

Discrete Mechanics and Optimal Control for Space Trajectory Design

Thesis by
Ashley Moore

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy



California Institute of Technology
Pasadena, California

2011

(Defended May 17, 2011)

© 2011
Ashley Moore
All rights Reserved

Acknowledgements

I would like to begin by thanking my advisor, Dr. Jerry Marsden. It was an honor and a pleasure to work under the guidance of such an inspiring, brilliant, and kind man.

A very special thanks is due to Dr. Sina Ober-Blöbaum, my closest collaborator and mentor for the past four years. Thanks for the Skype chats, for editing papers, for debugging code, and for teaching me just about everything I know about DMOC and how to do research. I can never really express my gratitude.

Thank you also to my committee members, Dr. Richard Murray, Dr. Mathieu Desbrun, and Dr. Michael Ortiz. In particular, thanks to Mathieu for many discussions on time adaption, and to Richard for serving as my advisor this year.

Furthermore, much appreciation is due to Wendy McKay for advice on many things, but particularly for every time another computer crashed on me. I also want to acknowledge the students with whom I had the pleasure of working and studying: Evan Gawlik, Natasha Bosanac, Patrick Mullen, Shane Ross, Stefano Campagnola, Marin Kobilarov, Molei Tao, Henry Jacobs, Paul Skerritt, David Pekarek, Philip Du Toit, Ari Stern, and Katie Broersma.

The trajectory design groups at JPL provided much practical guidance; many thanks to Jon Sims, Greg Whiffen, Stas Petropoulos, Damon Landau, Try Lam, Stephen Broschart, Ed Wright, and Ted Sweetser. Also, thank you to Dr. Philip Gill at UCSD for providing me with a MATLAB-compatible copy of SNOPT.

Finally, I must thank my friends and family. To Danielle Richey and Roseanna Zia, thanks for all the homework help and for being the best friends I didn't expect to meet in engineering school. To Mr. Brett Crock at PHS, thanks for convincing me that I'm not bad at math, that I'm actually pretty good at it. To my wonderful parents Brown and Kimberly Moore, thanks for always being there, for instilling in me a love of learning, and for believing in me every day. And to Dr. Brett Williams, thanks for supporting me on every step of this journey and for showing me the world along the way; I've learned much more than I ever expected.

**Discrete Mechanics and Optimal Control for Space Trajectory
Design**
by
Ashley Moore

In Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Abstract

Space trajectory design is often achieved through a combination of dynamical systems theory and optimal control. The union of trajectory design techniques utilizing invariant manifolds of the planar circular restricted three-body problem and the optimal control scheme Discrete Mechanics and Optimal Control (DMOC) facilitates the design of low-energy trajectories in the N -body problem. In particular, DMOC is used to optimize a trajectory from the Earth to the Moon in the 4-body problem, removing the mid-course change in velocity, ΔV , usually necessary for such a trajectory while still exploiting the structure from the invariant manifolds.

This thesis also focuses on how to adapt DMOC, a method devised with a constant step size, for the highly nonlinear dynamics involved in trajectory design. Mesh refinement techniques that aim to reduce discretization errors in the solution and energy evolution and their effect on DMOC optimization are explored and compared with trajectories created using time adaptive variational integrators.

Furthermore, a time adaptive form of DMOC is developed that allows for a variable step size that is updated throughout the optimization process. Time adapted DMOC is based on a discretization of Hamilton's principle applied to the time adapted Lagrangian of the optimal control problem. Variations of the discrete action of the optimal control Lagrangian lead to discrete Euler-Lagrange equations

that can be enforced as constraints for a boundary value problem. This new form of DMOC leads to the accurate and efficient solution of optimal control problems with highly nonlinear dynamics. Time adapted DMOC is tested on several space trajectory problems including the elliptical orbit transfer in the 2-body problem and the reconfiguration of a cubesat.

Contents

1	Introduction	1
2	Background	4
2.1	Trajectory Design Using Invariant Manifolds	4
2.1.1	The 3-Body Problem	5
2.1.2	Invariant Manifolds	7
2.1.3	History of Trajectory Design Using Invariant Manifolds . .	10
2.2	Variational Integrators	13
2.2.1	Variational Integrators with Forcing	16
2.2.2	Implementation	17
2.2.3	Time Adaptive Variational Integrators	18
2.3	Optimal Control	22
2.3.1	DMOC	24
3	Low Energy Earth-to-Moon Transfer Using DMOC and Invariant Manifolds	29
3.1	Introduction	29
3.2	Method 1—Shoot the Moon	30
3.2.1	Controlled 4-Body Model	32
3.2.2	Shoot the Moon Initial Guess Trajectory	35
3.3	Method 2—Invariant Manifold Endpoints	36
3.4	Step Size Considerations for Creation of Initial Guess	37
3.5	Initial Guesses	39

3.6 Optimization	41
3.6.1 Constraints and Cost Function	41
3.6.2 Optimization Results	44
3.7 Low Thrust	47
3.7.1 Formulation of Low Thrust Initial Guess	47
3.7.2 Optimization for Low Thrust Trajectory	50
3.8 Analysis and Comparison	57
3.9 Conclusion	61
4 Mesh Refinement for DMOC	63
4.1 Introduction	63
4.2 Motivation and Problem Formulation	64
4.2.1 Elliptical Orbit Transfer	65
4.3 Traditional Mesh Refinement	65
4.3.1 Energy Considerations for Mesh Refinement	68
4.4 Mesh Refinement Strategies	69
4.5 Mesh Refinement for Elliptical Orbit Transfer	72
4.6 Mesh Refinement for Shoot the Moon	77
4.7 Conclusion	82
5 Time Adaptive DMOC	86
5.1 Introduction	86
5.2 Lagrangian Mechanics with Time Adaption	87
5.2.1 Continuous System with Time Adaption	88
5.2.2 Continuous System with Time Adaption and Forces	90
5.2.3 Correspondence Between Original System and Time Adapted System	91
5.2.4 Discrete System with Time Adaption	92
5.3 Naive Time Adaption for DMOC	98
5.4 Time Adaption for Optimal Control Problem	99

5.4.1	Transformation from Optimal Control Hamiltonian to La-	
	grangian	100
5.4.2	Transformation of Time Adapted Optimal Control Hamilto-	
	nian and Lagrangian	102
5.4.3	Time Adapted DMOC: Discrete Time Adapted Euler La-	
	grange Equations	103
5.4.4	Time Adaptive DMOC: an Indirect Method	105
5.4.5	Results for Simple Example	106
5.5	Examples	111
5.5.1	Elliptical Orbit Transfer	112
5.5.2	Cubesat Reconfiguration	115
5.6	Conclusion	117
6	Conclusions and Future Work	120

Chapter 1

Introduction

Space trajectory design is a complicated endeavor that often combines dynamical systems theory, optimization, and numerical techniques. The importance of energy efficiency for space missions has motivated the development of many techniques for the design of low-energy trajectories. Such trajectories exploit the natural dynamics of the solar system to travel from one region of space to another using less fuel. For example, invariant manifolds of the planar circular restricted 3-body problem provide structure for control-free transport that can be utilized for some problems including Earth-to-Moon transfer or trajectories traveling between Jovian moons or in the Saturnian system. Many researchers have focused on using these techniques to design interesting, low-energy trajectories. In most instances, the resulting trajectories require some form of optimization to either reduce the change in velocity, ΔV , or to reconcile the dynamics with more accurate solar system models. Furthermore, optimal control schemes, particularly local optimal control methods, require a good initial guess for successful optimization. Therefore, using a trajectory that takes advantage of the natural forces in the solar system as an initial guess for an optimal control technique has the potential to generate highly energy-efficient trajectories not easily produced using either technique individually.

This thesis seeks to combine trajectory design techniques utilizing invariant manifolds of the planar circular restricted 3-body problem with the optimal control scheme Discrete Mechanics and Optimal Control (DMOC). DMOC is theoretically

formulated for use with a constant step size; however, it is possible to use DMOC with a variable step size. This is absolutely necessary when employing DMOC for the optimization of space trajectories due to the highly nonlinear nature of the dynamics. Therefore, this work also examines how to best adapt DMOC for use with nonlinear problems, first through step size refinement and then by considering full time adaption.

In Chapter 3, two different methods utilizing invariant manifolds are used to design initial guess trajectories from the Earth to the Moon that are then optimized using DMOC. This problem is solved considering both impulsive and low thrust maneuvers. The first method replicates the work done by Koon, Lo, Marsden, and Ross [27, 28] to design a trajectory in the patched 3-body problem. Invariant manifolds of the Sun–Earth and Earth–Moon 3-body systems are connected to create a trajectory that travels from the Earth to ballistic capture at the Moon requiring an impulsive ΔV at the manifold intersection. Next, a new method is devised in which the structure of the manifolds is exploited directly in the 4-body problem, generating a trajectory that travels from the Earth to the Moon with shorter flight time. In addition, the combination of DMOC with low thrust is considered for a trajectory that employs low thrust propulsion to spiral into an elliptical orbit at the Moon. Overall, DMOC is shown to be very successful at optimizing these trajectories.

Work in Chapter 3 reveals the necessity of a procedure for automated, dynamics-driven, step size design. Therefore, Chapter 4 focuses on refinement of the time grid. First, classical mesh refinement as described by Betts [6] is used to develop a step size profile that reduces discretization errors in the solution. Since DMOC generates optimal solutions with good energy behavior, a new mesh refinement method is proposed that seeks to reduce errors in the energy evolution. Finally, step size profiles are generated using time adapted variational integrators as described by Kharevych [24]. Each mesh refinement scheme is used to generate initial guess trajectories that are then optimized using DMOC. The results are compared analyzing the convergence rates for both solution and energy error. The mesh re-

finement schemes are tested on both an elliptical orbit transfer and the low-energy, Earth-to-Moon trajectory studied in Chapter 3.

Chapter 5 focuses on the development of a time adaptive form of DMOC. The chapter begins with a thorough derivation of variational integrators with time adaption. Even though DMOC follows directly from the derivation of regular variational integrators, the same is not true with time adaption. Naively translating time adaptive variational integrators to time adaptive DMOC leads to incorrect optimization results, demonstrating that time adaption within the optimal control problem is more complicated. First, it is necessary to consider how to properly write the time adapted version of the optimal control Lagrangian. Then, discretization of Hamilton’s principle applied to the optimal control Lagrangian leads to a different version of discrete Euler-Lagrange equations that serve as constraints for optimization. The proposed time adapted DMOC is now an indirect optimization method while regular DMOC is a direct method. The new method is tested on the elliptical orbit transfer problem and the reconfiguration of a formation flying cubesat.

Chapter 2

Background

This thesis combines and builds upon several topics within dynamical systems theory including invariant manifolds and their use in trajectory design, variational integrators, and optimal control, specifically, discrete mechanics and optimal control (DMOC). Therefore, an introduction to the theoretical background of each topic is warranted.

2.1 Trajectory Design Using Invariant Manifolds

Many techniques focus on the design of spacecraft trajectories. Traditionally, most trajectory design techniques are based on the 2-body problem and patched conics. Therefore, solution arcs are built based on possible solutions including ellipses, parabolas, and hyperbolas. As explained by Bate, Mueller, and White [3], this Keplerian approach to design was used for the trajectory that sent Voyager around the solar system. Even though no analytical solution exists, the 3-body problem is well understood and allows for the design of complicated trajectories not possible using patched conics. In particular, invariant manifolds of the planar circular restricted 3-body problem (PCR3BP) can be used to locate energy efficient trajectories that follow the natural dynamics of the solar system from one region of space to another. For trajectories involving more than two primary bodies, and since the N -body problem is notoriously difficult to solve, much work has focused on patching multiple 3-body systems together. Such trajectories typically include

impulsive control at the intersection of the invariant manifolds of the two systems.

2.1.1 The 3-Body Problem

The planar circular restricted 3-body problem describes the motion of a body P under the gravitational influence of two primary masses, m_1 and m_2 . The system is described in a rotating coordinate frame, and the mass is normalized with the mass parameter,

$$\mu = \frac{m_2}{m_1 + m_2}, \quad (2.1)$$

where $m_1 > m_2$. The normalized mass of the larger body is denoted by $m_1 = 1 - \mu$, and the normalized mass of the smaller body is $m_2 = \mu$. The two primary bodies rotate in circular, planar orbits about their common center of mass at the origin. The third body P , for example a spacecraft, is assumed to have negligible mass. The primary bodies, m_1 and m_2 , are positioned at $(-\mu, 0)$ and $(1 - \mu, 0)$, respectively. The geometry of the PCR3BP is shown in Figure 2.1. The equations of motion for the PCR3BP are [43]

$$\ddot{x} - 2\dot{y} = \frac{\partial\Omega}{\partial x}, \quad (2.2a)$$

$$\ddot{y} + 2\dot{x} = \frac{\partial\Omega}{\partial y}, \quad (2.2b)$$

where

$$\Omega = \frac{x^2 + y^2}{2} + \frac{1 - \mu}{\sqrt{(x + \mu)^2 + y^2}} + \frac{\mu}{\sqrt{(x - 1 + \mu)^2 + y^2}} + \frac{\mu(1 - \mu)}{2}. \quad (2.3)$$

The system, equations (2.2)–(2.3), has five equilibrium points L_1, \dots, L_5 (see Figure 2.1), also known as Lagrange points.

The equations of motion for the PCR3BP are Hamiltonian and time independent, so there exists the following energy integral,

$$E = \frac{1}{2}(\dot{x}^2 + \dot{y}^2) - \Omega(x, y). \quad (2.4)$$

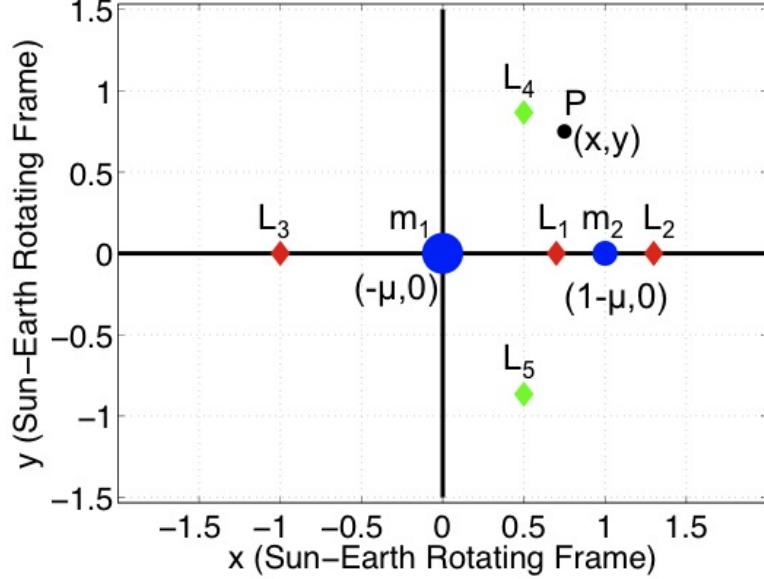


Figure 2.1: Geometry of PCR3BP in Sun-Earth rotating frame with two primary masses, m_1 and m_2 , and Lagrange points $\{L_i\}_{i=1}^5$.

Correspondingly, the Jacobi constant is given by $C = -2E$. The energy integral divides the phase space into regions of possible and forbidden motion. There are five possible cases, with the first four cases shown in Figure 2.2. Each plot shows the Hill's region, a projection of the energy surface $\mathcal{M}(\mu, e) = \{(x, y, \dot{x}, \dot{y}) | E(x, y, \dot{x}, \dot{y}) = e\}$ onto configuration space, for a particular energy level. The cases are distinguished by the critical energy $\{E_i\}_{i=1}^5$, which represents the energy of a particle at rest at the Lagrange point $\{L_i\}_{i=1}^5$.

If the energy of P is less than E_1 , then P is energetically trapped; it is impossible for P to reach either m_1 or m_2 from the exterior region, Figure 2.2(a). However, as the energy of P increases from $E < E_1$ to $E_1 < E < E_2$, a neck opens up at the L_1 Lagrange point so that motion between m_1 and m_2 is energetically possible, Figure 2.2(b). If $E_2 < E < E_3$, another neck opens at the L_2 Lagrange point making it energetically possible for P to travel to the exterior region as well, Figure 2.2(c). If $E_3 < E < E_4 = E_5$, more motion is possible, Figure 2.2(d). These openings near m_2 correspond to periodic orbits about L_1 and L_2 , and transport between m_1 , m_2 , and the exterior region must travel through these periodic orbits.

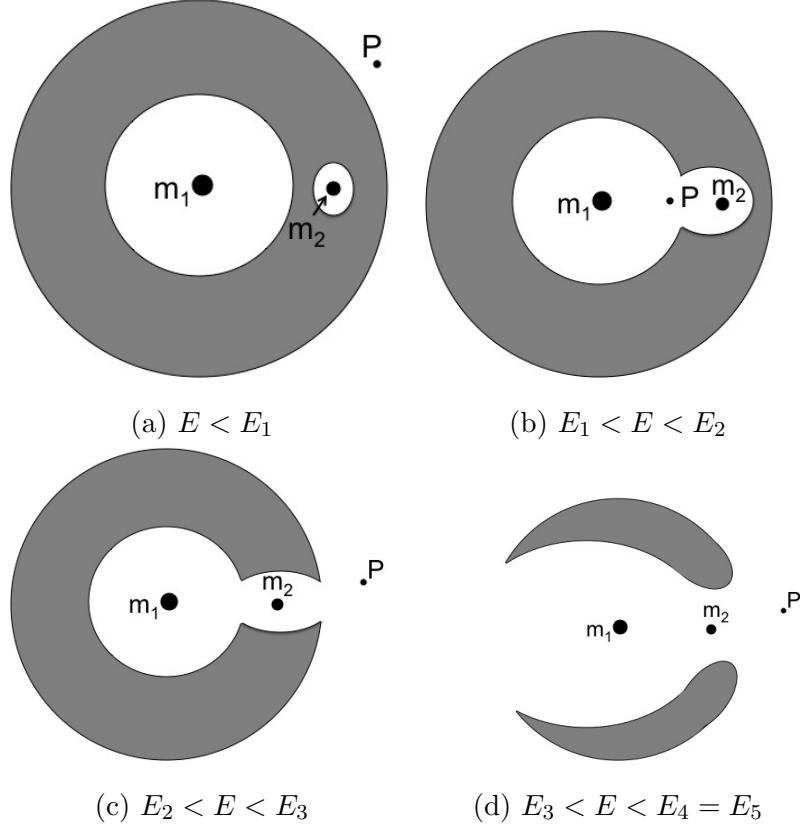


Figure 2.2: Regions of possible motion: (a) P cannot move between m_1 and m_2 , (b) P can move between m_1 and m_2 via L_1 , (c) P may move from m_1 to m_2 to exterior region via L_1 and L_2 , (d) P may travel past m_1 to exterior region via L_3 . Case 5, $E > E_5$, is not shown: P may move freely in x - y plane.

For a thorough treatment of the 3-body problem, see Szebehely [45], Ross [43], and Abraham and Marsden [1].

2.1.2 Invariant Manifolds

Invariant manifolds are tube-like structures comprised of trajectories originating on or leading to the periodic orbits of L_1 and L_2 . A particle, or spacecraft, may travel along or through an invariant manifold expending no energy. A method for computing invariant manifolds is described in Barden, Howell, and Lo [2] and Ross [43] and is briefly summarized here for manifolds of L_2 . First, the location of the L_2 Lagrange point and the initial condition for a periodic orbit are computed;

see [43] for details. Next, a family of periodic orbits is generated using differential correction and numerical continuation. The state transition matrix of a periodic orbit with period T , $\Phi(T, 0)$, is computed; this is also known as the monodromy matrix. From the monodromy matrix, the eigenvalues and associated eigenvectors are computed and used in the manifold initial condition near the first point on the periodic orbit, $x_{0,p.o.}$,

$$x_0^{u,\pm} = x_{0,p.o.} \pm \epsilon \frac{V_u}{\|V_u\|}, \quad (2.5)$$

$$x_0^{s,\pm} = x_{0,p.o.} \pm \epsilon \frac{V_s}{\|V_s\|}, \quad (2.6)$$

where ϵ is a small constant that moves the initial condition sufficiently far from the periodic orbit to avoid the asymptotic nature of the manifold while maintaining the linear estimate. For example, a normalized value of $\epsilon = 1e-4$ is used in this work for manifolds in the Earth–Moon 3-body system. Next, initial conditions can be generated corresponding to successive points on the periodic orbit, $x_{i,p.o.}$ with

$$x_i^{u,\pm} = x_{i,p.o.} \pm \epsilon \frac{\Phi(t_i, 0) \cdot V_u}{\|\Phi(t_i, 0) \cdot V_u\|}, \quad (2.7)$$

$$x_i^{s,\pm} = x_{i,p.o.} \pm \epsilon \frac{\Phi(t_i, 0) \cdot V_s}{\|\Phi(t_i, 0) \cdot V_s\|}. \quad (2.8)$$

Integrating each initial condition forwards (backwards) using the nonlinear equations of motion generates the trajectories that make up the unstable (stable) manifolds, $W_{L_2}^{u,\pm}$ and $W_{L_2}^{s,\pm}$, respectively.

The invariant manifolds for L_2 are shown in Figure 2.3. The plus and minus stable (unstable) manifolds are denoted by $W_{L_2}^{s,+}$ and $W_{L_2}^{s,-}$ ($W_{L_2}^{u,+}$ and $W_{L_2}^{u,-}$), respectively. Note that as time moves forward, a particle on the unstable manifold moves away from the periodic orbit while a point on the stable manifold travels toward the periodic orbit. The converse is true if time flows backwards.

Conley [11] and McGeehee [37] studied the orbit structures around L_1 and L_2 , classifying the trajectories as asymptotic orbits that are asymptotic to the periodic orbit, transit orbits that cross the equilibrium region around the Lagrange point

from one region to another, or non-transit orbits. They also considered how to use such orbits to travel from the Earth to the Moon.

A Poincaré section can be used to demonstrate the types of orbits identified by Conley and McGehee and the general behavior of trajectories near invariant manifolds. A Poincaré section is a plane transverse to the flow that gives the y and \dot{y} values for trajectories when they hit the x coordinate of the plane. The velocity in the x -direction, \dot{x} , may be computed using x , y , \dot{y} and the Jacobi constant of the trajectory. For example, Figure 2.4(a) shows the location of the Poincaré section at m_2 , $x = 1 - \mu$, for the invariant manifolds of L_2 , and Figure 2.4(b) displays the y and \dot{y} values for the unstable (-) and stable (+) manifolds on the Poincaré section.

Focusing on the unstable manifold, initial conditions $(x = 1 - \mu, \dot{x}, y, \dot{y})$ are selected inside the manifold, on the manifold, and outside the manifold, as shown in Figure 2.4(c). Integrating these initial conditions generates trajectories that flow towards the periodic orbit at L_2 . Figure 2.4(d) demonstrates the vastly different dynamical behavior of these trajectories. The trajectory with initial condition inside the manifold flows through the manifold tube and the periodic orbit toward the exterior region and is a transit orbit. The trajectory with initial condition on the manifold follows the manifold asymptotically to the periodic orbit. The initial condition just outside the invariant manifold generates a trajectory that hugs the manifold but then bounces off the periodic orbit, returning to the region surrounding m_2 ; this is a non-transit orbit.

Even though the initial conditions are very close to each other, they lead to dynamically different behavior, demonstrating that the invariant manifold acts as a separatrix. This idea is explored thoroughly in Gómez et al. [19]. It is this separatrix property that renders invariant manifolds useful in trajectory design. Near invariant manifolds, small changes in energy can lead to drastically different regions in space, providing energy-free transport via the manifold.

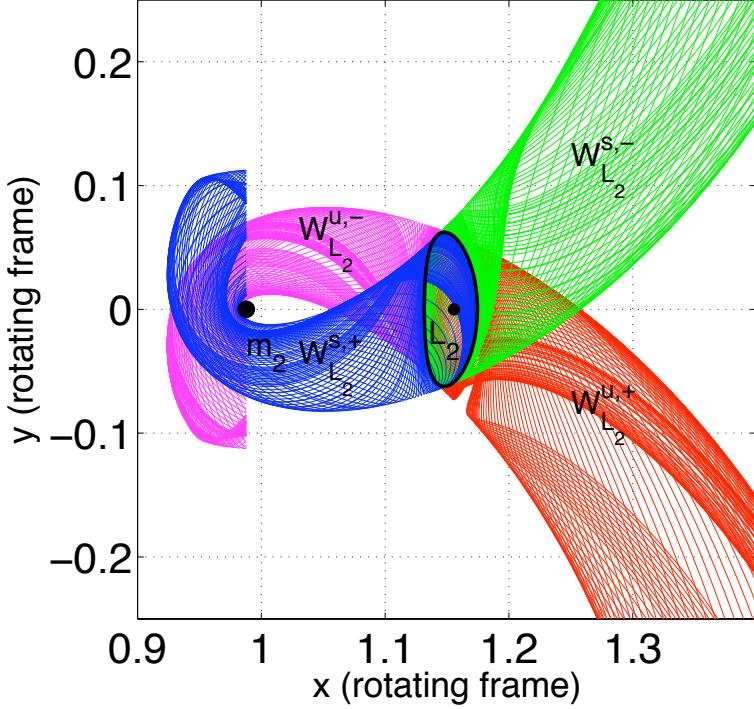


Figure 2.3: Stable and unstable manifolds emanate from the periodic orbit about the L_2 Lagrange point.

