to supply vectorized first and second derivatives (i.e. Jacobians and Hessians) to the direct method solver GPOPS-II.

Starting from the initial guess provided by the direct method, the controlled equations of motion (3.15), (3.16), and (3.17) are solved by predictor-corrector continuation in the parameter μ , as discussed in Appendices C and D. The MATLAB ODE TPBVP solvers `sbvp` [56] or `bvptwp` [57] are utilized by the predictor-corrector continuation method. By vectorized automatic differentiation of the regular Hamiltonian \hat{H} and non-vectorized automatic differentiation of the endpoint function G , ADiGator is used to numerically construct the normalized ODE velocity function (3.21), the Jacobians of the normalized ODE velocity function (3.22) and (3.23), the normalized BC function (3.39), and the Jacobians of the normalized BC function (3.41), (3.42), and (3.43), which are needed by the ODE TPBVP solvers `sbvp` and `bvptwp` to solve the controlled equations of motion (3.15), (3.16), and (3.17) by predictor-corrector continuation in the parameter μ .

In contrast to the direct method, the controlled equations of motion obtained via the indirect method have a very small radius of convergence. Therefore, the direct method is needed to initialize the predictor-corrector continuation of the controlled equations of motion. Predictor-corrector continuation is used in conjunction with the indirect, rather than direct, method, because a predictor-corrector continuation direct method requires a predictor-corrector continuation NLP solver. Even though predictor-corrector continuation NLP solver algorithms are provided in [58, 59], there do not seem to be any publicly available predictor-corrector continuation NLP solvers.

5.D.1 Simulations of the Rolling Disk

Numerical solutions of the controlled equations of motion for the rolling disk are presented here. A rolling disk of mass $m_0 = 1$, radius $r = 1$, polar moment of inertia $d_2 = 1$, and with the CM coinciding with the GC (i.e. $\zeta_0 = \mathbf{0}$) is simulated. There are $n = 4$ control masses, each of mass 1 so that $m_1 = m_2 = m_3 = m_4 = 1$, located on concentric circles centered on the GC of radii $r_1 = .9$, $r_2 = .6\bar{3}$, $r_3 = .3\bar{6}$, and $r_4 = .1$, as shown in Figure 5.9. The total system mass is $M = 5$. Gravity is $g = 1$. The initial time is fixed to $a = a_e = 0$ and the final time is fixed to $b = b_e = 2$. The disk's GC starts at rest at $z_a = 0$ at time $a = a_e = 0$ and stops at rest at $z_b = 1$ at time $b = b_e = 2$. Table 5.1 shows parameter values used in the rolling disk's initial conditions (5.92) and final conditions (5.93). Since the initial orientation of the disk is $\phi_a = 0$ and since the initial configurations of the control masses are given by $\theta_a = \left[-\frac{\pi}{2} \quad -\frac{\pi}{2} \quad -\frac{\pi}{2} \quad -\frac{\pi}{2}\right]^T$, all the control masses are initially located directly below the GC. Table 5.2 shows parameter values used in the rolling ball's final conditions (5.93). Moreover, for the final conditions (5.93), Π is projection onto the first coordinate. Δ_b and $\mathbf{\Pi}$ are selected so that the total system CM in the spatial frame translated to the GC is located above or below the GC at the final time b .

The desired GC path z_d in the integrand cost function (5.95) is depicted by the red curve in Figures 5.10a and 5.10b. z_d encourages the disk's GC to track a sinusoidally-modulated linear trajectory connecting $z = 0$ at $t = 0$ with $z = 1$ at $t = 2$. That is, the disk is encouraged to roll right, then left, then right, then left, and finally to the right, with the amplitude of each successive roll increasing from the previous one. Specifically,

Parameter	Value
θ_a	$\left[-\frac{\pi}{2} \quad -\frac{\pi}{2} \quad -\frac{\pi}{2} \quad -\frac{\pi}{2}\right]^\top$
$\dot{\theta}_a$	$\left[0 \quad 0 \quad 0 \quad 0\right]^\top$
ϕ_a	0
z_a	0
\dot{z}_a	0

Table 5.1: Initial condition parameter values for the rolling disk. Refer to (5.92) and (5.93).

Parameter	Value
Δ_b	0
$\dot{\theta}_b$	$\left[0 \quad 0 \quad 0 \quad 0\right]^\top$
z_b	1
\dot{z}_b	0

Table 5.2: Final condition parameter values for the rolling disk. Refer to (5.93).

z_d is given by (5.97), with $\epsilon = .01$ in (5.98) and where \tilde{z}_d is given by

$$\tilde{z}_d(t) = \left[z_a + (z_b - z_a) \frac{t - a_e}{b_e - a_e} \right] \sin\left(\frac{9\pi}{2} \frac{t - a_e}{b_e - a_e}\right). \quad (5.156)$$

Table 5.3 shows the values set for the integrand cost function coefficients in (5.95). Since the initial and final times are fixed, the integrand cost function coefficient δ in (5.95) and the endpoint cost function coefficients v_a and v_b in (5.94) are irrelevant. There is no external force acting on the disk's GC, so that $F_{e,1} = 0$ in (5.106).

Parameter	Value
$\alpha(\mu)$	$20 + \frac{.95 - \mu}{.95 - .00001} (5000 - 20)$
β	0
γ_1	.1
γ_2	.1
γ_3	.1
γ_4	.1

Table 5.3: Integrand cost function coefficient values for the rolling disk. Refer to (5.95).

The direct method solver GPOPS-II is used to solve the optimal control problem (5.103) when the integrand cost function coefficient is $\alpha = 20$. The predictor-corrector continuation indirect method is then used to solve the optimal control problem (5.103), starting from the direct method solution. The continuation parameter is μ , which is used to adjust α according to the linear homotopy given in Table 5.3, so that $\alpha = 20$ when $\mu = .95$ and $\alpha = 5,000$ when $\mu = .00001$. The predictor-corrector continuation indirect method begins at $\mu = .95$, which is consistent with the direct method solution obtained at $\alpha = 20$.

For the direct method, GPOPS-II was run using the IPOPT NLP solver with the MA57 linear solver. The

GPOPS-II mesh tolerance is $1e-6$ and the IPOPT error tolerance is $1e-7$. The sweep predictor-corrector method discussed in Appendix D was used by the indirect method. For the sweep predictor-corrector continuation method, the maximum tangent steplength is $\sigma_{\max} = 30$, there are 250 predictor-corrector steps, the direction of the initial unit tangent is determined by setting $d = -2$ to force the continuation parameter μ to initially decrease, the relative error tolerance is $1e-8$, the unit tangent solver is `twpbvpc_m`, and the monotonic “sweep” continuation solver is `acdcc`. The numerical results are shown in Figures 5.10 and 5.11. As μ decreases from .95 below -8 during continuation (see Figure 5.11a), α increases from 20 up to 54,866 (see Figure 5.11c). Since α is ratcheted up during continuation, thereby increasing the penalty in the integrand cost function (5.95) for deviation between the disk’s GC and z_d , by the end of continuation, the disk’s GC tracks z_d much more accurately (compare Figures 5.10a vs 5.10b), at the expense of wild control mass trajectories (compare Figures 5.10c vs 5.10d) and large magnitude controls (compare Figures 5.10e vs 5.10f). Note the turning points at solutions 14 and 19 in Figures 5.11a-5.11d. In Figure 5.11b, note that the predictor-corrector continuation method has to reduce the tangent steplength below $\sigma_{\max} = 30$ to get around the turning points encountered at solutions 14 and 19.

**Disk, Control Masses, and Control Rails
in the Body Frame Translated to the GC**

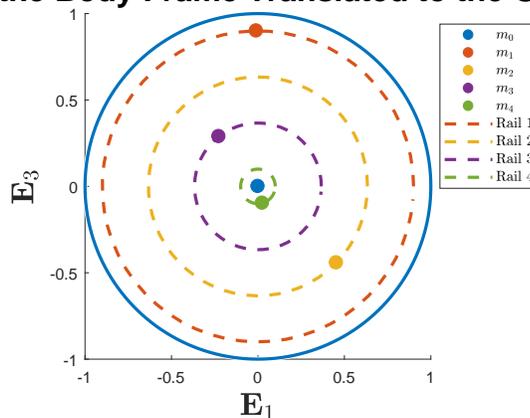


Figure 5.9: The disk of radius $r = 1$ actuated by 4 control masses, m_1 , m_2 , m_3 , and m_4 , each on its own circular control rail. The control rail radii are $r_1 = .9$, $r_2 = .63$, $r_3 = .36$, and $r_4 = .1$. The location of the disk’s CM is denoted by m_0 .

Based on the turning points, the disk is re-simulated using different parameters for the sweep predictor-corrector method in order to ratchet up α even more while decreasing the total simulation runtime. In particular, the maximum tangent steplength used by the sweep predictor-corrector method is adjusted based on the location of the turning points revealed by the previous simulation. Until the turning points (which occur at solutions 14 and 19) are passed, $\sigma_{\max} = 30$, after which σ_{\max} is increased linearly to a maximum of 3,000, as depicted in Figure 5.13b. Because σ_{\max} is increased dramatically, only 42 predictor-corrector steps are executed. The numerical results are shown in Figures 5.12 and 5.13. As μ decreases from .95 below -200 during continuation (see Figure 5.13a), α increases from 20 up to 1,238,285 (see Figure 5.13c). Since α is ratcheted up during continuation, thereby increasing the penalty in the integrand cost function (5.95) for deviation between the disk’s GC and z_d , by the end of continuation, the disk’s GC tracks z_d very accurately (compare Figures 5.12a vs 5.12b), at the expense of extremely wild control mass trajectories (compare Figures 5.12c vs 5.12d) and large magnitude controls (compare Figures 5.12e vs 5.12f).

5.D.2 Simulations of the Rolling Ball

Numerical solutions of the controlled equations of motion for the rolling ball are presented here. A rolling ball of mass $m_0 = 1$, radius $r = 1$, principle moments of inertia $d_1 = d_2 = d_3 = 1$, and with the CM coinciding with the GC (i.e. $\zeta_0 = \mathbf{0}$) is simulated. There are $n = 3$ control masses, each of mass 1 so that $m_1 = m_2 = m_3 = 1$, located on circular control rails centered on the GC of radii $r_1 = .95$, $r_2 = .9$, and $r_3 = .85$, oriented as shown in Figure 5.14. The total system mass is $M = 4$. Gravity is $g = 1$. The initial time is fixed to $a = a_e = 0$ and the final time is fixed to $b = b_e = .5$. The ball's GC starts at rest at $\mathbf{z}_a = [0 \ 0]^\top$ at time $a = a_e = 0$ and stops at rest at $\mathbf{z}_b = [1 \ 1]^\top$ at time $b = b_e = .5$. Table 5.4 shows the parameter values used in the rolling ball's initial conditions (5.132). The initial configurations of the control masses are selected so that the total system CM in the spatial frame translated to the GC is initially located above or below the GC. Table 5.5 shows the parameter values used in the rolling ball's final conditions (5.133). Moreover, for the final conditions in (5.133), $\mathbf{\Pi}$ is projection onto the first and second coordinates. $\mathbf{\Delta}_b$ and $\mathbf{\Pi}$ are selected so that the total system CM in the spatial frame translated to the GC is located above or below the GC at the final time b .

Parameter	Value
$\boldsymbol{\theta}_a$	$[0 \ 2.037 \ .7044]^\top$
$\dot{\boldsymbol{\theta}}_a$	$[0 \ 0 \ 0]^\top$
\mathbf{q}_a	$[1 \ 0 \ 0 \ 0]^\top$
$\boldsymbol{\Omega}_a$	$[0 \ 0 \ 0]^\top$
\mathbf{z}_a	$[0 \ 0]^\top$

Table 5.4: Initial condition parameter values for the rolling ball. Refer to (5.132).

Parameter	Value
$\mathbf{\Delta}_b$	$[0 \ 0]^\top$
$\dot{\boldsymbol{\theta}}_b$	$[0 \ 0 \ 0]^\top$
$\boldsymbol{\Omega}_b$	$[0 \ 0 \ 0]^\top$
\mathbf{z}_b	$[1 \ 1]^\top$

Table 5.5: Final condition parameter values for the rolling ball. Refer to (5.133).

The desired GC path \mathbf{z}_d in the integrand cost function (5.135) is depicted by the red curve in Figures 5.15a, 5.15b, and 5.8. \mathbf{z}_d encourages the ball's GC to track a trajectory in the \mathbf{e}_1 - \mathbf{e}_2 plane connecting $\mathbf{z} = [0 \ 0]^\top$ at $t = 0$ with $\mathbf{z} = [1 \ 1]^\top$ at $t = .5$. That is, the ball is encouraged to start at $\mathbf{z} = [0 \ 0]^\top$ at $t = 0$ with zero velocity and to stop at $\mathbf{z} = [1 \ 1]^\top$ at $t = .5$ with zero velocity. Specifically, \mathbf{z}_d is given by (5.137), with $\epsilon = .01$ in (5.98) and where $\tilde{\mathbf{z}}_d$ is given by (5.138).

Table 5.6 shows the values set for the integrand cost function coefficients in (5.135). Since the initial and final times are fixed, the integrand cost function coefficient δ in (5.135) and the endpoint cost function coefficients

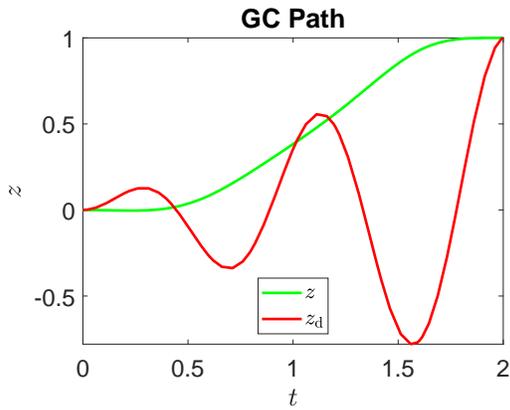
v_a and v_b in (5.134) are irrelevant. There are 2 circular obstacles which the ball’s GC should avoid, depicted in Figures 5.15a and 5.15b. The obstacles appearing in (5.135) are modeled by (5.140); the obstacle centers, radii, and heights are also shown in Table 5.6. There is no external force acting on the disk’s GC, so that $\tilde{\Gamma} = \Lambda^{-1}\mathbf{F}_e = \mathbf{0}$ in (5.131).

Parameter	Value
α	20
β	0
γ_1	10
γ_2	10
γ_3	10
\mathbf{v}_1	$\begin{bmatrix} .3 & .3 \end{bmatrix}^\top$
\mathbf{v}_2	$\begin{bmatrix} .7 & .7 \end{bmatrix}^\top$
ρ_1	.2
ρ_2	.2
$h_1(\mu)$	$.1 + \frac{.95-\mu}{.95-.00001} (1000 - .1)$
$h_2(\mu)$	$.1 + \frac{.95-\mu}{.95-.00001} (1000 - .1)$

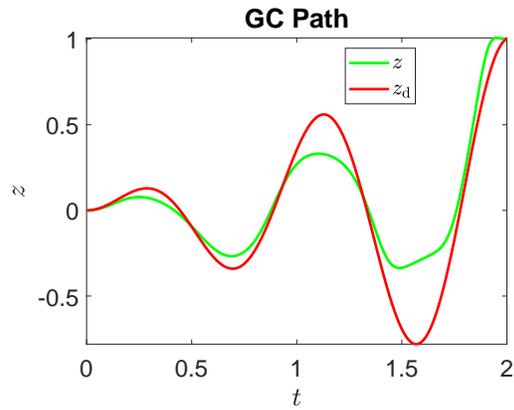
Table 5.6: Integrand cost function coefficient values for the rolling ball. Refer to (5.135) and (5.140).

The direct method solver `GPOPS-II` is used to solve the optimal control problem (5.141) when the obstacle heights appearing in the integrand cost function are $h_1 = h_2 = .1$. The predictor-corrector continuation indirect method is then used to solve the optimal control problem (5.141), starting from the direct method solution. The continuation parameter is μ , which is used to adjust $h_1 = h_2$ according to the linear homotopy shown in Table 5.6, so that $h_1 = h_2 = .1$ when $\mu = .95$ and $h_1 = h_2 = 1,000$ when $\mu = .00001$. The predictor-corrector continuation indirect method begins at $\mu = .95$, which is consistent with the direct method solution obtained at $h_1 = h_2 = .1$.

For the direct method, `GPOPS-II` was run using the IPOPT NLP solver with the MKL PARDISO linear solver. The `GPOPS-II` mesh tolerance is $1e-6$ and the IPOPT error tolerance is $1e-7$. The sweep predictor-corrector method discussed in Appendix D was used by the indirect method. For the sweep predictor-corrector continuation method, the maximum tangent steplength σ_{\max} is adjusted according to Figure 5.16b over the course of 6 predictor-corrector steps, the direction of the initial unit tangent is determined by setting $d = -2$ to force the continuation parameter μ to initially decrease, the relative error tolerance is $1e-6$, the unit tangent solver is `twpbvpc.m`, and the monotonic “sweep” continuation solver is `acdcc`. The numerical results are shown in Figures 5.15 and 5.16. As μ decreases from .95 below $-1,000$ during continuation (see Figure 5.16a), $h_1 = h_2$ increases from .1 up to 1,495,740 (see Figure 5.16c). Since $h_1 = h_2$ is ratcheted up during continuation, thereby increasing the penalty in the integrand cost function (5.135) when the GC intrudes into the obstacles, by the end of continuation, the ball’s GC veers tightly around both obstacles (compare Figures 5.15a vs 5.15b), at the expense of large magnitude controls (compare Figures 5.15e vs 5.15f). Note the turning points in the continuation parameter μ at solutions 3 and 5 in Figure 5.16a.

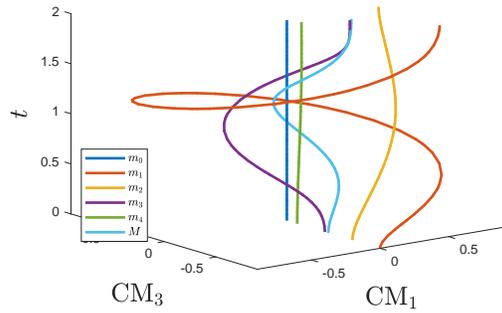


(a) The GC hardly tracks the desired path when $\alpha = 20$.



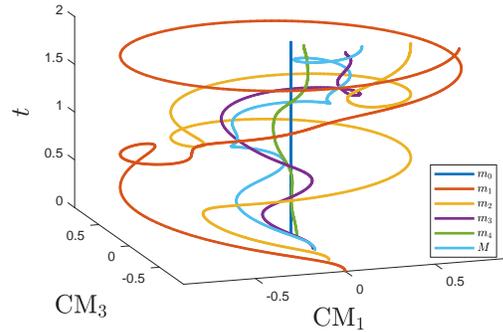
(b) The GC tracks the desired path much more accurately when $\alpha = 54,866$.

**Center of Masses
in the Body Frame Translated to the GC**



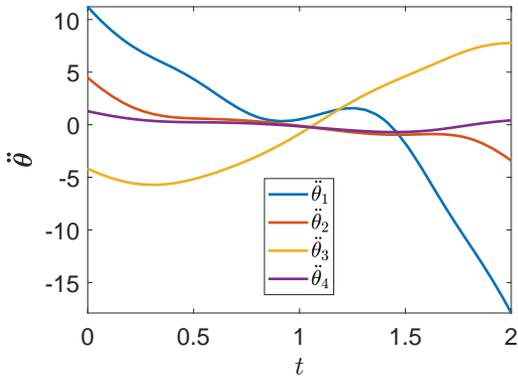
(c) The motion of the center of masses is modest when $\alpha = 20$.

**Center of Masses
in the Body Frame Translated to the GC**



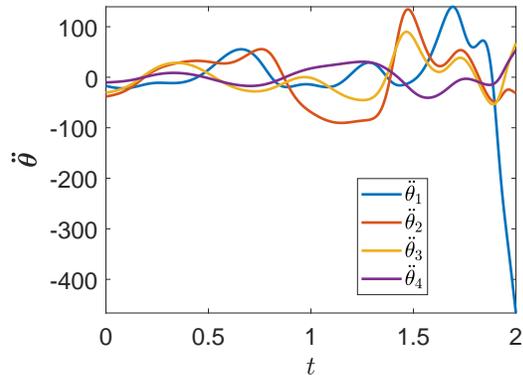
(d) The motion of the center of masses is much wilder when $\alpha = 54,866$.

Control Mass Param. Accelerations



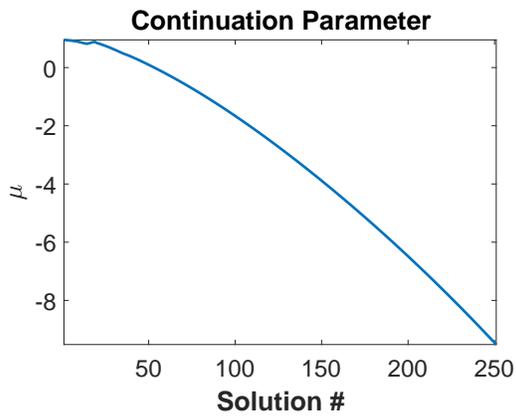
(e) The controls have relatively small magnitudes when $\alpha = 20$.

Control Mass Param. Accelerations

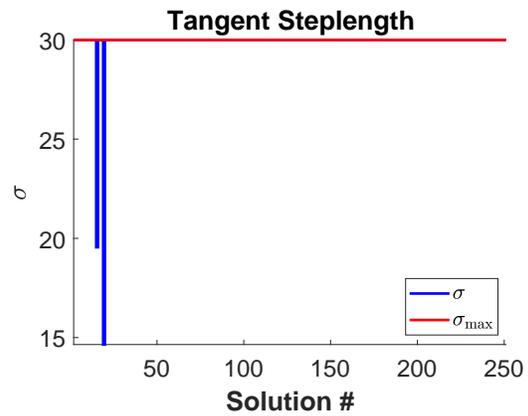


(f) The controls have large magnitudes when $\alpha = 54,866$.

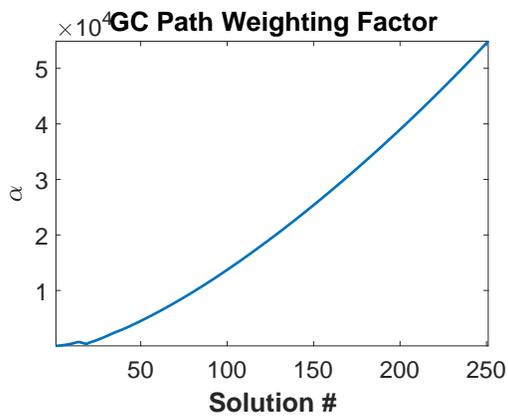
Figure 5.10: Numerical solutions of the rolling disk optimal control problem (5.103) using 4 control masses for $\beta = 0$, $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = .1$, and fixed initial and final times. The direct method results for $\alpha = 20$ are shown in the left column, while the predictor-corrector continuation indirect method results for $\alpha = 54,866$ are shown in the right column.



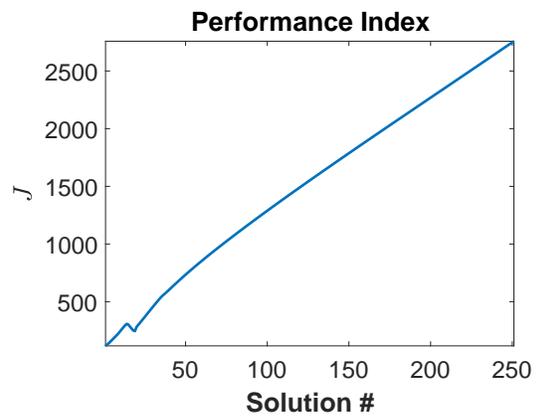
(a) Evolution of the continuation parameter μ .