2.1.3 History of Trajectory Design Using Invariant Manifolds

Since the discovery of the transport mechanism provided by invariant manifolds, many authors have investigated different ways of using the invariant manifolds to facilitate the design of low-energy trajectories. Marsden and Ross [35] and Koon et al. [30] offer a great overview of the subject. More specifically, transfer from the Earth to periodic orbits around L_1 and L_2 is described by Howell, Barden, and Lo [2] and Gómez et al. [18]. The formal existence of heteroclinic connections between periodic orbits of the same energy is investigated by Koon et al. [26].

Invariant manifolds are particularly useful for the design of trajectories from the Earth to the Moon, in the Jovian moon system, and in the Saturian moon system. The design of a multi-moon orbiter in the Jupiter system is studied in Gómez et al. [19]. The combination of resonance and gravity assists are used for trajectories to Titan in [16] and [9], and in the Jupiter system, Ross and

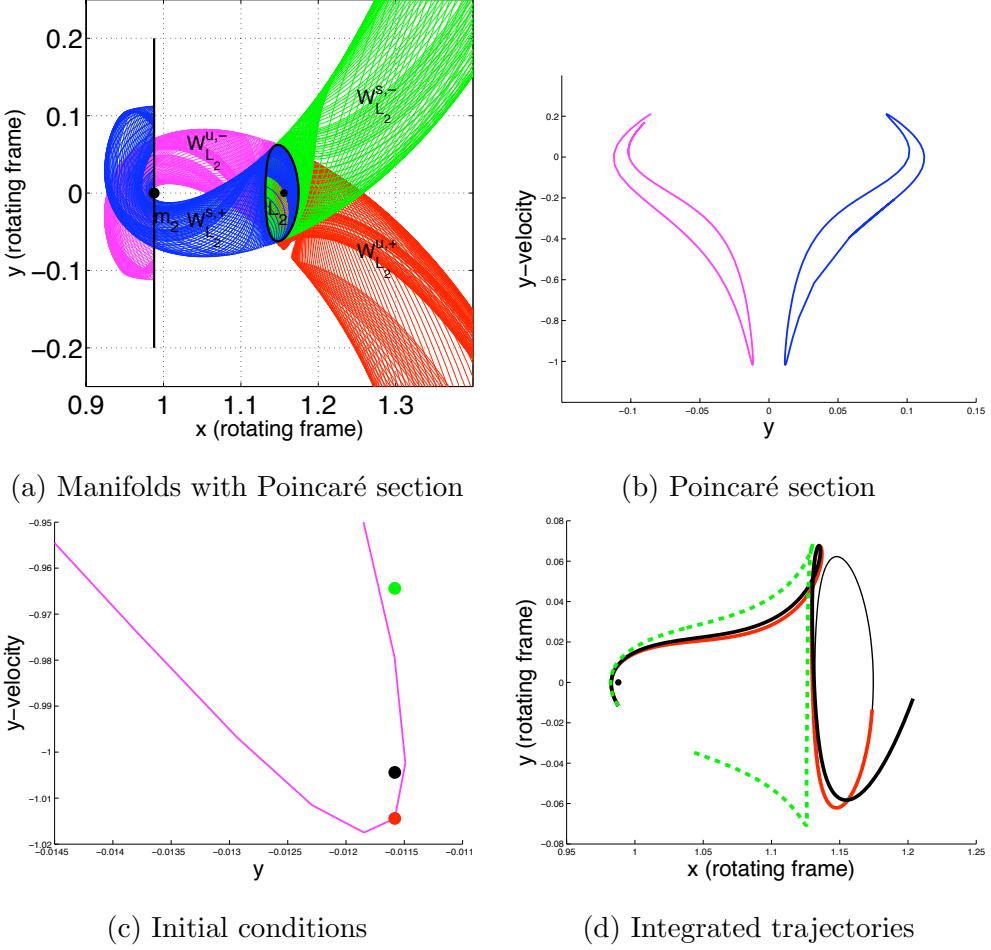


Figure 2.4: (a) A Poincaré section taken at $x = 1 - \mu$ on the manifolds of L_2 reveals (b) the y and \dot{y} coordinates of each manifold trajectory for $x = 1 - \mu$, and \dot{x} may be determined based on the energy. (c) Selecting initial conditions inside, on, or outside the manifold and (d) integrating reveals the vastly different trajectory behavior possible.

Scheeres [44]. Resonance and heteroclinic connections are shown to explain the motion of some comets, particularly those around Jupiter, presented by Koon et al. [29].

Belbruno and Miller [4] were the first to use the structure of the 3-body problem for a real mission. Taking advantage of the Weak Stability Boundary, an idea closely related to invariant manifolds, and by patching the Sun–Earth and Earth–Moon 3-body systems together, they designed a low-energy trajectory that sent the Japanese Hiten spacecraft to the Moon in 1991. Furthermore, the Genesis Dis-

covery Mission successfully flew on a trajectory following the invariant manifolds of the Sun–Earth system along a heteroclinic connection between L_1 and L_2 , as presented in Koon et al. [25]. Central to the research presented in Chapter 3 is the work of Koon, Lo, Marsden, and Ross [27, 28] on the Shoot the Moon problem, a low energy transfer from the Earth to the Moon facilitated by the patched 3-body problem combining the Sun–Earth and Earth–Moon systems. This problem was also studied by Howell and Kakoi [22] and Mingotti and Topputo [38]. Lo [32] proposes how invariant manifolds and their connections that make up the Interplanetary Superhighway may be used for future NASA missions as part of the Origins Program.

Invariant manifold techniques typically lead to trajectories that require some kind of control, whether it be impulsive or low thrust. For example, Mingotti et al. [39] explores the use of low thrust combined with invariant manifolds to reach an elliptical orbit at the moon. Lo et al. [33] further examines the role of invariant manifolds for low thrust trajectory design. Davis et al. [12] proposes an optimization technique for connecting periodic orbits about L_1 and L_2 of different energy. Marchand et al. [34] explores the use of optimal control for spacecraft formation keeping in orbits near L_1 and L_2 .

An extension of invariant manifold techniques in order to account for a continuously applied control force is presented in Dellnitz et al. [14] and employed to design a trajectory from Earth to Venus and from Earth to L_2 in [13]. However, techniques like this are only computationally reasonable for a constant one-dimensional control force. The research presented in Chapter 3 employs a time-dependent control law influencing all degrees of freedom of the spacecraft at each time node that are optimal with respect to a certain goal. Therefore, the application of a local optimal control scheme is indispensable for the design of trajectories with more complex control laws. Therefore, the thrust-less trajectories designed using invariant manifold techniques serve as initial guesses for the optimization of the controlled model. The emphasis in Chapters 3, 4, and 5 on the local optimal control scheme DMO_C warrants an introduction to DMO_C and optimal control.

DMOC is based on variational integrators, and since those ideas will be central to Chapter 5, a theoretical review of variational integrators is presented first.

2.2 Variational Integrators

Variational Integrators are symplectic, momentum-preserving integrators derived from variational mechanics. The full development and analysis of discrete mechanics and variational integrators is presented in Marsden and West [36]. Before discussing the derivation of variational integrators, it is useful to begin with some definitions. Consider a mechanical system with configuration manifold Q , associated state space TQ and Lagrangian $L : TQ \rightarrow \mathbb{R}$. Following the conventions of [36], given a time interval $[0, T]$, the *path space* is defined by

$$\mathcal{C}(Q) = \mathcal{C}([0, T], Q) = \{q : [0, T] \rightarrow Q | q \text{ is a } C^2 \text{ curve}\}, \quad (2.9)$$

and the *action map* $\mathfrak{G} : \mathcal{C}(Q) \rightarrow \mathbb{R}$ is

$$\mathfrak{G}(q) \equiv \int_0^T L(q(t), \dot{q}(t)) dt. \quad (2.10)$$

Hamilton's principle states that the evolution $q(t)$ of the system is a stationary point of the action. Therefore, variations of the action with fixed endpoints must be zero. For the Lagrangian system $L(q, \dot{q})$, this gives

$$\begin{aligned} \delta \int_0^T L(q(t), \dot{q}(t)) dt &= \int_0^T \left[\frac{\partial L}{\partial q} \cdot \delta q + \frac{\partial L}{\partial \dot{q}} \cdot \delta \dot{q} \right] dt \\ &= \int_0^T \left[\frac{\partial L}{\partial q} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \right] \cdot \delta q dt + \left. \frac{\partial L}{\partial \dot{q}} \delta q \right|_0^T \\ &= \int_0^T \left[\frac{\partial L}{\partial q} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \right] \cdot \delta q dt = 0, \end{aligned} \quad (2.11)$$

where integration by parts is used to reformulate the $\delta \dot{q}$ term and the boundary term disappears because $\delta q(T) = \delta q(0) = 0$. For this expression to be zero for all δq , then the integrand must be zero, resulting in the continuous Euler-Lagrange

equations

$$\frac{\partial L}{\partial q}(q, \dot{q}) - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}}(q, \dot{q}) \right) = 0. \quad (2.12)$$

The same derivation may be performed in the discrete framework using discrete variational mechanics. The state space TQ is replaced by $Q \times Q$ and the discretization grid is defined by $\Delta t = \{t_k = kh \mid k = 0, \dots, N\}$, $Nh = T$, where N is a positive integer and h is the step size. The path $q : [0, T] \rightarrow Q$ is replaced by a discrete path $q_d : \{t_k\}_{k=0}^N \rightarrow Q$, where $q_k = q_d(kh)$ is an approximation to $q(kh)$ [36, 40]. The continuous Lagrangian, $L(q, \dot{q})$, is replaced with a discrete Lagrangian, $L_d(q_k, q_{k+1}, h)$ using the midpoint rule

$$L_d(q_k, q_{k+1}, h) = hL \left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h} \right), \quad (2.13)$$

approximating the action integral along the curve between q_k and q_{k+1} . Thus it is possible to write

$$\int_0^T L(q, \dot{q}) \approx \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}, h) \quad (2.14)$$

where the integral has also been approximated using the midpoint rule. Note that it is possible to use more advanced quadrature rules to achieve integrators with a higher order of accuracy, but midpoint rule is exclusively used in this thesis.

Variations of the discrete action with respect to q_k gives

$$\begin{aligned} & \delta \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}, h) \\ &= \sum_{k=0}^{N-1} [D_1 L_d(q_k, q_{k+1}, h) \cdot \delta q_k + D_2 L_d(q_k, q_{k+1}, h) \cdot \delta q_{k+1}] \\ &= \sum_{k=0}^{N-1} [D_2 L_d(q_{k-1}, q_k, h) + D_1 L_d(q_k, q_{k+1}, h)] \cdot \delta q_k, \end{aligned}$$

where discrete integration by parts and the condition that $\delta q_0 = \delta q_N = 0$ is used to arrive at the final expression. Note that D_1 (D_2) denotes the derivative with respect to the first (second) argument. The discrete Euler-Lagrange equations are

obtained if the variations are required to vanish for all δq_k ,

$$D_2 L_d(q_{k-1}, q_k, h) + D_1 L_d(q_k, q_{k+1}, h) = 0. \quad (2.15)$$

The discrete Legendre transform, also called discrete fibre derivatives, gives the discrete version of the standard Legendre transform, $p = \frac{\partial L}{\partial \dot{q}}$,

$$\mathbb{F}^+ L_d : (q_0, q_1) \mapsto (q_1, p_1) = (q_1, D_2 L_d(q_0, q_1)), \quad (2.16)$$

$$\mathbb{F}^- L_d : (q_0, q_1) \mapsto (q_0, p_0) = (q_0, -D_1 L_d(q_0, q_1)). \quad (2.17)$$

The left and right momenta may now be defined as

$$\begin{aligned} p_{k,k+1}^+ &= p^+(q_k, q_{k+1}) = \mathbb{F}^+ L_d(q_k, q_{k+1}), \\ p_{k,k+1}^- &= p^-(q_k, q_{k+1}) = \mathbb{F}^- L_d(q_k, q_{k+1}). \end{aligned} \quad (2.18)$$

Recognizing that the Euler-Lagrange equations may be rewritten as

$$D_2 L_d(q_{k-1}, q_k) = -D_1 L_d(q_k, q_{k+1})$$

or

$$p_{k-1,k}^+ = p_{k,k+1}^-,$$

reveals that the Euler-Lagrange equations enforce momentum matching; that is, the momentum at a particular node k should be the same whether it is computed from above or below. Therefore, the momentum at each node k is given by

$$p_k = p_{k-1,k}^+ = p_{k,k+1}^-. \quad (2.19)$$

In addition to preserving the momentum, variational integrators display excellent energy behavior. In particular, symplecticity guarantees no energy dissipation or growth for constant time steps [36].

2.2.1 Variational Integrators with Forcing

For a Lagrangian system with external forces $f(q(t), \dot{q}(t), u(t))$, where $u(t) \in U$ is a control parameter, the motion $q(t)$ must satisfy the Lagrange-d'Alembert principle,

$$\delta \int_0^T L(q(t), \dot{q}(t)) dt + \int_0^T f(q(t), \dot{q}(t), u(t)) \cdot \delta q(t) dt = 0 \quad (2.20)$$

for all variations δq with $\delta q(0) = \delta q(T) = 0$. Integration by parts generates the forced Euler-Lagrange equations

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}}(q, \dot{q}) \right) - \frac{\partial L}{\partial q}(q, \dot{q}) = f(q, \dot{q}, u). \quad (2.21)$$

The path q is discretized as before, and the control path $u : [0, T] \rightarrow U$ is replaced by a discrete one. To this end, a refined grid, $\Delta\tilde{t}$, is generated via a set of control points $0 \leq c_1 < \dots < c_s \leq 1$ and $\Delta\tilde{t} = \{t_{k\ell} = t_k + c_\ell h \mid k = 0, \dots, N-1; \ell = 1, \dots, s\}$. With this notation, the discrete control path is defined to be $u_d : \Delta\tilde{t} \rightarrow U$. The intermediate control samples u_k on $[t_k, t_{k+1}]$ are defined as $u_k = (u_{k1}, \dots, u_{ks}) \in U^s$ to be the values of the control parameters guiding the system from $q_k = q_d(t_k)$ to $q_{k+1} = q_d(t_{k+1})$, where $u_{kl} = u_d(t_{kl})$ for $l \in \{1, \dots, s\}$. Then the continuous force $f(q, \dot{q}, u) : TQ \times U \rightarrow T^*Q$ is approximated by the discrete force $f_k(q_k, q_{k+1}, u_k)$ on the same time grid, $\Delta\tilde{t}$.

The continuous virtual work term in equation (2.20) is approximated by

$$f_k^- \cdot \delta q_k + f_k^+ \cdot \delta q_{k+1} \approx \int_{kh}^{(k+1)h} f(q(t), \dot{q}(t), u(t)) \cdot \delta q(t) dt, \quad (2.22)$$

where f_k^- , f_k^+ are the left and right discrete forces, respectively. The left and right discrete forces combine to represent the discrete force, f_k , such that

$$f_k(u_k)(q_k, q_{k+1}) \cdot (\delta q_k, \delta q_{k+1}) = f_k^+(u_k)(q_k, q_{k+1}) \cdot \delta q_{k+1} + f_k^-(u_k)(q_k, q_{k+1}) \cdot \delta q_k. \quad (2.23)$$

Note that f_{k-1}^+ may be viewed as the force acting on q_k during the time interval $[t_{k-1}, t_k]$, while f_k^- is the force on q_k applied during $[t_k, t_{k+1}]$. See [40] for more

details. Therefore, the discrete Lagrange-d'Alembert principle requires the discrete curve $\{q_k\}_{k=0}^N$ to satisfy

$$\delta \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}, h) + \sum_{k=0}^{N-1} [f_k^- \cdot \delta q_k + f_k^+ \cdot \delta q_{k+1}] = 0, \quad (2.24)$$

for all variations δq_k such that $\delta q_0 = \delta q_N = 0$. This is equivalent to the forced discrete Euler-Lagrange equations

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) + f_{k-1}^+ + f_k^- = 0. \quad (2.25)$$

The forced discrete Legendre transform,

$$\begin{aligned} \mathbb{F}^{f+} L_d : (q_{k-1}, q_k) &\mapsto (q_k, p_k) = (q_k, D_2 L_d(q_{k-1}, q_k) + f_{k-1}^+) \\ \mathbb{F}^{f-} L_d : (q_{k-1}, q_k) &\mapsto (q_{k-1}, p_{k-1}) = (q_{k-1}, -D_1 L_d(q_{k-1}, q_k) - f_{k-1}^-), \end{aligned} \quad (2.26)$$

provides the definition for the discrete momentum,

$$p_k = D_2 L_d(q_{k-1}, q_k) + f_{k-1}^+ \quad (2.27)$$

$$p_{k-1} = -D_1 L_d(q_{k-1}, q_k) - f_{k-1}^-. \quad (2.28)$$

Even with external forces, variational integrators preserve the energy rate better than non-symplectic integrators. Specifically, the Forced Noether's theorem relates the momentum evolution and applied forces, guaranteeing that the Lagrangian momentum map is preserved. See [36] for more details.

2.2.2 Implementation

Given an initial condition (q_0, p_0) , it is possible to compute q_1 from equation (2.28).

Next, the discrete Euler-Lagrange equations provide a recursive rule for computing $\{q_{k+1}\}_{k=1}^{N-1}$ based on (q_{k-1}, q_k) . The equations are most likely implicit and must be solved using an iterative solver such as Newton's method or Fsolve in MATLAB.

With knowledge of the $\{q_k\}_{k=0}^N$, the momenta $\{p_k\}_{k=1}^N$ can be computed using

equation (2.27).

2.2.3 Time Adaptive Variational Integrators

The variational integrators described above are valid for a constant step size, h . However, it is impractical to approach some systems using a constant step size. For example, the nonlinearity of the 3-body problem requires very small step size near bodies while coarser time stepping is sufficient elsewhere. Therefore, time adaption would be very useful in such a problem. However, if the step size is changed naively throughout the integration, the symplecticity can be destroyed. Therefore, care must be taken when including time adaption.

Hamiltonian Symplectic Integrators

Symplectic time adaptive integrators for Hamiltonian systems are proposed by Leimkuhler and Reich [31] and Hairer, Lubich, and Wanner [21] using a Sundman transformation,

$$\frac{dt}{d\tau} = \sigma(q, p), \quad (2.29)$$

where σ is a smooth function of position and momentum. Application of this transformation to a system with Hamiltonian $H(q, p)$ generates the equations of motion

$$\begin{aligned} q' &= \frac{dq}{d\tau} = \sigma(q, p) \nabla_p H(q, p) \\ p' &= \frac{dp}{d\tau} = -\sigma(q, p) \nabla_q H(q, p). \end{aligned} \quad (2.30)$$

In general, this system is no longer Hamiltonian. Therefore, the authors suggest a new Hamiltonian

$$\tilde{H}(q, p) = \sigma(q, p)(H(q, p) - H_0), \quad (2.31)$$

where H_0 is the energy and is constant along trajectories. The equations of motion for this system are given by

$$\begin{aligned} q' &= \sigma(q, p) \nabla_p H(q, p) + (H(q, p) - H_0) \nabla_p \sigma(q, p) \\ p' &= -\sigma(q, p) \nabla_q H(q, p) - (H(q, p) - H_0) \nabla_q \sigma(q, p). \end{aligned} \quad (2.32)$$

Since $H(q, p) - H_0 = 0$, this system reduces to the original system

$$\begin{aligned} q' &= \sigma(q, p) \nabla_p H(q, p) \rightarrow \dot{q} = \nabla_p H(q, p) \\ p' &= -\sigma(q, p) \nabla_q H(q, p) \rightarrow \dot{p} = -\nabla_q H(q, p). \end{aligned} \quad (2.33)$$

This idea will be very important for the derivation of DMOC with time adaption in Chapter 5. Integration of the transformed system using fixed time steps in τ is equivalent to using variable time steps in t .

Note that, in general, a Hamiltonian and Lagrangian are related by the equations

$$\begin{aligned} H &= \frac{\partial L}{\partial \dot{q}} \cdot \dot{q} - L \\ L &= \frac{\partial H}{\partial p} \cdot p - H, \end{aligned} \quad (2.34)$$

if they are hyper-regular. Therefore, it is possible to write the time adapted Lagrangian as

$$\begin{aligned} \tilde{L} &= \frac{\partial \tilde{H}}{\partial p} \cdot p - \tilde{H} = \frac{\partial \sigma(q, p)}{\partial p} (H - H_0) \cdot p + \sigma \left(\frac{\partial H}{\partial p} \cdot p - H + H_0 \right) \\ &= \sigma(L + H_0). \end{aligned} \quad (2.35)$$

This relationship will be useful in Chapter 5.

Time Adaption for Lagrangian Systems

In addition to a time-adaptive Hamiltonian formulation, it is desirable to develop the same ideas for a Lagrangian system. To this end, Kharevych [24] suggests adding a constraint to enforce the time step control directly into Hamilton's prin-

ciple. Consider the time adaption rule

$$t_{k+1} - t_k = h\sigma(q_k, q_{k+1}), \quad (2.36)$$

where t_k are the discrete time points in t , $h = \tau_{k+1} - \tau_k$ is the constant time step in τ , and τ_k are the discrete time nodes in τ . The discrete, constrained action may be written

$$\widehat{S}_0^N = \sum_{k=0}^{N-1} [L_d(q_k, q_{k+1}, t_{k+1} - t_k) + \lambda_k(t_{k+1} - t_k - h\sigma(q_k, q_{k+1}))], \quad (2.37)$$

where λ_k is a Lagrange multiplier that enforces the time constraint. Variations with respect to q_k , t_k , and λ_k give

$$\begin{aligned} \delta\widehat{S}_0^N = & \left[D_1 L_{k,k+1} + D_2 L_{k-1,k} - h\lambda_{k-1} \frac{\partial\sigma(q_{k-1}, q_k)}{\partial q_k} - h\lambda_k \frac{\partial\sigma(q_k, q_{k+1})}{\partial q_k} \right] \cdot \delta q_k \\ & + \left[\lambda_{k-1} - \lambda_k + E_{k+1} - E_k \right] \cdot \delta t_k + \left[t_{k+1} - t_k - h\sigma(q_k, q_{k+1}) \right] \cdot \delta\lambda_k, \end{aligned} \quad (2.38)$$

where $L_{k,k+1} = L_d(q_k, q_{k+1}, t_{k+1} - t_k)$, $L_{k-1,k} = L_d(q_{k-1}, q_k, t_k - t_{k-1})$, and E_{k+1} is the discrete energy given by

$$E_{k+1} = -D_3 L_d(q_k, q_{k+1}, t_{k+1} - t_k). \quad (2.39)$$

Since the discrete Hamilton's principle requires that $\delta\widehat{S}_0^N = 0$, the time adapted discrete Euler Lagrange equations are given by

$$D_1 L_{k,k+1} + D_2 L_{k-1,k} - h\lambda_{k-1} \frac{\partial\sigma(q_{k-1}, q_k)}{\partial q_k} - h\lambda_k \frac{\partial\sigma(q_k, q_{k+1})}{\partial q_k} = 0, \quad (2.40a)$$

$$\lambda_k = \lambda_{k-1} + E_{k+1} - E_k, \quad (2.40b)$$

$$t_{k+1} = t_k + h\sigma(q_k, q_{k+1}). \quad (2.40c)$$

Kharevych [24] claims that these new time adapted discrete Euler-Lagrange equations are only a slight modification of the regular, fixed time step equations,

and with λ_k sufficiently small, the integrator generates a discrete path with local flow near that of the original system while maintaining long time energy preservation. In particular, the discrete energy of the time adapted system,

$$\hat{E}_{k+1} - \hat{E}_1 = \lambda_k \sigma(q_k, q_{k+1}), \quad (2.41)$$

is preserved. This claim will be further explored in Chapter 5.

Lagrangian Systems with External and Dissipative Forces

A modification of the usual Lagrange-d'Alembert principle allows for the inclusion of external and dissipative forces in this time adaptive framework. The principle is now written

$$\delta \int_0^T L(q(t), \dot{q}(t)) dt + \int_0^T f(q(t), \dot{q}(t), u(t))(\delta q - \dot{q}\delta t) dt = 0, \quad (2.42)$$

where the term $-\dot{q}\delta t$ is necessary because variations with respect to time are also considered. Using this variational principle, the forced, time adaptive discrete Euler-Lagrange equations are

$$D_1 L_{k,k+1} + D_2 L_{k-1,k} - h\lambda_{k-1} \frac{\partial \sigma(q_{k-1}, q_k)}{\partial q_k} - h\lambda_k \frac{\partial \sigma(q_k, q_{k+1})}{\partial q_k} + f_{k-1}^+ + f_k^- = 0, \quad (2.43a)$$

$$\lambda_k = \lambda_{k-1} + E_{k+1} - E_k - f_{k-1}^+ \left(\frac{q_k - q_{k-1}}{h_{k-1}} \right) - f_k^- \left(\frac{q_{k+1} - q_k}{h_k} \right), \quad (2.43b)$$

$$t_{k+1} = t_k + h\sigma(q_k, q_{k+1}). \quad (2.43c)$$

where $f_k^+ = f_k^- = \frac{h_k}{2} f_k$, $h_k = t_{k+1} - t_k$, and $h_{k-1} = t_k - t_{k-1}$. Integration of the regular time adapted system, equation (2.40), or the forced time adapted system, equation (2.43), requires q_0, q_1, t_0, t_1 , and $\lambda_0 = 0$ to start. The implementation works as for a regular variational integrator with q_k , t_k , and λ_k computed simultaneously at each step.