(b) Evolution of the tangent steplength σ .

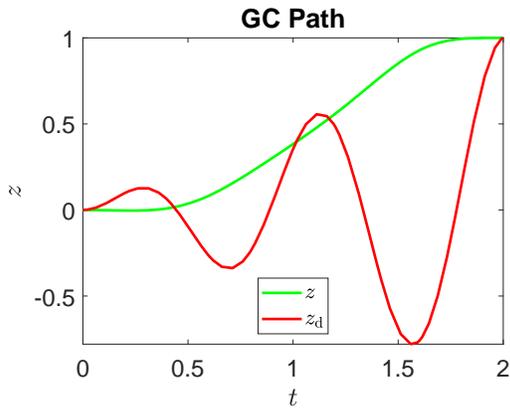


(c) Evolution of α from 20 to 54,866.

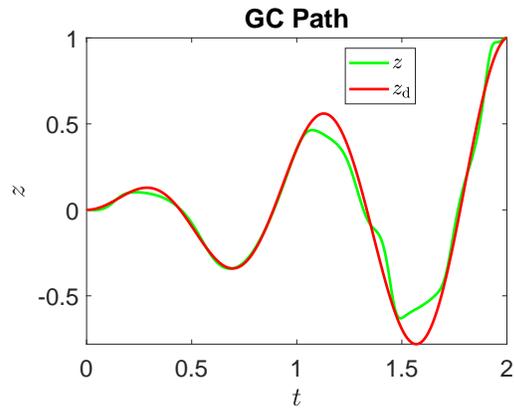


(d) Evolution of the performance index J .

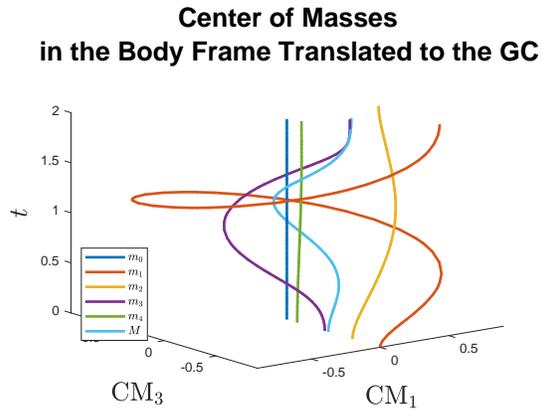
Figure 5.11: Evolution of various parameters during the predictor-corrector continuation indirect method, which starts from the direct method solution, used to solve the rolling disk optimal control problem (5.103). Note the pair of turning points at solutions 14 and 19. The maximum tangent steplength σ_{\max} is fixed to 30.



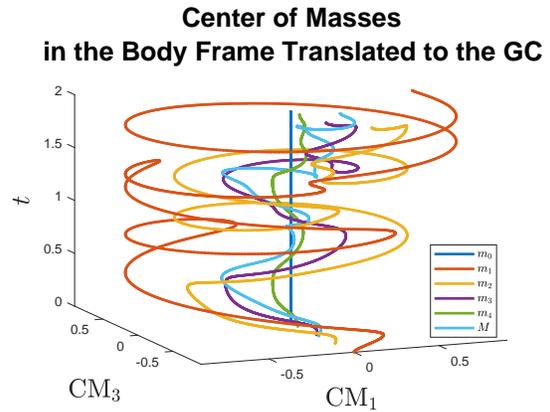
(a) The GC hardly tracks the desired path when $\alpha = 20$.



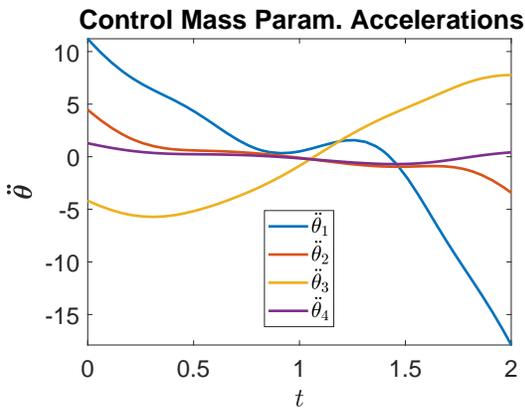
(b) The GC tracks the desired path very accurately when $\alpha = 1,238,285$.



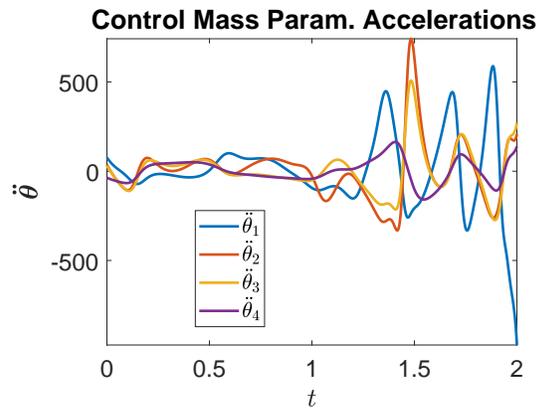
(c) The motion of the center of masses is modest when $\alpha = 20$.



(d) The motion of the center of masses is extremely wild when $\alpha = 1,238,285$.

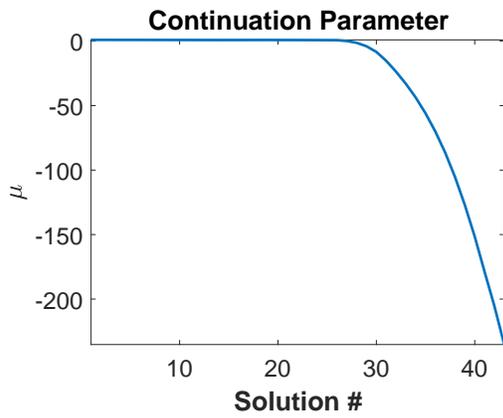


(e) The controls have relatively small magnitude when $\alpha = 20$.

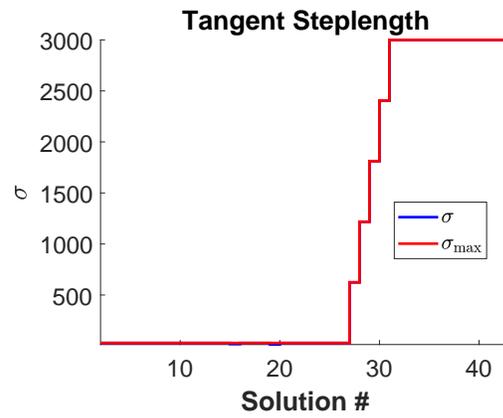


(f) The controls have large magnitudes when $\alpha = 1,238,285$.

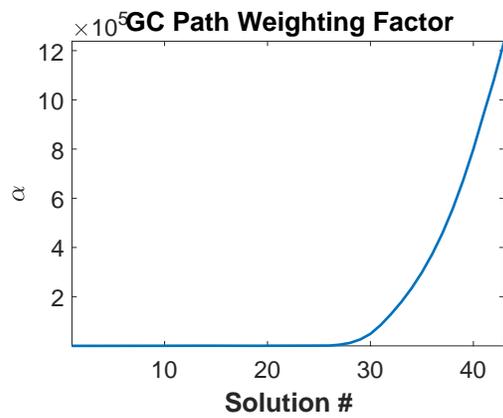
Figure 5.12: Numerical solutions of the rolling disk optimal control problem (5.103) using 4 control masses for $\beta = 0$, $\gamma_1 = \gamma_2 = \gamma_3 = \gamma_4 = .1$, and fixed initial and final times. The direct method results for $\alpha = 20$ are shown in the left column, while the predictor-corrector continuation indirect method results for $\alpha = 1,238,285$ are shown in the right column.



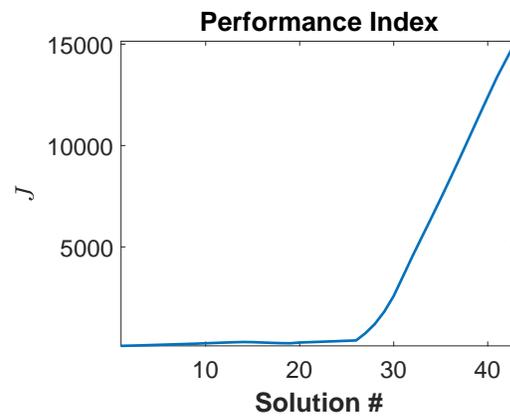
(a) Evolution of the continuation parameter μ .



(b) Evolution of the tangent steplength σ .



(c) Evolution of α from 20 to 1,238,285.



(d) Evolution of the performance index J .

Figure 5.13: Evolution of various parameters during the predictor-corrector continuation indirect method, which starts from the direct method solution, used to solve the rolling disk optimal control problem (5.103). The maximum tangent steplength σ_{\max} is increased linearly after passing the turning points.

Ball, Control Masses, and Control Rails in the Body Frame Translated to the GC

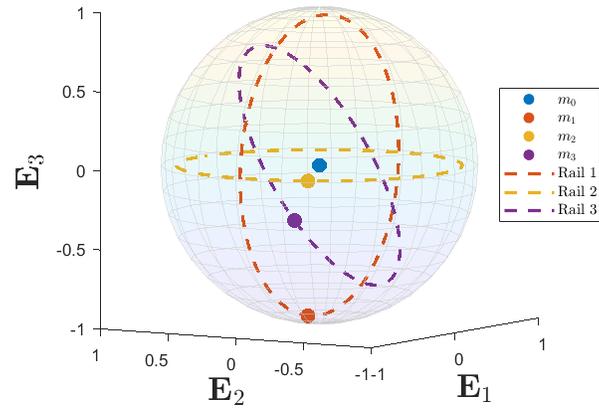
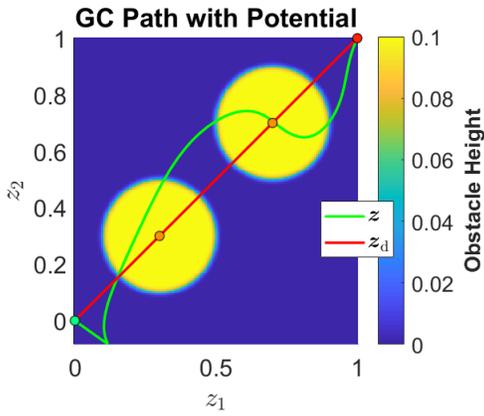
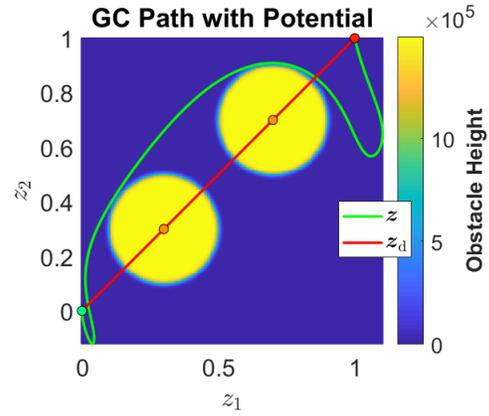


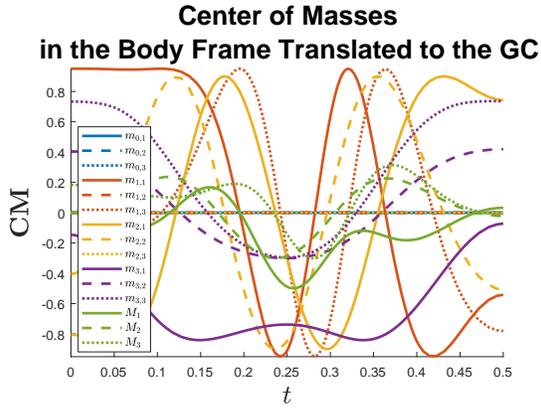
Figure 5.14: The ball of radius $r = 1$ actuated by 3 control masses, m_1 , m_2 , and m_3 , each on its own circular control rail. The control rail radii are $r_1 = .95$, $r_2 = .9$, and $r_3 = .85$. The location of the ball's CM is denoted by m_0 .



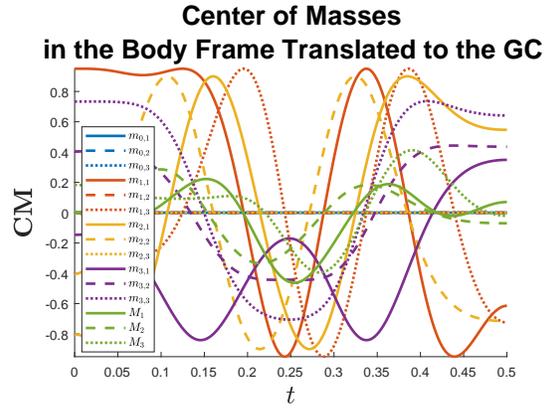
(a) The GC plows through the obstacles when $h_1 = h_2 = .1$.



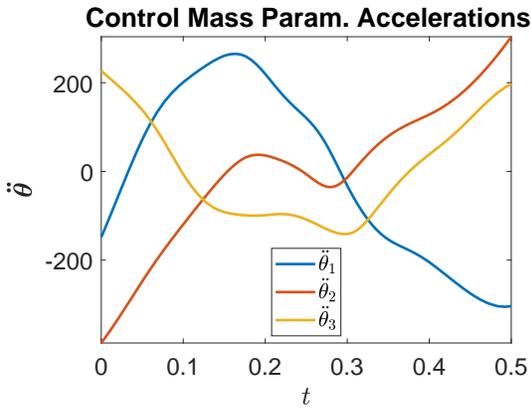
(b) The GC veers tightly around the obstacles when $h_1 = h_2 = 1,495,740$.



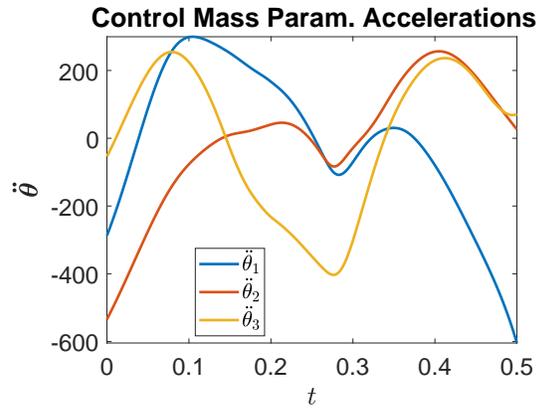
(c) Motion of the center of masses when $h_1 = h_2 = .1$.



(d) Motion of the center of masses when $h_1 = h_2 = 1,495,740$.

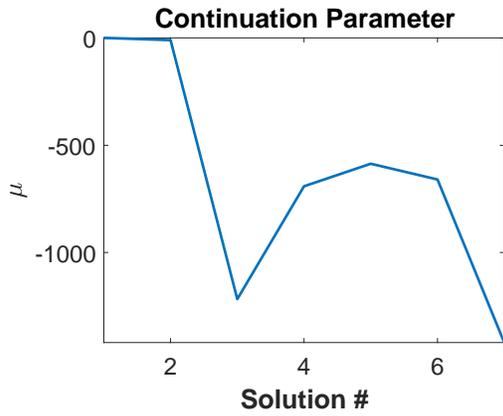


(e) The controls when $h_1 = h_2 = .1$.

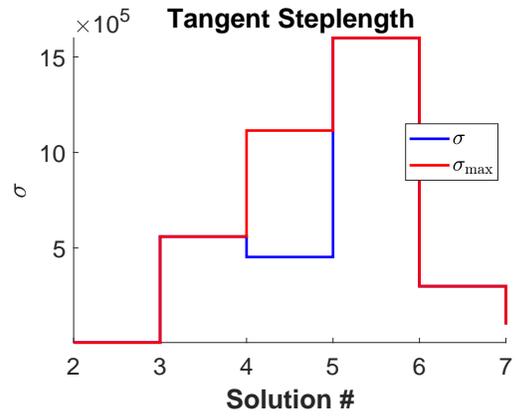


(f) The controls increase in magnitude when $h_1 = h_2 = 1,495,740$.

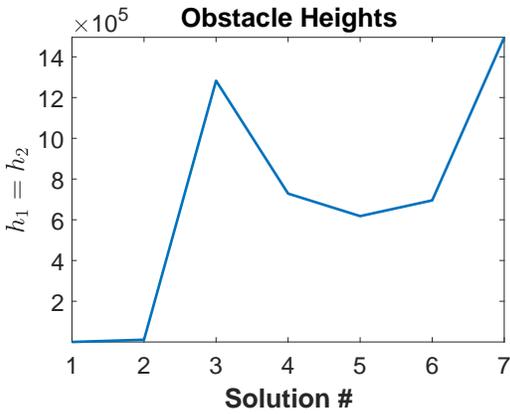
Figure 5.15: Numerical solutions of the rolling ball optimal control problem (5.141) using 3 control masses for $\alpha = 20$, $\beta = 0$, $\gamma_1 = \gamma_2 = \gamma_3 = 10$, and fixed initial and final times. The obstacle centers are located at $\mathbf{v}_1 = [v_{1,1} \ v_{1,2}]^T = [.3 \ .3]^T$ and $\mathbf{v}_2 = [v_{2,1} \ v_{2,2}]^T = [.7 \ .7]^T$ and the obstacle radii are $\rho_1 = \rho_2 = .2$. The direct method results for obstacle heights at $h_1 = h_2 = .1$ are shown in the left column, while the predictor-corrector continuation indirect method results for obstacle heights at $h_1 = h_2 = 1,495,740$ are shown in the right column.



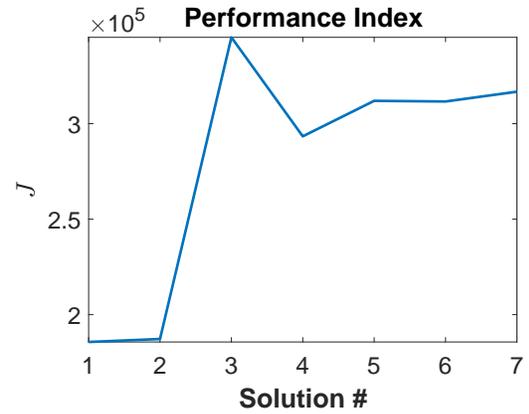
(a) Evolution of the continuation parameter μ . Note the turning points at solutions 3 and 5.



(b) Evolution of the tangent steplength σ . The maximum tangent steplength is not realized to obtain solution 4.



(c) Evolution of the obstacle heights $h_1 = h_2$ from .1 to 1,495,740.



(d) Evolution of the performance index J .

Figure 5.16: Evolution of various parameters during the predictor-corrector continuation indirect method, which starts from the direct method solution, used to solve the rolling ball optimal control problem (5.141).

Chapter 6

Conclusions

6.A Summary and Conclusions

Chapter 1 motivated this research with the rolling ball robots BB-8 and Rosphere, explained the origins of Suslov's problem, mentioned several methods to actuate the rolling ball, listed key contributions of this thesis, and provided a brief outline of the overall thesis.

Chapter 2 reviewed several methods from mechanics, namely Hamilton's principle, Euler-Poincaré's method, and Lagrange-d'Alembert's principle, needed to derive the uncontrolled equations of motion for Suslov's problem and the rolling ball. Euler-Poincaré's method was applied to derive the equations of motion for the free rigid body and the heavy top. By studying a simple nonholonomic particle, it was shown that Lagrange-d'Alembert's principle gives different equations of motion than the more natural and intuitive vakonomic approach.

Chapter 3 presented Pontryagin's minimum principle, which gives the controlled equations of motion corresponding to an optimal control problem. The Jacobians of the controlled equations of motion were derived because they are useful for the numerical solution of the controlled equations of motion. While the controlled equations of motion and their Jacobians are complicated, they were readily obtained in Chapters 4 and 5 to simulate the optimal control of Suslov's problem and the rolling ball by exploiting automatic differentiation. Moreover, these equations were constructed numerically very efficiently in MATLAB by exploiting vectorization; the non-vectorized version of these equations execute too slowly in MATLAB to complete timely simulations. The use of vectorized automatic differentiation to construct these equations numerically in MATLAB was key to obtaining results, because it is tedious to manually derive the non-vectorized version of these equations and terribly difficult to manually derive the vectorized version of these equations.

Chapter 4 considered the optimal control of Suslov's problem. The uncontrolled equations of motion for Suslov's problem were derived by applying Lagrange-d'Alembert's principle with Euler-Poincaré's method. Controllability of Suslov's problem was demonstrated. The controlled equations of motion were derived manually by applying Pontryagin's minimum principle and the Jacobians of the controlled equations of

motion were constructed numerically via automatic differentiation. With the aid of their Jacobians, the controlled equations of motion were solved numerically by using a monotonic continuation method, starting from the analytical solution of a singular optimal control problem.

Chapter 5 considered the optimal control of the rolling ball actuated by internal point masses that move along rails fixed within the ball. The uncontrolled equations of motion for the rolling ball were derived by applying Lagrange-d'Alembert's principle with Euler-Poincaré's method. The regular Hamiltonian \hat{H} and endpoint function G were formed based on the uncontrolled equations of motion, prescribed initial and final conditions, and a prescribed performance index that should be minimized. The controlled equations of motion (and their Jacobians) were constructed numerically via automatic differentiation of \hat{H} and G using the equations derived in Section 3.C. With the aid of their Jacobians, the controlled equations of motion were solved numerically by using a predictor-corrector continuation method detailed in Appendices C and D, starting from an initial solution obtained via a direct method. This process was applied to roll a disk back and forth several times, so that the disk's GC tracked a sinusoidally-modulated linear path, by performing continuation in the performance index parameter α which penalizes deviations between the disk's GC and the desired GC path. This process was again applied to roll a ball between a pair of points in the plane while avoiding a pair of obstacles, by performing continuation in the performance index parameters $h_1 = h_2$ which penalize intrusions of the ball's GC into the obstacle interiors. These results demonstrate the potential of this process to solve complicated trajectory-tracking and obstacle avoidance optimal control problems for the rolling ball.

This thesis focused on the indirect, rather than direct, method to numerically solve the optimal control problems. Because the indirect and direct methods only converge to a local minimum solution near the initial guess, a robust continuation algorithm capable of handling turning points is needed to obtain indirect and direct method solutions of complicated, nonconvex optimal control problems. A continuation indirect method requires a continuation ODE or DAE TPBVP solver, while a continuation direct method requires a continuation NLP solver. Predictor-corrector continuation ODE TPBVP algorithms were presented in Appendices C and D and implemented in MATLAB to realize the continuation indirect method used to solve the rolling ball optimal control problems. Because the predictor-corrector continuation ODE TPBVP algorithms had not been researched or implemented at the time when Suslov's problem was investigated, only monotonic continuation ODE TPBVP solvers (i.e. `acdc` and `acdcc`) were used to solve Suslov's optimal control problem. Even though predictor-corrector continuation NLP solver algorithms are provided in the literature (e.g. see [58, 59]), there do not seem to be any publicly available predictor-corrector continuation NLP solvers, which inhibited the use of a continuation direct method in this thesis. When compared against the direct method, the indirect method suffers from two major deficiencies:

1. Unlike the direct method, the indirect method has a very small radius of convergence and therefore requires a very accurate initial solution guess. Moreover, unlike the direct method, the indirect method requires a guess of the costates, which are unphysical.
2. Unlike the direct method, the indirect method is unable to construct the switching structure (i.e. the times when the states and/or controls enter and exit the boundary) of an optimal control problem having path inequality constraints.

To circumvent the first deficiency in the indirect method, the indirect method was provided a good initial

solution guess obtained analytically (in the case of Suslov’s problem) or via a direct method (in the case of the rolling ball). To circumvent the second deficiency in the indirect method, path inequality constraints were incorporated into the optimal control problems as soft constraints through penalty functions in the integrand and endpoint cost functions.