2.3 Optimal Control

The basic ideas behind optimal control are necessary for an understanding of DMOc and particularly the development in Chapter 5. Ober-Blöbaum [40] provides a nice introduction and is summarized here. The goal of optimal control is to modify the dynamics of a system such that some quantity, for example the control effort, is minimized. More precisely, the objective functional is to be minimized subject to the system dynamics, initial conditions, and final constraints. Therefore, the optimal control problem, as used in this work, is defined as

$$\min_{x(\cdot), u(\cdot), (T)} J(x, u) = \int_0^T C(x(t), u(t)) dt + \Phi(x(T)), \quad (2.44a)$$

$$\dot{x}(t) = f(x(t), u(t)), \quad (2.44b)$$

$$x(0) = x_0, \quad (2.44c)$$

$$0 = r(x(T)), \quad (2.44d)$$

where J is the objective functional, C is the cost function, $\Phi(x(T))$ is the Mayer term and is considered zero for this work, $\dot{x} = f(x(t), u(t))$ is the system of differential equations describing the dynamics, x_0 is a vector defining the initial condition, and $r(x(T))$ defines the final point constraint. Also note that the controls, $u(t)$, are constrained to the pointwise control constraint set $U = \{u(t) \in \mathbb{R}^{n_u} | h(u(t)) \geq 0\}$, and the final time T is held fixed.

The solution trajectory $\eta(t) = (x(\cdot), u(\cdot))$ is a feasible solution if the constraints, equations (2.44b)–(2.44d), are fulfilled. The solution trajectory $\eta(t) = (x^*, u^*)$ is an optimal solution of the optimal control problem if

$$J(x^*, u^*) \leq J(x, u) \quad (2.45)$$

for all feasible pairs (x, u) . The solution $\eta(t) = (x^*, u^*)$ is a locally optimal solution if there exists a neighborhood $B_\delta(x^*, u^*)$, $\delta > 0$ for which equation (2.45) is true for all feasible $(x, u) \in B_\delta(x^*, u^*)$. For such a solution, $x^*(t)$ is a locally optimal

trajectory, and $u^*(t)$ is the locally optimal control.

Definition The *Hamiltonian* of the optimal control problem is given by the function $\mathcal{H} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ and is defined by

$$\mathcal{H}(x, u, \lambda) = -C(x, u) + \lambda^T \cdot f(x, u), \quad (2.46)$$

where $\lambda_i, i = 1, \dots, n_x$ are the adjoint variables, and n_x and n_u are the dimensions of the state, x , and control, u , respectively.

Definition The *Lagrangian* of the optimal control problem, equation (2.44), is a function $\mathcal{L} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ given by

$$\mathcal{L}(\eta, \lambda) = C(x(t), u(t)) + \lambda^T(t) \cdot [\dot{x} - f(x(t), u(t))]. \quad (2.47)$$

The *action* of the optimal control Lagrangian is given by

$$\mathfrak{G}(\eta, \lambda) = \int_0^T (C(x(t), u(t)) + \lambda^T(t) \cdot [\dot{x} - f(x(t), u(t))]) dt. \quad (2.48)$$

The point $(\eta^*(t), \lambda^*(t))$ is a saddle point of the action if

$$(\eta(t), \lambda^*(t)) \leq \mathcal{L}(\eta^*(t), \lambda^*(t)) \leq \mathcal{L}(\eta^*(t), \lambda(t)) \quad \forall (\eta(t), \lambda(t)). \quad (2.49)$$

Local solutions of the optimal control problem, equation (2.44), are saddle points of the action of the Lagrangian \mathcal{L} . Therefore, setting variations of the action of \mathcal{L} with respect to η and λ to zero results in the Euler-Lagrange equations, which serve as necessary optimality conditions for the optimal control problem. This result is given by the Pontryagin Maximum Principle.

Theorem 2.3.1 (Pontryagin Maximum Principle) *Let (x^*, u^*) be an optimal solution of the optimal control problem, equation (2.44). Then, there exists a piecewise continuous differentiable function $\lambda : [0, T] \rightarrow \mathbb{R}^{n_x}$ and a vector $\alpha \in \mathbb{R}^{n_r}$*

such that

$$\mathcal{H}(x^*(t), u^*(t), \lambda(t)) = \max_{u(t) \in U} \mathcal{H}(x(t), u(t), \lambda(t)) \quad \forall t \in [0, T], \quad (2.50a)$$

$$\dot{x}^*(t) = \nabla_x \mathcal{H}(x^*(t), u^*(t), \lambda(t)), \quad x^*(0) = x_0, \quad (2.50b)$$

$$\dot{\lambda}(t) = -\nabla_x \mathcal{H}(x^*(t), u^*(t), \lambda(t)), \quad (2.50c)$$

$$\lambda(T) = \nabla_x (\Phi(x^*(T)) - \nabla_x r(x^*(T))\alpha). \quad (2.50d)$$

A proof of this theorem can be found in Pontryagin et al. [42]. Note that the proof is not based on the calculus of variations. Deriving the necessary optimality conditions via calculus of variations on the optimal control Lagrangian can be more intuitive and is valid only if the solution and controls are smooth enough.

There are many different approaches used for the numerical solution of optimal control problems. Most methods can be classified as either an indirect method or a direct method. Indirect methods are derived directly from the Pontryagin maximum principle and involve an explicit expression of the necessary conditions for optimality. For direct methods, the problem is transformed into a finite dimensional nonlinear programming problem. Some examples of indirect methods include *gradient methods*, *multiple shooting*, and *collocation*, while *direct shooting*, *direct multiple shooting*, and *direction collection* are examples of direct methods. Betts [5] and Binder et al. [8] provide good overviews of the algorithms used for different numerical optimization methods. DMOC (Discrete Mechanics and Optimal Control) can also be classified as a direct method.

2.3.1 DMOC

DMOC is an optimal control scheme closely related to variational integrators that was developed by Junge, Marsden, and Ober-Blöbaum [23, 40, 41]. It is based on a direct discretization of the Lagrange-d'Alembert principle of the mechanical system. The resulting forced discrete Euler-Lagrange equations are used as optimization constraints for a given cost function. The resulting restricted opti-

mization problem is solved with an SQP solver.

Consider a mechanical system to be moved along a curve $q(t) \in Q$ during the time interval $t \in [0, T]$ from an initial state (q^0, \dot{q}^0) to a final state (q^T, \dot{q}^T) under the influence of a force $f(q(t), \dot{q}(t), u(t))$. The curves q and u are chosen to minimize a given objective functional,

$$J(q, \dot{q}, u) = \int_0^T C(q(t), \dot{q}(t), f(q(t), \dot{q}(t), u(t))) dt, \quad (2.51)$$

such that the system satisfies the Lagrange-d'Alembert principle,

$$\delta \int_0^T L(q(t), \dot{q}(t)) dt + \int_0^T f(q(t), \dot{q}(t), u(t)) \cdot \delta q(t) dt = 0, \quad (2.52)$$

for all variations δq with $\delta q(0) = \delta q(T) = 0$.

The optimal control problem stated in equation (2.51) and equation (2.52) is transformed into a finite dimensional constrained optimization problem using a global discretization of the states and the controls, as described for variational integrators. Recall from §2.2, the discrete Lagrange-d'Alembert principle, equation (2.24), emerges using an approximation of the action integral in equation (2.52) by a discrete Lagrangian $L_d : Q \times Q \rightarrow \mathbb{R}$,

$$L_d(q_k, q_{k+1}) \approx \int_{kh}^{(k+1)h} L(q(t), \dot{q}(t)) dt,$$

and discrete forces

$$f_k^- \cdot \delta q_k + f_k^+ \cdot \delta q_{k+1} \approx \int_{kh}^{(k+1)h} f(q(t), \dot{q}(t), u(t)) \cdot \delta q(t) dt, \quad (2.53)$$

where the left and right discrete forces f_k^\pm now depend on (q_k, q_{k+1}, u_k) . Then the discrete Lagrange-d'Alembert principle requires that,

$$\delta \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}) + \sum_{k=0}^{N-1} (f_k^- \cdot \delta q_k + f_k^+ \cdot \delta q_{k+1}) = 0, \quad (2.54)$$

for all variations $\{\delta q_k\}_{k=0}^N$ with $\delta q_0 = \delta q_N = 0$.

The discrete cost function, C_d , approximates the continuous cost function, C , in a similar manner such that

$$C_d(q_k, q_{k+1}, f_k, f_{k+1}) \approx \int_{kh}^{(k+1)h} C(q, \dot{q}, f). \quad (2.55)$$

Therefore, the discrete objective functional is given by

$$J_d(q_d, f_d) = \sum_{k=0}^{N-1} C_d(q_k, q_{k+1}, f_k, f_{k+1}). \quad (2.56)$$

For the optimal control problem, it is also necessary to consider the boundary conditions. First, the discrete initial and final positions are required to match the continuous ones,

$$q_0 = q(0),$$

$$q_N = q(T).$$

The momentum boundary conditions require more care. The initial and final momentum of the continuous system is computed via the Legendre transform, $p = \frac{\partial L}{\partial \dot{q}}$,

$$p(0) = D_2 L(q_0, \dot{q}_0),$$

$$p(T) = D_2 L(q_N, \dot{q}_N).$$

Then requiring that $p(0) = p_0$ and $p(T) = p_N$, where p_0 and p_N are computed using the forced discrete Legendre transform, equations (2.27)–(2.28), generates the momentum boundary conditions,

$$\begin{aligned} D_2 L(q_0, \dot{q}_0) + D_1 L_d(q_0, q_1) + f_0^- &= 0, \\ -D_2 L(q_N, \dot{q}_N) + D_2 L_d(q_{N-1}, q_N) + f_{N-1}^+ &= 0. \end{aligned} \quad (2.57)$$

In summary, the discrete constrained optimization problem is given by

$$\min_{q_d, u_d} J_d(q_d, u_d) = \sum_{k=0}^{N-1} C_d(q_k, q_{k+1}, u_k), \quad (2.58a)$$

$$q_0 = q^0, \quad (2.58b)$$

$$q_N = q^T, \quad (2.58c)$$

$$D_2 L(q^0, \dot{q}^0) + D_1 L_d(q_0, q_1) + f_0^- = 0, \quad (2.58d)$$

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) + f_{k-1}^+ + f_k^- = 0, \quad (2.58e)$$

$$-D_2 L(q^T, \dot{q}^T) + D_2 L_d(q_{N-1}, q_N) + f_{N-1}^+ = 0, \quad (2.58f)$$

with $k = 1, \dots, N - 1$.

Balancing accuracy and efficiency, the discrete cost function, C_d , the discrete Lagrangian, L_d , and the discrete forces are approximated with the midpoint rule, and constant control parameters are assumed on each time interval with $l = 1$ and $c_1 = \frac{1}{2}$,

$$C_d(q_k, q_{k+1}, u_k) = hC\left(\frac{q_{k+1} + q_k}{2}, \frac{q_{k+1} - q_k}{h}, u_k\right), \quad (2.59)$$

$$L_d(q_k, q_{k+1}) = hL\left(\frac{q_{k+1} + q_k}{2}, \frac{q_{k+1} - q_k}{h}\right), \quad (2.60)$$

$$f_k^- = f_k^+ = \frac{h}{2} f\left(\frac{q_{k+1} + q_k}{2}, \frac{q_{k+1} - q_k}{h}, u_k\right). \quad (2.61)$$

The order of approximation of the discrete Lagrangian, equation (2.60), and the discrete forces, equation (2.61), determines the order of convergence of the optimal control scheme. Therefore, second-order convergence is expected with this form of DMOC.

Equation (2.58) describes a nonlinear optimization problem with equality constraints, which can be solved by standard optimization methods like SQP, such as SNOPT [17]. Optionally, inequality constraints on states and controls can be included. In contrast to other direct optimal control methods, DMOC is based on the discretization of the variational principle, equation (2.52), rather than a discretization of the ordinary differential equations. In Ober-Blöbaum, Junge, and

Marsden [41], a detailed analysis of DMOC resulting from this discrete variational approach is given. The optimization scheme is symplectic-momentum consistent, i.e., the symplectic structure and the momentum maps corresponding to symmetry groups are consistent with the control forces for the discrete solution independent of the step size h . Thus, the use of DMOC leads to a reasonable approximation to the continuous solution, also for large step sizes, i.e., a small number of discretization points. Also, the discrete solution inherits structural properties from the continuous system, e.g., good energy preservation or correct energy drift in the presence of external forces [36].

Chapter 3

Low Energy Earth-to-Moon Transfer Using DMOC and Invariant Manifolds

3.1 Introduction

As mentioned in Chapter 2, invariant manifolds of the planar circular restricted 3-body problem can be used to locate energy efficient trajectories that follow the natural dynamics of the solar system from one region of space to another. This chapter aims to extend the patched 3-body problem ideas for the design a trajectory using 4-body dynamics with local optimal control applied throughout the trajectory. The combination of invariant manifold techniques in the PCR3BP and the optimal control algorithm DMOC (Discrete Mechanics and Optimal Control) facilitates the design of low energy trajectories in the 4-body problem.

Since the 1950s countless missions have targeted the Moon, sending spacecraft along trajectories for fly-bys, lunar observation orbits, and both manned and unmanned lunar landings. More recently, propulsion technology and design techniques, including the use of invariant manifolds, have facilitated the design of creative, fuel efficient trajectories. For example, in addition to the Hiten mission mentioned in §2.1.3, ESA’s SMART-1, described by Camino et al. [10], was launched in 2003 to demonstrate the potential use of ion propulsion for future interplanetary and deep space missions. The sustained thrust provided by the ion thruster allowed the spacecraft to spiral out from an elliptical orbit around the

Earth to the Moon and then spiral in for lunar capture.

This chapter presents two methods for the design of locally optimal trajectories from the Earth to the Moon with initial guess trajectories based on the invariant manifolds of the PCR3BP. The first method, method 1, replicates a trajectory similar to the Shoot the Moon trajectory, following the procedure presented by Koon, Lo, Marsden, and Ross [27, 28]. Their trajectory begins in low Earth orbit, travels along the invariant manifolds of the Sun-Earth and Earth-Moon PCR3BPs, and ends in ballistic capture at the Moon, using a total change in velocity, ΔV , of approximately 3,245 m/s (3,211 m/s to leave Earth orbit and 34 m/s applied mid-course). Mingotti, Topputto, and Bernelli-Zazzera [39] describe a low thrust version of this trajectory.

The second method, method 2, explores a different way of exploiting the manifold structure directly in the 4-body problem. It focuses on the stable Sun-Earth manifold and the unstable Earth-Moon manifold and generates a trajectory directly in the 4-body problem. The necessary invariant manifolds are shown in Figure 3.1. Both methods create trajectories that are used as initial guesses for DMOC, which searches for a locally optimal trajectory in the 4-body system, applying control throughout the trajectory while minimizing the control effort, or the total ΔV .

3.2 Method 1—Shoot the Moon

To achieve transfer between the Earth and Moon using invariant manifolds, as presented in [27, 28], the first step is to locate a suitable intersection of the unstable Sun-Earth manifold with the stable Earth-Moon manifold. A Poincaré section is used to find this intersection in the Sun-Earth rotating frame. For the transformation between the Sun-Earth and Earth-Moon rotating frames, see [43]. As shown in Figure 3.2, the phase of the Earth-Moon frame with respect to the Sun-Earth frame can be adjusted to identify such an intersection; $\phi = 100$ degrees is used here.

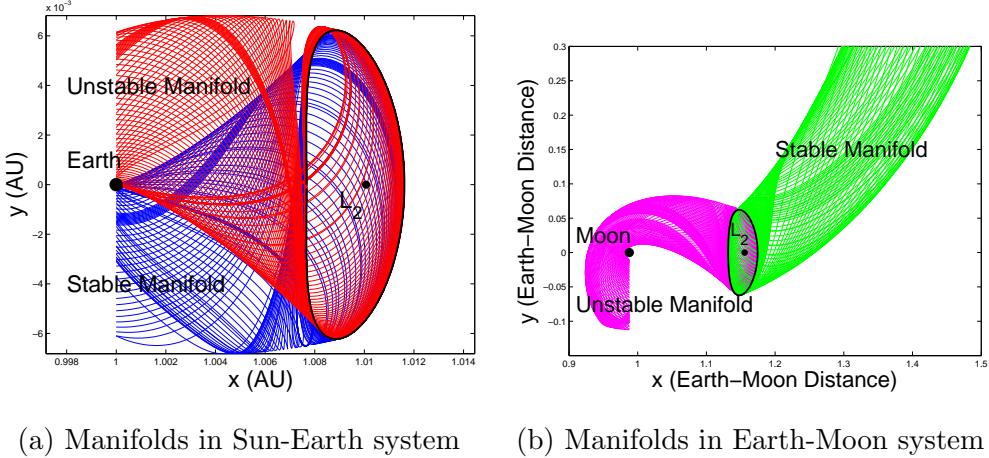


Figure 3.1: Manifolds emanate from the periodic orbit about L_2 . (a) Stable and unstable manifolds of the Sun-Earth L_2 Lagrange point. (b) Stable and unstable manifolds of Earth-Moon L_2 Lagrange point.

Using the Poincaré section, shown in Figure 3.3(b), a patch point is selected that falls within the stable manifold of the Earth-Moon system and outside the unstable manifold of the Sun-Earth system. From the Poincaré section, the patch point includes x , y , and \dot{y} . The x -velocity, \dot{x} , is selected to insure that the energy integral at the patch point equals that of the desired manifold. Forward integration of the conditions at the patch point (x, y, \dot{x}, \dot{y}) leads to a trajectory that flows along the stable Earth-Moon manifold and ends in ballistic capture at the Moon. The same initial conditions are modified slightly in \dot{x} and \dot{y} and integrated backwards, generating a trajectory that hugs the unstable Sun-Earth manifold and then twists, targeting back to the Earth. The modification in the velocity ensures that the energy of the spacecraft is at the appropriate level to travel along the Sun-Earth manifold in the desired manner.

The Sun-Earth and Earth-Moon trajectories are patched together to form a trajectory which begins at the Earth and ends in ballistic capture at the Moon. Ballistic capture here is defined to mean that the trajectory orbits the Moon at least once within its sphere of influence. Note that at the patch point, the energy is discontinuous; therefore, an impulsive ΔV is necessary to jump from the energy

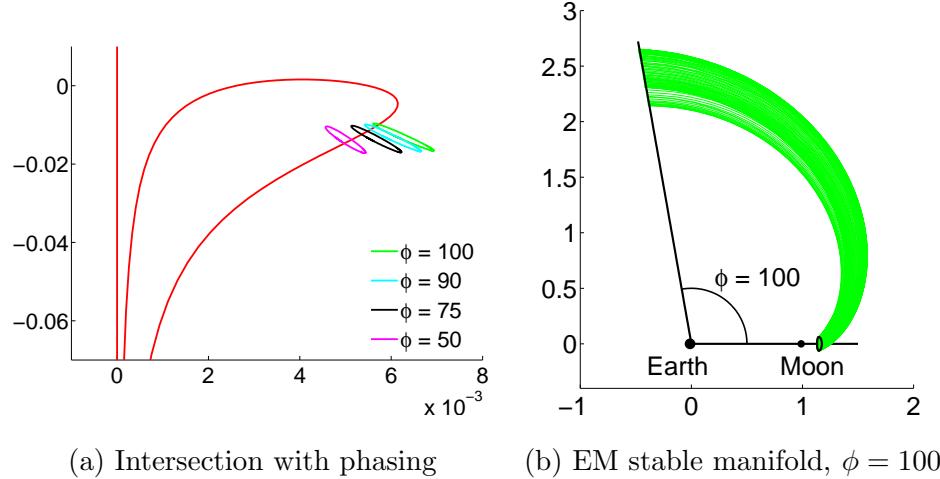


Figure 3.2: The phase, ϕ , of the Earth-Moon x -axis with respect to the Sun-Earth x -axis is varied until a suitable intersection of the Sun-Earth unstable manifold and Earth-Moon stable manifold is found.

level of the Sun-Earth manifold to the energy level of the Earth-Moon manifold. For mathematical details about this process, see [27, 43].

The patched trajectory is shown in Figure 3.4; it begins in a 135 km altitude circular orbit about the Earth and ends in a 11,785 km circular orbit about the Moon. An initial thrust of 3,227.8 m/s is required to escape Earth orbit along the trajectory, a mid-course ΔV of 60.6 m/s is applied at the patch point, and a final ΔV of 197.8 m/s is required to settle into a permanent circular orbit at the Moon. This trajectory is only valid for the patched 3-body problem; therefore, it is necessary to modify it to fulfill the dynamics of the 4-body problem.

3.2.1 Controlled 4-Body Model

The 4-body model used here is modeled in the Sun-Earth rotating frame, similar to the PCR3BP, with the Moon as a perturbation. Using the coordinates of the PCR3BP, the x -coordinates of the trajectory vary between $(0.995, 1.01)$, while the y -coordinates vary between $(-0.006, 0.006)$. The difference in scale between these numbers can cause problems for the optimization, so the convergence improves when both x and y variables hover around the origin [6]. Therefore, a simple

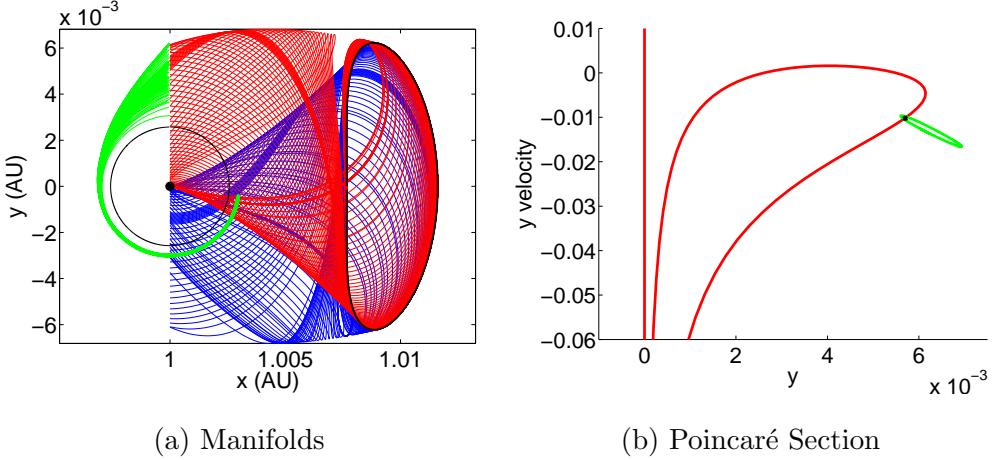


Figure 3.3: (a) Intersecting Earth-Moon and Sun-Earth manifolds in Sun-Earth rotating coordinates. (b) Poincaré section showing the intersection of the stable Earth-Moon manifold with the unstable Sun-Earth manifold. The patch point is chosen inside the stable Earth-Moon manifold and outside the unstable Sun-Earth manifold.

change of coordinates shifts the Earth so that it is centered at the origin. In general, the shifted problem converges faster and with smaller constraint residuals than the original problem.

This 4-body model describes the dynamics of the Sun, Earth, Moon, and spacecraft such that the Moon rotates around the Earth in planar circular motion. Then, the Earth and Sun rotate in planar circular motion about the center of mass of all three bodies. As before, the mass of the spacecraft is negligible. Figure 3.5 shows the geometry of this 4-body model. The controlled equations of motion for this model in Sun-Earth rotating coordinates are [43]

$$\ddot{x} - 2\dot{y} = \frac{\partial \Omega}{\partial x} + u_x, \quad (3.1a)$$

$$\ddot{y} + 2\dot{x} = \frac{\partial \Omega}{\partial y} + u_y, \quad (3.1b)$$

where

$$\Omega = \frac{x^2 + y^2}{2} + \frac{m_S}{\sqrt{(x+1)^2 + y^2}} + \frac{m_E}{\sqrt{x^2 + y^2}} + \frac{m_M}{\sqrt{(x - x_M)^2 + (y - y_M)^2}}, \quad (3.1c)$$

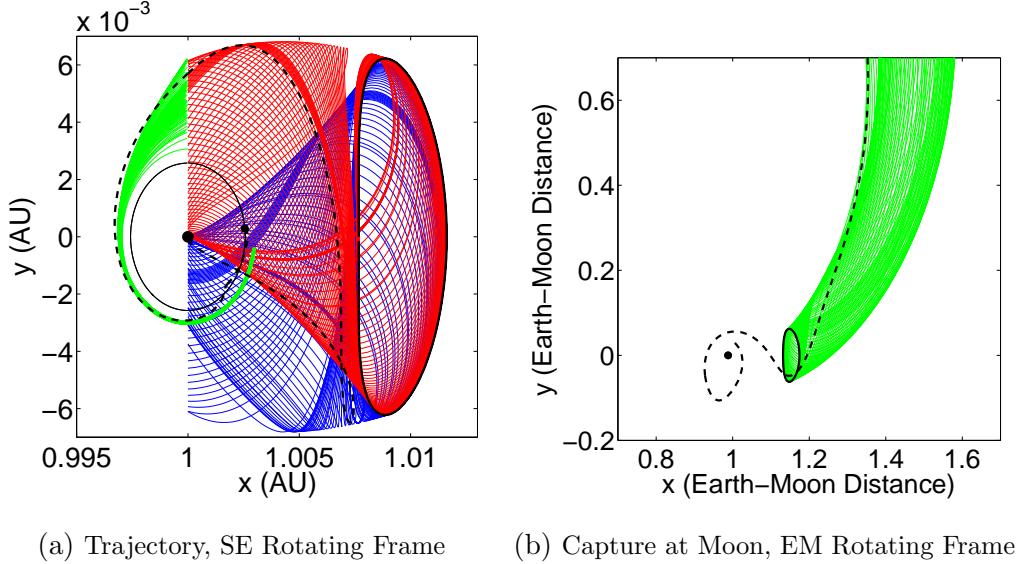


Figure 3.4: (a) Trajectory in 3-body problem (Sun-Earth rotating coordinates) begins near the Earth, hugs the Sun-Earth unstable manifold towards the periodic orbit of L_2 . It twists and then intersects the stable manifold of the Earth-Moon system, following that manifold to the realm of the Moon. (b) Ballistic capture at the Moon.

and m_S , m_E , and m_M are the normalized mass of the Sun, Earth, and Moon, respectively, given by

$$m_S = 1 - \mu, \quad (3.2a)$$

$$m_E = \mu, \quad (3.2b)$$

$$m_M = \frac{M_M}{M_M + M_E + M_S} = 3.734 \cdot 10^{-8}, \quad (3.2c)$$

and

$$\mu = \frac{M_E + M_M}{M_E + M_M + M_S} = 3.036 \cdot 10^{-6}. \quad (3.3)$$

Note that M_i , $i = E, M, S$, denotes the mass in kg. The variables u_x and u_y represent the control forces normalized by the negligible mass of the spacecraft in the x and y directions, respectively. Also, x_M and y_M represent the x - and

y -positions of the Moon as a function of time given by

$$\theta_M = \omega_M t + \theta_{M0}, \quad (3.4a)$$

$$x_M = a_M \cos \theta_M, \quad (3.4b)$$

$$y_M = a_M \sin \theta_M, \quad (3.4c)$$

where t is time, θ_{M0} is the initial angle of the Moon with respect to the x -axis in the Sun-Earth rotating frame, $a_M = 2.573 \cdot 10^{-3}$ is the normalized radius of the Moon's circular orbit, and $\omega_M = 12.369$ is the normalized rotation rate of the Moon.

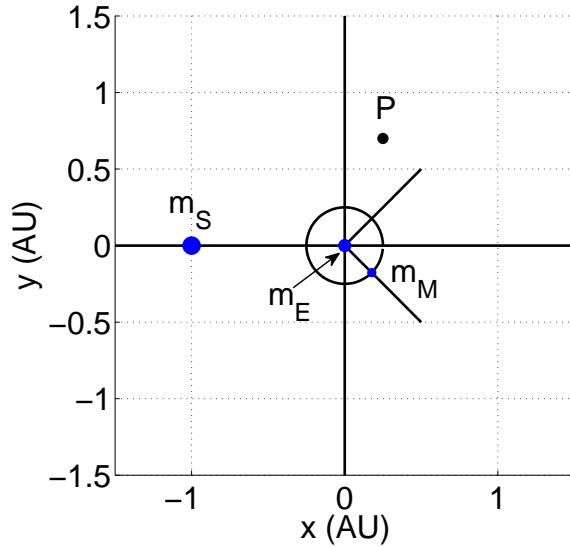


Figure 3.5: 4-body model: geometry in the Sun-Earth rotating frame with three primary masses, m_S , m_E , and m_M , and spacecraft, P . The Moon rotates relative to the Sun-Earth rotating frame, which is stationary.

3.2.2 Shoot the Moon Initial Guess Trajectory

Beginning with the same initial conditions from the patch point, \dot{x} and \dot{y} are modified slightly and integrated using the 4-body model described above. The modification is necessary due to the differences between the dynamics of the PCR3BP and the 4-body problem. The patch point is modified differently for the Sun-Earth

section and the Earth-Moon section because of the energy differences between the manifolds of the two systems. Thus, the initial conditions denoted by IC_{SE} and IC_{EM} , respectively, can be expressed as

$$IC_{SE} = \begin{bmatrix} x & y & \dot{x} + \Delta\dot{x}_{SE} & \dot{y} + \Delta\dot{y}_{SE} \end{bmatrix}, \quad (3.5)$$

$$IC_{EM} = \begin{bmatrix} x & y & \dot{x} + \Delta\dot{x}_{EM} & \dot{y} + \Delta\dot{y}_{EM} \end{bmatrix}. \quad (3.6)$$

IC_{SE} is integrated backwards to generate the Sun-Earth portion of the trajectory, and IC_{EM} is integrated forwards to generate the Earth-Moon portion of the trajectory. Note that the Δ s are adjusted until a good trajectory is found; that is, a trajectory which begins and ends at a desired distance from the Earth and Moon, respectively. Note that the initial and final momentum values may not be favorable. DMOC adjusts these momentum values according to the specified constraints and cost function during optimization. This trajectory serves as the initial guess for DMOC.

3.3 Method 2—Invariant Manifold Endpoints

To design a trajectory valid for the 4-body problem, the process starts with the unstable Earth-Moon manifold and stable Sun-Earth manifold. A point, IC_M , is selected on the unstable Earth-Moon manifold a desired distance from the Moon, and when integrated backwards in the 4-body problem (transformed to Sun-Earth rotating coordinates), generates $Traj_M$ which flows from the Moon towards, and then through, the Earth-Moon L_2 periodic orbit. Another point, IC_E , is selected on the stable Sun-Earth manifold some distance from the Earth. IC_E is integrated forwards along the manifold, generating $Traj_E$. In the 3-body problem, this trajectory would end on the periodic orbit; however, due to 4-body perturbations, the trajectory bounces off the periodic orbit and flows back towards the Moon's orbit. Figure 3.6(a) shows $Traj_M$ in Earth-Moon rotating coordinates, and Figure 3.6(b) shows $Traj_E$ in Sun-Earth rotating coordinates.

Next, the intersection of the resulting trajectories in Sun-Earth rotating coor-

dinates is located; this is the patch point, shown in Figure 3.6(c). The conditions of Traj_M at the patch point are integrated forwards in the 4-body problem to create Traj_{M2} (identical to Traj_M , but it flows in the opposite direction, towards the Moon instead of away from it). Note that even though the trajectories intersect in x - y space, they do not actually intersect in time or velocity. The discontinuity in velocity requires an impulsive ΔV . The lack of intersection in time means that the position of the Moon, given by equations (3.4), is different for Traj_M and Traj_E and requires more consideration. Consequently, the position of the Moon at the patch point for Traj_M is selected as the initial condition of the Moon for Traj_E . Next, the conditions of Traj_E at the patch point are modified slightly and integrated backwards toward the Earth to give Traj_{E2} , which is similar to Traj_E and ends in the desired location. Figure 3.6(d) shows Traj_{M2} and Traj_{E2} joined by an impulsive ΔV at the patch point. This trajectory serves as the initial guess for optimization. In Figure 3.6, the manifolds are labeled such that EM_U (EM_S) represents the Earth-Moon unstable (stable) manifold, and SE_U (SE_S) represents the Sun-Earth unstable (stable) manifold.

3.4 Step Size Considerations for Creation of Initial Guess

When creating the initial guess using the two methods described above, the non-linearity of the dynamics poses a problem. Ideally, a constant step size would be used throughout the trajectory, but this leads to two possible and undesirable scenarios. First, if a medium step size is used, e.g., $\mathcal{O}(10^{-2})$, there are not enough nodes near the Earth and Moon to accurately capture the dynamics and general accuracy and numerical problems arise. On the other hand, if a sufficiently small step size is used, e.g., $\mathcal{O}(10^{-5})$, there are too many nodes leading to unreasonable computation time and computer memory problems. To solve this problem, the trajectory is broken into sections of different step size. For example, five sections are used for the initial guess found with method 1. Selecting the step size and

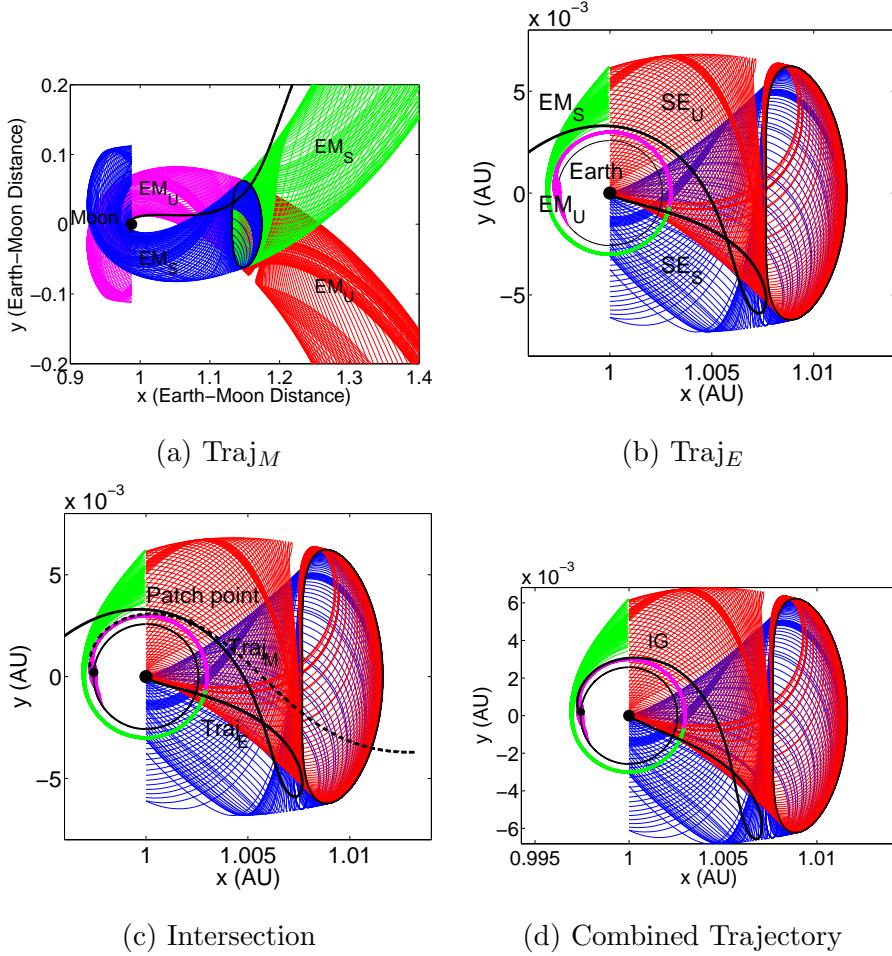


Figure 3.6: Process for method 2: (a) Integrate point on Earth-Moon unstable manifold backwards in 4-body problem. (b) Integrate point on Sun-Earth stable manifold forwards in 4-body problem. (c) Locate intersection of the two trajectories; this is the new patch point. (d) Integrate initial conditions at the patch point with consistent Moon position to generate initial guess with impulsive ΔV at the patch point.

number of nodes is an iterative, manual process that is repeated until a sufficiently accurate trajectory is produced. This variation in step size is accommodated in DMOC by supplying h as a vector containing the step size at each node.

3.5 Initial Guesses

The initial guess trajectories are plotted in Figure 3.7. Two initial guess trajectories are created using each method: IG1 and IG2 are created using method 1, while IG3 and IG4 are created using method 2. Note that IG3 and IG4 take less than 100 days to reach the Moon compared to IG1 and IG2, which take 168 and 161 days, respectively. Also, the final position of the Moon is different for each trajectory. IG1 and IG2 are very similar except that IG1 ends at the first encounter with the Moon while IG2 first loops around the Moon (ballistic capture) and then continues until the spacecraft encounters the Moon a second time. IG3 and IG4 differ most in the location of the path point. In Figure 3.7(b), notice the strong kink in IG4; this is at the patch point; there is a larger discontinuity in velocity than at IG3's patch point.

Table 3.1 displays the trajectory details including initial distance from the Earth, denoted $d_{E.O}$, final distance from the Moon, $d_{M.O.}$, the total ΔV , which is broken into: ΔV_E (the ΔV necessary to leave circular Earth orbit), ΔV_M (the ΔV necessary to inject the spacecraft into a circular orbit at the Moon), ΔV_{traj} (the ΔV applied throughout the trajectory, concentrated at the patch point for the initial guess), and the number of nodes. All four initial guesses begin within 500 km of the Earth, but they end at varying distances from the Moon: 2,614 km for IG1, 249 km for IG2, 685 km for IG3, and 267 km for IG4. Also, at 8,683 m/s and 9,250 m/s, ΔV_E is much larger for IG3 and IG4. Additionally, the impulsive ΔV_{traj} at the patch point is 60 m/s, 56 m/s, 174 m/s, and 269 m/s for IG1, IG2, IG3, and IG4, respectively. These different initial guess trajectories will demonstrate the local nature of DMOC as well as the effect different constraints have on the optimal solution.

Note that the number of nodes used for each initial guess is different. As mentioned before, each trajectory is broken into sections of constant step size to accurately capture the dynamics while maintaining a reasonable number of nodes. Smaller step sizes are used near the Earth and Moon, while larger step sizes are

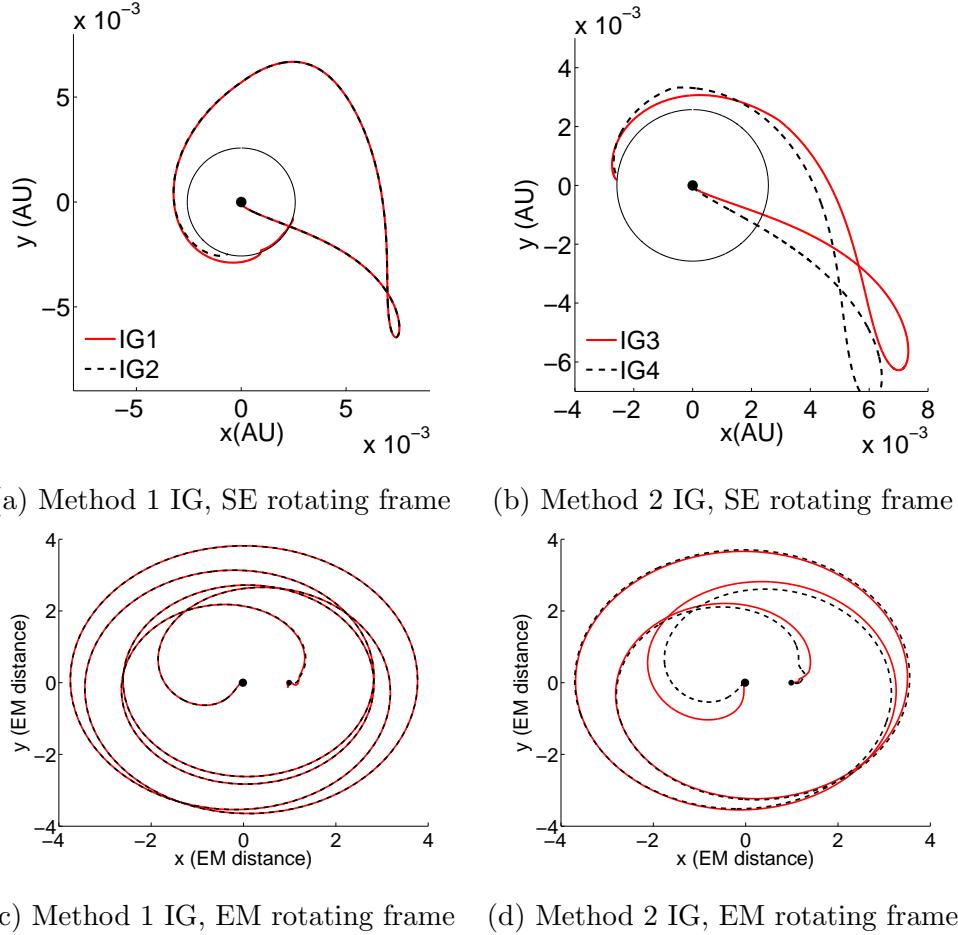


Figure 3.7: Initial guess trajectories: two initial guesses in Sun-Earth rotating frame created using (a) method 1 and (b) method 2. Initial guesses in Earth-Moon rotating frame for (c) method 1 and (d) method 2.

sufficient in regions far from the bodies. Note that IG3 and IG4 use more nodes than IG1 and IG2, even though IG1 and IG2 are longer trajectories. The maximum step size used for IG3 and IG4 is $1e-3$ while the maximum step size used for IG1 and IG2 is $5e-3$. The maximum step size is used in the middle portion of the trajectory, nearly in free space for IG1 and IG2, but closer to the Earth and the Moon's orbit for IG3 and IG4. Therefore, the dynamics dictate a smaller maximum step size for IG3 and IG4, leading to more nodes.

Even though the initial guesses have impulsive control applied at the patch point, when supplied to DMOc, the control force is assumed to be zero through-

out the trajectory. This allows DMOC to find a solution with a smooth control profile. Due to the local nature of the SQP (Sequential Quadratic Programming) algorithm, if the impulsive force is included, DMOC returns a control profile with an impulse and much higher ΔV .

Table 3.1: Details of Initial Guess Trajectories

	IG1	IG2	IG3	IG4
Time of flight (days)	168	161	98	95
Total ΔV (m/s)	3,966	3,992	9,951	10,193
ΔV_E (m/s)	3,214	3,214	8,683	9,250
ΔV_M (m/s)	692	722	1,094	674
ΔV_{traj} (m/s)	60	56	174	269
$d_{E.O.}$ (km)	195	195	217	378
$d_{M.O.}$ (km)	2,614	249	685	267
Number of nodes	1,810	1,190	2,271	3,018

3.6 Optimization

The optimization procedure begins with the formulation of cost function and constraints. Then, an SQP solver, SNOPT, performs the optimization. A number of optimization results are presented for the initial guesses described above.

3.6.1 Constraints and Cost Function

The next step before the optimization is the formulation of constraints. The primary constraint enforces the system dynamics, requiring that the forced discrete Euler-Lagrange equations, equation (2.58e), derived from the Lagrangian for the 4-body model, are fulfilled. The Lagrangian is given by

$$\begin{aligned} L = & \frac{1}{2} (\dot{x}^2 + \dot{y}^2) + \frac{1}{2} (x^2 + y^2) + xy - y\dot{x} + \frac{m_E}{\sqrt{x^2 + y^2}} \\ & + \frac{m_S}{\sqrt{(x+1)^2 + y^2}} + \frac{m_M}{\sqrt{(x-x_M)^2 + (y-y_M)^2}}. \end{aligned} \quad (3.7)$$

The control force, $f(q, \dot{q}, u) = u$, consisting of control parameters (u_x, u_y) , represents the control force applied in the x and y -direction, respectively, and is included in the forced discrete Euler-Lagrange equations.

In addition to the forced discrete Euler-Lagrange equations, a variety of other boundary conditions are enforced, including the initial and final distance from the Earth and Moon, respectively, the initial and final radial velocity, and a capture condition at the Moon. For each optimization, the initial distance from the Earth is required to be $d_{E,O} = 200$ km. The final distance from the Moon is different for each optimization, either equaling that of the initial guess or being allowed to vary within some specified range. If the radial velocity is required to be zero, the velocity of the initial (final) node is tangential to the trajectory, which allows the spacecraft to move out of (or into) a circular or elliptical orbit using less ΔV . If the final distance from the Moon is required to match that of the initial guess, ballistic capture happens naturally. However, if the final distance is allowed to vary, usually to decrease the final distance from that of the initial condition, ballistic capture may not be maintained by the optimization. As described by Belbruno and Miller [4], ballistic capture may be enforced with the condition,

$$E_M = \frac{1}{2} ((\dot{x}_N - \dot{x}_{M_N})^2 + (\dot{y}_N - \dot{y}_{M_N})^2) - \frac{m_M}{r_{M_N}} < 0, \quad (3.8)$$

where \dot{x}_N and \dot{y}_N are the x and y velocity components of the spacecraft at the final node, and \dot{x}_{M_N} and \dot{y}_{M_N} are the velocity components of the Moon at the final node. This equation gives the Keplerian energy with respect to the Moon and states that the kinetic energy of the spacecraft relative to the Moon at the final node must be less than the gravitational potential energy of the Moon. Due to gravitational effects of the Earth and Sun, this condition is not enough for permanent capture, but it does generate ballistic capture, as desired.

For each of the four initial guesses, two optimizations are performed with a slightly different set of constraints. Each optimal result is named according to the convention *DMOC i - j*, where $i = 1, \dots, 4$ is the initial guess number ($i = 1, 2$

correspond to method 1 and $i = 3, 4$ correspond to method 2), and $j = 1, 2$ is the optimization run number. For all the $j = 1$ runs, the final distance from the Moon is required to match that of the initial guess, except for DMOC 1-1 which is required to be 500 km instead of the 2,614 km of the initial guess. Also, for these trials, the initial and final radial velocity must be zero. For all the $j = 2$ runs, the final distance from the Moon is allowed to vary within some range, only the initial radial velocity is zero, and the capture condition is enforced. The constraints for each run are outlined in Table 3.2.

Table 3.2: Optimization Constraints

	$d_{E.O}$ (km)	$d_{M.O}$ (km)	$v_{r_0} = 0$	$v_{r_N} = 0$ or capture
DMOC 1-1	200	500	yes	$v_{r_N} = 0$
DMOC 1-2	200	[100 : 5,000]	yes	capture
DMOC 2-1	200	249	yes	$v_{r_N} = 0$
DMOC 2-2	200	[400 : 1,000]	yes	capture
DMOC 3-1	200	685	yes	$v_{r_N} = 0$
DMOC 3-2	200	[500 : 1,000]	yes	capture
DMOC 4-1	200	267	yes	$v_{r_N} = 0$
DMOC 4-2	200	[500 : 1,000]	yes	capture

The goal of the optimization is to minimize the control effort; correspondingly, the discrete cost function is

$$J_d(u_d) = \sum_k h_k \|(u_k)_d\|^2, \quad (3.9)$$

where $(u_k)_d = \{(u_{x,k}, u_{y,k})\}_{k=0}^{N-1}$ is a vector of length $2N$, corresponding to $N + 1$ total discretization points (nodes), and $\|\cdot\|$ denotes the 2-norm. The total ΔV applied throughout the trajectory, based on the control forces computed with DMOC, is given by

$$\Delta V_{traj} = \alpha_V \sum_k h_k \|(u_k)_d\|, \quad (3.10)$$

where α_V scales the velocity to units of m/s.

The optimization process is performed using SNOPT [17]. The primary advantage of SNOPT is its ability to handle large, sparse, nonlinear programming problems. This problem is large, requiring more than 1000 nodes for basic results. A trajectory with $N + 1$ nodes results in $2N + 2$ constraints and $8N^2 + 12N + 4$ constraint derivatives. Obviously, this is a huge endeavor in terms of memory. Fortunately, this problem is very sparse: most of the constraint derivatives are zero. SNOPT allows the user to provide analytical expressions for the derivatives, and to store these as sparse matrices, saving a great deal of memory and speeding up the process.

3.6.2 Optimization Results

The optimization results for methods 1 and 2 are summarized in Table 3.3, and the initial guess and optimal trajectories are shown in Figure 3.8 and Figure 3.9. Each optimal trajectory is different, demonstrating the local nature of DMOC; each optimized trajectory is similar to its initial guess. Most importantly, ΔV_{traj} is reduced to zero. With the correct initial condition, it travels to the Moon using no fuel. Also, both ΔV_E and ΔV_M are reduced because the radial velocity constraint forces the initial and final node of the trajectory to be tangent to a circular orbit at the same distance. These impulsive ΔV are included for completeness, but are not explicitly included in the optimization or cost function.