In summary, this thesis has utilized Lagrange-d’Alembert’s principle, Euler-Poincaré’s method, and continuation indirect methods to solve challenging optimal control problems for two nonholonomic mechanical systems, Suslov’s problem and the rolling ball.

6.B Future Work

Topics for future work are listed below.

- Pontryagin’s minimum principle provides necessary but not sufficient conditions for a local minimum solution of an optimal control problem. Sufficient conditions which ensure the local minimality of a solution that satisfies the necessary conditions provided by Pontryagin’s minimum principle are given in [44] within the context of geometric control theory. It would be useful to understand and implement these sufficient conditions to verify the local minimality of a solution obtained by the indirect method used to solve an optimal control problem. COTCOT [60] and HamPath [61] are indirect method optimal control software packages that have already implemented these sufficient conditions.
- The direct and indirect methods only provide local minimum solutions to an optimal control problem. In contrast, dynamic programming offers a global minimum solution, but is impractical to implement due to the curse of dimensionality. References [62, 63, 64, 65] present recent research results that seek to overcome the curse of dimensionality in certain special cases, and it may be useful to understand and try to apply these methods to optimal control problems such as those investigated in this thesis.
- The indirect method used in this thesis only numerically solves ODE TPBVP, rather than more general DAE TPBVP. This is because a DAE TPBVP solver is not readily available in MATLAB. It may be worthwhile to develop a MATLAB DAE TPBVP solver based on the Fortran code COLDAE [66] and the MATLAB code `bvpsuite1.1` [67]. COLDAE is capable of solving index-2 DAE TPBVPs, while `bvpsuite1.1` is capable of solving index-1 DAE TPBVPs. While written in MATLAB, `bvpsuite1.1` relies on an awkward graphical user interface for input of the DAE and boundary condition functions, does not accept user-supplied Jacobians of the DAE and boundary condition functions, and is not vectorized. A new version of `bvpsuite`, `bvpsuite2.0` [68, 69, 70], is in preparation that addresses the first two deficiencies but not the last one. Since the predictor-corrector continuation methods discussed in Appendices C and D only apply to ODE TPBVP, they would need to be adapted to handle DAE TPBVP.
- Solving the rolling ball controlled equations of motion in MATLAB is quite slow, even when vectorization is used. Another path is to use Fortran or C/C++, which are an order of magnitude faster than MATLAB. BOCOP [71] and *PSOPT* [72] are free C++ direct method optimal control solvers. `bvpSolve` [73] is an R library that wraps the Fortran solvers TWPBVP, TWPBVPC, TWPBVPL, TWPBVPLC, ACDC, and ACDCC and COLSYS, COLNEW, COLMOD, and COLDAE, which could be utilized to

numerically solve the controlled equations of motion; moreover, the Fortran codes wrapped by `bvpSolve` are able to execute compiled Fortran or C/C++ code implementing the ODEs/DAEs, BCs, ODE/DAE Jacobians, and BC Jacobians. `bvptwp`, which offers the algorithms `twpbvp_m`, `twpbvpc_m`, `twpbvp_l`, `twpbvpc_l`, `acdc`, and `acdcc`, is a `MATLAB` reimplement of the Fortran solvers `TWPBVP`, `TWPBVPC`, `TWPBVPL`, `TWPBVPLC`, `ACDC`, and `ACDCC`. The `MATLAB` solver `sbvp` has capabilities similar to the Fortran solvers `COLSYS` and `COLNEW`. Using `bvpSolve` has the added benefit in that it provides access to the monotonic continuation ODE BVP solver `COLMOD` and the DAE BVP solver `COLDAE`. Since `Tapenade` [74] is able to automatically differentiate Fortran and C code, `ADOL-C` [75] is able to differentiate C/C++ code, and `CppAD` [76] is able to differentiate C++ code, these automatic differentiation software packages could be used to numerically construct the controlled equations of motion (and their Jacobians) from Fortran or C/C++ implementations of the Hamiltonian H (or the regular Hamiltonian \hat{H}) and endpoint function G . Another alternative to `MATLAB` is `Julia`, a relatively new high-level programming language. `Julia` is just as easy to program in as other high-level programming languages like `Python`, `R`, and `MATLAB`, but at the same time `Julia` executes almost as fast as `C` due to a just-in-time compiler. `Julia` offers several automatic differentiation packages and the `ODEInterface` package, which provides an interface to the Fortran ODE BVP solvers `COLNEW` and `BVP_SOLVER-2`.

- As an alternative to automatic differentiation, the first and second derivatives required by the direct and indirect methods could be supplied by dual/hyper-dual numbers [77, 78, 79] or by complex-step/bicomplex-step differentiation [80, 81, 82, 83]. Dual numbers and complex-step differentiation are two different techniques to construct first derivatives, while hyper-dual numbers and bicomplex-step differentiation are two different techniques to construct second derivatives. These alternatives are possible because the functions being differentiated are real analytic.
- One difficulty in the predictor-corrector continuation method is adapting the tangent steplength so that the solution curve is efficiently traced. The tangent steplength minimum, maximum, increase scale factor, and reduction scale factor must be chosen wisely in order to efficiently trace the solution curve. The sweep predictor-corrector continuation method used in this thesis only manually changed the maximum tangent steplength by trial and error.
- In this thesis, continuation was used to adjust weighting factors that scale penalty functions in the optimal control problem. Another approach that should be investigated is to perform continuation in the final states starting from the trivial case when the final states match or nearly match the prescribed initial states.
- Aside from continuation, another method to construct multiple solutions of a nonlinear operator (such as an ODE TPBVP, a DAE TPBVP, or a PDE) is deflation [84]. Reference [84] uses deflation to construct multiple solutions of ODE TPBVPs, while [85] uses deflation to construct multiple solutions of PDEs. Reference [86] combines deflation and continuation to construct multiple solutions of several different kinds of nonlinear operators. It may be fruitful to apply a combined deflation and continuation algorithm to solve the controlled equations of motion corresponding to optimal control problems.
- In this thesis, the mechanical systems are encouraged to track a prescribed trajectory, with prescribed time-parameterization, in a fixed or minimum time. A more general optimal control problem, called time-optimal path parameterization (TOPP) [87, 88], is for a dynamical system to track a prescribed

trajectory, whose time-parameterization is not prescribed, in a fixed or minimum time. It may be worthwhile to investigate and apply TOPP algorithms to optimal control problems such as those investigated in this thesis.

- In this thesis, the technique used to actuate the motion of the rolling ball is by moving internal point masses along rails fixed within the ball. It would be useful to apply the methodology presented in this thesis to investigate the optimal control of the rolling ball where the motion is actuated by other techniques, for example by rotating internal rotors or by swinging an internal spherical pendulum [89, 27, 28].
- Controllability in the sense of Definition 4.3 was demonstrated for Suslov's problem in Subsection 4.B.2, but controllability was not demonstrated for the rolling ball. Controllability of the rolling ball actuated by moving internal point masses along rails fixed within the ball should be demonstrated.

Bibliography

- [1] Vakhtang Putkaradze and Stuart Rogers. “Constraint Control of Nonholonomic Mechanical Systems”. In: *Journal of Nonlinear Science* (2017). DOI: [10.1007/s00332-017-9406-1](https://doi.org/10.1007/s00332-017-9406-1).
- [2] Vakhtang Putkaradze and Stuart Rogers. “Optimal Control of a Rolling Ball Robot Actuated by Internal Point Masses”. In: *arXiv preprint arXiv:1708.03829* (2017).
- [3] Sphero, ed. *BB-8 By Sphero*. [Online; accessed August 24, 2017]. Aug. 4, 2015. URL: https://brandfolder.com/bb8/attachments/926upng2/bb-8-by-sphero-bb-8-rolling-2-genericfile.jpg?dl=true&resource_key=mln615o60934&resource_type=Brandfolder.
- [4] J.D. Hernández et al. “Moisture measurement in crops using spherical robots”. In: *Industrial Robot: An International Journal* 40.1 (2013), pp. 59–66.
- [5] Henri Poincaré. “Sur une forme nouvelle des équations de la mécanique”. In: *CR Acad. Sci* 132 (1901), pp. 369–371.
- [6] A. A. Bloch et al. “A geometric approach to the optimal control of nonholonomic mechanical systems”. In: *Analysis and Geometry in Control Theory and its Applications*. Switzerland: Springer INdAM Series, 2015, pp. 35–64.
- [7] GK Suslov. “Theoretical mechanics”. In: *Gostekhizdat, Moscow* 3 (1946), pp. 40–43.
- [8] Alexey V Borisov, Alexander A Kilin, and Ivan S Mamaev. “Hamiltonicity and integrability of the Suslov problem”. In: *Regular and Chaotic Dynamics* 16.1-2 (2011), pp. 104–116.
- [9] V Vagner. “Geometrical interpretation of the motion of nonholonomic dynamical systems”. In: *Proc. Seminar on Vector and Tensor Analysis*. Vol. 5. 1941, pp. 301–327.
- [10] Dmitry V Zenkov and Anthony M Bloch. “Dynamics of the n-dimensional Suslov problem”. In: *Journal of Geometry and Physics* 34.2 (2000), pp. 121–136.
- [11] Valerii Vasil’evich Kozlov. “On the integration theory of equations of nonholonomic mechanics”. In: *Regular and Chaotic Dynamics* 7.2 (2002), pp. 161–176.
- [12] AJ Maciejewski and M Przybylska. “Non-integrability of the Suslov problem”. In: *Regul. Chaotic Dyn* 7.1 (2002), pp. 73–80.
- [13] Yuri N Fedorov, Andrzej J Maciejewski, and Maria Przybylska. “The Poisson equations in the non-holonomic Suslov problem: integrability, meromorphic and hypergeometric solutions”. In: *Nonlinearity* 22.9 (2009), p. 2231.
- [14] Luis C García-Naranjo et al. “The inhomogeneous Suslov problem”. In: *Physics Letters A* 378.32 (2014), pp. 2389–2394.

- [15] Jinglai Shen, David A Schneider, and Anthony M Bloch. “Controllability and motion planning of a multibody Chaplygin’s sphere and Chaplygin’s top”. In: *International Journal of Robust and Non-linear Control* 18.9 (2008), pp. 905–945.
- [16] D.D. Holm. *Geometric Mechanics: Rotating, translating, and rolling*. Geometric Mechanics. Imperial College Press, 2011. ISBN: 9781848167773.
- [17] SA Chaplygin. “On a motion of a heavy body of revolution on a horizontal plane”. In: *Regular and Chaotic Dynamics* 7.2 (2002), pp. 119–130.
- [18] SA Chaplygin. “On a ball’s rolling on a horizontal plane”. In: *Regular and Chaotic Dynamics* 7.2 (2002), pp. 131–148.
- [19] E Routh. *Advanced Rigid Body Dynamics*. MacMillan and Co., London, 1884.
- [20] John Hewitt Jellett. *A Treatise on the Theory of Friction*. Hodges, Foster, and Company, 1872.
- [21] Tuhin Das, Ranjan Mukherjee, and H Yuksel. “Design considerations in the development of a spherical mobile robot”. In: *Proc. 15th SPIE Annual International Symposium on Aerospace/Defense Sensing, Simulation, and Controls*. Vol. 4364. 2001, pp. 61–71.
- [22] Puyan Mojabi et al. “Introducing August: a novel strategy for an omnidirectional spherical rolling robot”. In: *Robotics and Automation, 2002. Proceedings. ICRA ’02. IEEE International Conference on*. Vol. 4. IEEE. 2002, pp. 3527–3533.
- [23] Alexey V Borisov, Alexander A Kilin, and Ivan S Mamaev. “How to control Chaplygin’s sphere using rotors”. In: *Regular and Chaotic Dynamics* 17.3 (2012), pp. 258–272.
- [24] Sergey Bolotin. “The problem of optimal control of a Chaplygin ball by internal rotors”. In: *Regular and Chaotic Dynamics* 17.6 (2012), pp. 559–570.
- [25] Sneha Gajbhiye and Ravi N Banavar. “Geometric tracking control for a nonholonomic system: a spherical robot”. In: *IFAC-PapersOnLine* 49.18 (2016), pp. 820–825.
- [26] Sneha Gajbhiye and Ravi N. Banavar. “Geometric modeling and local controllability of a spherical mobile robot actuated by an internal pendulum”. In: *International Journal of Robust and Nonlinear Control* (2015), n/a–n/a. ISSN: 1099-1239. DOI: [10.1002/rnc.3457](https://doi.org/10.1002/rnc.3457). URL: <http://dx.doi.org/10.1002/rnc.3457>.
- [27] Alexander A Kilin, Elena N Pivovarova, and Tatyana B Ivanova. “Spherical robot of combined type: Dynamics and control”. In: *Regular and Chaotic Dynamics* 20.6 (2015), pp. 716–728.
- [28] Matt Burkhardt and Joel W Burdick. “Reduced dynamical equations for barycentric spherical robots”. In: *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE. 2016, pp. 2725–2732.
- [29] J Osborne and D Zenkov. “Steering the Chaplygin sleigh by a moving mass”. In: *IEEE CONFERENCE ON DECISION AND CONTROL*. Vol. 44. 2. IEEE; 1998. 2005, p. 1114.
- [30] Anthony M Bloch. *Nonholonomic mechanics and control*. Vol. 24. Springer Science & Business Media, 2003.
- [31] Sphero, ed. *Sphero Edu*. [Online; accessed August 24, 2017]. May 8, 2017. URL: https://brandfolder.com/spheroedu/attachments/opnbxt-5z1lfs-30z63c/sphero-edu-sprkplus-hero-genericfile.png?dl=true&resource_key=ooxisb-816c4o-f9z224&resource_type=Brandfolder.

- [32] Wikipedia, the free encyclopedia. *Tangent bundle*. [Online; accessed June 16, 2017]. 2017. URL: https://en.wikipedia.org/wiki/File:Tangent_bundle.svg.
- [33] Greg Egan. *Orthogonal: Geometry and Waves*. [Online; accessed June 16, 2017]. 2011. URL: <http://www.gregegan.net/ORTHOGONAL/03/014.png>.
- [34] Greg Egan. *Orthogonal: Geometry and Waves*. [Online; accessed June 16, 2017]. 2011. URL: <http://www.gregegan.net/ORTHOGONAL/03/012.png>.
- [35] Andrew D Lewis and Richard M Murray. “Variational principles for constrained systems: theory and experiment”. In: *International Journal of Non-Linear Mechanics* 30.6 (1995), pp. 793–815.
- [36] Larry Bates et al. “Nonholonomic reduction”. In: *Reports on Mathematical Physics* 32.1 (1993), pp. 99–115.
- [37] Vladimir I Arnold, Valery V Kozlov, and Anatoly I Neishtadt. *Mathematical aspects of classical and celestial mechanics*. Vol. 3. Springer Science & Business Media, 2007.
- [38] David G Hull. *Optimal control theory for applications*. Springer Science & Business Media, 2013.
- [39] VG Boltyanskiy et al. *Mathematical theory of optimal processes*. Interscience Publishers, 1962.
- [40] Matthew J Weinstein and Anil V Rao. “Algorithm: ADiGator, a Toolbox for the Algorithmic Differentiation of Mathematical Functions in MATLAB Using Source Transformation via Operator Overloading”. In: *ACM Transactions on Mathematical Software (in revision)* (2016).
- [41] Matthew J Weinstein, Michael A Patterson, and Anil V Rao. “Utilizing the Algorithmic Differentiation Package ADiGator for Solving Optimal Control Problems Using Direct Collocation”. In: *AIAA Guidance, Navigation, and Control Conference*. 2015, p. 1085.
- [42] Jerrold E Marsden and Tudor Ratiu. *Introduction to mechanics and symmetry: a basic exposition of classical mechanical systems*. Vol. 17. Springer Science & Business Media, 2013.
- [43] Arthur Earl Bryson. *Applied optimal control: optimization, estimation and control*. CRC Press, 1975.
- [44] A. Agrachev and Y. Sachkov. *Control theory from the geometric viewpoint*. Berlin Heidelberg New York: Springer, 2004.
- [45] Roger W Brockett. “System theory on group manifolds and coset spaces”. In: *SIAM Journal on control* 10.2 (1972), pp. 265–284.
- [46] Henk Nijmeijer and Arjan Van der Schaft. *Nonlinear dynamical control systems*. Springer Science & Business Media, 2013.
- [47] Alberto Isidori. *Nonlinear control systems*. Springer Science & Business Media, 2013.
- [48] John M Lee. *Smooth manifolds*. Springer, 2003.
- [49] G Dahlquist and A Björck. *Numerical Methods*. Series in Automatic Computation. Prentice-Hall, 1974.
- [50] Eugene L Allgower and Kurt Georg. *Introduction to numerical continuation methods*. Vol. 45. SIAM, 2003.
- [51] Bernard Bonnard, Jean-Baptiste Caillaud, and Emmanuel Trélat. “Second order optimality conditions in the smooth case and applications in optimal control”. In: *ESAIM: Control, Optimisation and Calculus of Variations* 13.2 (2007), pp. 207–236.

- [52] Wikipedia, the free encyclopedia. *Portrait of Newton at 46 in 1689 by Godfrey Kneller*. [Online; accessed June 18, 2017]. 2017. URL: <https://upload.wikimedia.org/wikipedia/commons/thumb/3/39/GodfreyKneller-IsaacNewton-1689.jpg/330px-GodfreyKneller-IsaacNewton-1689.jpg>.
- [53] Wikipedia, the free encyclopedia. *Joseph-Louis (Giuseppe Luigi), comte de Lagrange*. [Online; accessed June 18, 2017]. 2017. URL: https://upload.wikimedia.org/wikipedia/commons/1/19/Lagrange_portrait.jpg.
- [54] John T Betts. *Practical methods for optimal control and estimation using nonlinear programming*. Vol. 19. Siam, 2010.
- [55] Michael A Patterson and Anil V Rao. “GPOPS-II: A MATLAB software for solving multiple-phase optimal control problems using hp-adaptive Gaussian quadrature collocation methods and sparse nonlinear programming”. In: *ACM Transactions on Mathematical Software (TOMS)* 41.1 (2014), p. 1.
- [56] Winfried Auzinger et al. “A collocation code for singular boundary value problems in ordinary differential equations”. In: *Numerical Algorithms* 33.1-4 (2003), pp. 27–39.
- [57] Jeff R Cash et al. “Algorithm 927: the MATLAB code bvptwp. m for the numerical solution of two point boundary value problems”. In: *ACM Transactions on Mathematical Software (TOMS)* 39.2 (2013), p. 15.
- [58] Willard I Zangwill and CB Garcia. *Pathways to solutions, fixed points, and equilibria*. Prentice Hall, 1981.
- [59] Vyacheslav Kungurtsev and Johannes Jaschke. “A Predictor-Corrector Path-Following Algorithm for Dual-Degenerate Parametric Optimization Problems”. In: *SIAM Journal on Optimization* 27.1 (2017), pp. 538–564.
- [60] Bernard Bonnard, Jean-Baptiste Caillaud, and Emmanuel Trélat. “Computation of conjugate times in smooth optimal control: the COTCOT algorithm”. In: *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference 2005 (CDC-ECC'05), Séville, Spain (2005)*. IEEE. 2005, 6–pages.
- [61] J-B Caillaud, O Cots, and J Gergaud. “Differential continuation for regular optimal control problems”. In: *Optimization Methods and Software* 27.2 (2012), pp. 177–196.
- [62] Jérôme Darbon and Stanley Osher. “Algorithms for Overcoming the Curse of Dimensionality for Certain Hamilton-Jacobi Equations Arising in Control Theory and Elsewhere”. In: *arXiv preprint arXiv:1605.01799* (2016).
- [63] Yat Tin Chow et al. “Algorithm for Overcoming the Curse of Dimensionality for Certain Non-convex Hamilton-Jacobi Equations, Projections and Differential Games”. In: *Annals of Mathematical Sciences and Applications (to appear)* (2016).
- [64] Yat Tin Chow et al. “Algorithm for Overcoming the Curse of Dimensionality for Time-dependent Non-convex Hamilton-Jacobi Equations Arising from Optimal Control and Differential Games Problems”. In: ().
- [65] Yat Tin Chow et al. “Algorithm for Overcoming the Curse of Dimensionality for State-dependent Hamilton-Jacobi equations”. In: *arXiv preprint arXiv:1704.02524* (2017).

- [66] Uri M Ascher and Raymond J Spiteri. “Collocation software for boundary value differential-algebraic equations”. In: *SIAM Journal on Scientific Computing* 15.4 (1994), pp. 938–952.
- [67] G Kitzhofer et al. “The new Matlab code bvpsuite for the solution of singular implicit BVPs”. In: *J. Numer. Anal. Indust. Appl. Math* 5 (2010), pp. 113–134.
- [68] Stefan Wurm. “BVPsuite 2.0 - a new version of a collocation code for singular BVPs in ODEs, EVPs, and DAEs”. PhD thesis. Vienna University of Technology, 2016.
- [69] Markus Schöbinger. “A new version of a collocation code for singular BVPs: Nonlinear solver and its application to m-Laplacians”. In: ().
- [70] JANA BURKOTOV et al. “Systems of ODEs with a singularity of the first kind and nonsmooth nonlinearity”. In: ().
- [71] Inria Saclay Team Commands. *BOCOP: an open source toolbox for optimal control*. <http://bocop.org>. 2017.
- [72] Victor M Becerra. “Solving complex optimal control problems at no cost with PSOPT”. In: *2010 IEEE International Symposium on Computer-Aided Control System Design*. IEEE. 2010, pp. 1391–1396.
- [73] Francesca Mazzia, Jeff R Cash, and Karline Soetaert. “Solving boundary value problems in the open source software R: Package bvpSolve”. In: *Opuscula mathematica* 34.2 (2014), pp. 387–403.
- [74] L. Hascoët and V. Pascual. “The Tapenade Automatic Differentiation tool: Principles, Model, and Specification”. In: *ACM Transactions On Mathematical Software* 39.3 (2013). URL: <http://dx.doi.org/10.1145/2450153.2450158>.
- [75] Andrea Walther and Andreas Griewank. “Getting started with ADOL-C”. In: *Combinatorial scientific computing*. Chapman and Hall/CRC, 2012, pp. 181–202.
- [76] Bradley M Bell. “CppAD: a package for C++ algorithmic differentiation”. In: *Computational Infrastructure for Operations Research* 57 (2012).
- [77] Jeffrey A Fike and Juan J Alonso. “The development of hyper-dual numbers for exact second-derivative calculations”. In: *AIAA paper* 886 (2011), p. 124.
- [78] Jeffrey A Fike et al. “Optimization with gradient and hessian information calculated using hyper-dual numbers”. In: *AIAA paper* 3807 (2011), p. 2011.
- [79] Jeffrey A Fike and Juan J Alonso. “Automatic differentiation through the use of hyper-dual numbers for second derivatives”. In: *Recent Advances in Algorithmic Differentiation*. Springer, 2012, pp. 163–173.
- [80] William Squire and George Trapp. “Using complex variables to estimate derivatives of real functions”. In: *Siam Review* 40.1 (1998), pp. 110–112.
- [81] JRRR Martins, Peter Sturdza, and Juan J Alonso. “The connection between the complex-step derivative approximation and algorithmic differentiation”. In: *AIAA paper* 921 (2001), p. 2001.
- [82] Joaquim RRA Martins, Peter Sturdza, and Juan J Alonso. “The complex-step derivative approximation”. In: *ACM Transactions on Mathematical Software (TOMS)* 29.3 (2003), pp. 245–262.
- [83] Gregory Lantoiné, Ryan P Russell, and Thierry Dargent. “Using multicomplex variables for automatic computation of high-order derivatives”. In: *ACM Transactions on Mathematical Software (TOMS)* 38.3 (2012), p. 16.

- [84] Ásgeir Birkisson. “Numerical solution of nonlinear boundary value problems for ordinary differential equations in the continuous framework”. PhD thesis. University of Oxford, 2013.
- [85] Patrick E Farrell, A Birkisson, and Simon W Funke. “Deflation techniques for finding distinct solutions of nonlinear partial differential equations”. In: *SIAM Journal on Scientific Computing* 37.4 (2015), A2026–A2045.
- [86] Patrick E Farrell, Casper HL Beentjes, and Ásgeir Birkisson. “The computation of disconnected bifurcation diagrams”. In: *arXiv preprint arXiv:1603.00809* (2016).
- [87] Quang-Cuong Pham. “A general, fast, and robust implementation of the time-optimal path parameterization algorithm”. In: *IEEE Transactions on Robotics* 30.6 (2014), pp. 1533–1540.
- [88] Huy Nguyen and Quang-Cuong Pham. “Time-optimal path parameterization of rigid-body motions: applications to spacecraft reorientation”. In: *Journal of Guidance, Control, and Dynamics* (2016), pp. 1–5.
- [89] Sneha Gajbhiye and Ravi N Banavar. “Geometric modeling and local controllability of a spherical mobile robot actuated by an internal pendulum”. In: *International Journal of Robust and Nonlinear Control* (2015).
- [90] Arthur Earl Bryson. *Dynamic optimization*. Vol. 1. Prentice Hall, 1999.
- [91] John T Betts. “Survey of numerical methods for trajectory optimization”. In: *Journal of guidance, control, and dynamics* 21.2 (1998), pp. 193–207.
- [92] Anil V Rao. “A survey of numerical methods for optimal control”. In: *Advances in the Astronautical Sciences* 135.1 (2009), pp. 497–528.
- [93] Matthias Gerdts. *Optimal control of ODEs and DAEs*. Walter de Gruyter, 2012.
- [94] Lorenz T Biegler. *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. Vol. 10. SIAM, 2010.
- [95] Lawrence C. Evans. *Partial differential equations*. Providence, R.I.: American Mathematical Society, 2010. ISBN: 9780821849743 0821849743.
- [96] Frédéric Bonnans et al. “BocopHJB 1.0. 1–User Guide”. PhD thesis. INRIA, 2015.
- [97] Andreas Wächter and Lorenz T Biegler. “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming”. In: *Mathematical programming* 106.1 (2006), pp. 25–57.
- [98] Tim Nikolayzik, Christof Büskens, and Matthias Gerdts. “Nonlinear large-scale Optimization with WORHP”. In: *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*. Vol. 13. 15.09. 2010.
- [99] Philip E Gill, Walter Murray, and Michael A Saunders. “SNOPT: An SQP algorithm for large-scale constrained optimization”. In: *SIAM review* 47.1 (2005), pp. 99–131.
- [100] Richard H Byrd, Jorge Nocedal, and Richard A Waltz. “KNITRO: An integrated package for nonlinear optimization”. In: *Large-scale nonlinear optimization*. Springer, 2006, pp. 35–59.
- [101] Matlab Documentation Center. *Optimization Toolbox, Constrained optimization, fmincon*.
- [102] YangQuan Chen and Adam L Schwartz. “RIOTS95: a MATLAB toolbox for solving general optimal control problems and its applications to chemical processes”. In: (2002).

- [103] M Cizniar et al. “Dynopt—dynamic optimisation code for MATLAB”. In: *Technical Computing Prague 2005* (2005).
- [104] Paola Falugi, Eric Kerrigan, and Eugene Van Wyk. “Imperial college london optimal control software user guide (ICLOCS)”. In: *Department of Electrical and Electronic Engineering, Imperial College London, London, England, UK* (2010).
- [105] Anil V Rao et al. “Algorithm 902: Gpops, a matlab software for solving multiple-phase optimal control problems using the gauss pseudospectral method”. In: *ACM Transactions on Mathematical Software (TOMS)* 37.2 (2010), p. 22.
- [106] M Rieck et al. “FALCON.m User Guide”. In: *Institute of Flight System Dynamics, Technische Universität München* (2016).
- [107] Matthew P. Kelly. “OptimTraj User’s Guide, Version 1.5”. In: (2016).
- [108] I Michael Ross. “Users manual for DIDO: A MATLAB application package for solving optimal control problems”. In: *Tomlab Optimization, Sweden* (2004).
- [109] Per E Rutquist and Marcus M Edvall. “Propt-matlab optimal control software”. In: *Tomlab Optimization Inc* 260 (2010).
- [110] Boris Houska, Hans Joachim Ferreau, and Moritz Diehl. “ACADO toolkitAn open-source framework for automatic control and dynamic optimization”. In: *Optimal Control Applications and Methods* 32.3 (2011), pp. 298–312.
- [111] CJ Goh and KL Teo. “MISER: a FORTRAN program for solving optimal control problems”. In: *Advances in Engineering Software (1978)* 10.2 (1988), pp. 90–99.
- [112] Oskar von Stryk. “Users Guide for DIRCOL”. In: *Technische Universität Darmstadt* (2000).
- [113] Philip E Gill and Elizabeth Wong. “Users Guide for SNCTRL”. In: (2015).
- [114] WG Vlases et al. “Optimal trajectories by implicit simulation”. In: *Boeing Aerospace and Electronics, Technical Report WRDC-TR-90-3056, Wright-Patterson Air Force Base* (1990).
- [115] GL Brauer, DEj Cornick, and R. Stevenson. “Capabilities and applications of the Program to Optimize Simulated Trajectories (POST). Program summary document”. In: (1977).
- [116] C Jansch, KH Well, and K Schnepfer. “GESOP-Eine Software Umgebung Zur Simulation Und Optimierung”. In: *Proceedings des SFB* (1994).
- [117] JT Betts. “Sparse optimization suite (SOS)”. In: *Applied Mathematical Analysis, LLC.(Based on the Algorithms Published in Betts, JT, Practical Methods for Optimal Control and Estimation Using Nonlinear Programming. SIAM Press, Philadelphia, PA.(2010).)* (2013).
- [118] Peter Kunkel. *Differential-algebraic equations: analysis and numerical solution*. European Mathematical Society, 2006.
- [119] Uri M Ascher, Robert MM Mattheij, and Robert D Russell. *Numerical solution of boundary value problems for ordinary differential equations*. Vol. 13. Siam, 1994.
- [120] Lawrence F Shampine, Jacek Kierzenka, and Mark W Reichelt. “Solving boundary value problems for ordinary differential equations in MATLAB with bvp4c”. In: *Tutorial notes* (2000), pp. 437–448.
- [121] Jacek Kierzenka and Lawrence F Shampine. “A BVP solver that controls residual and error”. In: *JNAIAM J. Numer. Anal. Ind. Appl. Math* 3.1-2 (2008), pp. 27–41.

- [122] Nicholas Hale and Daniel R Moore. “A sixth-order extension to the MATLAB package bvp4c of J. Kierzenka and L. Shampine”. In: (2008).
- [123] Luigi Brugnano and Donato Trigiante. “A new mesh selection strategy for ODEs”. In: *Applied Numerical Mathematics* 24.1 (1997), pp. 1–21.
- [124] F Mazzia and I Sgura. “Numerical approximation of nonlinear BVPs by means of BVMs”. In: *Applied Numerical Mathematics* 42.1 (2002), pp. 337–352.
- [125] Lidia Aceto, Francesca Mazzia, and Donato Trigiante. “The performances of the code TOM on the Holt problem”. In: *Modeling, Simulation, and Optimization of Integrated Circuits*. Springer, 2003, pp. 349–360.
- [126] Francesca Mazzia and Donato Trigiante. “A hybrid mesh selection strategy based on conditioning for boundary value ODE problems”. In: *Numerical Algorithms* 36.2 (2004), pp. 169–187.
- [127] JR Cash et al. “The role of conditioning in mesh selection algorithms for first order systems of linear two point boundary value problems”. In: *Journal of computational and applied mathematics* 185.2 (2006), pp. 212–224.
- [128] Asgeir Birkisson and Tobin A Driscoll. “Automatic Fréchet differentiation for the numerical solution of boundary-value problems”. In: *ACM Transactions on Mathematical Software (TOMS)* 38.4 (2012), p. 26.
- [129] Asgeir Birkisson and Tobin A Driscoll. “Automatic linearity detection”. In: (2013).
- [130] Tobin A Driscoll, Nicholas Hale, and Lloyd N Trefethen. *Chebfun guide*. 2014.
- [131] Hans Joachim Oberle and Werner Grimm. *BNDSCO: A program for the numerical solution of optimal control problems*. Inst. für Angewandte Math. der Univ. Hamburg Germany, 2001.
- [132] WH Enright and PH Muir. “Runge-Kutta software with defect control for boundary value ODEs”. In: *SIAM Journal on Scientific Computing* 17.2 (1996), pp. 479–497.
- [133] LF Shampine, PH Muir, and H Xu. “A User-Friendly Fortran BVP Solver1”. In: *JNAIAM* 1.2 (2006), pp. 201–217.
- [134] Jason J Boisvert, Paul H Muir, and Raymond J Spiteri. “A Runge-Kutta BVODE Solver with Global Error and Defect Control”. In: *ACM Transactions on Mathematical Software (TOMS)* 39.2 (2013), p. 11.
- [135] JR Cash and Margaret H Wright. “A deferred correction method for nonlinear two-point boundary value problems: implementation and numerical evaluation”. In: *SIAM journal on scientific and statistical computing* 12.4 (1991), pp. 971–989.
- [136] JR Cash and F Mazzia. “A new mesh selection algorithm, based on conditioning, for two-point boundary value codes”. In: *Journal of Computational and Applied Mathematics* 184.2 (2005), pp. 362–381.
- [137] Z Bashir-Ali, JR Cash, and HHM Silva. “Lobatto deferred correction for stiff two-point boundary value problems”. In: *Computers & Mathematics with Applications* 36.10 (1998), pp. 59–69.
- [138] Jeff R Cash and Francesca Mazzia. “Hybrid mesh selection algorithms based on conditioning for two-point boundary value problems”. In: *JNAIAM J. Numer. Anal. Indust. Appl. Math* 1.1 (2006), pp. 81–90.

- [139] Jeff R Cash, Gerald Moore, and Ross W Wright. “An automatic continuation strategy for the solution of singularly perturbed nonlinear boundary value problems”. In: *ACM Transactions on Mathematical Software (TOMS)* 27.2 (2001), pp. 245–266.
- [140] U Ascher, J Christiansen, and Robert D Russell. “Algorithm 569: COLSYS: Collocation software for boundary-value ODEs [D2]”. In: *ACM Transactions on Mathematical Software (TOMS)* 7.2 (1981), pp. 223–229.
- [141] Georg Bader and Uri Ascher. “A new basis implementation for a mixed order boundary value ODE solver”. In: *SIAM journal on scientific and statistical computing* 8.4 (1987), pp. 483–500.
- [142] Jason J Boisvert, Paul H Muir, and Raymond J Spiteri. “py_bvp: a universal Python interface for BVP codes”. In: *Proceedings of the 2010 Spring Simulation Multiconference*. Society for Computer Simulation International. 2010, p. 95.
- [143] The Numerical Algorithms Group (NAG). *The NAG Library*. URL: www.nag.com.
- [144] Kenneth Holmström, Anders Göran, and Marcus M Edvall. “Users Guide for Tomlab 7”. In: (2010).
- [145] *Community Portal for Automatic Differentiation*. 2016. URL: <http://www.autodiff.org/> (visited on 10/08/2016).
- [146] Harry Dankowicz and Frank Schilder. *Recipes for continuation*. SIAM, 2013.
- [147] Eusebius J Doedel et al. “AUTO-07P: Continuation and bifurcation software for ordinary differential equations”. In: (2007).
- [148] Georg Kitzhofer, Othmar Koch, and Ewa B Weinmüller. “Pathfollowing for essentially singular boundary value problems with application to the complex Ginzburg-Landau equation”. In: *BIT Numerical Mathematics* 49.1 (2009), pp. 141–160.
- [149] Eugene L Allgower and Kurt Georg. “Continuation and path following”. In: *Acta numerica* 2 (1993), pp. 1–64.
- [150] LV Kantorovich. “On Newtons method for functional equations”. In: *Dokl. Akad. Nauk SSSR*. Vol. 59. 7. 1948, pp. 1237–1240.
- [151] DF Davidenko. “On a new method of numerical solution of systems of nonlinear equations”. In: *Dokl. Akad. Nauk SSSR*. Vol. 88. 4. 1953, pp. 601–602.
- [152] Peter Deuffhard. *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*. Vol. 35. Springer Science & Business Media, 2011.
- [153] Basile Graf. “Quaternions and dynamics”. In: *arXiv preprint arXiv:0811.2889* (2008).
- [154] Brian L Stevens, Frank L Lewis, and Eric N Johnson. *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons, 2015.
- [155] David Baraff. “Physically based modeling: Rigid body simulation”. In: *SIGGRAPH Course Notes, ACM SIGGRAPH* 2.1 (2001), pp. 2–1.

Appendix A

Survey of Numerical Methods for Solving Optimal Control Problems: Dynamic Programming, the Direct Method, and the Indirect Method

There are three approaches to solving an optimal control problem: 1) dynamic programming, 2) the direct method, and 3) the indirect method. References [43, 90] present an introduction to dynamic programming. References [91, 92] are thorough survey articles on the direct and indirect methods. Reference [93] is a recent treatise providing detailed descriptions of both the direct and indirect methods, [94] is a comprehensive reference on the direct method, while [54] provides a comprehensive, modern treatment of the local collocation technique of the direct method.

In dynamic programming, a PDE, called the Hamilton-Jacobi-Bellman equation [95], is formulated and solved. However, due to the curse of dimensionality, solution of this PDE is only practical for very simple problems. Therefore, very few numerical solvers implement dynamic programming to solve optimal control problems. For example, BOCOPHJB [96] is free C++ software implementing the dynamic programming approach. References [62, 63, 64, 65] constitute recent research that seeks to overcome the curse of dimensionality in certain special cases.

Note that because the control function \mathbf{u} is an unknown function of time, an optimal control problem is infinite-dimensional. In the direct method, the infinite-dimensional optimal control problem is approximated by a finite-dimensional nonlinear programming (NLP) problem by parameterizing the control function \mathbf{u} as a finite linear combination of basis functions. In the sequential approach of the direct method, the state is reconstructed from a guess of the unknown coefficients for the control basis functions, the unknown parameters, the unknown initial states, and the unknown final time by multiple shooting or collocation. In the simultaneous approach of the direct method, the state is also parameterized as a finite linear combination

of basis functions. In the direct method, the ODE, initial conditions, final conditions, and path constraints are represented as a system of algebraic inequalities, and the objective function is minimized subject to satisfying the system of algebraic inequalities, with the unknowns being the coefficients for the control and/or state basis functions, parameters, initial states, and the final time. In the Lagrange and Bolza formulations, the objective function is approximated via numerical quadrature. There are many NLP solvers available, such as IPOPT [97], WORHP [98], SNOPT [99], KNITRO [100], and MATLAB's `fmincon` [101]; of these NLP solvers, only IPOPT and WORHP are free. Most direct method solvers utilize one of these NLP solvers.

The packages RIOTS [102], DYNOPT [103], ICLOCS [104], GPOPS [105], FALCON.m [106], and `OptimTraj` [107] are free MATLAB implementations of the direct method, while DIDO [108], PROPT [109], and GPOPS-II [55] are commercial MATLAB implementations of the direct method. BOCOP [71], ACADO [110], and *PSOPT* [72] are free C++ implementations of the direct method. MISER [111], DIRCOL [112], SNCTRL [113], OTIS [114], and POST [115] are free Fortran implementations of the direct method, while GESOP [116] and SOS [117] are commercial Fortran implementations of the direct method.

In the indirect method, Pontryagin's minimum principle uses the calculus of variations to formulate necessary conditions for a minimum solution to the optimal control problem. These necessary conditions take the form of a differential algebraic equation (DAE) with boundary conditions; such a problem is called a DAE boundary value problem (BVP). In some cases, through algebraic manipulation, it is possible to convert the DAE to an ordinary differential equation (ODE), thereby producing an ODE BVP.

A DAE BVP can be solved numerically by multiple shooting, collocation, or quasilinearization [118]. `bvpsuite` [67] is a free MATLAB collocation DAE BVP solver. COLDAE [66] is a free Fortran quasilinearization DAE BVP solver, which solves each linearized problem via collocation. The commercial Fortran code SOS, mentioned previously, also has the capability to solve DAE BVPs arising from optimal control problems via multiple shooting or collocation.

An ODE BVP can be solved numerically by multiple shooting, Runge-Kutta methods, collocation (which is a special subset of Runge-Kutta methods), finite-differences, or quasilinearization [119]. `bvp4c` [120], `bvp5c` [121], `bvp6c` [122], and `sbvp` [56] are MATLAB collocation ODE BVP solvers; `bvp4c` and `bvp5c` come standard with MATLAB, while `bvp6c` and `sbvp` are free. `bvptwp` [57] is a free MATLAB ODE BVP solver package implementing 6 algorithms: `twpbvp_m`, `twpbvpc_m`, `twpbvp_l`, `twpbvpc_l`, `acdc`, and `acdcc`; `acdc` and `acdcc` perform automatic continuation. `twpbvp_m` and `twpbvpc_m` rely on Runge-Kutta methods, while the other 4 algorithms rely on collocation. TOM [123, 124, 125, 126, 127] is a free MATLAB quasilinearization ODE BVP solver, which uses finite-differences to solve each linearized problem. `solvebvp` [84, 128, 129] is a MATLAB quasilinearization ODE BVP solver available in the free MATLAB toolbox `Chebfun` [130]; `solvebvp` uses spectral collocation to solve each linearized problem. COTCOT [60], `HamPath` [61], and BNDSCO [131] are free Fortran indirect method optimal control solvers that use multiple shooting to solve the ODE BVPs.

MIRKDC [132], BVP_SOLVER [133], and BVP_SOLVER-2 [134] and TWPBVP [135] and TWPBVPC [136] are free Fortran Runge-Kutta method ODE BVP solvers. TWPBVPL [137], TWPBVPLC [138], ACDC [139], and ACDCC [57] are free Fortran collocation ODE BVP solvers. `bvptwp`, mentioned previously, is a MATLAB reimplement of the Fortran solvers TWPBVP, TWPBVPC, TWPBVPL, TWPBVPLC, ACDC, and ACDCC. COLSYS [140], COLNEW [141], and COLMOD [139] are free Fortran collocation quasilinearization

ODE BVP solvers; COLMOD is an automatic continuation version of COLNEW. `bvpSolve` [73] is an R library that wraps the Fortran solvers TWPBVP, TWPBVPC, TWPBVPL, TWPBVPLC, ACDC, and ACDC and COLSYS, COLNEW, COLMOD, and COLDAE. `py_bvp` [142] is a Python library that wraps the Fortran solvers TWPBVPC, COLNEW, and BVP_SOLVER. The NAG Library [143] is a commercial Fortran library consisting of several multiple shooting, collocation, and finite-difference ODE BVP solvers. The Fortran solvers in the NAG Library are accessible from other languages (like C, Python, MATLAB, and .NET) via wrappers.

The NLP solver utilized by a direct method and the ODE/DAE BVP solver utilized by an indirect method must compute Jacobians and/or Hessians (i.e. first and/or second derivatives) of the functions involved in the optimal control problem. These derivatives may be approximated by finite-differences, but for increased accuracy and in many cases increased efficiency, exact (to machine precision) derivatives are desirable. These exact derivatives may be computed through symbolic or automatic differentiation. Usually, a symbolic derivative evaluates much more rapidly than an automatic derivative; however, due to expression explosion, symbolic derivatives cannot always be obtained for very complicated functions. The `Symbolic Math Toolbox` and `TomSym` [144] are commercial MATLAB toolboxes that compute symbolic derivatives. While the `Symbolic Math Toolbox` only computes non-vectorized symbolic derivatives, `TomSym` computes both non-vectorized and vectorized symbolic derivatives. `ADiGator` [40, 41] is a free MATLAB toolbox capable of computing both non-vectorized and vectorized automatic derivatives. Usually in MATLAB, a vectorized automatic derivative evaluates much more rapidly than a non-vectorized symbolic derivative (wrapped within a `for` loop). Only the MATLAB `Symbolic Math Toolbox` and `ADiGator` were utilized in this research. For other automatic differentiation packages available in many programming languages see [145].

A dynamic programming solution satisfies necessary and sufficient conditions for a global minimum solution of an optimal control problem. A direct method solution satisfies necessary and sufficient conditions for a local minimum solution of a finite-dimensional approximation of an optimal control problem, while an indirect method solution only satisfies necessary conditions for a local minimum solution of an optimal control problem. Thus, the dynamic programming approach is the holy grail for solving an optimal control problem; however, as mentioned previously, dynamic programming is impractical due to the curse of dimensionality. Therefore, in practice only direct and indirect methods are used to solve optimal control problems.

Since the direct method solves a finite-dimensional approximation of the original optimal control problem, the direct method is not as accurate as the indirect method. Moreover, the indirect method converges much more rapidly than the direct method. However, in addition to solving for the states and controls, the indirect method must also solve for the costates. Since the costates are unphysical, they are very difficult to guess initially. Therefore, though the direct method may be slower than the indirect method and may not be quite as accurate as the indirect method, the direct method is much more robust to poor initial guesses of the states and controls. Therefore, the preferred method of solution for many practical applications tends to be the direct method.

In some cases, it is possible to surmount the problem of providing a good initial guess required to obtain convergence via the indirect method. In such cases, the indirect method will converge substantially faster than the direct method. If a solution of a simpler optimal control problem is known and if the simpler and original optimal control problems are related by a continuous parameter, it may be possible to perform numerical continuation in the parameter from the solution of the simpler optimal control problem to a so-

lution of the original optimal control problem. In the literature, numerical continuation is also sometimes called the differential path following method or the homotopy method. Reference [50] is a comprehensive treatise on numerical continuation methods. `bvpsuite` implements a continuation algorithm to solve DAE BVPs, where the continuation parameter may have turning points (i.e. the continuation parameter need not monotonically increase or decrease). `COCO` [146], a free collection of MATLAB toolboxes, and `AUTO` [147], free Fortran software, implement sophisticated algorithms for the numerical continuation (permitting turning points) of ODE BVPs. `followpath`, available in the MATLAB toolbox `Chebfun`, is able to utilize `Chebfun`'s quasilinearization ODE BVP solver `bvpsolve` to solve ODE BVPs via continuation, where the continuation parameter may have turning points. Appendices C and D discuss predictor-corrector continuation algorithms for solving ODE TPBVPs, where the continuation parameter may have turning points; the algorithm presented in Appendix C is very similar to `followpath`. `acdc` / `ACDC`, `acdcc` / `ACDCC`, and `COLMOD` implement continuation algorithms to solve ODE BVPs, but the continuation parameter is assumed to monotonically increase or decrease. `HamPath`, mentioned previously, is a free Fortran indirect method optimal control solver which uses continuation (permitting turning points) in concert with multiple shooting. Of these numerical continuation tools, only `acdc`, `acdcc`, and the predictor-corrector continuation algorithms discussed in Appendices C and D were used in this research.

In order to converge to the solution of the true optimal control problem rather than a finite-dimensional approximation, a direct method may use h , p , or hp methods. In the h method, the degree of the approximating polynomial on each mesh interval is held fixed while the mesh is adaptively refined until the solution meets given error tolerances. In the p method, the mesh is held fixed while the degree of the approximating polynomial on each mesh interval is adaptively increased until the solution meets given error tolerances. In the hp method, which is implemented by `GPOPS-II`, the mesh and the degree of the approximating polynomial on each mesh interval are adaptively refined until the solution meets given error tolerances.