Table 3.3: Details of Optimization

	1-1	1-2	2-1	2-2	3-1	3-2	4-1	4-2
Flight time (days)	168	168	161	161	98	98	95	95
Total ΔV (m/s)	3,844	3,803	3,853	4,025	3,764	3,790	3,824	3,787
ΔV_E (m/s)	3,240	3,240	3,240	3,240	3,212	3,212	3,212	3,212
ΔV_M (m/s)	604	563	613	785	552	578	612	575
ΔV_{traj} (m/s)	0	0	0	0	0	0	0	0
$d_{E.O.}$ (km)	200	200	200	200	200	200	200	200
$d_{M.O.}$ (km)	500	5,000	249	931	685	500	267	500

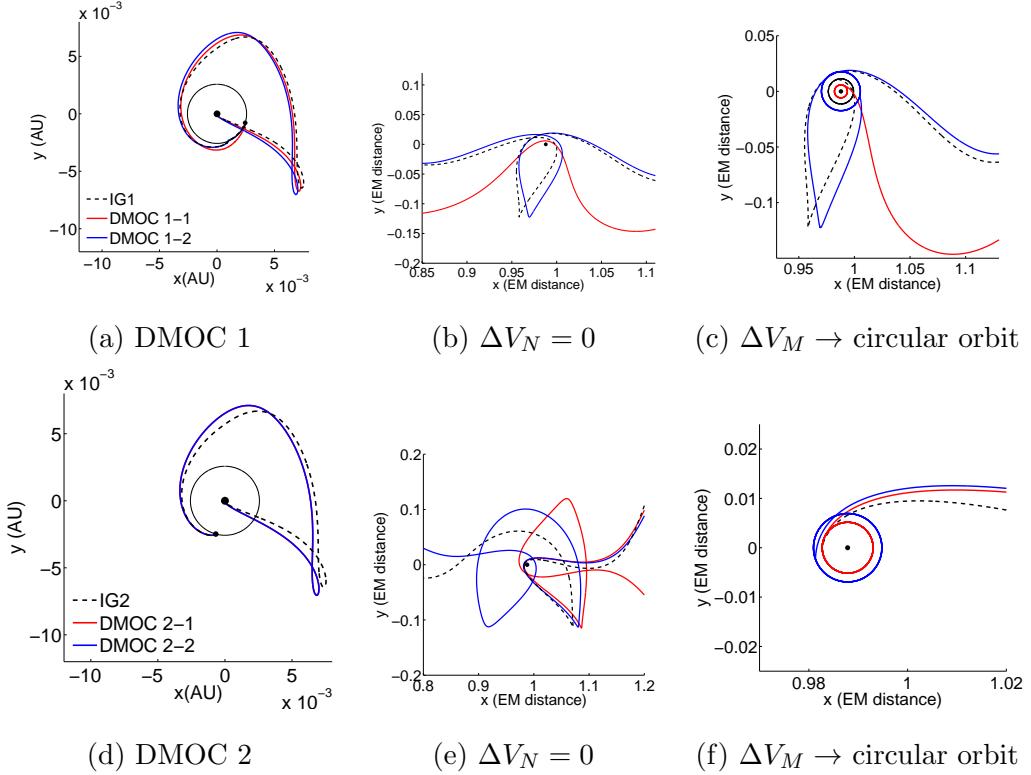


Figure 3.8: DMOC results generated using different constraints for method 1. Optimal trajectories in SE rotating frame for (a) IG1 and (d) IG2. Natural capture behavior at the Moon in EM rotating frame for (b) IG1 and (e) IG2. Circular orbit at the Moon facilitated by ΔV_M applied at final node of optimal trajectory for (c) IG1 and (f) IG2.

In Figure 3.8, (a) and (d) show the optimal trajectories and initial guesses for IG1 and IG2, (b) and (e) display the trajectory integrated beyond the final node with zero ΔV in the EM rotating frame. As shown in the Figure 3.8(b), DMOC 1-1 is not ballistically captured, even though the initial guess had capture because the final distance at the Moon is reduced from 2,614 km to just 500 km without enforcing capture. For DMOC 1-2, capture is maintained without any additional ΔV . Figure 3.8(c) shows the circular orbit at the Moon when ΔV_M is applied at the final node of the optimal trajectory. Notice that DMOC 1-2 is ballistically captured before the final node, so the trajectory loops around the Moon once before entering the permanent circular orbit. As shown in Figure 3.8(e) and (f), continuation of the trajectory shows that ballistic capture is naturally maintained

for both DMOC 2-1 and DMOC 2-2, and the application of ΔV_M at the final node leads to a permanent circular orbit.

Figure 3.9 shows similar results for the optimal trajectories generated for method 2 initial guesses. Unlike the results shown in Figure 3.8, the different constraints have less of an impact on the optimal results. DMOC 3-1 and DMOC 3-2 are very similar, as are DMOC 4-1 and DMOC 4-2. All four optimal trajectories naturally maintain ballistic capture at the Moon, as shown in Figure 3.9(b) and (e). As before, ΔV_M applied at the final node of the trajectory injects the spacecraft into a permanent circular orbit.

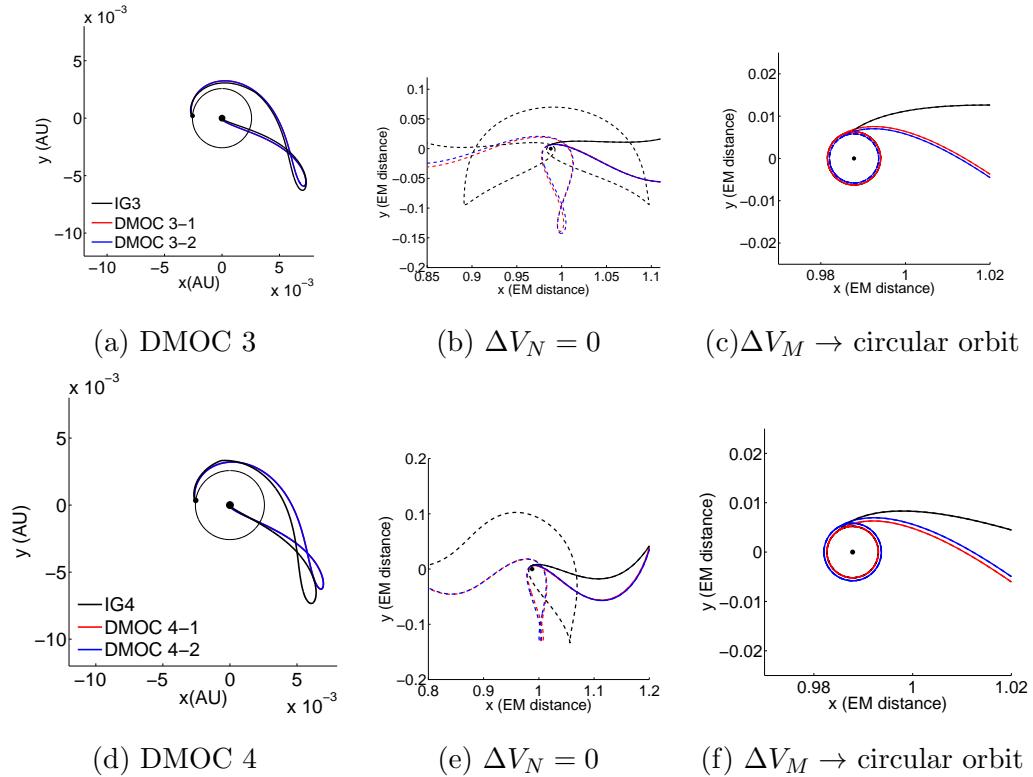


Figure 3.9: DMOC results generated using different constraints for method 2. Optimal trajectories in SE rotating frame for (a) IG3 and (d) IG4. Natural capture behavior at the Moon in EM rotating frame for (b) IG3 and (e) IG4. Circular orbit at the Moon facilitated by ΔV_M applied at final node of optimal trajectory for (c) IG3 and (f) IG4.

3.7 Low Thrust

The lack of control throughout the trajectory means that this is not actually optimal control. DMOC essentially adjusts the boundary conditions until a free transfer is found. These results are interesting nonetheless. Also, as shown in the previous sections, an impulsive maneuver is still required for permanent capture at the Moon. Therefore, this section explores the use of low thrust propulsion to spiral into an elliptical orbit at the Moon.

3.7.1 Formulation of Low Thrust Initial Guess

Since this problem focuses on the trajectory behavior near the Moon, it is natural to consider the dynamics of the 4-body problem in the Earth-Moon rotating frame. The low thrust equations of motion for this model in Earth-Moon rotating coordinates are [39, 43]

$$\ddot{x} - 2\dot{y} = \frac{\partial \Omega}{\partial x} + \frac{T_x}{m}, \quad (3.11a)$$

$$\ddot{y} + 2\dot{x} = \frac{\partial \Omega}{\partial y} + \frac{T_y}{m}, \quad (3.11b)$$

$$\dot{m} = -\frac{T}{I_{sp}^{LT} g_0}, \quad (3.11c)$$

$$0 \leq T = \sqrt{T_x^2 + T_y^2} \leq T_{max}, \quad (3.11d)$$

$$\begin{aligned} \Omega = & \frac{x^2 + y^2}{2} + \frac{m_E}{\sqrt{(x + \mu)^2 + y^2}} + \frac{m_M}{\sqrt{(x + (1 - \mu))^2 + y^2}} \\ & + \frac{m_S}{\sqrt{(x - x_S)^2 + (y - y_S)^2}} - \frac{m_S}{a_S^3}(x_S + y_S), \end{aligned} \quad (3.11e)$$

and m_S , m_E , and m_M are the normalized mass of the Sun, Earth, and Moon, respectively, given by

$$m_E = 1 - \mu, \quad (3.12a)$$

$$m_M = \mu, \quad (3.12b)$$

$$m_S = \frac{M_S}{M_M + M_E} = 3.2890 \times 10^5, \quad (3.12c)$$

and

$$\mu = \frac{M_M}{M_E + M_M} = 0.01215. \quad (3.13)$$

As before, M_i , $i = E, M, S$, denotes the mass in kg, and x_S and y_S represent the x - and y -positions of the Sun as a function of time given by

$$\theta_S = -\omega_S t + \theta_{S0}, \quad (3.14a)$$

$$x_S = a_S \cos \theta_S, \quad (3.14b)$$

$$y_S = a_S \sin \theta_S, \quad (3.14c)$$

where t is time, θ_{S0} is the initial angle of the Sun with respect to the x -axis in the Earth-Moon rotating frame, $a_S = 3.8881 \times 10^2$ is the normalized radius of the Sun's circular orbit, and $\omega_S = 0.9251$ is the normalized rotation rate of the Sun. Note that $I_{sp}^{LT} = 3000$ s is the specific impulse of the thruster, $g_0 = 9.81$ m/s² is the acceleration due to gravity at sea level, and T_{max} is the maximum allowable thrust; in this case, 0.5 N is used. As before all of these values are normalized and non-dimensionalized.

The performance of a low-thrust trajectory is measured based on the fuel mass consumption, m_p , and the mass fraction, $\frac{m_p}{m_0}$, where

$$m_p = m_0 - m_N, \quad (3.15)$$

the difference of the initial and final mass. To determine the mass fraction for an impulsive ΔV ,

$$\frac{m_p}{m_0} = 1 - \exp \left(-\frac{\sum_i \Delta V_i}{I_{sp}^{HT} g_0} \right), \quad (3.16)$$

where $I_{sp}^{HT} = 300$ s is the specific impulse of an impulsive thruster, and ΔV_i are the impulsive ΔV .

The desired optimal trajectory ends in an elliptical orbit at the Moon. Therefore, the conditions (x, y, \dot{x}, \dot{y}) of an elliptical orbit with eccentricity $e = 0.65$ and desired distance from the Moon are integrated backwards applying the maximum

value of thrust. At a specified time, T_t , the thrust is set to zero and the integration continues. By varying T_t and Φ , the phase angle of the ellipse with respect to the Earth-Moon rotating frame's x -axis (shown in Figure 3.10), it is possible to generate a trajectory that spirals out from the elliptical orbit at the Moon, flows through the periodic orbit at L_2 , shown in Figure 3.11(a),(b), and follows the stable invariant manifold toward the intersection with the Sun-Earth unstable manifold, the patch point, shown in Figure 3.11(c),(d).

For this low thrust initial guess, time $t = 0$ is set at the desired final point of the trajectory near the Moon, and time flows backwards. This differs from the original case when designing an initial guess in the Sun-Earth rotating frame using method 1 for which time $t = 0$ occurs at the patch point. To ensure the same manifold intersection as before, θ_{S_0} , the initial angle of the Sun with respect to the Earth-Moon x -axis, is chosen such that θ_M at the patch point equals θ_{M_0} from method 1. This is achieved using an iterative process in which a guess is posed for θ_{S_0} , the trajectory is integrated backwards to the patch point, θ_M is computed and θ_{S_0} is refined. This process continues until $\theta_M(\text{patch point}) - \theta_{M_0(\text{method 1})} < \text{tolerance}$.

When the patch point is reached, an impulsive ΔV is applied and integration continues until the trajectory reaches the Earth. This ΔV may be adjusted until the desired radius at the Earth is reached. The full trajectory is shown in Figure 3.11(c) in Earth-Moon rotating coordinates and (d) in Sun-Earth rotating coordinates.

If the initial conditions at the Moon are integrated forwards with no thrust, the desired elliptical orbit at the Moon results. Recall that the majority of the trajectory, except for the small portion near the Moon and at the patch point, is achieved using zero thrust. For optimization purposes, the trajectory is reorganized such that it begins at the Earth and ends near the Moon. The initial guess ends at the last node before low thrust is applied, excluding the low thrust spiral towards the Moon.

Two initial guesses are tested. Both trajectories begin approximately 167 km from the surface of the Earth. IG LT1 requires an impulsive $\Delta V = 68$ m/s

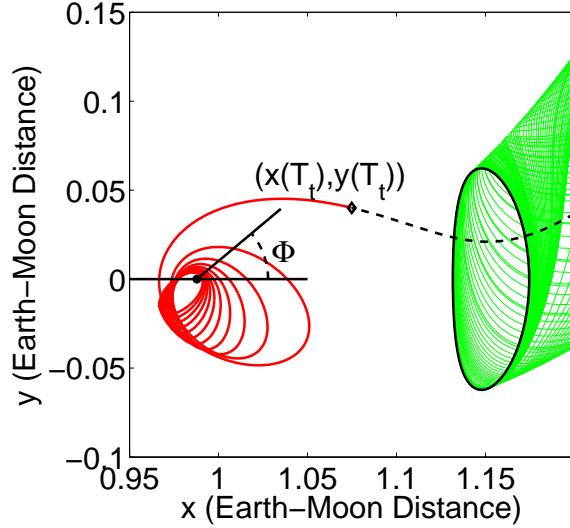


Figure 3.10: Low thrust spiral with variable parameters Φ and T_t . Φ is the angle of the semi-major axis of the elliptical orbit at the Moon with respect to the Earth-Moon x -axis. T_t marks the transition from no thrust to low thrust.

at the patch point and with low thrust spiral, ends in an elliptical orbit with $d_p = 1000$ km at periapsis. The mass fraction for the low-thrust portion of the trajectory is 0.0158, and according to equation (3.16), the mass fraction for the impulsive ΔV is 0.0230, giving a total mass fraction of 0.0388. The second initial guess, IG LT2, requires an impulsive $\Delta V = 33$ m/s at the patch point and ends in an elliptical orbit at the Moon with periapsis $d_p = 100$ km. The total mass fraction is 0.0356 with 0.0243 for the low thrust spiral and 0.0113 for the impulsive ΔV .

3.7.2 Optimization for Low Thrust Trajectory

The optimization process differs from that of method 1 and method 2. In particular, the optimized solution should be achievable using only low thrust. As before, the primary constraints are the discrete Euler Lagrange equations, equation (2.58e), derived from the Lagrangian

$$L = \frac{1}{2} (\dot{x}^2 + \dot{y}^2) + \frac{1}{2} (x^2 + y^2) + xy - y\dot{x} + \frac{m_E}{r_E} + \frac{m_M}{r_M} + \frac{m_S}{r_S} - \frac{m_S}{a_S^3} (x \cdot x_S + y \cdot y_S), \quad (3.17)$$

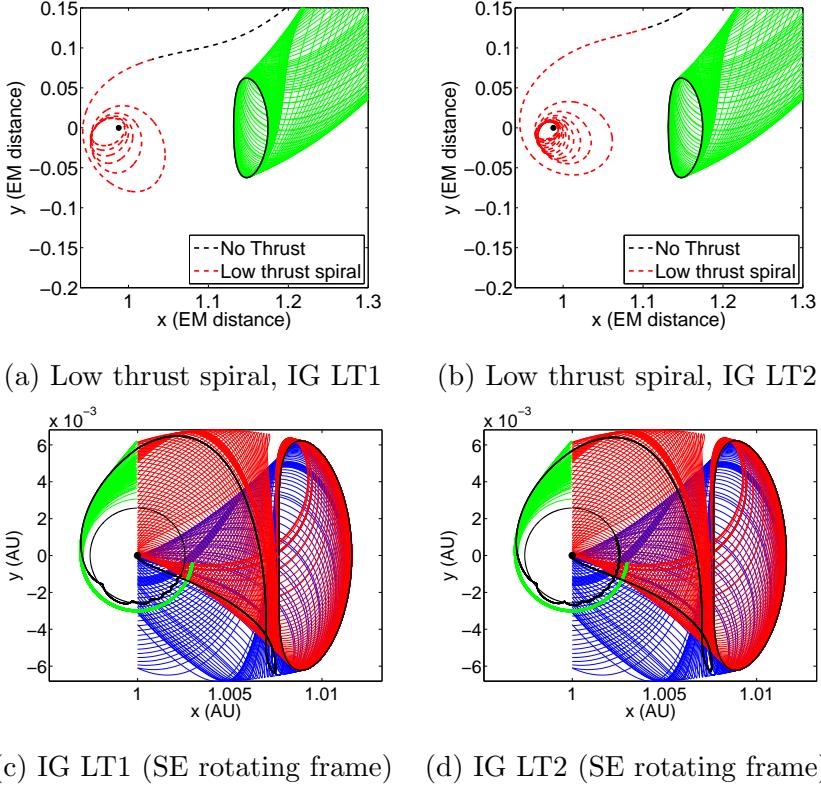


Figure 3.11: Initial guess trajectories with low thrust are achieved by integrating backwards from desired elliptical orbit at the Moon. Maximum thrust is applied in the direction opposite the velocity until reaching time, T_t , and then integration continues without thrust for (a) IG LT1 and (b) IG LT2. The full trajectory, including a small impulsive ΔV at the patch point, ends 167 km from the Earth in the Sun-Earth rotating frame for (c) IG LT1 and (d) IG LT2.

where r_E , r_M , and r_S are the distance of the spacecraft from the center of the Earth, Moon, and Sun, respectively. Since mass is included in the equations of motion, equation (3.11c), the mass at each node, m_k , is an optimization variable, and the mass dynamics are enforced by the constraint equation,

$$m_{k+1} - m_k + h_k \left(\frac{T_k}{I_{sp}^{LT} g_0} \right) = 0. \quad (3.18)$$

where T_k is the thrust magnitude applied at each node, k . There are several options for the definition of $T_{x,k}$ and $T_{y,k}$.

1. $T_{x,k}$ and $T_{y,k}$ are optimization variables that vary within the limits $[-T_{max}, T_{max}]$

subject to the constraint $T_k = \sqrt{T_{x,k}^2 + T_{y,k}^2} \leq T_{max}$.

2. The optimization variable is τ_k , and $T_{x,k}$ and $T_{y,k}$ are defined such that the thrust, $T_k = |\tau_k|$, is applied in the direction parallel to or opposite the velocity vector, depending on the sign of τ_k ,

$$T_{x,k} = \tau_k \left(\frac{v_{x,k}}{\sqrt{v_{x,k}^2 + v_{y,k}^2}} \right), \quad (3.19)$$

$$T_{y,k} = \tau_k \left(\frac{v_{y,k}}{\sqrt{v_{x,k}^2 + v_{y,k}^2}} \right), \quad (3.20)$$

$$-T_{max} \leq \tau_k \leq T_{max}. \quad (3.21)$$

There are problems with both of these options that arise in the computation of the derivatives for the constraint Jacobian. For option 1, if $T_{x,k} = 0$ and $T_{y,k} = 0$, which is allowable, the derivative of the mass constraint, equation (3.18), with respect to $T_{x,k}$ or $T_{y,k}$ does not exist. For option 2, since T_k is not differentiable when $\tau_k = 0$, the derivative of the mass constraint with respect to τ_k also does not exist everywhere. There are possible tricks in the problem setup that avoid these existence problems, but both strategies display poor convergence results. Ultimately, the best results are achieved when the thrust is defined similar to option 2, but with a restriction.

Let T_k be the control optimization variable such that the thrust is applied in the direction opposite the velocity (this means that the thrust acts to slow the spacecraft)

$$T_{x,k} = T_k \left(-\frac{v_{x,k}}{\sqrt{v_{x,k}^2 + v_{y,k}^2}} \right), \quad (3.22)$$

$$T_{y,k} = T_k \left(-\frac{v_{y,k}}{\sqrt{v_{x,k}^2 + v_{y,k}^2}} \right), \quad (3.23)$$

$$0 \leq T_k \leq T_{max}, \quad (3.24)$$

where $v_{x,k}$ and $v_{y,k}$ are the velocity in the x - and y -directions at node k , respectively. The discrete left and right control forces based on this thrust definition are

$$f_{x,k}^+ = f_{x,k}^- = \frac{h_k}{2} \frac{T_k}{\left(\frac{m_{k+1}+m_k}{2}\right)} \left(-\frac{\left(\frac{x_{k+1}-x_k}{h_k}\right)}{\sqrt{\left(\frac{x_{k+1}-x_k}{h_k}\right)^2 + \left(\frac{y_{k+1}-y_k}{h_k}\right)^2}} \right), \quad (3.25)$$

$$f_{y,k}^+ = f_{y,k}^- = \frac{h_k}{2} \frac{T_k}{\left(\frac{m_{k+1}+m_k}{2}\right)} \left(-\frac{\left(\frac{y_{k+1}-y_k}{h_k}\right)}{\sqrt{\left(\frac{x_{k+1}-x_k}{h_k}\right)^2 + \left(\frac{y_{k+1}-y_k}{h_k}\right)^2}} \right). \quad (3.26)$$

In summary, the optimization variables are x_k, y_k , and m_k , for $k = 1, \dots, N$ and T_k for $k = 1, \dots, N-1$, and the constraints that enforce dynamics are

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) + f_{k-1}^+ + f_k^- = 0, \quad (3.27)$$

$$m_{k+1} - m_k + h_k \left(\frac{T_k}{I_{sp}^{LT} g_0} \right) = 0. \quad (3.28)$$

Since the optimized solution should naturally flow into the low thrust spiral determined for the initial guess, the final mass m_N is required to be 1000 kg (the initial mass assumed for the spiral). Additionally, the initial altitude of a circular orbit at the Earth must be $d_{E,O} = 167$ km and $v_{r0} = 0$. To ensure that the spiraling, low thrust portion remains possible, the position and momentum at the final node must match that of the initial guess.

Two different objective functions are considered. The first aims to minimize the initial mass (since the final mass is held fixed),

$$J_{d1} = m_1. \quad (3.29)$$

The second objective function seeks to minimize the overall control effort, or thrust,

$$J_{d2} = \sum_k h_k \left(\frac{T_k}{\frac{m_{k+1}+m_k}{2}} \right)^2. \quad (3.30)$$

These objective functions lead to very similar optimal trajectories with different thrust profiles. Optimization with J_{d1} , equation (3.29), converges faster and leads to a thrust profile for which the thrust is turned off for most of the trajectory. Minimizing the control effort, equation (3.30), leads to a continuous thrust profile, with magnitude much less than T_{max} , applied for most of the trajectory. Results for both initial guesses, IG LT1 and IG LT2, are presented for J_{d1} , denoted DMOC LT1-1 and DMOC LT2-1, respectively. Only IG LT1 is optimized with J_{d2} and since the trajectory results, denoted by DMOC LT1-2, are nearly identical to those of J_{d1} , only the thrust profile and thrust location are shown.

Figure 3.12 shows the optimization results for DMOC LT1-1. The transition from no thrust to low thrust near the Moon is shown in Earth-Moon rotating coordinates in Figure 3.12(a). Figures 3.12(c) and (d) show the entire trajectory with the location of thrust arcs in red in Sun-Earth rotating coordinates and Earth-centered inertial coordinates, respectively. If the final conditions at the end of the low-thrust spiral are integrated with no thrust, the trajectory settles into the desired orbit at the Moon, shown in Moon-centered inertial coordinates in Figure 3.12(b). Figure 3.13 shows the same plots for DMOC LT2-1. Figures 3.14(a) and (b) show the thrust arcs in Sun-Earth rotating coordinates and Earth-centered inertial coordinates for DMOC LT1-2.

The entire thrust profile for DMOC LT1-1 is shown in Figure 3.15(a). The majority of the profile consists of zero thrust, with a small maximum thrust arc applied for about 9.5 hours on day 33. Also, the thruster turns on approximately 7 hours before reaching the beginning of the low thrust spiral. This optimal trajectory from Earth to elliptical orbit at the Moon requires just 17 kg of fuel, giving a total mass fraction of $m_p/m_0 = 0.0168$. This thrust profile is nearly bang-bang control; the thrust is either off or on at the maximum value.

The entire thrust profile for DMOC LT2-1 is shown in Figure 3.15(b). This thrust profile contains two smaller, continuous thrust arcs. The first arc begins on day 31, rising continuously to a maximum of 0.1 N before decreasing back to zero by day 32. The second thrust arc begins on day 146, reaches a maximum of

0.29 N, and ends approximately 1 day later. The mass fraction for this trajectory is $m_p/m_0 = 0.0249$, and uses just 25 kg of fuel.