We have used the indirect method to numerically solve Suslov's optimal control problem, as it is vastly superior in speed compared to other methods and is capable of dealing with solutions having sharp gradients, if an appropriate BVP solver is utilized. This was made possible by constructing an analytical state and control solution to a singular optimal control problem and then by using continuation in the integrand cost function coefficients to solve the actual optimal control problem. The singular optimal control problem can be solved analytically since the initial conditions $\boldsymbol{\xi}(a)$ are not prescribed, because the constraint $\langle \boldsymbol{\Omega}(a), \boldsymbol{\xi}(a) \rangle = 0$ is not explicitly enforced, and because it is easy to solve Suslov's uncontrolled equations of motion for $\boldsymbol{\xi}$ in terms of $\boldsymbol{\Omega}$. Since the necessary conditions obtained by applying Pontryagin's minimum principle to Suslov's optimal control problem can be formulated as an ODE BVP, ODE rather than DAE BVP solvers were used. The MATLAB solvers `bvp4c`, `bvp5c`, `bvp6c`, `sbvp`, `bvptwp`, and `TOM` were used to solve the ODE BVP. `sbvp` and `bvptwp`, which have up to 8th-order accuracy, were found to be the most robust in solving the ODE BVP for Suslov's optimal control problem; the other MATLAB solvers were found to be very inefficient, requiring many thousands of mesh points due to their lower accuracy. The numerical results presented in Subsection 4.C.3 were obtained via `bvptwp`'s automatic continuation solver `acdc` and `sbvp`.

Appendix B

Calculation Connecting Classical and Reduced Costates for Suslov's Optimal Control Problem

This appendix verifies (4.88), which relates the classical and reduced costates for Suslov's optimal control problem. Recall the uncontrolled equations of motion for Suslov's problem (4.43):

$$\mathbf{q} = \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \left[\mathbb{I} \dot{\boldsymbol{\Omega}} - (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} \right] + \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} = \mathbf{0}. \quad (\text{B.1})$$

Solving for $\dot{\boldsymbol{\Omega}}$ in (B.1) yields

$$\dot{\boldsymbol{\Omega}} = \frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} \left\{ (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} - \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} \right\}. \quad (\text{B.2})$$

Define the right-hand side of (B.2) to be the function \mathbf{h} :

$$\mathbf{h}(\boldsymbol{\Omega}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}) = \frac{\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} \left\{ (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega} - \left[\langle \boldsymbol{\Omega}, \dot{\boldsymbol{\xi}} \rangle + \langle (\mathbb{I} \boldsymbol{\Omega}) \times \boldsymbol{\Omega}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \right] \boldsymbol{\xi} \right\}, \quad (\text{B.3})$$

so that $\dot{\boldsymbol{\Omega}} = \mathbf{h}(\boldsymbol{\Omega}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}})$. As it will be used later, observe that

$$\frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\xi}}} = \frac{-\mathbb{I}^{-1}}{\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle} \boldsymbol{\xi} \boldsymbol{\Omega}^T. \quad (\text{B.4})$$

Without loss of generality, assume that both the initial time a and final time b are fixed. Recall the reduced, augmented performance index S for Suslov's optimal control problem from (4.70). Using (B.1) and (B.3),

this performance index can be expressed as

$$\begin{aligned}
S &= \langle \boldsymbol{\rho}, \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \rangle + \langle \boldsymbol{\nu}, \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \rangle + \int_a^b [C + \langle \boldsymbol{\kappa}, \mathbf{q} \rangle] dt \\
&= \langle \boldsymbol{\rho}, \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \rangle + \langle \boldsymbol{\nu}, \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \rangle + \int_a^b \left[C - \left\langle \boldsymbol{\kappa}, \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} [\mathbf{h} - \dot{\boldsymbol{\Omega}}] \right\rangle \right] dt.
\end{aligned} \tag{B.5}$$

The variation of S with respect to $\boldsymbol{\Omega}$ is

$$\begin{aligned}
\delta_{\boldsymbol{\Omega}} S &= \int_a^b \left[\left\langle \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} \right)^\top, \delta \boldsymbol{\Omega} \right\rangle + \left\langle \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top, \delta \dot{\boldsymbol{\Omega}} \right\rangle - \left\langle \boldsymbol{\kappa}, \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \left[\frac{\partial \mathbf{h}}{\partial \boldsymbol{\Omega}} \delta \boldsymbol{\Omega} - \delta \dot{\boldsymbol{\Omega}} \right] \right\rangle \right] dt \\
&\quad + \langle \boldsymbol{\rho}, \delta \boldsymbol{\Omega}(a) \rangle + \langle \boldsymbol{\nu}, \delta \boldsymbol{\Omega}(b) \rangle \\
&= \int_a^b \left\langle \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} \right)^\top - \left(\frac{\partial \mathbf{h}}{\partial \boldsymbol{\Omega}} \right)^\top \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} - \frac{d}{dt} \left[\left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top + \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} \right], \delta \boldsymbol{\Omega} \right\rangle dt \\
&\quad + \left\langle \left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top + \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa}, \delta \boldsymbol{\Omega} \right\rangle \Big|_a^b + \langle \boldsymbol{\rho}, \delta \boldsymbol{\Omega}(a) \rangle + \langle \boldsymbol{\nu}, \delta \boldsymbol{\Omega}(b) \rangle.
\end{aligned} \tag{B.6}$$

Requiring that $\delta_{\boldsymbol{\Omega}} S = 0$ for all variations $\delta \boldsymbol{\Omega}$ such that $\delta \boldsymbol{\Omega}(a) = \delta \boldsymbol{\Omega}(b) = 0$ gives the controlled equations of motion for the reduced costates $\boldsymbol{\kappa}$:

$$\frac{d}{dt} \left[\left(\frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \right)^\top + \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} \right] = \left(\frac{\partial C}{\partial \boldsymbol{\Omega}} \right)^\top - \left(\frac{\partial \mathbf{h}}{\partial \boldsymbol{\Omega}} \right)^\top \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa}. \tag{B.7}$$

Recalling (3.9), the classical, augmented performance index for Suslov's optimal control problem is

$$\tilde{J} = \langle \boldsymbol{\rho}, \boldsymbol{\Omega}(a) - \boldsymbol{\Omega}_a \rangle + \langle \boldsymbol{\nu}, \boldsymbol{\Omega}(b) - \boldsymbol{\Omega}_b \rangle + \int_a^b \left[L + \langle \boldsymbol{\pi}_d, \mathbf{h} - \dot{\boldsymbol{\Omega}} \rangle + \langle \boldsymbol{\pi}_e, \mathbf{u} - \dot{\boldsymbol{\xi}} \rangle \right] dt, \tag{B.8}$$

with classical Hamiltonian

$$H(t, \mathbf{x}, \boldsymbol{\lambda}, \mathbf{u}) = L(\boldsymbol{\Omega}, \boldsymbol{\xi}, \mathbf{u}, t) + \left\langle \begin{bmatrix} \boldsymbol{\pi}_d \\ \boldsymbol{\pi}_e \end{bmatrix}, \begin{bmatrix} \mathbf{h}(\boldsymbol{\Omega}, \boldsymbol{\xi}, \mathbf{u}) \\ \mathbf{u} \end{bmatrix} \right\rangle, \tag{B.9}$$

classical states

$$\mathbf{x} = \begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\xi} \end{bmatrix}, \tag{B.10}$$

classical costates

$$\boldsymbol{\lambda} = \boldsymbol{\pi} = \begin{bmatrix} \boldsymbol{\pi}_d \\ \boldsymbol{\pi}_e \end{bmatrix}, \tag{B.11}$$

and where

$$L(\boldsymbol{\Omega}, \boldsymbol{\xi}, \mathbf{u}, t) = C(\boldsymbol{\Omega}, \dot{\boldsymbol{\Omega}}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}, t) = C(\boldsymbol{\Omega}, \mathbf{h}(\boldsymbol{\Omega}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}}), \boldsymbol{\xi}, \mathbf{u}, t) \tag{B.12}$$

since $\dot{\boldsymbol{\Omega}} = \mathbf{h}(\boldsymbol{\Omega}, \boldsymbol{\xi}, \dot{\boldsymbol{\xi}})$ and $\mathbf{u} = \dot{\boldsymbol{\xi}}$. Recall the latter two necessary conditions in the controlled equations of

motion (3.12)

$$\begin{aligned}\dot{\boldsymbol{\lambda}} &= -H_{\boldsymbol{x}}^{\top}(t, \boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u}) \\ \mathbf{0} &= H_{\boldsymbol{u}}^{\top}(t, \boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u}),\end{aligned}\tag{B.13}$$

where the continuation parameter μ has been omitted. For Suslov's optimal control problem, $\mathbf{0} = H_{\boldsymbol{u}}^{\top}(t, \boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u})$ translates into

$$\begin{aligned}\boldsymbol{\pi}_e &= - \left[\begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \boldsymbol{u}} \end{pmatrix}^{\top} \boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial L}{\partial \boldsymbol{u}} \end{pmatrix}^{\top} \right] \\ &= - \left[\begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} + \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \right] \\ &= - \left[\begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \left(\boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right) + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \right] \\ &= - \left[\begin{pmatrix} -\mathbb{I}^{-1} \\ \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \boldsymbol{\xi} \boldsymbol{\Omega}^{\top} \end{pmatrix}^{\top} \left(\boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right) + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \right] \\ &= - \left[\begin{pmatrix} -1 \\ \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \boldsymbol{\Omega} \boldsymbol{\xi}^{\top} \mathbb{I}^{-1} \end{pmatrix} \left(\boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right) + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \right]\end{aligned}\tag{B.14}$$

and $\dot{\boldsymbol{\lambda}} = -H_{\boldsymbol{x}}^{\top}(t, \boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{u})$ translates into

$$\begin{bmatrix} \dot{\boldsymbol{\pi}}_d \\ \dot{\boldsymbol{\pi}}_e \end{bmatrix} = - \begin{bmatrix} \begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} & 0 \\ \begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\pi}_d \\ \boldsymbol{\pi}_e \end{bmatrix} - \begin{bmatrix} \begin{pmatrix} \frac{\partial L}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \\ \begin{pmatrix} \frac{\partial L}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \end{bmatrix}.\tag{B.15}$$

The upper half of (B.15) is

$$\begin{aligned}\dot{\boldsymbol{\pi}}_d &= - \left[\begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial L}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right] \\ &= - \left[\begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} + \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right] \\ &= - \left[\begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} + \begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \left(\boldsymbol{\pi}_d + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right) \right].\end{aligned}\tag{B.16}$$

Matching (B.16) with (B.7) gives the relationship between $\boldsymbol{\pi}_d$ and $\boldsymbol{\kappa}$

$$\boldsymbol{\pi}_d = - \left[\langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \right].\tag{B.17}$$

From (B.14) and (B.17), the relationship between the classical costates, $\boldsymbol{\pi}_d$ and $\boldsymbol{\pi}_e$, and the reduced costates $\boldsymbol{\kappa}$ is

$$\boldsymbol{\pi} = \begin{bmatrix} \boldsymbol{\pi}_d \\ \boldsymbol{\pi}_e \end{bmatrix} = - \begin{bmatrix} \langle \boldsymbol{\xi}, \mathbb{I}^{-1} \boldsymbol{\xi} \rangle \mathbb{I} \boldsymbol{\kappa} + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\Omega}}} \end{pmatrix}^{\top} \\ \boldsymbol{\Omega} \boldsymbol{\xi}^{\top} \boldsymbol{\kappa} + \begin{pmatrix} \frac{\partial C}{\partial \dot{\boldsymbol{\xi}}} \end{pmatrix}^{\top} \end{bmatrix},\tag{B.18}$$

which verifies (4.88).

Appendix C

Predictor-Corrector Continuation Method for Solving an ODE TPBVP

C.1 Introduction

Suppose it is desired to solve the ODE TPBVP:

$$\begin{aligned}\frac{d}{ds}\mathbf{y}(s) &= \mathbf{F}(s, \mathbf{y}(s), \lambda) \\ \mathbf{G}(\mathbf{y}(a), \mathbf{y}(b), \lambda) &= \mathbf{0}_{n \times 1},\end{aligned}\tag{C.1}$$

where $a, b \in \mathbb{R}$ are prescribed with $a < b$, $s \in [a, b] \subset \mathbb{R}$ is the independent variable, $n \in \mathbb{N}$ is the prescribed number of dependent variables in \mathbf{y} , $\mathbf{y}: [a, b] \rightarrow \mathbb{R}^n$ is an unknown function which must be solved for, $\lambda \in \mathbb{R}$ is a prescribed scalar parameter, $\mathbf{F}: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ is a prescribed ODE velocity function defining the velocity of \mathbf{y} , and $\mathbf{G}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ is a prescribed two-point boundary condition function. Observe that if $n = 1$, \mathbf{y} , \mathbf{F} , and \mathbf{G} are scalar-valued functions, while if $n > 1$, \mathbf{y} , \mathbf{F} , and \mathbf{G} are vector-valued functions. The Jacobian of \mathbf{F} with respect to \mathbf{y} is $\mathbf{F}_{\mathbf{y}}: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$ and the Jacobian of \mathbf{F} with respect to λ is $\mathbf{F}_{\lambda}: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times 1}$. The Jacobian of \mathbf{G} with respect to $\mathbf{y}(a)$ is $\mathbf{G}_{\mathbf{y}(a)}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, the Jacobian of \mathbf{G} with respect to $\mathbf{y}(b)$ is $\mathbf{G}_{\mathbf{y}(b)}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, and the Jacobian of \mathbf{G} with respect to λ is $\mathbf{G}_{\lambda}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times 1}$. If \mathbf{F} is linear in \mathbf{y} and \mathbf{G} is linear in $\mathbf{y}(a)$ and $\mathbf{y}(b)$, then (C.1) is said to be a linear ODE TPBVP; otherwise, (C.1) is said to be a nonlinear ODE TPBVP.

Note that a solution \mathbf{y} to (C.1) depends on the given value of the scalar parameter λ , so a solution to (C.1) will be denoted by the pair (\mathbf{y}, λ) . Usually it is not possible to solve (C.1) analytically. Instead, a numerical method such as a shooting, finite-difference, or Runge-Kutta method (collocation is a special kind of Runge-Kutta method) must be utilized to construct an approximate solution to (C.1). All such numerical methods require an initial solution guess and convergence to a solution is guaranteed only if the initial solution guess is sufficiently near the solution. Thus, solving (C.1) numerically requires construction of a good initial solution guess.

One way to construct a good initial solution guess for (C.1) is through continuation in the scalar parameter λ . If $(\mathbf{y}_I, \lambda_I)$ solves (C.1) and it desired to solve (C.1) for $\lambda = \lambda_F$, it may be possible to construct a finite sequence of solutions $\{(\mathbf{y}_j, \lambda_j)\}_{j=1}^J$ starting at the known solution $(\mathbf{y}_1, \lambda_1) = (\mathbf{y}_I, \lambda_I)$ and ending at the desired solution $(\mathbf{y}_J, \lambda_J) = (\mathbf{y}_F, \lambda_F)$, using the previous solution $(\mathbf{y}_j, \lambda_j)$ as an initial solution guess for the numerical solver to obtain the next solution $(\mathbf{y}_{j+1}, \lambda_{j+1})$, $1 \leq j < J$, in the sequence. $J \in \mathbb{N}$ denotes the number of solutions in the sequence.

This appendix describes a particular such continuation method, called predictor-corrector continuation, for solving (C.1). The treatment given here follows [84]. In the literature, predictor-corrector continuation is also called pseudo-arclength continuation [84], path-following [148], predictor-corrector path-following [149], and differential path-following [61]. Before delving into the details, some functional analysis is reviewed which is necessary to understand how the predictor-corrector continuation method is applied to solve (C.1).

C.2 A Hilbert Space

Let $\mathcal{H} = \{(\mathbf{y}, \lambda) : \mathbf{y} \in L^2([a, b], \mathbb{R}^n), \lambda \in \mathbb{R}\}$. \mathcal{H} is a Hilbert space over \mathbb{R} . If $\alpha, \beta \in \mathbb{R}$ and $(\mathbf{y}, \lambda), (\tilde{\mathbf{y}}, \tilde{\lambda}) \in \mathcal{H}$, then

$$\alpha(\mathbf{y}, \lambda) + \beta(\tilde{\mathbf{y}}, \tilde{\lambda}) = (\alpha\mathbf{y} + \beta\tilde{\mathbf{y}}, \alpha\lambda + \beta\tilde{\lambda}), \quad (\text{C.2})$$

the inner product on \mathcal{H} is

$$\langle (\mathbf{y}, \lambda), (\tilde{\mathbf{y}}, \tilde{\lambda}) \rangle = \int_a^b \mathbf{y}^\top(s) \tilde{\mathbf{y}}(s) ds + \lambda \tilde{\lambda}, \quad (\text{C.3})$$

and the norm on \mathcal{H} , induced by the inner product, is

$$\|(\mathbf{y}, \lambda)\| = \langle (\mathbf{y}, \lambda), (\mathbf{y}, \lambda) \rangle^{\frac{1}{2}} = \left[\int_a^b \mathbf{y}^\top(s) \mathbf{y}(s) ds + \lambda^2 \right]^{\frac{1}{2}}. \quad (\text{C.4})$$

$(\mathbf{y}, \lambda) \in \mathcal{H}$ and $(\tilde{\mathbf{y}}, \tilde{\lambda}) \in \mathcal{H}$ are said to be orthogonal if

$$\langle (\mathbf{y}, \lambda), (\tilde{\mathbf{y}}, \tilde{\lambda}) \rangle = \int_a^b \mathbf{y}^\top(s) \tilde{\mathbf{y}}(s) ds + \lambda \tilde{\lambda} = 0. \quad (\text{C.5})$$

$(\mathbf{y}, \lambda) \in \mathcal{H}$ is said to be of unit length if

$$\|(\mathbf{y}, \lambda)\| = \langle (\mathbf{y}, \lambda), (\mathbf{y}, \lambda) \rangle^{\frac{1}{2}} = \left[\int_a^b \mathbf{y}^\top(s) \mathbf{y}(s) ds + \lambda^2 \right]^{\frac{1}{2}} = 1. \quad (\text{C.6})$$

C.3 The Fréchet Derivative and Newton's Method

Given a function $\mathbf{F} : \mathbb{R}^n \rightarrow \mathbb{R}^m$, recall that ordinary vector calculus defines the Jacobian of \mathbf{F} as the function $\mathbf{F}' : \mathbb{R}^n \rightarrow \mathbb{R}^{m \times n}$ such that $\mathbf{F}'(\mathbf{x})$ is the linearization of \mathbf{F} at $\mathbf{x} \in \mathbb{R}^n$. Given normed spaces V and W and an open subset U of V , the Fréchet derivative is an extension of the Jacobian to an operator $\mathcal{F} : U \rightarrow W$.

Before giving the definition of the Fréchet derivative, recall that $L(V, W)$ denotes the space of continuous linear operators from V to W . Now for the definition of the Fréchet derivative, which comes from Definition 2.2.4 of [84].

Definition C.1. *Suppose that V and W are normed spaces, and let U be an open subset of V . Then the operator $\mathcal{F} : U \rightarrow W$ is said to be Fréchet differentiable at $u \in U$ if and only if there exists an operator $\mathcal{L} \in L(V, W)$ such that*

$$\lim_{\|h\|_V \rightarrow 0} \frac{\|\mathcal{F}(u+h) - \mathcal{F}(u) - \mathcal{L}h\|_W}{\|h\|_V} = 0. \quad (\text{C.7})$$

The operator \mathcal{L} is then called the Fréchet derivative of \mathcal{F} at u , often denoted by $\mathcal{F}'(u)$. If \mathcal{F} is Fréchet differentiable at all points in U , \mathcal{F} is said to be Fréchet differentiable in U .

Given a function $\mathbf{H} : \mathbb{R}^m \rightarrow \mathbb{R}^m$, Newton's method is an algorithm to solve $\mathbf{H}(\mathbf{x}) = \mathbf{0}$ for $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{0} \in \mathbb{R}^m$ when \mathbf{H} satisfies certain mild conditions. Starting from an initial solution guess $\mathbf{x}_0 \in \mathbb{R}^m$ sufficiently close to a solution, Newton's method converges to a solution of $\mathbf{H}(\mathbf{x}) = \mathbf{0}$ by iteratively solving the equations

$$\mathbf{H}'(\mathbf{x}_k)\delta\mathbf{x}_k = -\mathbf{H}(\mathbf{x}_k), \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \delta\mathbf{x}_k, \quad (\text{C.8})$$

starting at $k = 0$, where \mathbf{H}' denotes the Jacobian of \mathbf{H} and $\mathbf{x}_k, \delta\mathbf{x}_k \in \mathbb{R}^m$ for $k \geq 0$. The iteration in (C.8) continues until $\mathbf{H}(\mathbf{x}_k) \approx \mathbf{0}$ (or $\delta\mathbf{x}_k \approx 0$) or until k exceeds a maximum iteration threshold. Now consider an operator $\mathcal{H} : U \subset V \rightarrow W$, where V and W are Banach spaces and U is an open subset of V . Kantorovich [150] provided an extension of Newton's method to solve $\mathcal{H}(u) = 0$ for $u \in U$ and $0 \in W$ when \mathcal{H} satisfies certain mild conditions. Starting from an initial solution guess $u_0 \in U$ sufficiently close to a solution, Kantorovich's extension of Newton's method converges to a solution of $\mathcal{H}(u) = 0$ by iteratively solving the equations

$$\mathcal{H}'(u_k)\delta u_k = -\mathcal{H}(u_k), \quad u_{k+1} = u_k + \delta u_k, \quad (\text{C.9})$$

starting at $k = 0$, where \mathcal{H}' denotes the Fréchet derivative of \mathcal{H} and $u_k, \delta u_k \in U$ for $k \geq 0$. The iteration in (C.9) continues until $\mathcal{H}(u_k) \approx 0$ (or $\delta u_k \approx 0$) or until k exceeds a maximum iteration threshold.

C.4 The Davidenko ODE IVP

To motivate the predictor-corrector continuation method, the Davidenko ODE IVP is first presented. Let $\mathcal{C} = \{(\mathbf{y}, \lambda) : (\mathbf{y}, \lambda) \text{ solves (C.1)}\}$ denote the solution manifold of (C.1). Suppose the solution manifold \mathcal{C} is parameterized by arclength ν , so that an element of \mathcal{C} is $(\mathbf{y}(\nu), \lambda(\nu))$, the tangent $(\mathbf{v}(\nu), \tau(\nu))$ to \mathcal{C} at $(\mathbf{y}(\nu), \lambda(\nu))$ satisfies $\|(\mathbf{v}(\nu), \tau(\nu))\|^2 = \int_a^b \mathbf{v}^\top(s, \nu)\mathbf{v}(s, \nu)ds + [\tau(\nu)]^2 = 1$ (i.e. $(\mathbf{v}(\nu), \tau(\nu))$ is a unit tangent), and the solution manifold \mathcal{C} can be described as a solution curve. With this arclength parameterization, $\mathbf{y} : [a, b] \times \mathbb{R} \rightarrow \mathbb{R}^n$, $\lambda : \mathbb{R} \rightarrow \mathbb{R}$, $\mathbf{v} : [a, b] \times \mathbb{R} \rightarrow \mathbb{R}^n$, $\tau : \mathbb{R} \rightarrow \mathbb{R}$, $\mathbf{y}(\nu)$ is shorthand for $\mathbf{y}(\cdot, \nu) : [a, b] \rightarrow \mathbb{R}^n$, and $\mathbf{v}(\nu)$ is shorthand for $\mathbf{v}(\cdot, \nu) : [a, b] \rightarrow \mathbb{R}^n$. Note that the components of the unit tangent $(\mathbf{v}(\nu), \tau(\nu))$ to \mathcal{C} at $(\mathbf{y}(\nu), \lambda(\nu))$ are given explicitly by $\mathbf{v}(s, \nu) = \frac{\partial \mathbf{y}(s, \nu)}{\partial \nu}$ and $\tau(\nu) = \frac{d\lambda(\nu)}{d\nu}$.