The thrust profile for DMOC LT1-2, excluding the low thrust spiral, is shown in Figure 3.15(c). The thrust is continuous, except for the drop-off to zero at the final node. The thrust turns on approximately 1 day into the trajectory. The thrust steadily increases to a maximum of 0.0065 N on day 34 and then decreases back to zero on day 79. The thrust is then off until day 113 when it begins to increase to a maximum of 0.016 N, decreases slightly and increases again to the global maximum of 0.0371 N on day 153 before shutting off. This trajectory requires 18 kg of fuel and its total mass fraction is $m_p/m_0 = 0.0179$. Notice the difference

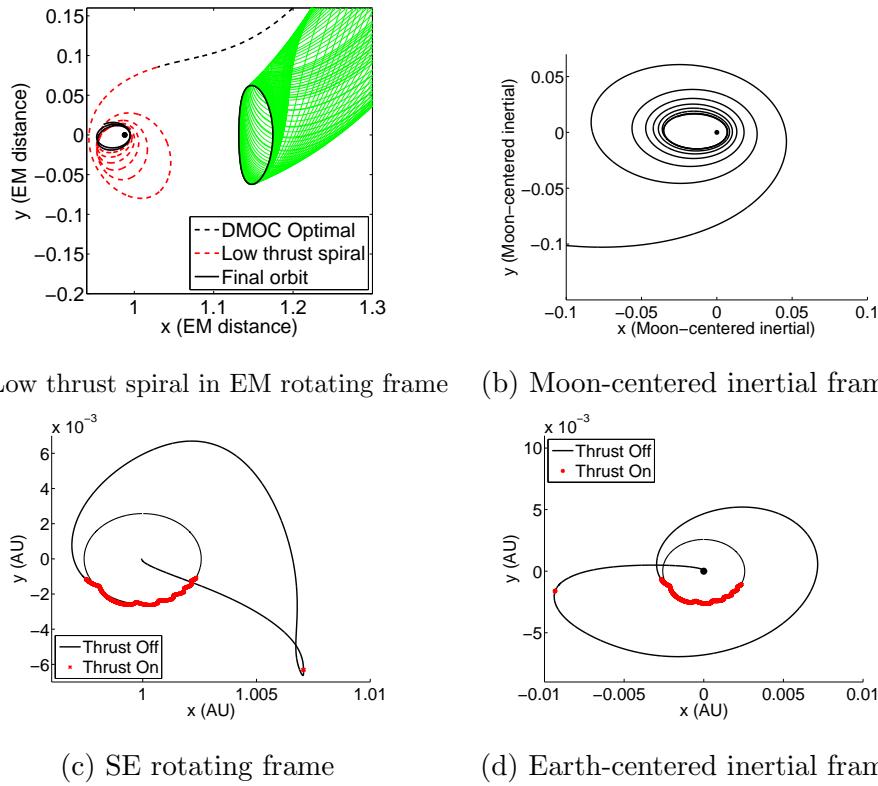


Figure 3.12: DMOC LT1: optimal trajectory with low thrust. (a) The optimal trajectory is shown near the Moon in the Earth-Moon rotating frame, with low thrust spiral and final orbit added. (b) Low thrust spiral and final orbit in Moon-centered inertial frame. (c) Optimized trajectory in Sun-Earth rotating frame with low thrust arcs indicated in red, and (d) the optimized trajectory in Earth-centered inertial frame with low thrust arcs.

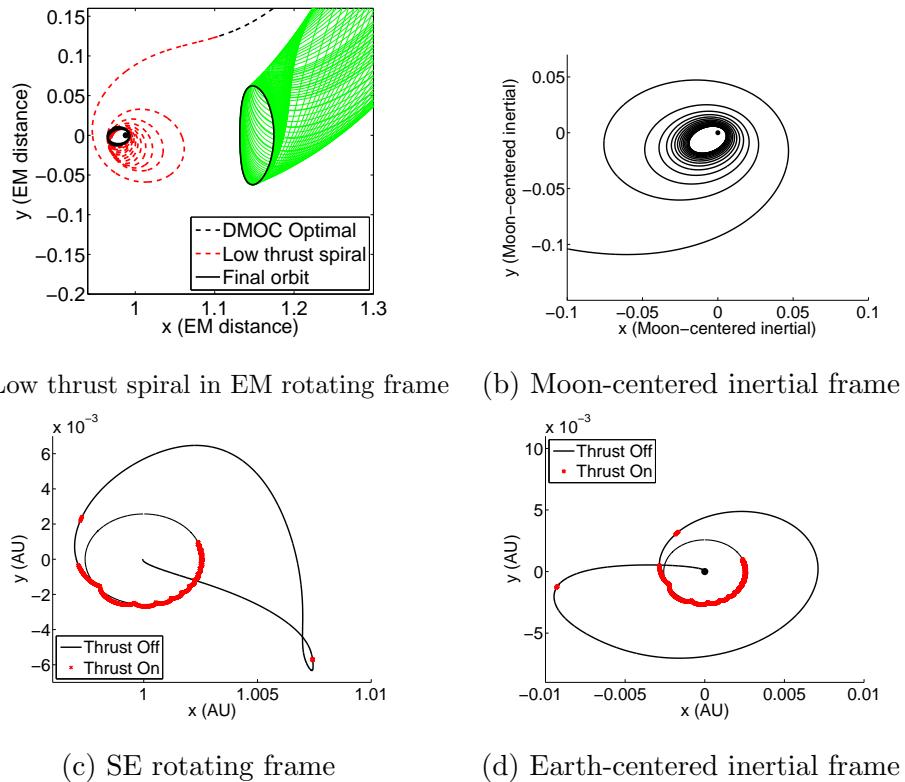


Figure 3.13: DMOC LT2: optimal trajectory with low thrust. (a) The optimal trajectory is shown near the Moon in the Earth-Moon rotating frame, with low thrust spiral and final orbit added. (b) Low thrust spiral and final orbit in Moon-centered inertial frame. (c) Optimized trajectory in Sun-Earth rotating frame with low thrust arcs indicated in red, and (d) the optimized trajectory in Earth-centered inertial frame with low thrust arcs.

in scale between Figures 3.15(a) and (b) and Figure 3.15(c); the thrust profile resulting from J_{d2} leads to thrust magnitudes more than an order of magnitude less than those generated by J_{d1} .

Table 3.4 summarizes the optimization results. Of the three optimal results, DMOC LT1-1 results in the smallest mass fraction and takes 4 days less than DMOC LT2-1. Figure 3.16 shows the mass consumption for each trajectory. DMOC LT1-2 requires the highest starting mass. DMOC LT2-1 requires the smallest initial mass, but the low thrust spiral burns more fuel than the other low thrust spirals.

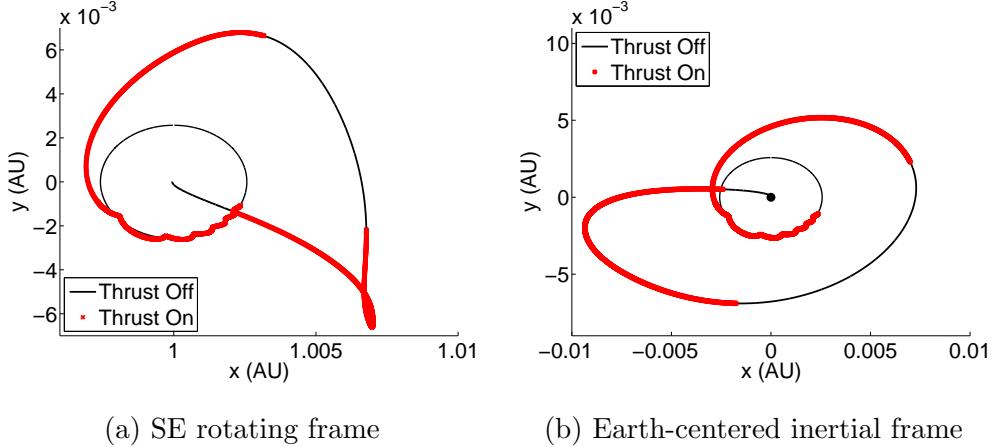


Figure 3.14: DMOC LT1-2: optimal trajectory with low thrust, generated with cost function J_{d2} . (a) Optimized trajectory in Sun-Earth rotating frame with low thrust arcs indicated in red, and (b) the optimized trajectory in Earth-centered inertial frame with low thrust arcs.

Table 3.4: Details of Low Thrust Optimization

	LT IG1	DMOC LT1-1	DMOC LT1-2	LT IG2	DMOC LT2-1
Flight Time	164	164	164	168	168
ΔV_E (m/s)	3,190	3,190	3,190	3,189	3,189
ΔV_M (m/s)	381	381	381	465	465
ΔV_{traj} (m/s)	68	-	-	33	-
$d_{E.O.}$ (km)	167	167	167	167	167
$d_{E.O.}$ (km)	1000	1000	1000	100	100
m_p/m_0	0.0388	0.0168	0.0179	0.0356	0.0249

3.8 Analysis and Comparison

Table 3.5 displays several trajectory results from literature to compare with the DMOC solutions. The first four trajectories are presented by Belbruno and Miller [4] where WSB, BP, H, and BE stand for Weak Stability Boundary, biparabolic, bielliptic, and Hohmann transfers, respectively. All of these trajectories begin in a 167 km circular Earth orbit, and end in a 100 km circular orbit at the Moon. The WSB trajectory is very similar to that of Shoot the Moon and produces a low energy transfer trajectory that connects the weak stability boundaries (closely

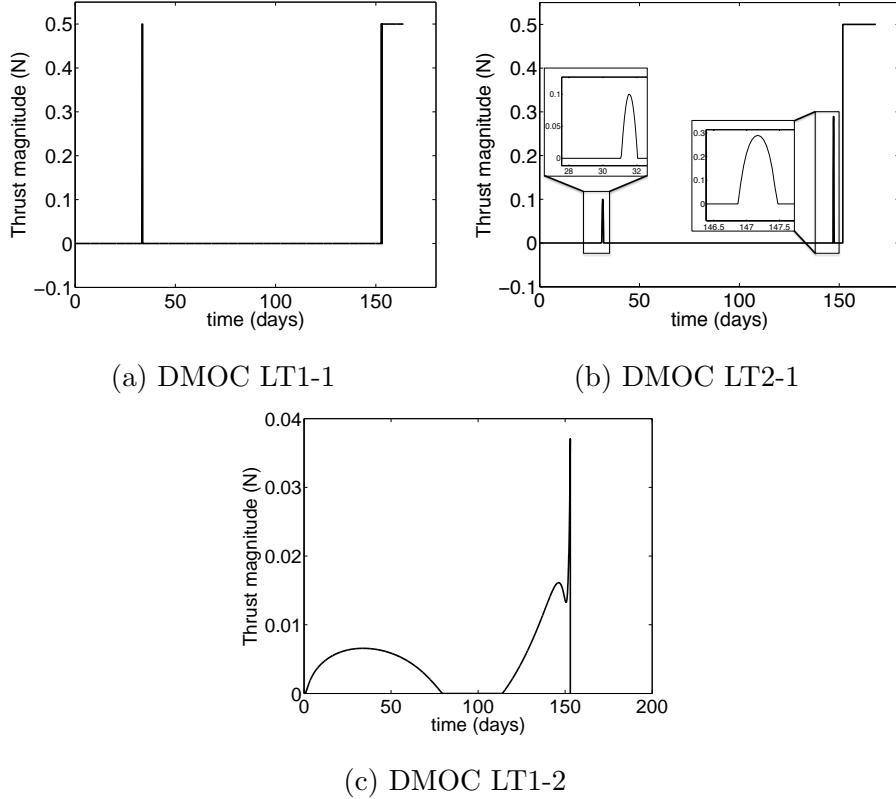


Figure 3.15: Thrust profile: magnitude of thrust throughout the trajectory for (a) DMOC LT1-1, (b) DMOC LT2-1, and (c) DMOC LT1-2.

related to invariant manifolds) of the Sun-Earth and Earth-Moon systems. Additionally, the Shoot the Moon trajectory presented by Koon, Lo, Marsden, and Ross [28] is denoted by SM. This trajectory begins in a 200 km circular orbit, just like the DMOC solutions and ends in ballistic capture. The final orbit and ΔV_M required to circularize the orbit at the Moon are not provided for SM.

For each trajectory, certain metrics are compared, as done in Belbruno and Miller [4]. First, it is assumed that the launch vehicle provides an injection ΔV equal to that necessary for a Hohmann transfer from that particular altitude orbit. The first four trajectories begin in a 167 km orbit, requiring $\Delta V_H = 3.143 \text{ km/s}$ to leave Earth orbit using a Hohmann transfer. Any additional ΔV required for injection, denoted $\Delta V_E - \Delta V_H$, must be included in the mission ΔV . The DMOC and SM results begin in 200 km orbits; such orbits require $\Delta V_H = 3.149 \text{ km/s}$ for

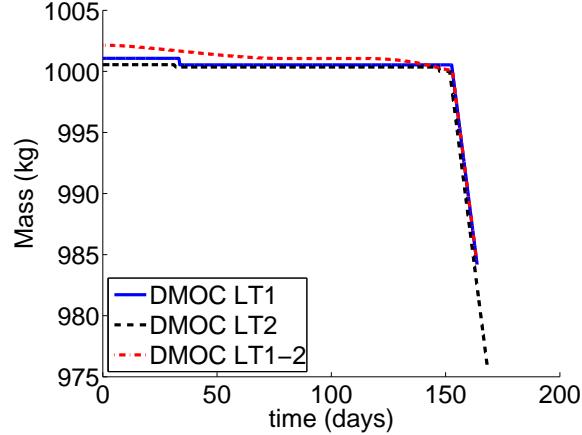


Figure 3.16: Mass consumption for DMOC LT1, DMOC LT2, and DMOC LT1-2

a Hohmann transfer. Next, mid-course ΔV are compared, as denoted by ΔV_{traj} . For the WSB, BP, H, and BE trajectories, it is assumed that the spacecraft is first captured at the Moon in an elliptical orbit with eccentricity $e = 0.95$, and periapsis radius $r_p = r_M + 100$ km where r_M is the radius of the Moon. The ΔV necessary to enter this elliptical capture orbit is denoted by ΔV_C ,

$$\Delta V_C = \sqrt{V_\infty^2 + \frac{2m_M}{r_p}} - \sqrt{\frac{2m_M r_a}{(r_a + r_p)r_p}}, \quad (3.31)$$

where r_a is the radius of the orbit at apopasis and

$$V_\infty = \sqrt{m_M \left(\frac{2}{r_{M_H}} - \frac{1}{a_H} \right)} - \sqrt{\frac{m_M}{r_{M_H}}}, \quad (3.32)$$

where r_{M_H} and a_H are the periapsis and semi-major axis of the Hohmann transfer ellipse, respectively. Finally, ΔV_M is necessary to circularize the orbit at the periapsis. The DMOC solutions do not include ΔV_C , they are circularized directly from the final node. The total measure of ΔV performance is given by $\sum \Delta V = \Delta V_E - \Delta V_H + \Delta V_{traj} + \Delta V_C + \Delta V_M$. All numbers in the table are given in km/s.

As shown by the % change from the corresponding Hohmann transfer, the DMOC results are at most 19% better than the Hohmann transfer (DMOC 3-1). Two of the DMOC results, DMOC 1-2 and DMOC 2-2, are actually worse than the

Hohmann transfer, requiring 7% and 19% more ΔV , respectively. Both of these results lack the constraint requiring the radial velocity at the final node to be zero, demonstrating the importance of that constraint. The other DMOC results are competitive with those from the literature. An emphasis on ΔV_E and ΔV_M for the initial guess or in the optimization may lead to even better results.

Table 3.5: Comparison for Trajectories with Impulsive ΔV

Type	$\Delta V_E - \Delta V_H$	ΔV_{traj}	ΔV_C	ΔV_M	$\sum \Delta V$	% change from H
WSB	0.018	0.029	0	0.648	0.695	-18
BP	0.089	0	0.073	0.648	0.810	-4
H	0.000	0	0.2	0.648	0.848	0
BE	0.018	0.287	0.052	0.648	1.005	19
DMOC 1-1	0.090	0	-	0.604	0.694	-11
DMOC 1-2	0.091	0	-	0.562	0.653	7
DMOC 2-1	0.091	0	-	0.613	0.704	-13
DMOC 2-2	0.091	0	-	0.785	0.876	19
DMOC 3-1	0.062	0	-	0.552	0.614	-19
DMOC 3-2	0.062	0	-	0.578	0.640	-17
DMOC 4-1	0.063	0	-	0.612	0.675	-16
DMOC 4-2	0.063	0	-	0.575	0.638	-18
SM	0.062	0.034	-	-	-	-

Table 3.6 compares the DMOC low thrust results with low thrust reference trajectories from literature. These reference trajectories, denoted by LT ref 1 and LT ref 2, presented by Mingotti et al. [39], were created using a shooting optimization method. All trajectories begin in a 167 km circular orbit at the Earth and end in an elliptical orbit at the Moon. Comparing the trajectories, DMOC trajectories require less fuel and flight time than the trajectories created using shooting methods. Due to similar end conditions, DMOC LT1-1 and DMOC LT1-2 may be compared directly with LT ref 1, and DMOC LT2-1 may be compared with LT ref 2. The mass fractions of DMOC LT1-1 and DMOC LT1-2 are an improvement of 46% and 42%, respectively, over the mass fraction of LT ref 1, and DMOC LT2-1 is a 59% improvement over the mass fraction of LT ref 2. Note that

these comparisons exclude the small differences in ΔV_E necessary to start on the optimal trajectory.

Table 3.6: Comparison for Trajectories with Low Thrust

Type	ΔV_E (m/s)	d_p (km)	e	m_p/m_0	flight time (days)
LT ref 1	3,195	1000	0.65	0.031	236
LT ref 2	3,203	100	0.65	0.061	228
DMOC LT1	3,190	1000	0.65	0.0168	164
DMOC LT2	3,189	100	0.65	0.0249	178
DMOC LT1-2	3,190	1000	0.65	0.0179	164

3.9 Conclusion

This chapter describes how to combine dynamical systems theory with discrete mechanics and optimal control to design interesting, low-energy trajectories from the Earth to the Moon. It should be noted that the optimal trajectories produced here are accurate to second order. For a higher fidelity solution, these solutions could act as initial guesses for a higher-order method.

First, two methods are presented to create initial guess trajectories. Method 1 utilizes the invariant manifolds of the Sun-Earth and Earth-Moon PCR3BP, patching the two 3-body systems together to generate a trajectory that begins at the Earth and is ballistically captured at the Moon. Slight modification of the velocity at the intersection of the manifolds produces a trajectory valid for the 4-body problem. Method 2 exploits the dynamics of the problem, designing the trajectory directly in the 4-body problem, using the invariant manifolds of the PCR3BP as a guide. Next, the initial guess trajectories are optimized using DMOC, removing the impulsive mid-course ΔV and reducing the ΔV necessary to leave Earth orbit and to circularize the orbit at the Moon, following the natural ballistic capture. The combination of invariant manifolds and DMOC successfully produces optimized trajectories from the Earth to the Moon that are competitive

with the literature, requiring up to 19% ΔV less than a Hohmann transfer.

DMOC is also shown to be very effective for the design of low thrust trajectories. After an impulsive ΔV that sets the spacecraft on a trajectory influenced by invariant manifolds, the trajectory requires as little as 17 kg of fuel to reach an elliptical orbit at the Moon.

Chapter 4

Mesh Refinement for DMOC

4.1 Introduction

Optimal trajectory planning in space mission design provides a challenging task regarding accuracy requirements. These problems can be formulated as optimal control problems and solved via direct transcription methods, i.e., the trajectory is approximated by a discrete path using appropriate integration schemes based on a discrete time mesh. To improve the accuracy of the discrete solution, finer time-stepping is required near planets due to the strong influence of gravity, while for a transfer in nearly free space, fewer discretization points are necessary to accurately reflect the dynamics. The design of the variable step size profile, or mesh, is a manual process; hence it is favorable to use an automated process such as mesh refinement. In this chapter, the effect of mesh refinement strategies for the optimal control scheme Discrete Mechanics and Optimal Control (DMOC) is investigated by means of an elliptical orbit transfer and an energy-optimal transfer from the Earth to the Moon.

The primary idea of mesh refinement, as described by Betts [6], is to iteratively add nodes to the mesh to reduce the discretization error. The results presented here exhibit the techniques' ability to improve the accuracy of a solution as well as to improve the qualitative behavior of a solution, specifically the energy behavior, while maintaining a fast computation time.

4.2 Motivation and Problem Formulation

Consider the energy-optimal transfer from the Earth to the Moon presented in Chapter 3. Since the dynamics of the 4-body problem are very nonlinear, a variable step size is necessary. If a sufficiently small, constant step size is used, fully capturing the nonlinear dynamics, a prohibitive number of nodes are necessary. Conversely, restricting the number of nodes, the step size is too large for the dynamics, leading to poor accuracy and convergence problems. A compromise leads to a step size profile that consists of sections of constant step size, as shown in Figure 4.1(b). The design of this step size profile is a manual and tedious process based on experimentation, motivating the need for an automated mesh refinement process.

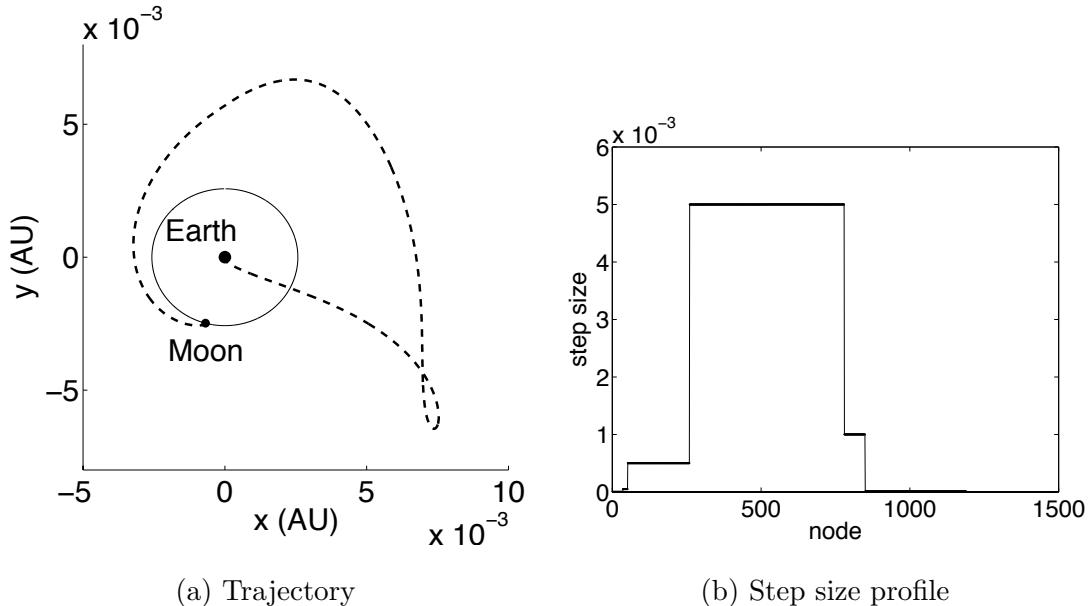


Figure 4.1: Motivating example: optimization of the Shoot the Moon trajectory requires the discretization mesh to be broken into sections of constant step size. (a) Optimized trajectory and (b) step size profile.

4.2.1 Elliptical Orbit Transfer

The mesh refinement algorithms are first demonstrated by means of an optimal elliptical orbit transfer. Consider the 2-body problem for which a spacecraft orbits the Earth in an elliptical orbit. The controlled equations of motion are

$$\ddot{x} = -\frac{x}{(x^2 + y^2)^{\frac{3}{2}}} + \frac{u_x}{m}, \quad (4.1a)$$

$$\ddot{y} = -\frac{y}{(x^2 + y^2)^{\frac{3}{2}}} + \frac{u_y}{m}, \quad (4.1b)$$

where u_x and u_y are the control forces in the x - and y -directions, respectively, and m is the mass of the satellite. The problem is normalized such that GM_E , usually seen in the 2-body problem, equals one. At the periapsis an impulsive ΔV is applied to double the apoapsis of the orbit. DMOC is used to optimize the transfer, allowing continuous control throughout the trajectory and minimizing the control effort. Although this problem may be solved with desired accuracy and reasonable computation time using a constant step size, various step size profiles generated with the different mesh refinement strategies are tested and compared. Figure 4.2(a) shows the initial guess and DMOC optimal trajectory for the 2-body problem with step size $h = 5e-4$ and 4203 total nodes. Figure 4.2(b) shows the optimal control magnitude $\frac{|u|}{m}$. Instead of one impulsive thrust applied at the periapsis, the control is distributed throughout the trajectory, with the maximum control applied at the periapsis. Henceforth the terms $\frac{u_x}{m}$ and $\frac{u_y}{m}$ will simply be denoted by u_x and u_y .