The Fréchet derivative of the ODE TPBVP (C.1) with respect to ν about the solution $(\mathbf{y}(\nu), \lambda(\nu))$, in conjunction with the arclength constraint and the initial condition $(\mathbf{y}_I, \lambda_I)$, gives the nonlinear ODE IVP

in the independent arclength variable ν :

$$\begin{aligned}
\frac{d}{ds}\mathbf{v}(s, \nu) &= \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}(s, \nu), \lambda(\nu)) \mathbf{v}(s, \nu) + \mathbf{F}_{\lambda}(s, \mathbf{y}(s, \nu), \lambda(\nu)) \tau(\nu), \\
\mathbf{0}_{n \times 1} &= \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}(a, \nu), \mathbf{y}(b, \nu), \lambda(\nu)) \mathbf{v}(a, \nu) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}(a, \nu), \mathbf{y}(b, \nu), \lambda(\nu)) \mathbf{v}(b, \nu) \\
&\quad + \mathbf{G}_{\lambda}(\mathbf{y}(a, \nu), \mathbf{y}(b, \nu), \lambda(\nu)) \tau(\nu), \\
\|(\mathbf{v}(\nu), \tau(\nu))\|^2 &= \langle (\mathbf{v}(\nu), \tau(\nu)), (\mathbf{v}(\nu), \tau(\nu)) \rangle = \int_a^b \mathbf{v}^T(s, \nu) \mathbf{v}(s, \nu) ds + [\tau(\nu)]^2 = 1, \\
(\mathbf{y}(\nu_0), \lambda(\nu_0)) &= (\mathbf{y}_I, \lambda_I),
\end{aligned} \tag{C.10}$$

which must be solved for $(\mathbf{y}(\nu), \lambda(\nu))$ starting at ν_0 from an initial solution $(\mathbf{y}_I, \lambda_I)$ of (C.1). (C.10) is called the Davidenko ODE IVP and its solution is called the Davidenko flow [151]. The first two equations in (C.10) constitute the Fréchet derivative of the ODE TPBVP (C.1), the third equation is the arclength constraint, and the final equation is the initial condition. By introducing a dummy scalar-valued function w to represent the integrand of the arclength constraint, (C.10) can be re-written:

$$\begin{aligned}
\frac{d}{ds}\mathbf{v}(s, \nu) &= \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}(s, \nu), \lambda(\nu)) \mathbf{v}(s, \nu) + \mathbf{F}_{\lambda}(s, \mathbf{y}(s, \nu), \lambda(\nu)) \tau(\nu), \\
\frac{d}{ds}w(s, \nu) &= \mathbf{v}^T(s, \nu) \mathbf{v}(s, \nu), \\
\mathbf{0}_{n \times 1} &= \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}(a, \nu), \mathbf{y}(b, \nu), \lambda(\nu)) \mathbf{v}(a, \nu) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}(a, \nu), \mathbf{y}(b, \nu), \lambda(\nu)) \mathbf{v}(b, \nu) \\
&\quad + \mathbf{G}_{\lambda}(\mathbf{y}(a, \nu), \mathbf{y}(b, \nu), \lambda(\nu)) \tau(\nu), \\
w(a, \nu) &= 0, \\
w(b, \nu) + [\tau(\nu)]^2 - 1 &= 0, \\
(\mathbf{y}(\nu_0), \lambda(\nu_0)) &= (\mathbf{y}_I, \lambda_I).
\end{aligned} \tag{C.11}$$

Again, letting ν vary, (C.11) is a nonlinear ODE IVP which must be solved for $(\mathbf{y}(\nu), \lambda(\nu))$ (i.e. $\mathbf{y}: [a, b] \times \mathbb{R} \rightarrow \mathbb{R}^n$ and $\lambda: \mathbb{R} \rightarrow \mathbb{R}$) starting at ν_0 from an initial solution $(\mathbf{y}_I, \lambda_I)$ of (C.1). However, for a fixed ν , (C.11) is a nonlinear ODE TPBVP which must be solved for $\mathbf{v}(\cdot, \nu): [a, b] \rightarrow \mathbb{R}^n$, $\tau(\nu) \in \mathbb{R}$, and $w(\cdot, \nu): [a, b] \rightarrow \mathbb{R}$ and where the independent variable is $s \in [a, b]$.

As explained in Chapter 5 of [152], it is inadvisable to integrate the Davidenko ODE IVP (C.10), or equivalently (C.11). Instead, a predictor-corrector continuation method, depicted in Figure C.1 and explained in detail in the following sections, is used to generate a solution sequence $\{(\mathbf{y}_j, \lambda_j)\}_{j=1}^J$ which is a discrete subset of the Davidenko flow such that $(\mathbf{y}_1, \lambda_1) = (\mathbf{y}_I, \lambda_I)$.

C.5 Construct the Tangent

Given a solution $(\mathbf{y}_j, \lambda_j)$ to (C.1) and a unit tangent $(\mathbf{v}_{j-1}, \tau_{j-1})$ to the previous solution $(\mathbf{y}_{j-1}, \lambda_{j-1})$ to (C.1), we seek to construct a tangent (\mathbf{v}_j, τ_j) to the solution curve \mathcal{C} at $(\mathbf{y}_j, \lambda_j)$ which is roughly of unit length. The arclength constraint is

$$\|(\mathbf{v}_j, \tau_j)\|^2 = \langle (\mathbf{v}_j, \tau_j), (\mathbf{v}_j, \tau_j) \rangle = \int_a^b \mathbf{v}_j^T(s) \mathbf{v}_j(s) ds + \tau_j^2 = 1, \tag{C.12}$$

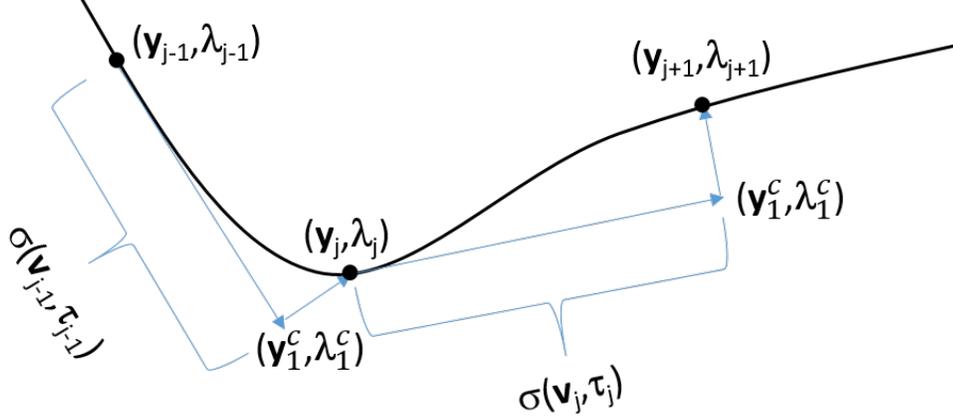


Figure C.1: Predictor-corrector continuation.

which is nonlinear in the tangent (\mathbf{v}_j, τ_j) . An alternative constraint, the pseudo-arclength constraint, is

$$\langle (\mathbf{v}_{j-1}, \tau_{j-1}), (\mathbf{v}_j, \tau_j) \rangle = \int_a^b \mathbf{v}_{j-1}^\top(s) \mathbf{v}_j(s) ds + \tau_{j-1} \tau_j = 1, \quad (\text{C.13})$$

which, in contrast to the arclength constraint (C.12), is linear in the tangent (\mathbf{v}_j, τ_j) . The linearization (i.e. Fréchet derivative) of the ODE TPBVP (C.1) about the solution $(\mathbf{y}_j, \lambda_j)$, in conjunction with the pseudo-arclength condition (C.13), gives the linear ODE TPBVP:

$$\begin{aligned} \frac{d}{ds} \mathbf{v}_j(s) &= \mathbf{F}_y(s, \mathbf{y}_j(s), \lambda_j) \mathbf{v}_j(s) \\ &\quad + \mathbf{F}_\lambda(s, \mathbf{y}_j(s), \lambda_j) \tau_j \\ \frac{d}{ds} \tau_j &= 0 \\ \frac{d}{ds} w(s) &= \mathbf{v}_{j-1}^\top(s) \mathbf{v}_j(s) \end{aligned} \quad (\text{C.14})$$

$$\begin{aligned} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(a) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(b) \\ + \mathbf{G}_\lambda(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \tau_j &= \mathbf{0}_{n \times 1} \\ w(a) &= 0 \\ w(b) + \tau_{j-1} \tau_j - 1 &= 0, \end{aligned}$$

which must be solved for $\mathbf{v}_j: [a, b] \rightarrow \mathbb{R}^n$, $\tau_j \in \mathbb{R}$, and $w: [a, b] \rightarrow \mathbb{R}$ and where (\mathbf{v}_j, τ_j) is a tangent to \mathcal{C} at $(\mathbf{y}_j, \lambda_j)$. Note that the first, second, and third equations in (C.14) are the ODEs, while the fourth, fifth, and sixth equations constitute the boundary conditions. The first, second, and fourth equations in (C.14) are the linearization (i.e. Fréchet derivative) of (C.1) about the solution $(\mathbf{y}_j, \lambda_j)$ and ensure that a tangent is produced, while the third, fifth, and sixth equations in (C.14) enforce the pseudo-arclength condition (C.13). The initial solution guess to solve (C.14) is $(\mathbf{v}_j, \tau_j) = (\mathbf{v}_{j-1}, \tau_{j-1})$ and $w(s) = \int_a^s \mathbf{v}_{j-1}^\top(\tilde{s}) \mathbf{v}_{j-1}(\tilde{s}) d\tilde{s}$, $s \in [a, b]$, for $j \geq 1$. For $j = 1$, define $(\mathbf{v}_0, \tau_0) = (\mathbf{0}, 1)$. Note that the construction of the initial guess for w can be realized efficiently via the MATLAB routine `cumtrapz`.

Note that the linear ODE TPBVP (C.14) can be solved numerically via the MATLAB routines `sbvp` or `bvptwp`,

which offers 4 algorithms: `twpbvp-m`, `twpbvpc-m`, `twpbvp-l`, and `twpbvpc-l`; moreover, `sbvp` and `bvptwp` have special algorithms to solve linear ODE TPBVP. Since \mathbf{y}_j and \mathbf{v}_{j-1} are usually only known for a discrete set of points in $[a, b]$, the values of these functions at the other points in $[a, b]$ must be obtained through interpolation in order to numerically solve (C.14). The MATLAB routine `interp1` performs linear, cubic, `pchip`, and spline interpolation and may be utilized to interpolate \mathbf{y}_j and \mathbf{v}_{j-1} while solving (C.14).

Because the numerical solvers usually converge faster when provided Jacobians of the ODE velocity function and of the two-point boundary condition function, these are computed below. Let

$$\mathbf{x} = \begin{bmatrix} \mathbf{v}_j \\ \tau_j \\ w \end{bmatrix}. \quad (\text{C.15})$$

The ODE velocity function in (C.14) is

$$\mathbf{H}^t(s, \mathbf{x}(s)) = \mathbf{H}^t(s, \mathbf{v}_j(s), \tau_j, w(s)) = \begin{bmatrix} \mathbf{F}_y(s, \mathbf{y}_j(s), \lambda_j) \mathbf{v}_j(s) + \mathbf{F}_\lambda(s, \mathbf{y}_j(s), \lambda_j) \tau_j \\ 0 \\ \mathbf{v}_{j-1}^\top(s) \mathbf{v}_j(s) \end{bmatrix}. \quad (\text{C.16})$$

The Jacobian of the ODE velocity function \mathbf{H}^t with respect to \mathbf{x} is

$$\begin{aligned} \mathbf{H}_x^t(s, \mathbf{x}(s)) &= \mathbf{H}_x^t(s, \mathbf{v}_j(s), \tau_j, w(s)) \\ &= \begin{bmatrix} \mathbf{F}_y(s, \mathbf{y}_j(s), \lambda_j) & \mathbf{F}_\lambda(s, \mathbf{y}_j(s), \lambda_j) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{v}_{j-1}^\top(s) & 0 & 0 \end{bmatrix}. \end{aligned} \quad (\text{C.17})$$

The two-point boundary condition in (C.14) is

$$\mathbf{K}^t(\mathbf{x}(a), \mathbf{x}(b)) = \mathbf{0}_{(n+2) \times 1}, \quad (\text{C.18})$$

where \mathbf{K}^t is the two-point boundary condition function

$$\mathbf{K}^t(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(a) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(b) + \mathbf{G}_\lambda(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \tau_j \\ w(a) \\ w(b) + \tau_{j-1} \tau_j - 1 \end{bmatrix}. \quad (\text{C.19})$$

The Jacobians of the two-point boundary condition function \mathbf{K}^t with respect to $\mathbf{x}(a)$ and $\mathbf{x}(b)$ are

$$\mathbf{K}_{\mathbf{x}(a)}^t(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{G}_\lambda(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 1 \\ \mathbf{0}_{1 \times n} & \tau_{j-1} & 0 \end{bmatrix} \quad (\text{C.20})$$

and

$$\mathbf{K}_{\mathbf{x}(b)}^t(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{G}_\lambda(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{0}_{1 \times n} & \tau_{j-1} & 1 \end{bmatrix}. \quad (\text{C.21})$$

Special care must be taken when implementing the Jacobians (C.20) and (C.21). Since the unknown constant τ_j appears as the second to last element of both $\mathbf{x}(a)$ and $\mathbf{x}(b)$, τ_j from only one of $\mathbf{x}(a)$ and $\mathbf{x}(b)$ is actually used to construct each term in \mathbf{K}^t involving τ_j . The middle column of (C.20) is actually the derivative of \mathbf{K}^t with respect to the τ_j in $\mathbf{x}(a)$, while the middle column of (C.21) is actually the derivative of \mathbf{K}^t with respect to the τ_j in $\mathbf{x}(b)$. Thus, the middle columns in (C.20) and (C.21) corresponding to the derivative of \mathbf{K}^t with respect to τ_j should not coincide in a software implementation. For example, if \mathbf{K}^t is constructed from the τ_j in $\mathbf{x}(a)$, $\mathbf{K}_{\mathbf{x}(a)}^t$ is as shown in (C.20) while the middle column of (C.21) corresponding to the derivative of \mathbf{K}^t with respect to the τ_j in $\mathbf{x}(b)$ is all zeros. Alternatively, if \mathbf{K}^t is constructed from the τ_j in $\mathbf{x}(b)$, $\mathbf{K}_{\mathbf{x}(b)}^t$ is as shown in (C.21) while the middle column of (C.20) corresponding to the derivative of \mathbf{K}^t with respect to the τ_j appearing in $\mathbf{x}(a)$ is all zeros.

C.6 Normalize the Tangent

The tangent (\mathbf{v}_j, τ_j) at $(\mathbf{y}_j, \lambda_j)$ obtained by solving (C.14) in the previous step is only roughly of unit length. A unit tangent at $(\mathbf{y}_j, \lambda_j)$ is obtained from (\mathbf{v}_j, τ_j) through normalization:

$$(\mathbf{v}_j, \tau_j) \leftarrow \frac{1}{\kappa} (\mathbf{v}_j, \tau_j), \quad (\text{C.22})$$

where

$$\kappa = \|(\mathbf{v}_j, \tau_j)\| = \langle (\mathbf{v}_j, \tau_j), (\mathbf{v}_j, \tau_j) \rangle^{\frac{1}{2}} = \left[\int_a^b \mathbf{v}_j^\top(s) \mathbf{v}_j(s) ds + \tau_j^2 \right]^{\frac{1}{2}}. \quad (\text{C.23})$$

The integration operator to construct the normalization scalar κ in (C.23) can be realized via the MATLAB routine `trapz`.

C.7 Construct the Tangent Predictor

The unit tangent (\mathbf{v}_j, τ_j) constructed in (C.22) is used to obtain a guess (the so-called ‘‘tangent predictor’’) $(\mathbf{y}_1^c, \lambda_1^c)$ for the next solution $(\mathbf{y}_{j+1}, \lambda_{j+1})$ as follows:

$$(\mathbf{y}_1^c, \lambda_1^c) = (\mathbf{y}_j, \lambda_j) + \sigma (\mathbf{v}_j, \tau_j), \quad (\text{C.24})$$

where $\sigma \in [\sigma_{\min}, \sigma_{\max}]$ is a steplength and where $0 < \sigma_{\min} \leq \sigma_{\max}$. Concretely, σ_{\min} might be .0001 and σ_{\max} might be $\frac{1}{2}$. σ is adapted during the predictor-corrector continuation method based on the corrector step, discussed in the next section. Initially, the value of σ is set to $\sigma_{\text{init}} \in [\sigma_{\min}, \sigma_{\max}]$. The notation $(\mathbf{y}_1^c, \lambda_1^c)$ is used to denote the tangent predictor in (C.24) because, as discussed in the next section, the tangent predictor is used as the initial corrector in an iterative Newton’s method that projects the tangent predictor onto \mathcal{C} .

C.8 Construct the Corrector

Since the tangent predictor $(\mathbf{y}_1^c, \lambda_1^c)$ constructed in (C.24) does not necessarily lie on \mathcal{C} , $(\mathbf{y}_1^c, \lambda_1^c)$ must be projected onto \mathcal{C} to obtain the next solution (the so-called ‘‘corrector’’) $(\mathbf{y}_{j+1}, \lambda_{j+1})$. This projection process is the corrector step. In order to perform the projection efficiently, the difference between the next solution and the tangent predictor, $(\mathbf{y}_{j+1}, \lambda_{j+1}) - (\mathbf{y}_1^c, \lambda_1^c)$, should be orthogonal to the unit tangent (\mathbf{v}_j, τ_j) . That is, the orthogonality constraint is

$$\begin{aligned} \langle (\mathbf{v}_j, \tau_j), (\mathbf{y}_{j+1}, \lambda_{j+1}) - (\mathbf{y}_1^c, \lambda_1^c) \rangle &= \langle (\mathbf{v}_j, \tau_j), (\mathbf{y}_{j+1} - \mathbf{y}_1^c, \lambda_{j+1} - \lambda_1^c) \rangle \\ &= \int_a^b \mathbf{v}_j^\top(s) [\mathbf{y}_{j+1}(s) - \mathbf{y}_1^c(s)] ds + \tau_j [\lambda_{j+1} - \lambda_1^c] = 0. \end{aligned} \quad (\text{C.25})$$

The tangent predictor $(\mathbf{y}_1^c, \lambda_1^c)$ can be iteratively corrected by applying Newton’s method to (C.1), while enforcing the orthogonality constraint (C.25), to generate a sequence of correctors $\{(\mathbf{y}_k^c, \lambda_k^c)\}_{k=1}^K$. Applying Newton’s method to the ODE TPBVP (C.1) about the current corrector $(\mathbf{y}_k^c, \lambda_k^c)$, in conjunction with the orthogonality constraint (C.25), gives the linear ODE TPBVP:

$$\begin{aligned} \frac{d}{ds} \delta \mathbf{y}_k^c(s) &= \mathbf{F}_y(s, \mathbf{y}_k^c(s), \lambda_k^c) \delta \mathbf{y}_k^c(s) \\ &\quad + \mathbf{F}_\lambda(s, \mathbf{y}_k^c(s), \lambda_k^c) \delta \lambda_k^c \\ &\quad - \frac{d}{ds} \mathbf{y}_k^c(s) + \mathbf{F}(s, \mathbf{y}_k^c(s), \lambda_k^c) \\ \frac{d}{ds} \delta \lambda_k^c &= 0 \\ \frac{d}{ds} w(s) &= \mathbf{v}_j^\top(s) \delta \mathbf{y}_k^c(s) \end{aligned} \quad (\text{C.26})$$

$$\begin{aligned} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \delta \mathbf{y}_k^c(a) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \delta \mathbf{y}_k^c(b) \\ + \mathbf{G}_\lambda(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \delta \lambda_k^c + \mathbf{G}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) = \mathbf{0}_{n \times 1} \\ w(a) = 0 \\ w(b) + \tau_j \delta \lambda_k^c = 0, \end{aligned}$$

which must be solved for $\delta \mathbf{y}_k^c: [a, b] \rightarrow \mathbb{R}^n$, $\delta \lambda_k^c \in \mathbb{R}$, and $w: [a, b] \rightarrow \mathbb{R}$ and where $(\delta \mathbf{y}_k^c, \delta \lambda_k^c)$ represents a correction to the current corrector $(\mathbf{y}_k^c, \lambda_k^c)$. Note that the first, second, and third equations in (C.26) are the ODEs, while the fourth, fifth, and sixth equations constitute the boundary conditions. The first, second, and fourth equations in (C.26) are the result of applying Newton’s method to (C.1) about the current corrector $(\mathbf{y}_k^c, \lambda_k^c)$, while the third, fifth, and sixth equations in (C.26) enforce the orthogonality constraint (C.25). (C.26) must be solved iteratively for at most K iterations, so that $1 \leq k \leq K$. The initial guess at the beginning of each iteration is $(\delta \mathbf{y}_k^c, \delta \lambda_k^c) = (\mathbf{0}, 0)$ and $w(s) = 0$, $s \in [a, b]$. The initial corrector about which Newton’s method is applied in the first iteration is the tangent predictor $(\mathbf{y}_1^c, \lambda_1^c)$. At the end of each iteration, the corrector about which Newton’s method is applied for the next iteration is updated via $(\mathbf{y}_{k+1}^c, \lambda_{k+1}^c) = (\mathbf{y}_k^c, \lambda_k^c) + (\delta \mathbf{y}_k^c, \delta \lambda_k^c)$. At the end of each iteration, convergence to \mathcal{C} should be tested via:

$$\frac{\|(\delta \mathbf{y}_k^c, \delta \lambda_k^c)\|}{\|(\mathbf{y}_1^c, \lambda_1^c)\|} = \frac{\left[\int_a^b [\delta \mathbf{y}_k^c(s)]^\top \delta \mathbf{y}_k^c(s) ds + [\delta \lambda_k^c]^2 \right]^{\frac{1}{2}}}{\left[\int_a^b [\mathbf{y}_1^c(s)]^\top \mathbf{y}_1^c(s) ds + [\lambda_1^c]^2 \right]^{\frac{1}{2}}} < \gamma, \quad (\text{C.27})$$

where γ is a small threshold such as .001. Since Newton's method enjoys quadratic convergence near a solution, only a few (say $K = 5$) iterative solves of (C.26) should be attempted. If convergence has not been attained in K iterations, the steplength σ should be reduced:

$$\sigma \leftarrow \sigma_r \sigma, \quad (\text{C.28})$$

where σ_r is a reduction scale factor such as $\frac{1}{4}$ and the corrector step should be restarted at the new tangent predictor $(\mathbf{y}_1^c, \lambda_1^c) = (\mathbf{y}_j, \lambda_j) + \sigma(\mathbf{v}_j, \tau_j)$, based on the updated value of σ realized in (C.28). If, as a result of the reduction realized in (C.28), $\sigma < \sigma_{\min}$, the algorithm should halt and predictor-corrector continuation failed. However, if convergence has been achieved in $k + 1 \leq K$ iterations, the next solution can be taken to be $(\mathbf{y}_{j+1}, \lambda_{j+1}) = (\mathbf{y}_{k+1}^c, \lambda_{k+1}^c)$ or the corrector can be further polished as explained in the next section. Moreover, if convergence has been achieved rapidly in no more than k_{fast} iterations, where $1 \leq k_{\text{fast}} \leq K$ and, concretely, k_{fast} might be 3, then the steplength σ may be increased:

$$\sigma \leftarrow \min \{ \sigma_i \sigma, \sigma_{\max} \}, \quad (\text{C.29})$$

where σ_i is an increase scale factor such as 2.