4.3 Traditional Mesh Refinement

Mesh refinement for the optimal control problem is thoroughly described by Betts [5, 7, 6]. Betts' method, summarized here, describes how to refine the discretization mesh used for the solution of an optimal control problem to reduce errors caused by the discretization. Suppose the dynamical system to be optimized consists of

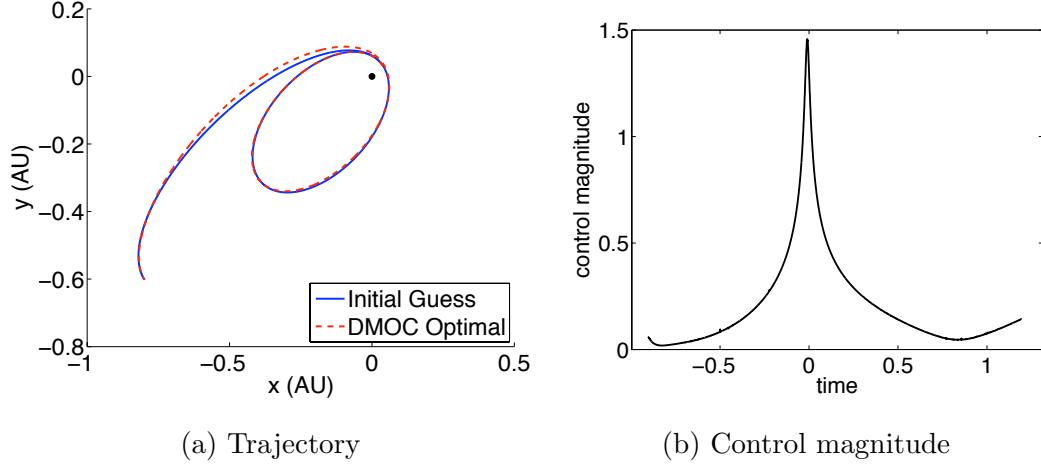


Figure 4.2: Elliptical orbit transfer example problem with (a) initial guess trajectory and DMOC optimized trajectory and (b) the optimal control magnitude.

the differential equation

$$\dot{\mathbf{y}}(t) = \mathbf{f}[\mathbf{y}(t), \mathbf{u}(t), t]. \quad (4.2)$$

The first step is to construct an approximation to the continuous solution using cubic B-splines

$$\mathbf{y}(t) \approx \tilde{\mathbf{y}}(t), \quad (4.3)$$

subject to

$$\tilde{\mathbf{y}}(t_k) = \mathbf{y}_k, \quad (4.4)$$

$$\frac{d}{dt}\tilde{\mathbf{y}}(t_k) = \mathbf{f}_k, \quad (4.5)$$

where $\tilde{\mathbf{y}}(t)$ is the B-spline approximation, $k = 0, \dots, N$, $N + 1$ is the number of discrete points, and $\mathbf{f}_k = \mathbf{f}(\mathbf{y}_k, \mathbf{u}_k, t_k)$. In the same manner, a linear B-spline approximation of the control is formulated,

$$\mathbf{u}(t) \approx \tilde{\mathbf{u}}(t) \quad \text{with} \quad (4.6)$$

$$\tilde{\mathbf{u}}(t_k) = \mathbf{u}_k, \quad k = 0, \dots, N. \quad (4.7)$$

Next, it is necessary to estimate the discretization error for the current mesh. Begin by defining the absolute local error at step k by

$$\eta_{i,k} = \int_{t_k}^{t_{k+1}} |\varepsilon_i(s)| ds, \quad (4.8)$$

where

$$\varepsilon(t) = \dot{\tilde{\mathbf{y}}}(t) - \mathbf{f}[\tilde{\mathbf{y}}(t), \tilde{\mathbf{u}}(t), t]. \quad (4.9)$$

Since $\tilde{\mathbf{y}}$ represents splines for x , y , v_x , and v_y , $\varepsilon(t)$ consists of four components, reading for the 2-body problem

$$\varepsilon_1(t) = \dot{\tilde{x}}(t) - \tilde{v}_x(t), \quad (4.10a)$$

$$\varepsilon_2(t) = \dot{\tilde{y}}(t) - \tilde{v}_y(t), \quad (4.10b)$$

$$\varepsilon_3(t) = \dot{\tilde{v}}_x(t) - \left(-\frac{\tilde{x}(t)}{(\tilde{x}(t)^2 + \tilde{y}(t)^2)^{3/2}} + \frac{\tilde{u}_x(t)}{m} \right), \quad (4.10c)$$

$$\varepsilon_4(t) = \dot{\tilde{v}}_y(t) - \left(-\frac{\tilde{y}(t)}{(\tilde{x}(t)^2 + \tilde{y}(t)^2)^{3/2}} + \frac{\tilde{u}_y(t)}{m} \right), \quad (4.10d)$$

where $\dot{\tilde{x}}(t)$, $\dot{\tilde{y}}(t)$, $\dot{\tilde{v}}_x(t)$, and $\dot{\tilde{v}}_y(t)$ are the time derivatives of the respective splines.

Based on the absolute error, the solution error is approximated by

$$\epsilon_k \approx \max_i \frac{\eta_{i,k}}{(\omega_i + 1)}, \quad (4.11)$$

with the weights

$$\omega_i = \max_{k=1}^N [|\tilde{y}_{i,k}|, |\dot{\tilde{y}}_{i,k}|]. \quad (4.12)$$

Note that the integral in equation (4.8) is evaluated using adaptive Simpson quadrature. Now a new mesh is created based on the solution error, equation (4.11). Nodes are added such that a given time interval is subdivided, so the new mesh still contains the original nodes. Let I_k represent the number of nodes added to any time interval k . Determine the interval α with the maximum error,

$$\epsilon_\alpha = \max_k \epsilon_k, \quad (4.13)$$

and subdivide this interval, adding one node. Next, compute the predicted error based on the new mesh. From Betts [6], the predicted error is

$$\epsilon_{p,k} \approx \max_i \frac{\eta_{i,k}}{(\omega_i + 1)} \left(\frac{1}{1 + I_k} \right)^{p-r_k+1}, \quad (4.14)$$

where p is the order of accuracy of the method ($p = 2$ for this form of DMOC) and r_k is the order of reduction, assumed to be zero for this work because no additional path or state constraints are considered. Now, the maximum error is again computed based on the predicted error, equation (4.14), and a node is added to the corresponding interval. This process is repeated until a termination condition is met.

The mesh refinement performed for this work follows Betts' procedure excluding the termination condition. Betts [6] suggests a termination condition that depends upon a combination of the predicted error, the total nodes added, and the nodes added to a single interval. Here the process terminates when the discretization mesh contains a maximum number of total nodes. After a new mesh is generated, an initial guess with the new discretization mesh is optimized using DMOC. The new optimal result serves as the starting point for a new round of mesh refinement. This iterative process continues until either the error has been sufficiently reduced or until the total number of nodes becomes computationally cumbersome. Note that the weight, ω_i , is calculated only on the first iteration of the mesh refinement process and is then used for subsequent iterations.

4.3.1 Energy Considerations for Mesh Refinement

The classic form of mesh refinement is based entirely on the solution error. It is desirable to formulate a metric by which the qualitative behavior of the solution, such as energy behavior, can be judged. As mentioned before, the use of discrete variational principles for the approximation of the continuous trajectories (as for DMOC) leads to good energy behavior using a constant time step. However, arbitrarily adding nodes to the discretization mesh may destroy this property.

Since accurate energy behavior is the goal, it is important to examine how the mesh refinement affects the energy, and thus, construct meshes in such a way that the error in the energy evolution is reduced.

In the presence of control, the energy injected into the system at time t is $E_u(t) = \int_0^t \dot{q}(\tau) \cdot u(\tau) d\tau$, leading to energy change. Thus the controlled energy $E_c(t) = E(t) - E_u(t)$, the difference of the total energy and the energy induced by external control forces, should be preserved. The energy difference,

$$\Delta E_c = E_c(t) - E_0, \quad (4.15)$$

where E_0 is the energy of the initial condition used to create the initial guess, and

$$E_c(t) = \frac{1}{2}(\dot{x}^2 + \dot{y}^2) - \frac{1}{\sqrt{x^2 + y^2}} - \int_{t_0}^t (\dot{x}(\tau)u_x(\tau) + \dot{y}(\tau)u_y(\tau))d\tau, \quad (4.16)$$

should be zero for all time, t , for the continuous system. The discrete version of equation (4.16) is computed using the discrete variables x_k , y_k , $v_{x,k}$, and $v_{y,k}$,

$$E_{c,k} = \frac{1}{2}(v_{x,k}^2 + v_{y,k}^2) - \frac{1}{\sqrt{x_k^2 + y_k^2}} - \sum_{i=0}^k h_i (v_{x,i}u_{x,i} + v_{y,i}u_{y,i}), \quad (4.17)$$

and $k = 0, \dots, N$. DMOC uses state variables x_k and y_k only, so the velocities, $v_{x,k}$ and $v_{y,k}$ are computed using the discrete Legendre transform, equation (2.26). Figure 4.3 shows the energy difference, the discrete version of equation (4.15), for the optimal solution of the 2-body problem shown in Figure 4.2.

4.4 Mesh Refinement Strategies

Mesh refinement is approached using three different strategies. The first strategy follows the procedure introduced by Betts [6] and adds nodes to decrease discretization errors in the solution.

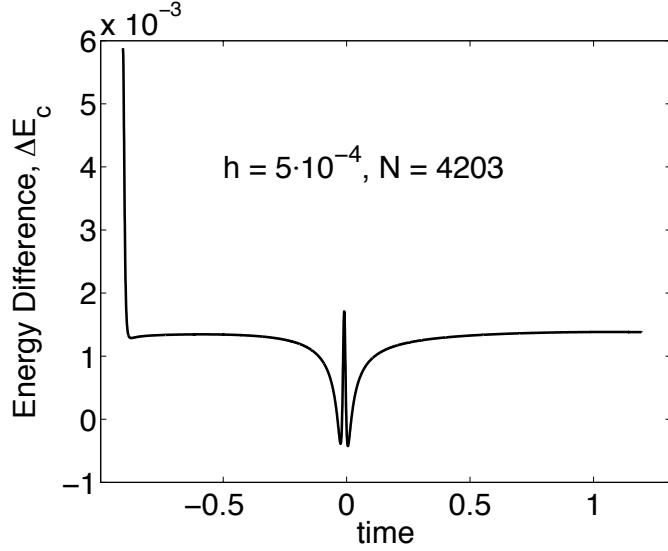


Figure 4.3: Energy difference ΔE_c : The energy difference increases in regions where the potential is greatest, when the spacecraft is closest to the Earth, indicating that a finer discretization mesh is required in that region.

Energy-Based Mesh Refinement

The second strategy also follows Betts' basic idea, but instead of reducing the solution error, the goal is to minimize the energy difference given by equation (4.15). To this aim, an expression is formulated for the discretization error in the energy. The energy error is defined by

$$\epsilon_{E,k} = \int_{t_k}^{t_{k+1}} |\varepsilon_E(s)| ds, \quad (4.18)$$

where

$$\varepsilon_E(t) = \tilde{E}_c(t) - E_0. \quad (4.19)$$

$\tilde{E}_c(t)$ approximates $E_c(t)$ using linear B-splines that fulfill the condition

$$\tilde{E}_c(t_k) = E_{c,k}, \quad (4.20)$$

where $E_{c,k}$ is computed according to equation (4.17). The mesh refinement then proceeds as described before with the maximum energy error

$$\epsilon_{E,\alpha} = \max_k \epsilon_{E,k}, \quad (4.21)$$

and the predicted error

$$\epsilon_{E_p,k} \approx \epsilon_{E,k} \left(\frac{1}{1 + I_k} \right)^{p-r_k+1}, \quad (4.22)$$

where $p = 2$ is the order of the optimization scheme and $r_k = 0$ is the order of reduction. As before, a node is added to the interval with the maximum error, and the process repeats until the mesh contains a maximum number of nodes.

Time Adaption

The third strategy is based on the time adaptive variational integrators described in §2.2.3. This integrator aims to maintain the good energy behavior of the discrete solution also for non-constant time steps. The approach directly considers the system's dynamics for the construction of a discrete adaptive time grid. In particular, the time update is determined such that

$$t_{k+1} = t_k + h\sigma(q_k, q_{k+1}), \quad (4.23)$$

where h is the initial step size and σ is a function of the dynamics, specifically the potential and/or energy. It follows that the variable step size is given by

$$h_k = h\sigma(q_k, q_{k+1}). \quad (4.24)$$

Two update strategies are tested in this work:

$$\sigma_1(q_k, q_{k+1}) = \frac{1}{\sqrt{E_0 - W\left(\frac{q_k + q_{k+1}}{2}\right) + \nu}} \quad (\text{equispaced poses}) \quad (4.25)$$

$$\sigma_2(q_k, q_{k+1}) = \frac{1}{\|\nabla W(q_k) + \nabla W(q_{k+1}) + \nu\|_2} \quad (\text{acceleration based}) \quad (4.26)$$

where W is the potential, E_0 is the initial energy, and ν is a small constant that prevents division by zero. Employing time adaption in this manner ensures that the nodes are arranged according to the dynamics. If time is adapted based on equispaced poses, given by equation (4.25) and denoted by σ_1 , time evolves such that the points are equally spaced in x - y space. If time is adapted according to acceleration, given by equation (4.26) and denoted by σ_2 , there are more nodes when the dynamics change quickly and fewer nodes when the dynamics change slowly. Unlike energy- or solution-based mesh refinement, additional nodes are not added iteratively; the mesh is generated based on the dynamics and h only. Thus each solution stands alone and does not depend on the previous solutions.

These time adaption strategies are first presented for the 2-body problem. Figure 4.4(a) shows the potential and node placement for a constant step size of $h = 0.01$ and 255 nodes. There are a dearth of nodes in the regions where the potential is strongest. Figures 4.4(b) and 4.4(c) show the potential for time adaption according to equispaced poses and acceleration, respectively. For the acceleration-based time adaption, the nodes are concentrated where the potential is strongest, capturing the dynamics more accurately with fewer total nodes than the constant time step version.

4.5 Mesh Refinement for Elliptical Orbit Transfer

Using each strategy, nodes are added to the mesh, approximately 500–1000 per iteration. In this way, a variety of step size profiles are created for each strategy. The step size profiles for energy- and solution-based mesh refinement (also denoted by MR) are not immediately ready for use with DMOC. Due to convergence

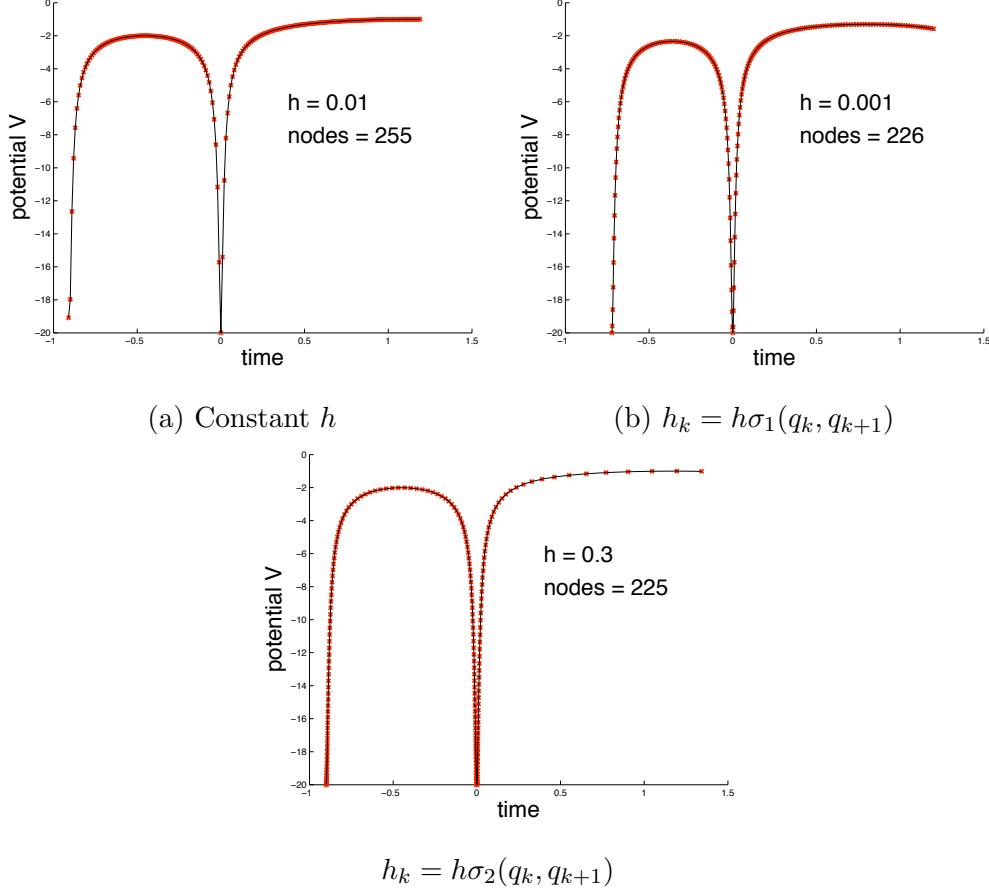


Figure 4.4: Potential energy for the 2-body elliptical orbit transfer. Node placement along potential energy for (a) constant step size, (b) time adaption with equispaced poses, σ_1 , and (c) acceleration-based time adaption, σ_2 . Time adaption leads to better node placement.

problems, any spikes in the step size profiles are removed.

Representative profiles are shown for all mesh refinement strategies in Figure 4.5. The maximum step size for the energy- and solution-based mesh refinement is 5e-3 for the first iteration while the time adapted schemes (also denoted by TA) generate much larger maximum step sizes at nearly 0.04 for the acceleration-based time adaption and approximately 0.01 for equispaced poses. Also, the time adapted step size profiles are smooth and continuous and require no manipulation.

Each discretization mesh is used to create an initial guess for DMOC optimization. The optimization constraints include the discrete Euler-Lagrange equations,

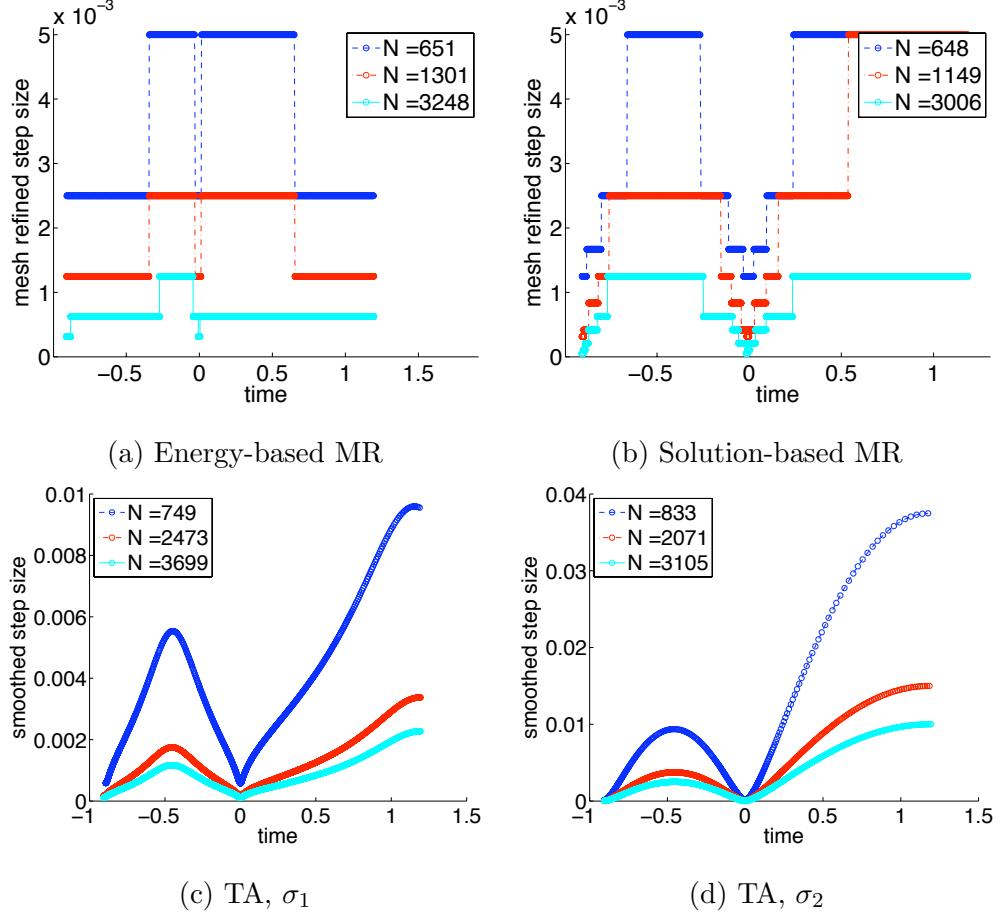


Figure 4.5: Elliptical orbit transfer step size profiles for (a) energy-based MR, (b) solution-based MR, (c) time adaption with σ_1 , and (d) time adaption with σ_2 .

equation (2.58e), which enforce the dynamics, and the initial and final position and momentum, equations (2.58d) and (2.58f), are held fixed. The objective function corresponds to the control effort,

$$J_d = \sum_k h_k(u_{x,k}^2 + u_{y,k}^2), \quad (4.27)$$

and has to be minimized.

The resulting discretization energy error, equation (4.18), is shown for each strategy in Figure 4.6. As expected, the energy error consistently improves for the energy-based mesh refinement. However, the energy error for the solution-

based mesh refinement is even smaller. Two factors contribute to this result. The energy-based mesh refinement adds nodes uniformly at the beginning, middle, and end of the trajectory, compared with the graduated changes in step size that occur for solution-based mesh refinement. Also the minimum step size is not nearly as small compared with an equivalent number of nodes in a solution-based mesh. The energy error for both time adaption schemes is smooth and improves quickly with the addition of nodes.

The solution error for each strategy, computed according to Betts, equation (4.11), is shown in Figure 4.7. For all schemes except the acceleration-based time adaption, the largest errors are concentrated at the beginning and middle of the trajectory; these points occur when the spacecraft is closest to the Earth and subject to the greatest potential force. The solution error for the acceleration-based time adaption is more evenly distributed and grows toward the end of the trajectory.

To put the results of each scheme into context with the others, log-log plots of the maximum norm of the energy error and the solution error versus the total number of nodes, respectively, are shown in Figure 4.8(a) and Figure 4.8(b). The plots demonstrate the convergence rates, clearly displaying how quickly the error improves with the addition of nodes to the mesh. DMOC optimal solutions with constant step size attain nearly third-order convergence, as expected for the local error using the midpoint rule approximation (black line) for both energy error and solution error.

Figure 4.8(a) shows that the energy error decreases with nearly third order convergence for the energy-based mesh refinement (blue line). The solution-based mesh refinement (green) results in smaller energy errors with the same number of nodes and a similar convergence rate of approximately 2.8. For both time adaption schemes, equispaced poses (red line) and acceleration-based (magenta line), the energy error decreases with a convergence rate of 2.6 and 2.1, respectively, and with error magnitude less than those of the constant time step solutions. With larger total number of nodes, the energy-based mesh refinement performs slightly better than the acceleration-based time adaption. Figure 4.8(b) shows

that the best solution error behavior occurs for solution-based mesh refinement. For small number of total nodes, the time adaption schemes produce smaller errors than the solution-based mesh refinement, but the faster convergence rate of the solution-based mesh refinement quickly leads to smaller errors as nodes are added. The energy-based mesh refinement shows an error behavior rather similar to the constant time stepping solutions with a slightly faster convergence rate. Overall, the solution-based mesh refinement generates the best results for both energy error and solution error, even though the solutions generated with time adaption look promising.

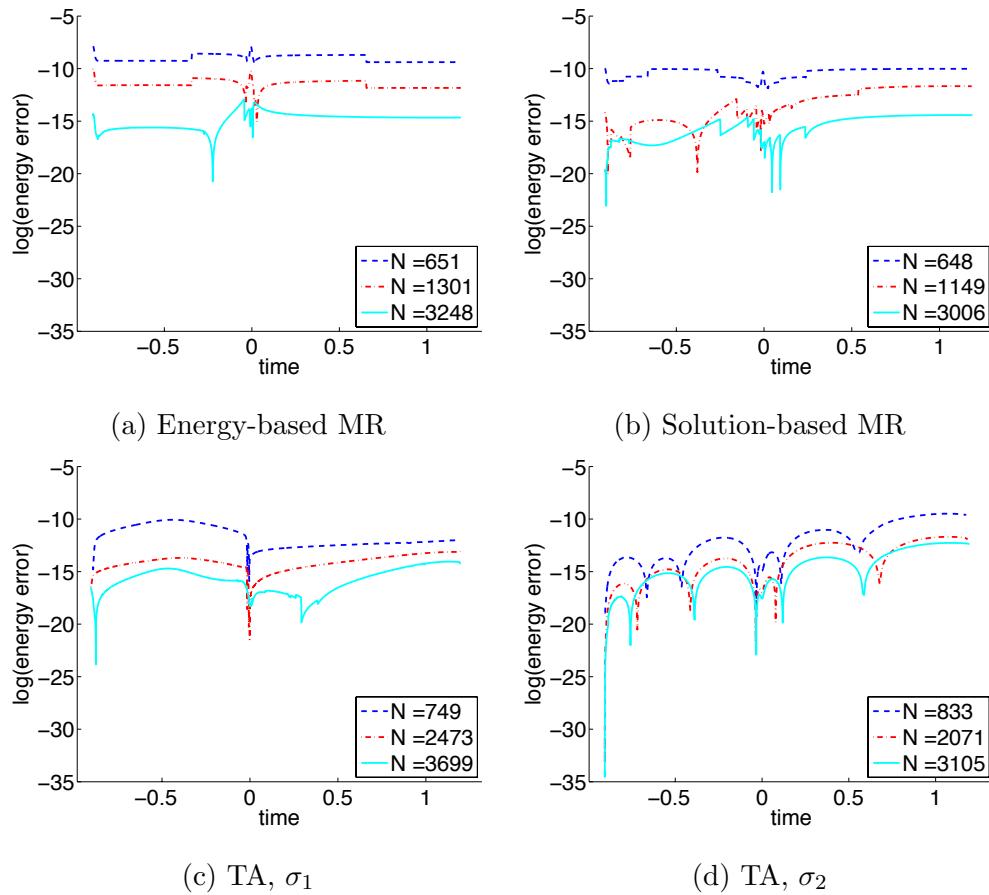


Figure 4.6: Elliptical orbit transfer discretization energy error for (a) energy-based MR, (b) solution-based MR, (c) time adaption with σ_1 , and (d) time adaption with σ_2 .