Note that the linear ODE TPBVP (C.26) can be solved numerically via the MATLAB routines `sbvp` or `bvptwp`, which offers 4 algorithms: `twpbvp_m`, `twpbvpc_m`, `twpbvp_l`, and `twpbvpc_l`; moreover, `sbvp` and `bvptwp` have special algorithms to solve linear ODE TPBVP. Since \mathbf{y}_k^c , $\frac{d}{ds}\mathbf{y}_k^c$, and \mathbf{v}_j are usually only known for a discrete set of points in $[a, b]$, the values of these functions at the other points in $[a, b]$ must be obtained through interpolation in order to numerically solve (C.26). The MATLAB routine `interp1` performs linear, cubic, `pchip`, and spline interpolation and may be utilized to interpolate \mathbf{y}_k^c , $\frac{d}{ds}\mathbf{y}_k^c$, and \mathbf{v}_j while solving (C.26).

Because the numerical solvers usually converge faster when provided Jacobians of the ODE velocity function and of the two-point boundary condition function, these are computed below. Let

$$\mathbf{x} = \begin{bmatrix} \delta \mathbf{y}_k^c \\ \delta \lambda_k^c \\ w \end{bmatrix}. \quad (\text{C.30})$$

The ODE velocity function in (C.26) is

$$\begin{aligned} \mathbf{H}^c(s, \mathbf{x}(s)) &= \mathbf{H}^c(s, \delta \mathbf{y}_k^c(s), \delta \lambda_k^c, w(s)) \\ &= \begin{bmatrix} \mathbf{F}_y(s, \mathbf{y}_k^c(s), \lambda_k^c) \delta \mathbf{y}_k^c(s) + \mathbf{F}_\lambda(s, \mathbf{y}_k^c(s), \lambda_k^c) \delta \lambda_k^c - \frac{d}{ds} \mathbf{y}_k^c(s) + \mathbf{F}(s, \mathbf{y}_k^c(s), \lambda_k^c) \\ 0 \\ \mathbf{v}_j^\top(s) \delta \mathbf{y}_k^c(s) \end{bmatrix}. \end{aligned} \quad (\text{C.31})$$

The Jacobian of the ODE velocity function \mathbf{H}^c with respect to \mathbf{x} is

$$\begin{aligned}\mathbf{H}_{\mathbf{x}}^c(s, \mathbf{x}(s)) &= \mathbf{H}_{\mathbf{x}}^c(s, \delta\mathbf{y}_k^c(s), \delta\lambda_k^c, w(s)) \\ &= \begin{bmatrix} \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}_k^c(s), \lambda_k^c) & \mathbf{F}_{\lambda}(s, \mathbf{y}_k^c(s), \lambda_k^c) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{v}_j^T(s) & 0 & 0 \end{bmatrix}.\end{aligned}\quad (\text{C.32})$$

The two-point boundary condition in (C.26) is

$$\mathbf{K}^c(\mathbf{x}(a), \mathbf{x}(b)) = \mathbf{0}_{(n+2) \times 1}, \quad (\text{C.33})$$

where \mathbf{K}^c is the two-point boundary condition function

$$\mathbf{K}^c(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \delta\mathbf{y}_k^c(a) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \delta\mathbf{y}_k^c(b) \\ + \mathbf{G}_{\lambda}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \delta\lambda_k^c + \mathbf{G}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) \\ w(a) \\ w(b) + \tau_j \delta\lambda_k^c \end{bmatrix}. \quad (\text{C.34})$$

The Jacobians of the two-point boundary condition function \mathbf{K}^c with respect to $\mathbf{x}(a)$ and $\mathbf{x}(b)$ are

$$\mathbf{K}_{\mathbf{x}(a)}^c(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) & \mathbf{G}_{\lambda}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 1 \\ \mathbf{0}_{1 \times n} & \tau_j & 0 \end{bmatrix} \quad (\text{C.35})$$

and

$$\mathbf{K}_{\mathbf{x}(b)}^c(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) & \mathbf{G}_{\lambda}(\mathbf{y}_k^c(a), \mathbf{y}_k^c(b), \lambda_k^c) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{0}_{1 \times n} & \tau_j & 1 \end{bmatrix}. \quad (\text{C.36})$$

Special care must be taken when implementing the Jacobians (C.35) and (C.36). Since the unknown constant $\delta\lambda_k^c$ appears as the second to last element of both $\mathbf{x}(a)$ and $\mathbf{x}(b)$, $\delta\lambda_k^c$ from only one of $\mathbf{x}(a)$ and $\mathbf{x}(b)$ is actually used to construct each term in \mathbf{K}^c involving $\delta\lambda_k^c$. The middle column of (C.35) is actually the derivative of \mathbf{K}^c with respect to the $\delta\lambda_k^c$ in $\mathbf{x}(a)$, while the middle column of (C.36) is actually the derivative of \mathbf{K}^c with respect to the $\delta\lambda_k^c$ in $\mathbf{x}(b)$. Thus, the middle columns in (C.35) and (C.36) corresponding to the derivative of \mathbf{K}^c with respect to $\delta\lambda_k^c$ should not coincide in a software implementation. For example, if \mathbf{K}^c is constructed from the $\delta\lambda_k^c$ in $\mathbf{x}(a)$, $\mathbf{K}_{\mathbf{x}(a)}^c$ is as shown in (C.35) while the middle column of (C.36) corresponding to the derivative of \mathbf{K}^c with respect to the $\delta\lambda_k^c$ in $\mathbf{x}(b)$ is all zeros. Alternatively, if \mathbf{K}^c is constructed from the $\delta\lambda_k^c$ in $\mathbf{x}(b)$, $\mathbf{K}_{\mathbf{x}(b)}^c$ is as shown in (C.36) while the middle column of (C.35) corresponding to the derivative of \mathbf{K}^c with respect to the $\delta\lambda_k^c$ appearing in $\mathbf{x}(a)$ is all zeros.

C.9 Polish the Corrector

The final corrector $(\mathbf{y}_{k+1}^c, \lambda_{k+1}^c)$ from the previous step can be further polished by finding $(\mathbf{y}_{j+1}, \lambda_{j+1})$ that solves (C.1) while satisfying the orthogonality constraint (C.25). This yields the ODE TPBVP:

$$\begin{aligned}
 \frac{d}{ds} \mathbf{y}_{j+1}(s) &= \mathbf{F}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) \\
 \frac{d}{ds} \lambda_{j+1} &= 0 \\
 \frac{d}{ds} w(s) &= \mathbf{v}_j^\top(s) [\mathbf{y}_{j+1}(s) - \mathbf{y}_1^c(s)] \\
 \mathbf{G}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) &= \mathbf{0}_{n \times 1} \\
 w(a) &= 0 \\
 w(b) + \tau_j [\lambda_{j+1} - \lambda_1^c] &= 0,
 \end{aligned} \tag{C.37}$$

which must be solved for $\mathbf{y}_{j+1}: [a, b] \rightarrow \mathbb{R}^n$, $\lambda_{j+1} \in \mathbb{R}$, and $w: [a, b] \rightarrow \mathbb{R}$. Note that the first, second, and third equations in (C.37) are the ODEs, while the fourth, fifth, and sixth equations constitute the boundary conditions. The first, second, and fourth equations in (C.37) ensure that the solution lies on \mathcal{C} (i.e. satisfies (C.1)), while the third, fifth, and sixth equations in (C.37) enforce the orthogonality constraint (C.25). The initial solution guess to solve (C.37) is the final corrector $(\mathbf{y}_{k+1}^c, \lambda_{k+1}^c)$ from the previous step and $w(s) = 0$, $s \in [a, b]$.

Note that the ODE TPBVP (C.37) can be solved numerically via the MATLAB routines `sbvp` or `bvptwp`, which offers 4 algorithms: `twpbvp_m`, `twpbvpc_m`, `twpbvp_l`, and `twpbvpc_l`. Since \mathbf{y}_1^c and \mathbf{v}_j are usually only known for a discrete set of points in $[a, b]$, the values of these functions at the other points in $[a, b]$ must be obtained through interpolation in order to numerically solve (C.37). The MATLAB routine `interp1` performs linear, cubic, `pchip`, and spline interpolation and may be utilized to interpolate \mathbf{y}_1^c and \mathbf{v}_j while solving (C.37).

Because the numerical solvers usually converge faster when provided Jacobians of the ODE velocity function and of the two-point boundary condition function, these are computed below. Let

$$\mathbf{x} = \begin{bmatrix} \mathbf{y}_{j+1} \\ \lambda_{j+1} \\ w \end{bmatrix}. \tag{C.38}$$

The ODE velocity function in (C.37) is

$$\mathbf{H}^p(s, \mathbf{x}(s)) = \mathbf{H}^p(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}, w(s)) = \begin{bmatrix} \mathbf{F}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) \\ 0 \\ \mathbf{v}_j^\top(s) [\mathbf{y}_{j+1}(s) - \mathbf{y}_1^c(s)] \end{bmatrix}. \tag{C.39}$$

The Jacobian of the ODE velocity function \mathbf{H}^p with respect to \mathbf{x} is

$$\begin{aligned}\mathbf{H}_{\mathbf{x}}^p(s, \mathbf{x}(s)) &= \mathbf{H}_{\mathbf{x}}^p(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}, w(s)) \\ &= \begin{bmatrix} \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) & \mathbf{F}_{\lambda}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{v}_j^{\top}(s) & 0 & 0 \end{bmatrix}.\end{aligned}\quad (\text{C.40})$$

The two-point boundary condition in (C.37) is

$$\mathbf{K}^p(\mathbf{x}(a), \mathbf{x}(b)) = \mathbf{0}_{(n+2) \times 1}, \quad (\text{C.41})$$

where \mathbf{K}^p is the two-point boundary condition function

$$\mathbf{K}^p(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) \\ w(a) \\ w(b) + \tau_j [\lambda_{j+1} - \lambda_1^c] \end{bmatrix}. \quad (\text{C.42})$$

The Jacobians of the two-point boundary condition function \mathbf{K}^p with respect to $\mathbf{x}(a)$ and $\mathbf{x}(b)$ are

$$\mathbf{K}_{\mathbf{x}(a)}^p(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{G}_{\lambda}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 1 \\ \mathbf{0}_{1 \times n} & \tau_j & 0 \end{bmatrix} \quad (\text{C.43})$$

and

$$\mathbf{K}_{\mathbf{x}(b)}^p(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{G}_{\lambda}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{0}_{1 \times n} & \tau_j & 1 \end{bmatrix}. \quad (\text{C.44})$$

Special care must be taken when implementing the Jacobians (C.43) and (C.44). Since the unknown constant λ_{j+1} appears as the second to last element of both $\mathbf{x}(a)$ and $\mathbf{x}(b)$, λ_{j+1} from only one of $\mathbf{x}(a)$ and $\mathbf{x}(b)$ is actually used to construct each term in \mathbf{K}^p involving λ_{j+1} . The middle column of (C.43) is actually the derivative of \mathbf{K}^p with respect to the λ_{j+1} in $\mathbf{x}(a)$, while the middle column of (C.44) is actually the derivative of \mathbf{K}^p with respect to the λ_{j+1} in $\mathbf{x}(b)$. Thus, the middle columns in (C.43) and (C.44) corresponding to the derivative of \mathbf{K}^p with respect to λ_{j+1} should not coincide in a software implementation. For example, if \mathbf{K}^p is constructed from the λ_{j+1} in $\mathbf{x}(a)$, $\mathbf{K}_{\mathbf{x}(a)}^p$ is as shown in (C.43) while the middle column of (C.44) corresponding to the derivative of \mathbf{K}^p with respect to the λ_{j+1} in $\mathbf{x}(b)$ is all zeros. Alternatively, if \mathbf{K}^p is constructed from the λ_{j+1} in $\mathbf{x}(b)$, $\mathbf{K}_{\mathbf{x}(b)}^p$ is as shown in (C.44) while the middle column of (C.43) corresponding to the derivative of \mathbf{K}^p with respect to the λ_{j+1} appearing in $\mathbf{x}(a)$ is all zeros.

C.10 Pseudocode for Predictor-Corrector Continuation

Below is pseudocode that realizes the predictor-corrector continuation method.

Algorithm C.1 Predictor-Corrector Continuation for Nonlinear ODE TPBVPs. Part 1.

Input: ODE velocity function $\mathbf{F}: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$, two-point boundary condition function $\mathbf{G}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$, and their Jacobians $\mathbf{F}_y: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, $\mathbf{F}_\lambda: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times 1}$, $\mathbf{G}_{y(a)}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, $\mathbf{G}_{y(b)}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, and $\mathbf{G}_\lambda: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times 1}$. Initial point on the solution curve \mathcal{C} , $(\mathbf{y}_1, \lambda_1)$. Maximum number of points not including the initial point to be computed on \mathcal{C} , J . Initial tangent steplength, σ_{init} . Minimum and maximum tangent steplengths permitted, σ_{min} and σ_{max} . Tangent steplength reduction and increase scale factors, σ_r and σ_i . Maximum number of Newton correction steps permitted, K . Maximum number of Newton correction steps for which a tangent steplength increase may occur if convergence is obtained, k_{fast} . Newton correction convergence threshold, γ . Tangent direction at the first solution, d . d may be -2 , -1 , 1 , or 2 . If d is -1 or 1 , the first tangent is scaled by d . If d is -2 (2), the first tangent is scaled so that λ decreases (increases) in the first step. `polish` is a Boolean that determines whether the Newton corrector solution is polished by solving (C.37).

Output: A solution curve \mathbf{c} or a `flag` indicating that the curve could not be traced.

```

1: function PAC_BVP( $\mathbf{F}, \mathbf{G}, \mathbf{F}_y, \mathbf{F}_\lambda, \mathbf{G}_{y(a)}, \mathbf{G}_{y(b)}, \mathbf{G}_\lambda, (\mathbf{y}_1, \lambda_1), J, \sigma_{\text{init}}, \sigma_{\text{min}}, \sigma_{\text{max}}, \sigma_r, \sigma_i, K, k_{\text{fast}}, \gamma, d, \text{polish}$ )
2:    $\sigma \leftarrow \sigma_{\text{init}}$  ▷ Set the initial tangent steplength.
3:    $\mathbf{c}(1) \leftarrow (\mathbf{y}_1, \lambda_1)$  ▷ Store the initial solution on  $\mathcal{C}$ .
4:    $(\mathbf{v}_0, \tau_0) \leftarrow (\mathbf{0}, 1)$  ▷ Select an initial unit tangent. This choice forces  $\tau_1 = 1$ .
5:   for  $j = 1$  to  $J$  do ▷ Trace the solution curve  $\mathcal{C}$ .
6:     Obtain a tangent  $(\mathbf{v}_j, \tau_j)$  to  $\mathcal{C}$  at  $(\mathbf{y}_j, \lambda_j)$  by solving (C.14) starting from  $(\mathbf{v}_{j-1}, \tau_{j-1})$ .
7:      $\kappa \leftarrow \|(\mathbf{v}_j, \tau_j)\|$ 
8:     if  $j == 1$  then ▷ Choose the direction of the tangent at the initial solution, based on  $d$ .
9:       if  $(d == -2$  OR  $d == 2)$  AND  $\tau_1 < 0$  then
10:         $d \leftarrow -d$  ▷ Flip the sign of  $d$  to get the desired tangent direction.
11:      end if
12:       $\kappa \leftarrow \text{sgn}(d) \kappa$ 
13:    end if
14:     $(\mathbf{v}_j, \tau_j) \leftarrow \frac{1}{\kappa} (\mathbf{v}_j, \tau_j)$  ▷ Normalize the tangent.
15:    reject  $\leftarrow$  TRUE
16:    while reject do
17:       $(\mathbf{y}_1^c, \lambda_1^c) \leftarrow (\mathbf{y}_j, \lambda_j) + \sigma (\mathbf{v}_j, \tau_j)$  ▷ Take a tangent step of length  $\sigma$ .
18:      for  $k = 1$  to  $K$  do ▷ Newton correction counter.
19:        Obtain a Newton correction  $(\delta \mathbf{y}_k^c, \delta \lambda_k^c)$  to  $(\mathbf{y}_k^c, \lambda_k^c)$  by solving (C.26).
20:         $(\mathbf{y}_{k+1}^c, \lambda_{k+1}^c) \leftarrow (\mathbf{y}_k^c, \lambda_k^c) + (\delta \mathbf{y}_k^c, \delta \lambda_k^c)$  ▷ Construct the Newton corrector.
21:        if  $\frac{\|(\delta \mathbf{y}_k^c, \delta \lambda_k^c)\|}{\|(\mathbf{y}_1^c, \lambda_1^c)\|} < \gamma$  then ▷ Test for convergence to  $\mathcal{C}$ .
22:          reject  $\leftarrow$  FALSE
23:          if polish then
24:            Obtain the next solution  $(\mathbf{y}_{j+1}, \lambda_{j+1})$  on  $\mathcal{C}$  by solving (C.37) starting from  $(\mathbf{y}_{k+1}^c, \lambda_{k+1}^c)$ .
25:          else
26:             $(\mathbf{y}_{j+1}, \lambda_{j+1}) \leftarrow (\mathbf{y}_{k+1}^c, \lambda_{k+1}^c)$  ▷ Accept the Newton corrector solution.
27:          end if
28:           $\mathbf{c}(j+1) \leftarrow (\mathbf{y}_{j+1}, \lambda_{j+1})$  ▷ Store the new solution on  $\mathcal{C}$ .
29:          if  $k \leq k_{\text{fast}}$  then ▷ Test for rapid Newton convergence.
30:             $\sigma \leftarrow \min \{\sigma_i \sigma, \sigma_{\text{max}}\}$  ▷ Rapid Newton convergence, so increase the tangent steplength.
31:          end if
32:          break ▷ Break out of the for loop since convergence to  $\mathcal{C}$  has been achieved.
33:        end if
34:      end for

```

Algorithm C.1 Predictor-Corrector Continuation for Nonlinear ODE TPBVPs. Part 2.

```
35:         if reject then
36:              $\sigma \leftarrow \sigma_r \sigma$             $\triangleright$  Too many Newton steps taken, so reduce the tangent steplength.
37:             if  $\sigma < \sigma_{\min}$  then
38:                 print "Unable to trace  $\mathcal{C}$  because the tangent steplength is too small:  $\sigma < \sigma_{\min}$ ."
39:                 return flag
40:             end if
41:         end if
42:     end while
43: end for
44:     return  $c$ 
45: end function
```

Appendix D

Sweep Predictor-Corrector Continuation Method for Solving an ODE TPBVP

D.1 Introduction

In this section, an alternative predictor-corrector continuation method is presented that exploits a monotonic continuation ODE TPBVP solver, such as `bvptwp`'s `acdc` or `acdcc`, to monotonically increase (i.e. sweep) the tangent steplength σ from 0 up until a maximum threshold σ_{\max} is reached or until the next turning point is reached.

D.2 Construct the Tangent

Given a solution $(\mathbf{y}_j, \lambda_j)$ to (C.1), we seek to construct a unit tangent (\mathbf{v}_j, τ_j) to the solution curve \mathcal{C} at $(\mathbf{y}_j, \lambda_j)$. Recall the arclength constraint

$$\|(\mathbf{v}_j, \tau_j)\|^2 = \langle (\mathbf{v}_j, \tau_j), (\mathbf{v}_j, \tau_j) \rangle = \int_a^b \mathbf{v}_j^T(s) \mathbf{v}_j(s) ds + \tau_j^2 = 1. \quad (\text{D.1})$$

The linearization (i.e. Fréchet derivative) of the ODE TPBVP (C.1) about the solution $(\mathbf{y}_j, \lambda_j)$, in conjunction with the arclength constraint (D.1), gives the nonlinear ODE TPBVP:

$$\begin{aligned}
\frac{d}{ds} \mathbf{v}_j(s) &= \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}_j(s), \lambda_j) \mathbf{v}_j(s) \\
&\quad + \mathbf{F}_{\lambda}(s, \mathbf{y}_j(s), \lambda_j) \tau_j \\
\frac{d}{ds} \tau_j &= 0 \\
\frac{d}{ds} w(s) &= \mathbf{v}_j^T(s) \mathbf{v}_j(s) \\
\mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(a) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(b) \\
&\quad + \mathbf{G}_{\lambda}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \tau_j = \mathbf{0}_{n \times 1} \\
w(a) &= 0 \\
w(b) + \tau_j^2 - 1 &= 0,
\end{aligned} \tag{D.2}$$

which must be solved for $\mathbf{v}_j: [a, b] \rightarrow \mathbb{R}^n$, $\tau_j \in \mathbb{R}$, and $w: [a, b] \rightarrow \mathbb{R}$ and where (\mathbf{v}_j, τ_j) is a unit tangent to \mathcal{C} at $(\mathbf{y}_j, \lambda_j)$. Note that the first, second, and third equations in (D.2) are the ODEs, while the fourth, fifth, and sixth equations constitute the boundary conditions. The first, second, and fourth equations in (D.2) are the linearization (i.e. Fréchet derivative) of (C.1) about the solution $(\mathbf{y}_j, \lambda_j)$ and ensure that a tangent is produced, while the third, fifth, and sixth equations in (D.2) enforce the arclength constraint (D.1) ensuring that the tangent is of unit length. The initial solution guess to solve (D.2) is $(\mathbf{v}_j, \tau_j) = (\mathbf{0}, 1)$ and $w(s) = 0$, $s \in [a, b]$.

Note that the ODE TPBVP (D.2) can be solved numerically via the MATLAB routines `sbvp` or `bvptwp`, which offers 4 algorithms: `twpbvp_m`, `twpbvpc_m`, `twpbvp_l`, and `twpbvpc_l`. Since \mathbf{y}_j is usually only known for a discrete set of points in $[a, b]$, the values of this function at the other points in $[a, b]$ must be obtained through interpolation in order to numerically solve (D.2). The MATLAB routine `interp1` performs linear, cubic, `pchip`, and spline interpolation and may be utilized to interpolate \mathbf{y}_j while solving (D.2).

Because the numerical solvers usually converge faster when provided Jacobians of the ODE velocity function and of the two-point boundary condition function, these are computed below. Let

$$\mathbf{x} = \begin{bmatrix} \mathbf{v}_j \\ \tau_j \\ w \end{bmatrix}. \tag{D.3}$$

The ODE velocity function in (D.2) is

$$\mathbf{H}^t(s, \mathbf{x}(s)) = \mathbf{H}^t(s, \mathbf{v}_j(s), \tau_j, w(s)) = \begin{bmatrix} \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}_j(s), \lambda_j) \mathbf{v}_j(s) + \mathbf{F}_{\lambda}(s, \mathbf{y}_j(s), \lambda_j) \tau_j \\ 0 \\ \mathbf{v}_j^T(s) \mathbf{v}_j(s) \end{bmatrix}. \tag{D.4}$$

The Jacobian of the ODE velocity function \mathbf{H}^t with respect to \mathbf{x} is

$$\begin{aligned} \mathbf{H}_{\mathbf{x}}^t(s, \mathbf{x}(s)) &= \mathbf{H}_{\mathbf{x}}^t(s, \mathbf{v}_j(s), \tau_j, w(s)) \\ &= \begin{bmatrix} \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}_j(s), \lambda_j) & \mathbf{F}_{\lambda}(s, \mathbf{y}_j(s), \lambda_j) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ 2\mathbf{v}_j^T(s) & 0 & 0 \end{bmatrix}. \end{aligned} \quad (\text{D.5})$$

The two-point boundary condition in (D.2) is

$$\mathbf{K}^t(\mathbf{x}(a), \mathbf{x}(b)) = \mathbf{0}_{(n+2) \times 1}, \quad (\text{D.6})$$

where \mathbf{K}^t is the two-point boundary condition function

$$\mathbf{K}^t(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(a) + \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \mathbf{v}_j(b) + \mathbf{G}_{\lambda}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) \tau_j \\ w(a) \\ w(b) + \tau_j^2 - 1 \end{bmatrix}. \quad (\text{D.7})$$

The Jacobians of the two-point boundary condition function \mathbf{K}^t with respect to $\mathbf{x}(a)$ and $\mathbf{x}(b)$ are

$$\mathbf{K}_{\mathbf{x}(a)}^t(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{G}_{\lambda}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 1 \\ \mathbf{0}_{1 \times n} & 2\tau_j & 0 \end{bmatrix} \quad (\text{D.8})$$

and

$$\mathbf{K}_{\mathbf{x}(b)}^t(\mathbf{x}(a), \mathbf{x}(b)) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{G}_{\lambda}(\mathbf{y}_j(a), \mathbf{y}_j(b), \lambda_j) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{0}_{1 \times n} & 2\tau_j & 1 \end{bmatrix}. \quad (\text{D.9})$$

Special care must be taken when implementing the Jacobians (D.8) and (D.9). Since the unknown constant τ_j appears as the second to last element of both $\mathbf{x}(a)$ and $\mathbf{x}(b)$, τ_j from only one of $\mathbf{x}(a)$ and $\mathbf{x}(b)$ is actually used to construct each term in \mathbf{K}^t involving τ_j . The middle column of (D.8) is actually the derivative of \mathbf{K}^t with respect to the τ_j in $\mathbf{x}(a)$, while the middle column of (D.9) is actually the derivative of \mathbf{K}^t with respect to the τ_j in $\mathbf{x}(b)$. Thus, the middle columns in (D.8) and (D.9) corresponding to the derivative of \mathbf{K}^t with respect to τ_j should not coincide in a software implementation. For example, if \mathbf{K}^t is constructed from the τ_j in $\mathbf{x}(a)$, $\mathbf{K}_{\mathbf{x}(a)}^t$ is as shown in (D.8) while the middle column of (D.9) corresponding to the derivative of \mathbf{K}^t with respect to the τ_j in $\mathbf{x}(b)$ is all zeros. Alternatively, if \mathbf{K}^t is constructed from the τ_j in $\mathbf{x}(b)$, $\mathbf{K}_{\mathbf{x}(b)}^t$ is as shown in (D.9) while the middle column of (D.8) corresponding to the derivative of \mathbf{K}^t with respect to the τ_j appearing in $\mathbf{x}(a)$ is all zeros.

D.3 Determine the Tangent Direction

The unit tangent (\mathbf{v}_j, τ_j) at $(\mathbf{y}_j, \lambda_j)$ obtained by solving (D.2) must be scaled so that the sweep predictor-corrector continuation method does not reverse direction. As shown in [148], the correct direction for the

unit tangent is obtained via:

$$(\mathbf{v}_j, \tau_j) \leftarrow \text{sgn}(\kappa) (\mathbf{v}_j, \tau_j), \quad (\text{D.10})$$

where κ is the inner product of the previous and current unit tangents:

$$\kappa = \langle (\mathbf{v}_{j-1}, \tau_{j-1}), (\mathbf{v}_j, \tau_j) \rangle = \int_a^b \mathbf{v}_{j-1}^\top(s) \mathbf{v}_j(s) ds + \tau_{j-1} \tau_j. \quad (\text{D.11})$$

The integration operator to construct the inner product κ in (D.11) can be realized via the MATLAB routine `trapz`. With the sign direction selected by (D.10), the inner product of the previous and current unit tangents is positive:

$$\langle (\mathbf{v}_{j-1}, \tau_{j-1}), (\mathbf{v}_j, \tau_j) \rangle = \int_a^b \mathbf{v}_{j-1}^\top(s) \mathbf{v}_j(s) ds + \tau_{j-1} \tau_j > 0. \quad (\text{D.12})$$

D.4 Sweep along the Tangent

By monotonically increasing (or sweeping) the tangent steplength σ from 0, the current solution $(\mathbf{y}_j, \lambda_j)$ and its unit tangent (\mathbf{v}_j, τ_j) can be used to find the next solution $(\mathbf{y}_{j+1}, \lambda_{j+1})$ that solves (C.1) while satisfying the orthogonality constraint:

$$\begin{aligned} & \langle (\mathbf{v}_j, \tau_j), (\mathbf{y}_{j+1}, \lambda_{j+1}) - ((\mathbf{y}_j, \lambda_j) + \sigma (\mathbf{v}_j, \tau_j)) \rangle = \langle (\mathbf{v}_j, \tau_j), (\mathbf{y}_{j+1} - (\mathbf{y}_j + \sigma \mathbf{v}_j), \lambda_{j+1} - (\lambda_j + \sigma \tau_j)) \rangle \\ & = \int_a^b \mathbf{v}_j^\top(s) [\mathbf{y}_{j+1}(s) - (\mathbf{y}_j(s) + \sigma \mathbf{v}_j(s))] ds + \tau_j [\lambda_{j+1} - (\lambda_j + \sigma \tau_j)] = 0. \end{aligned} \quad (\text{D.13})$$

This yields the ODE TPBVP:

$$\begin{aligned} \frac{d}{ds} \mathbf{y}_{j+1}(s) &= \mathbf{F}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) \\ \frac{d}{ds} \lambda_{j+1} &= 0 \\ \frac{d}{ds} w(s) &= \mathbf{v}_j^\top(s) [\mathbf{y}_{j+1}(s) - (\mathbf{y}_j(s) + \sigma \mathbf{v}_j(s))] \\ \mathbf{G}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) &= \mathbf{0}_{n \times 1} \\ w(a) &= 0 \\ w(b) + \tau_j [\lambda_{j+1} - (\lambda_j + \sigma \tau_j)] &= 0, \end{aligned} \quad (\text{D.14})$$

which must be solved for $\mathbf{y}_{j+1}: [a, b] \rightarrow \mathbb{R}^n$, $\lambda_{j+1} \in \mathbb{R}$, and $w: [a, b] \rightarrow \mathbb{R}$ by monotonically increasing (or sweeping) σ . Note that the first, second, and third equations in (D.14) are the ODEs, while the fourth, fifth, and sixth equations constitute the boundary conditions. The first, second, and fourth equations in (D.14) ensure that the solution lies on \mathcal{C} (i.e. satisfies (C.1)), while the third, fifth, and sixth equations in (D.14) enforce the orthogonality constraint (D.13). The initial solution guess to solve (D.14) is the current solution $(\mathbf{y}_j, \lambda_j)$ and $w(s) = 0$, $s \in [a, b]$. σ starts at 0, since the initial solution guess for $(\mathbf{y}_{j+1}, \lambda_{j+1})$ is $(\mathbf{y}_j, \lambda_j)$, and increases monotonically until the maximum threshold σ_{\max} is reached or until the ODE TPBVP solver halts (due to reaching a turning point).

Note that the ODE TPBVP (D.14) can be solved numerically via the MATLAB routine `bvptwp`, which offers 2 continuation algorithms: `acdc` and `acdcc`. The continuation algorithms `acdc` and `acdcc` assume that the continuation parameter (in this case σ) is monotonically increasing or decreasing, so that they will halt at a turning point in the continuation parameter. Since \mathbf{y}_j and \mathbf{v}_j are usually only known for a discrete set of points in $[a, b]$, the values of these functions at the other points in $[a, b]$ must be obtained through interpolation in order to numerically solve (D.14). The MATLAB routine `interp1` performs linear, cubic, `pchip`, and spline interpolation and may be utilized to interpolate \mathbf{y}_j and \mathbf{v}_j while solving (D.14).

Because the numerical solvers usually converge faster when provided Jacobians of the ODE velocity function and of the two-point boundary condition function, these are computed below. Let

$$\mathbf{x} = \begin{bmatrix} \mathbf{y}_{j+1} \\ \lambda_{j+1} \\ w \end{bmatrix}. \quad (\text{D.15})$$

The ODE velocity function in (D.14) is

$$\mathbf{H}^q(s, \mathbf{x}(s), \sigma) = \mathbf{H}^q(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}, w(s), \sigma) = \begin{bmatrix} \mathbf{F}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) \\ 0 \\ \mathbf{v}_j^\top(s) [\mathbf{y}_{j+1}(s) - (\mathbf{y}_j(s) + \sigma \mathbf{v}_j(s))] \end{bmatrix}. \quad (\text{D.16})$$

The Jacobian of the ODE velocity function \mathbf{H}^q with respect to \mathbf{x} is

$$\begin{aligned} \mathbf{H}_{\mathbf{x}}^q(s, \mathbf{x}(s), \sigma) &= \mathbf{H}_{\mathbf{x}}^q(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}, w(s), \sigma) \\ &= \begin{bmatrix} \mathbf{F}_{\mathbf{y}}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) & \mathbf{F}_{\lambda}(s, \mathbf{y}_{j+1}(s), \lambda_{j+1}) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{v}_j^\top(s) & 0 & 0 \end{bmatrix}. \end{aligned} \quad (\text{D.17})$$

The two-point boundary condition in (D.14) is

$$\mathbf{K}^q(\mathbf{x}(a), \mathbf{x}(b), \sigma) = \mathbf{0}_{(n+2) \times 1}, \quad (\text{D.18})$$

where \mathbf{K}^q is the two-point boundary condition function

$$\mathbf{K}^q(\mathbf{x}(a), \mathbf{x}(b), \sigma) = \begin{bmatrix} \mathbf{G}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) \\ w(a) \\ w(b) + \tau_j [\lambda_{j+1} - (\lambda_j + \sigma \tau_j)] \end{bmatrix}. \quad (\text{D.19})$$

The Jacobians of the two-point boundary condition function \mathbf{K}^q with respect to $\mathbf{x}(a)$ and $\mathbf{x}(b)$ are

$$\mathbf{K}_{\mathbf{x}(a)}^q(\mathbf{x}(a), \mathbf{x}(b), \sigma) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(a)}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{G}_{\lambda}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 1 \\ \mathbf{0}_{1 \times n} & \tau_j & 0 \end{bmatrix} \quad (\text{D.20})$$

and

$$\mathbf{K}_{\mathbf{x}(b)}^q(\mathbf{x}(a), \mathbf{x}(b), \sigma) = \begin{bmatrix} \mathbf{G}_{\mathbf{y}(b)}(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{G}_\lambda(\mathbf{y}_{j+1}(a), \mathbf{y}_{j+1}(b), \lambda_{j+1}) & \mathbf{0}_{n \times 1} \\ \mathbf{0}_{1 \times n} & 0 & 0 \\ \mathbf{0}_{1 \times n} & \tau_j & 1 \end{bmatrix}. \quad (\text{D.21})$$

Special care must be taken when implementing the Jacobians (D.20) and (D.21). Since the unknown constant λ_{j+1} appears as the second to last element of both $\mathbf{x}(a)$ and $\mathbf{x}(b)$, λ_{j+1} from only one of $\mathbf{x}(a)$ and $\mathbf{x}(b)$ is actually used to construct each term in \mathbf{K}^q involving λ_{j+1} . The middle column of (D.20) is actually the derivative of \mathbf{K}^q with respect to the λ_{j+1} in $\mathbf{x}(a)$, while the middle column of (D.21) is actually the derivative of \mathbf{K}^q with respect to the λ_{j+1} in $\mathbf{x}(b)$. Thus, the middle columns in (D.20) and (D.21) corresponding to the derivative of \mathbf{K}^q with respect to λ_{j+1} should not coincide in a software implementation. For example, if \mathbf{K}^q is constructed from the λ_{j+1} in $\mathbf{x}(a)$, $\mathbf{K}_{\mathbf{x}(a)}^q$ is as shown in (D.20) while the middle column of (D.21) corresponding to the derivative of \mathbf{K}^q with respect to the λ_{j+1} in $\mathbf{x}(b)$ is all zeros. Alternatively, if \mathbf{K}^q is constructed from the λ_{j+1} in $\mathbf{x}(b)$, $\mathbf{K}_{\mathbf{x}(b)}^q$ is as shown in (D.21) while the middle column of (D.20) corresponding to the derivative of \mathbf{K}^q with respect to the λ_{j+1} appearing in $\mathbf{x}(a)$ is all zeros.

D.5 Pseudocode for Sweep Predictor-Corrector Continuation

Below is pseudocode that realizes the sweep predictor-corrector continuation method.

Algorithm D.2 Sweep Predictor-Corrector Continuation for Nonlinear ODE TPBVPs.

Input: ODE velocity function $\mathbf{F}: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$, two-point boundary condition function $\mathbf{G}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$, and their Jacobians $\mathbf{F}_y: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, $\mathbf{F}_\lambda: [a, b] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times 1}$, $\mathbf{G}_{y(a)}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, $\mathbf{G}_{y(b)}: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times n}$, and $\mathbf{G}_\lambda: \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^{n \times 1}$. Initial point on the solution curve \mathcal{C} , $(\mathbf{y}_1, \lambda_1)$. Maximum number of points not including the initial point to be computed on \mathcal{C} , J . σ_{\max} is a vector of length J such that $\sigma_{\max}(j)$ is the maximum tangent steplength permitted to obtain solution $j + 1$. Tangent direction at the first solution, d . d may be -2 , -1 , 1 , or 2 . If d is -1 or 1 , the first tangent is scaled by d . If d is -2 (2), the first tangent is scaled so that λ decreases (increases) in the first step.

Output: A solution curve \mathbf{c} .

```
1: function PAC_s3_BVP( $\mathbf{F}, \mathbf{G}, \mathbf{F}_y, \mathbf{F}_\lambda, \mathbf{G}_{y(a)}, \mathbf{G}_{y(b)}, \mathbf{G}_\lambda, (\mathbf{y}_1, \lambda_1), J, \sigma_{\max}, d$ )
2:    $\mathbf{c}(1) \leftarrow (\mathbf{y}_1, \lambda_1)$  ▷ Store the initial solution on  $\mathcal{C}$ .
3:   for  $j = 1$  to  $J$  do ▷ Trace the solution curve  $\mathcal{C}$ .
4:     Obtain a unit tangent  $(\mathbf{v}_j, \tau_j)$  to  $\mathcal{C}$  at  $(\mathbf{y}_j, \lambda_j)$  by solving (D.2) starting from  $(\mathbf{0}, 1)$ .
5:     if  $j == 1$  then ▷ Choose the direction of the tangent at the initial solution, based on  $d$ .
6:       if  $(d == -2$  OR  $d == 2)$  AND  $\tau_1 < 0$  then
7:          $d \leftarrow -d$  ▷ Flip the sign of  $d$  to get the desired tangent direction.
8:       end if
9:        $\kappa \leftarrow d$ 
10:      else
11:         $\kappa \leftarrow \langle (\mathbf{v}_{j-1}, \tau_{j-1}), (\mathbf{v}_j, \tau_j) \rangle$  ▷ Ensure that the traced solution does not reverse direction.
12:      end if
13:       $(\mathbf{v}_j, \tau_j) \leftarrow \text{sgn}(\kappa) (\mathbf{v}_j, \tau_j)$  ▷ Choose the correct tangent direction.
14:      Obtain the next solution  $(\mathbf{y}_{j+1}, \lambda_{j+1})$  on  $\mathcal{C}$  by solving (D.14) starting from  $(\mathbf{y}_j, \lambda_j)$  and monotonically increasing  $\sigma$  starting from 0 and without exceeding  $\sigma_{\max}(j)$ .
15:       $\mathbf{c}(j + 1) \leftarrow (\mathbf{y}_{j+1}, \lambda_{j+1})$  ▷ Store the new solution on  $\mathcal{C}$ .
16:    end for
17:    return  $\mathbf{c}$ 
18: end function
```

Appendix E

Quaternions

Quaternions were invented by William Rowan Hamilton in 1843. Good references on quaternions and how they are used to model rigid body dynamics are [16, 153, 154, 155]. The set of quaternions, which is isomorphic to \mathbb{R}^4 , is denoted by \mathbb{H} . A quaternion $\mathbf{p} \in \mathbb{H}$ can be expressed as the column vector

$$\mathbf{p} = \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 \end{bmatrix}^T = [p_0 ; p_1 ; p_2 ; p_3]. \quad (\text{E.1})$$

Given a column vector $\mathbf{v} \in \mathbb{R}^3$, \mathbf{v}^\sharp is the quaternion $[0 ; \mathbf{v}] \in \mathbb{H}$; that is,

$$\mathbf{v}^\sharp = \begin{bmatrix} 0 \\ \mathbf{v} \end{bmatrix} = [0 ; \mathbf{v}]. \quad (\text{E.2})$$

Given a quaternion $\mathbf{p} \in \mathbb{H}$, $\mathbf{p}^b \in \mathbb{R}^3$ is the column vector such that

$$\mathbf{p} = \begin{bmatrix} p_0 \\ \mathbf{p}^b \end{bmatrix} = [p_0 ; \mathbf{p}^b]. \quad (\text{E.3})$$

Given a column vector $\mathbf{v} \in \mathbb{R}^3$, note that

$$(\mathbf{v}^\sharp)^b = \mathbf{v}. \quad (\text{E.4})$$

However, given a quaternion $\mathbf{p} \in \mathbb{H}$,

$$(\mathbf{p}^b)^\sharp = \mathbf{p} \quad \text{iff} \quad \mathbf{p} = \begin{bmatrix} 0 \\ \mathbf{p}^b \end{bmatrix} = [0 ; \mathbf{p}^b]. \quad (\text{E.5})$$

Given quaternions $\mathbf{p} = [p_0 ; \mathbf{p}^b]$, $\mathbf{q} = [q_0 ; \mathbf{q}^b] \in \mathbb{H}$, their sum is

$$\mathbf{p} + \mathbf{q} = [p_0 ; \mathbf{p}^b] + [q_0 ; \mathbf{q}^b] = [p_0 + q_0 ; \mathbf{p}^b + \mathbf{q}^b], \quad (\text{E.6})$$

their product is

$$\mathbf{p}\mathbf{q} = [p_0; \mathbf{p}^b] [q_0; \mathbf{q}^b] = [p_0q_0 - \mathbf{p}^b \cdot \mathbf{q}^b; p_0\mathbf{q}^b + q_0\mathbf{p}^b + \mathbf{p}^b \times \mathbf{q}^b], \quad (\text{E.7})$$

and their dot product is

$$\mathbf{p} \cdot \mathbf{q} = [p_0; \mathbf{p}^b] \cdot [q_0; \mathbf{q}^b] = [p_0; p_1; p_2; p_3] \cdot [q_0; q_1; q_2; q_3] = p_0q_0 + \mathbf{p}^b \cdot \mathbf{q}^b = p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3. \quad (\text{E.8})$$

It may be shown that multiplication in \mathbb{H} is associative (i.e. $\mathbf{p}(\mathbf{q}\mathbf{r}) = (\mathbf{p}\mathbf{q})\mathbf{r} \quad \forall \mathbf{p}, \mathbf{q}, \mathbf{r} \in \mathbb{H}$) but not commutative (i.e. $\mathbf{p}\mathbf{q} \neq \mathbf{q}\mathbf{p}$ for general $\mathbf{p}, \mathbf{q} \in \mathbb{H}$). Given $c \in \mathbb{R}$ and a quaternion $\mathbf{p} = [p_0; \mathbf{p}^b] \in \mathbb{H}$, scalar multiplication of \mathbf{p} by c is

$$c\mathbf{p} = c [p_0; \mathbf{p}^b] = [cp_0; c\mathbf{p}^b]. \quad (\text{E.9})$$

Given a quaternion $\mathbf{p} = [p_0; \mathbf{p}^b] \in \mathbb{H}$, its conjugate is

$$\mathbf{p}^* = [p_0; -\mathbf{p}^b], \quad (\text{E.10})$$

its magnitude is

$$|\mathbf{p}| = (\mathbf{p} \cdot \mathbf{p})^{\frac{1}{2}} = (p_0^2 + \mathbf{p}^b \cdot \mathbf{p}^b)^{\frac{1}{2}}, \quad (\text{E.11})$$

and its inverse is

$$\mathbf{p}^{-1} = \frac{\mathbf{p}^*}{|\mathbf{p}|^2}. \quad (\text{E.12})$$

In the language of abstract algebra, \mathbb{H} is a four-dimensional associative normed division algebra over the real numbers.

$\mathcal{S} \subset \mathbb{H}$ denotes the set of unit quaternions, also called versors, which is isomorphic to $\mathbb{S}^3 \subset \mathbb{R}^4$. That is,

$$\mathcal{S} = \left\{ \mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \in \mathbb{R}^4 : |\mathbf{q}|^2 = \mathbf{q} \cdot \mathbf{q} = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \right\} \subset \mathbb{H}. \quad (\text{E.13})$$

The set of versors \mathcal{S} is useful because it may be utilized to parameterize the set of rotation matrices $SO(3)$.

Given a versor

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \in \mathcal{S}, \quad (\text{E.14})$$

the corresponding rotation matrix $\Lambda \in SO(3)$ is

$$\Lambda = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \in SO(3). \quad (\text{E.15})$$

It is easy to see from (E.15), that the versors

$$\mathfrak{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \in \mathcal{S} \quad \text{and} \quad -\mathfrak{q} = \begin{bmatrix} -q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix} \in \mathcal{S} \quad (\text{E.16})$$

correspond to the same rotation matrix $\Lambda \in SO(3)$, so that \mathcal{S} is a double covering of $SO(3)$. Given a vector $\mathbf{Y} \in \mathbb{R}^3$, the rotation of \mathbf{Y} by $\Lambda \in SO(3)$ can be realized using the versor $\mathfrak{q} \in \mathcal{S}$ via the Euler-Rodrigues formula

$$\Lambda \mathbf{Y} = [\mathfrak{q} \mathbf{Y}^\# \mathfrak{q}^{-1}]^b. \quad (\text{E.17})$$

Since $\mathfrak{q}^{-1} \in \mathcal{S}$ parameterizes $\Lambda^{-1} \in SO(3)$, (E.17) says that the rotation of $\mathbf{y} \in \mathbb{R}^3$ by $\Lambda^{-1} \in SO(3)$ can be realized using the versor $\mathfrak{q}^{-1} \in \mathcal{S}$ via

$$\Lambda^{-1} \mathbf{y} = [\mathfrak{q}^{-1} \mathbf{y}^\# \mathfrak{q}]^b. \quad (\text{E.18})$$

Now consider a rigid body, such as a free rigid body, a heavy top, Suslov's problem, a rolling disk, a rolling ball, etc., with orientation matrix $\Lambda \in SO(3)$ (i.e. Λ maps the body frame into the spatial frame) and body angular velocity

$$\boldsymbol{\Omega} \equiv [\Lambda^{-1} \dot{\Lambda}]^\vee = [\Lambda^\top \dot{\Lambda}]^\vee \in \mathbb{R}^3, \quad (\text{E.19})$$

so that

$$\dot{\Lambda} = \Lambda \widehat{\boldsymbol{\Omega}}. \quad (\text{E.20})$$

Let $\mathfrak{q} \in \mathcal{S}$ denote a versor corresponding to Λ . Then it may be shown that

$$\dot{\mathfrak{q}} = \frac{1}{2} \mathfrak{q} \boldsymbol{\Omega}^\# \mathfrak{q}. \quad (\text{E.21})$$