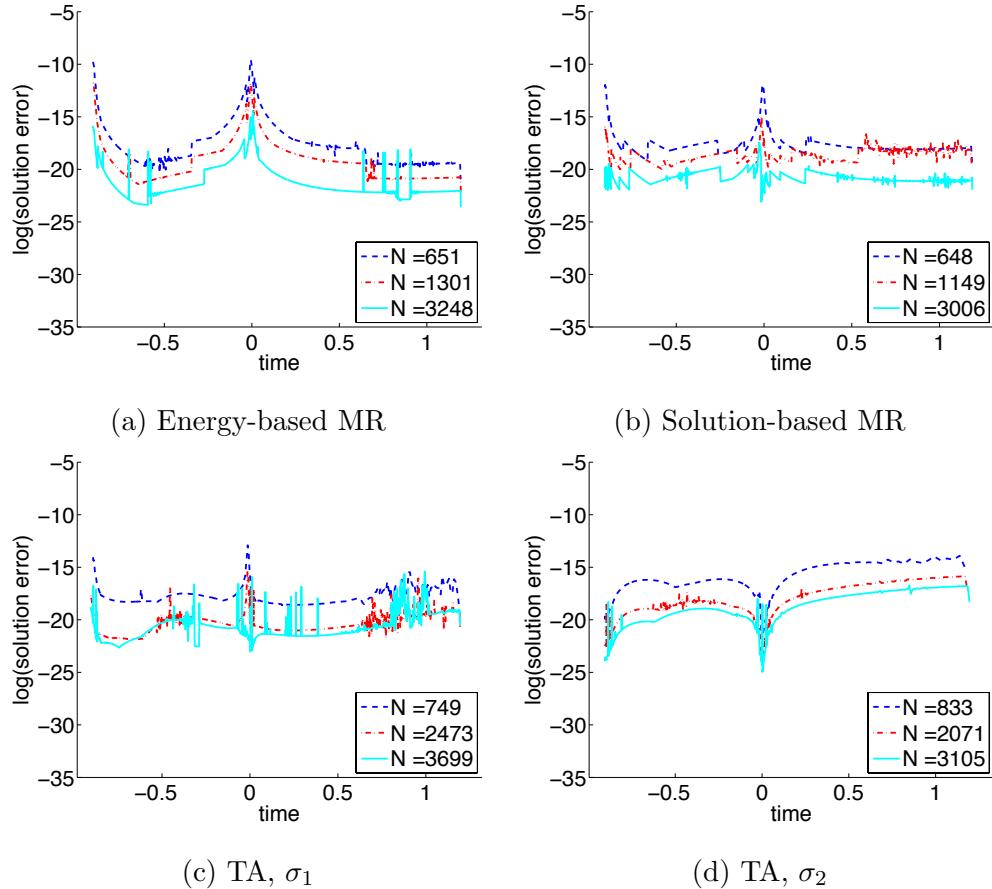


Figure 4.7: Elliptical orbit transfer discretization solution error for (a) energy-based MR, (b) solution-based MR, (c) time adaption with σ_1 , and (d) time adaption with σ_2 .

4.6 Mesh Refinement for Shoot the Moon

Now that the success of mesh refinement has been demonstrated for the elliptical orbit transfer, the same ideas are tested on the motivating problem, Shoot the Moon. Solution-based mesh refinement proceeds as before using the 4-body dynamics. When considering energy-based mesh refinement, the problem is more complicated because it is time dependent; the energy is not preserved. A new metric for the energy error must be developed. In a time-independent problem like the 2-body problem, it holds that $\frac{dE_c}{dt} = 0 \Rightarrow E_c(t) - E_0 = 0$. Since $\frac{dE_c}{dt} \neq 0$ for the Shoot the Moon problem, it is useful to consider errors in the energy derivative.

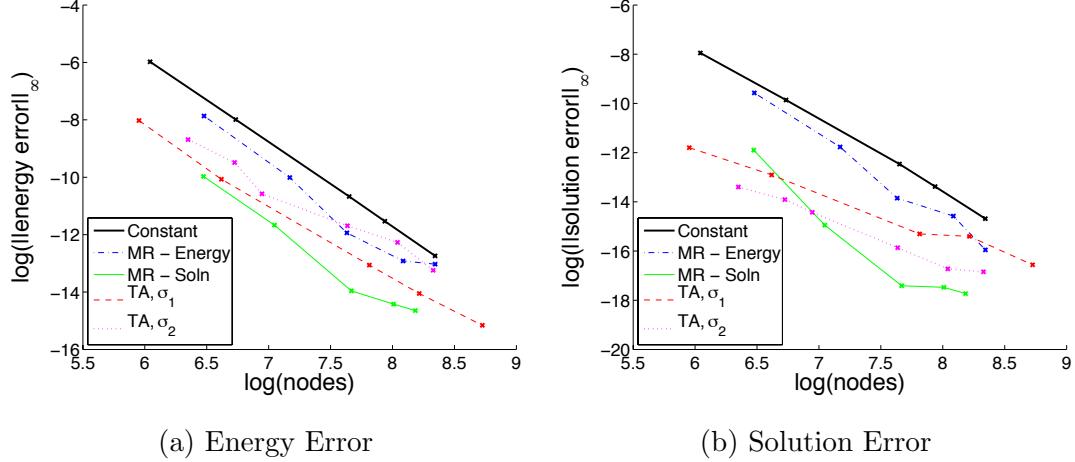


Figure 4.8: Convergence rate of DMOC optimized solution for elliptical orbit transfer when considering the (a) energy error and (b) solution error. The log of the maximum norm of the error is plotted versus the log of the total number of nodes.

The controlled energy evolves according to the equation,

$$E_c = \frac{1}{2}(\dot{x}^2 + \dot{y}^2) - \frac{1}{2}((x + (1 - \mu))^2 + y^2) - \frac{\mu_S}{r_S} - \frac{\mu_E}{r_E} - \frac{\mu_M}{r_M} - \int_{t_0}^t (\dot{x}(\tau)u_x(\tau) + \dot{y}(\tau)u_y(\tau))d\tau, \quad (4.28)$$

where $(1 - \mu)$ shifts the problem so that the Earth is at the origin, μ_S , μ_E , and μ_M are the normalized masses of the Sun, Earth, and Moon, respectively, and r_S , r_E , and r_M are the distances of the spacecraft from the center of each body. For the time derivative of the controlled energy, all terms cancel except for those relating to the time-dependent portion of the problem, the position of the Moon,

$$\begin{aligned} \frac{dE_c}{dt} &= \frac{d}{dt} \left(\frac{-\mu_M}{r_M(t)} \right) = \\ &- \frac{\mu_M a_M \omega_M}{r_M^3} (\cos(\theta_M(t)) \cdot (y(t) - y_M(t)) - \sin(\theta_M(t)) \cdot (x(t) - x_M(t))), \end{aligned} \quad (4.29)$$

where

$$r_M(t) = \sqrt{(x(t) - x_M(t))^2 + (y(t) - y_M(t))^2}, \quad (4.30a)$$

$$x_M(t) = a_M \cos \theta_M(t), \quad (4.30b)$$

$$y_M(t) = a_M \sin \theta_M(t), \quad (4.30c)$$

$$\theta_M(t) = \omega_M t + \theta_{M_0}, \quad (4.30d)$$

and a_M is the normalized radius of the Moon's circularized orbit, θ_{M_0} is the initial angle of the Moon with respect to the Sun-Earth line, and ω_M is the normalized rate of rotation of the Moon. For ease of notation, let the right-hand side of equation (4.29) be denoted by $\Psi(t)$. Thus the difference of the time derivative of the controlled energy and the time-dependent $\Psi(t)$,

$$\Delta \dot{E}_c = \frac{d}{dt} E_c(t) - \Psi(t), \quad (4.31)$$

should be zero for all time t . The discrete version of $dE_c(t) := \frac{d}{dt} E_c(t)$ on each interval $[t_k, t_{k+1}]$ is determined using finite differences of the discrete controlled energy values $E_{c,k}$, reading

$$dE_{c,k} = \frac{E_{c,k+1} - E_{c,k}}{t_{k+1} - t_k}, \quad (4.32)$$

for $k = 0, \dots, N - 1$. The discrete version of $\Psi(t)$ is given by

$$\Psi_k = -\frac{\mu_M a_M \omega_M}{r_M(t_k)^3} (\cos(\theta_M(t_k)) (y_k - y_M(t_k)) - \sin(\theta_M(t_k)) (x_k - x_M(t_k))), \quad (4.33)$$

for $k = 0, \dots, N - 1$. In this way, the energy error is given as in equation (4.18) where ϵ_E is replaced with

$$\varepsilon_{\dot{E}}(t) = \widetilde{dE}_c(t) - \tilde{\Psi}(t), \quad (4.34)$$

where the linear splines \tilde{dE}_c and $\tilde{\Psi}$ fulfill $\tilde{dE}_c(t_k) = dE_{c,k}$ and $\tilde{\Psi}(t_k) = \Psi_k$, respectively. The mesh refinement proceeds as for the 2-body problem.

Note that optimization of the Shoot the Moon problem leads to a solution with zero control because the boundary conditions are adjusted until a free transfer is found. Therefore, the effect of control terms in equation (4.28) are henceforth neglected. Figure 4.9 displays the energy, equation (4.28), for the Shoot the Moon trajectory. Far from the Moon, the energy oscillates around the constant energy value of the Earth-Sun-spacecraft 3-body problem. The oscillation corresponds to the time-dependent perturbation from the Moon. As the spacecraft approaches the Moon, the Moon's potential force dominates, leading to the rapid change in energy at the end.

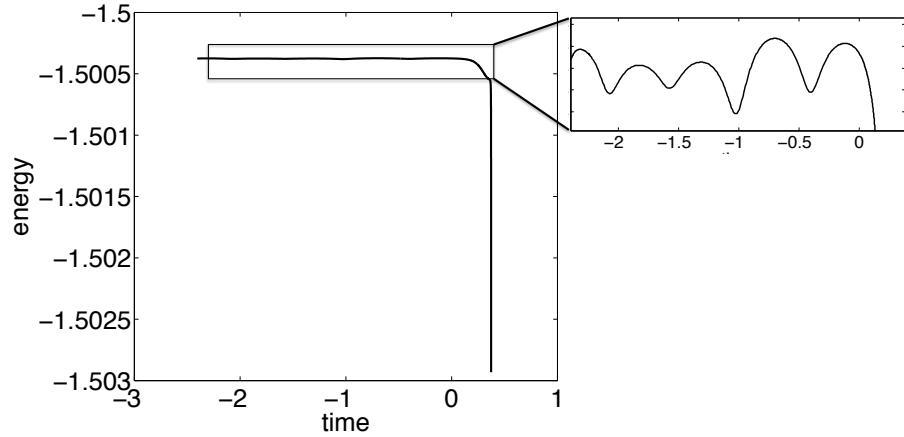


Figure 4.9: The energy for the Shoot the Moon problem is time-dependent and evolves based on the constant energy value of the Sun-Earth-spacecraft planar circular restricted 3-body problem and the time-dependent motion of the Moon.

Time adaption also works as before; however, only the acceleration-based σ , denoted σ_2 , is used. Due to the non-linearity of this problem, equispaced poses are not appropriate. Such time adaption places too many nodes in the mesh.

Figure 4.10 shows the representative step size profiles for energy-based mesh refinement, solution-based mesh refinement, and acceleration-based time adaption. The energy- and solution-based mesh refinement schemes begin by adding nodes to an optimized trajectory with maximum step size $h = 0.01$. As shown in the

plots, energy- and solution-based mesh refinement primarily add nodes at the beginning and end of the trajectory, near the Earth and Moon, respectively. The acceleration-based time adaption generates smooth profiles with a maximum step size in the region furthest from the Earth and the Moon and very small step sizes near the bodies. Also, notice that the maximum step size is larger compared with the energy- and solution-based profiles.

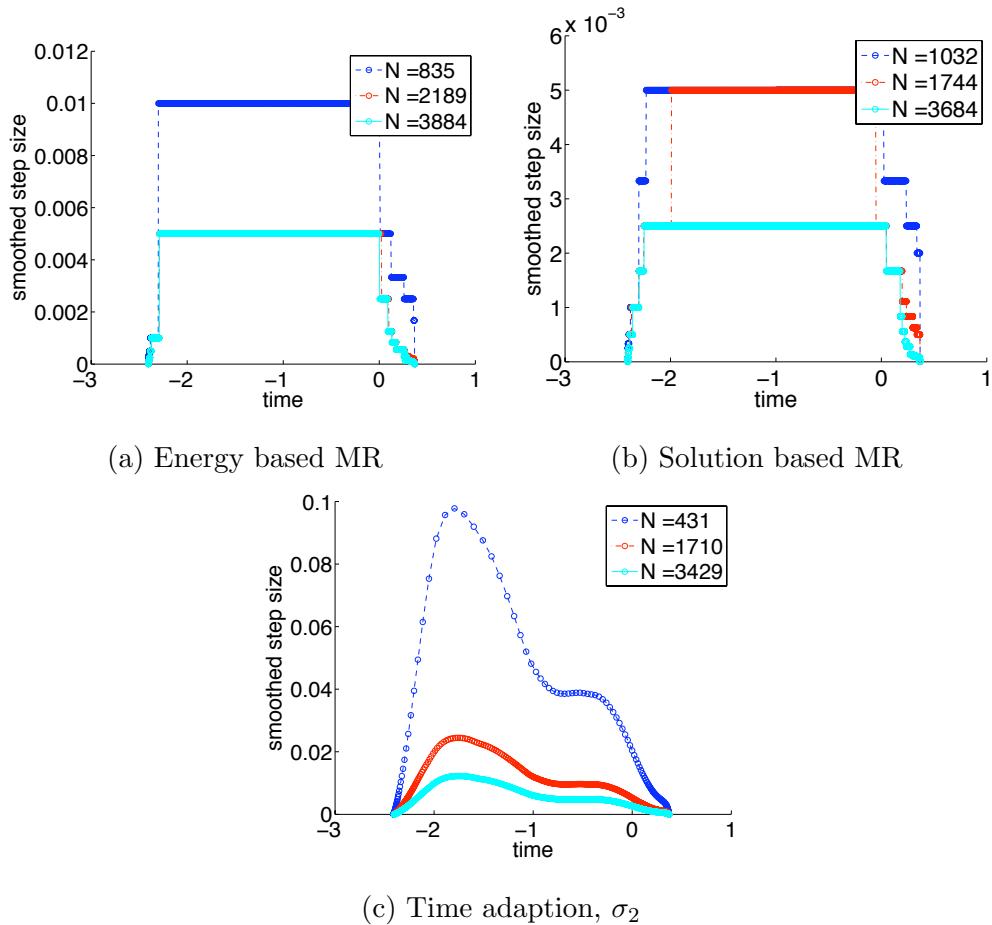


Figure 4.10: Shoot the Moon step size profiles for (a) energy based MR, (b) solution based MR, and (c) time adaption with σ_2 .

Figure 4.11 shows the energy error for each mesh refinement strategy. For all strategies, the discretization energy error oscillates around zero with a large spike near the Moon (end of trajectory). The solution error is shown in Figure 4.12. The solution error looks very similar for both energy- and solution-based mesh re-

finement. The acceleration-based time adaption looks a bit different, with smaller magnitude errors near the Earth.

The log-log convergence plots fully demonstrate the behavior of the different strategies for Shoot the Moon. As shown in Figure 4.13(a), the maximum norm of energy error for acceleration-based time adaption is significantly smaller than for the energy- or solution-based mesh refinement. The energy-based mesh refinement line has a faster convergence rate, but the magnitude is higher. While the acceleration-based scheme converges at a slower rate of 2.2, the rate is consistent, and the value of the error is orders of magnitude smaller.

The trends are not as clear when examining the solution error shown in Figure 4.13(b). For the first several points, the solution error is significantly better for the acceleration-based scheme even though the energy-based mesh refinement converges with a much faster rate. In the middle, the convergence lines cross: the energy-based convergence rate levels off, but the magnitude of error is still less than that of acceleration-based time adaption. Even though the solution-based mesh refinement leads to a convergence rate similar to that of acceleration-based time adaption, the magnitude of error is higher. The convergence rates for all solutions are displayed in Table 4.1.

Based on these results, it appears that both energy-based mesh refinement and acceleration-based time adaption perform better than solution-based mesh refinement for this time-dependent problem. Overall, the acceleration-based time adaption produces the best results for Shoot the Moon regarding energy and solution error. Additionally, the time adaption approach is much easier to use and requires less computation time for convergence to optimal solutions.

4.7 Conclusion

This chapter examines the effect of different mesh refinement schemes on the energy error and the solution error of optimal solutions produced using DMOC. First, four different approaches are tested on an elliptical orbit transfer. The energy-

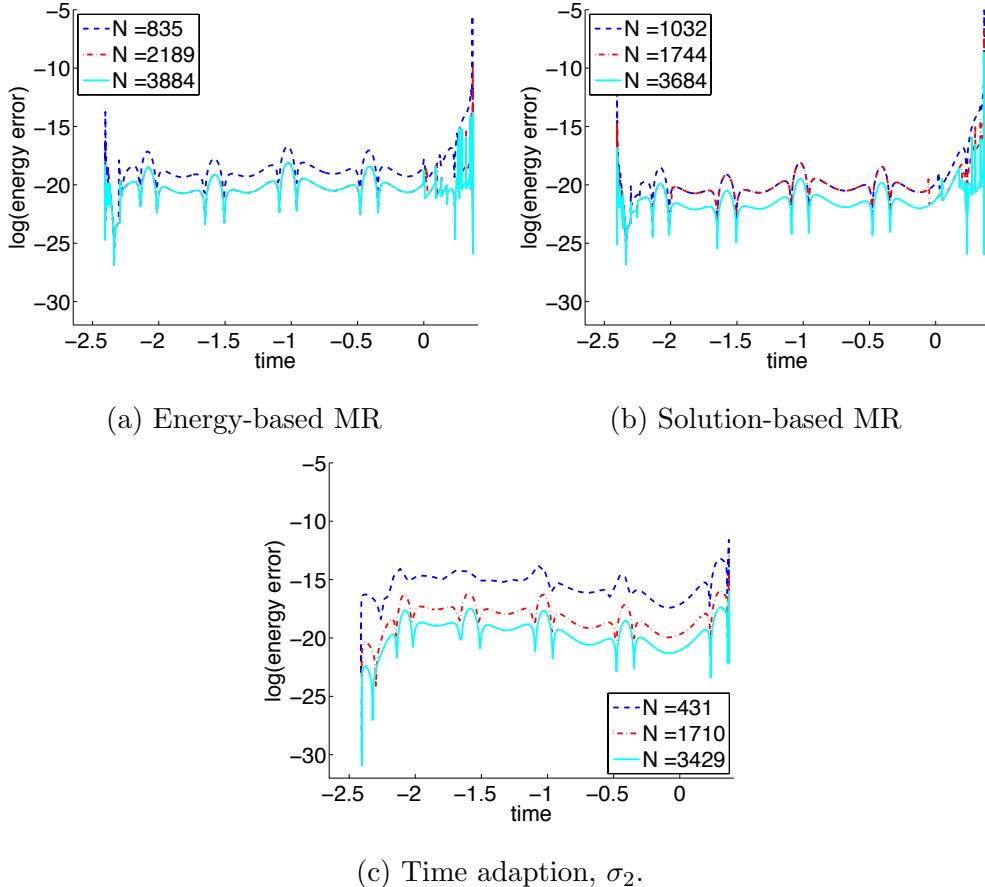


Figure 4.11: Shoot the Moon discretization energy error: the energy error decreases with increasing nodes for (a) energy-based MR, (b) solution-based MR, and (c) time adaption with σ_2 .

and the solution-based approaches add nodes to the discretization mesh to reduce the energy error and the solution error, respectively. Additionally, time adaption based on equispaced poses and acceleration lead to continuous step size profiles. Overall, solution-based mesh refinement leads to the best energy error and solution error results for the 2-body problem with the time adaption schemes producing results that are nearly as good.

Mesh refinement is also explored for the Shoot the Moon problem. The energy- and solution-based schemes are tested again, as well as the acceleration-based time adaption. It should be noted that all schemes lead to meshes that accurately and efficiently produce optimal solutions using DMOC. While the convergence rates

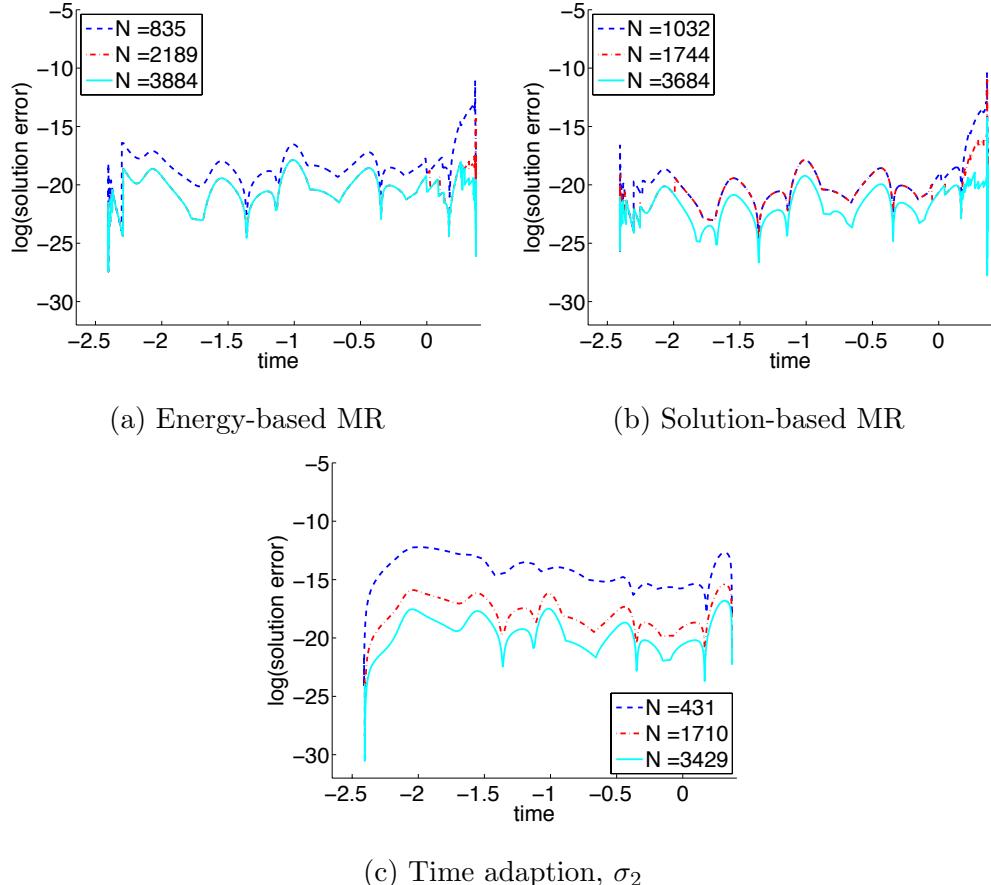


Figure 4.12: Shoot the Moon discretization solution error: the solution error decreases with increasing nodes for (a) energy-based MR, (b) solution-based MR, and (c) time adaption with σ_1 .

may be faster for the energy- and solution-based schemes, the acceleration-based scheme leads to the best energy and solution error that improves consistently with the addition of more nodes. Based on the competitive results and ease of use, time adaption shows the most promise. It is the easiest to use and requires no iteration from one solution to the next; each solution stands on its own and is not influenced by the results of previous solutions. Also, the smooth step size profile leads to fast convergence of the optimizer. Overall, the success of time adaption motivates exploration of a time adaptive form of DMOC. Time adaption schemes as described here have already been used for time adaptive variational integrators [24]. Since DMOC is based on the same discretization schemes as variational integrators,

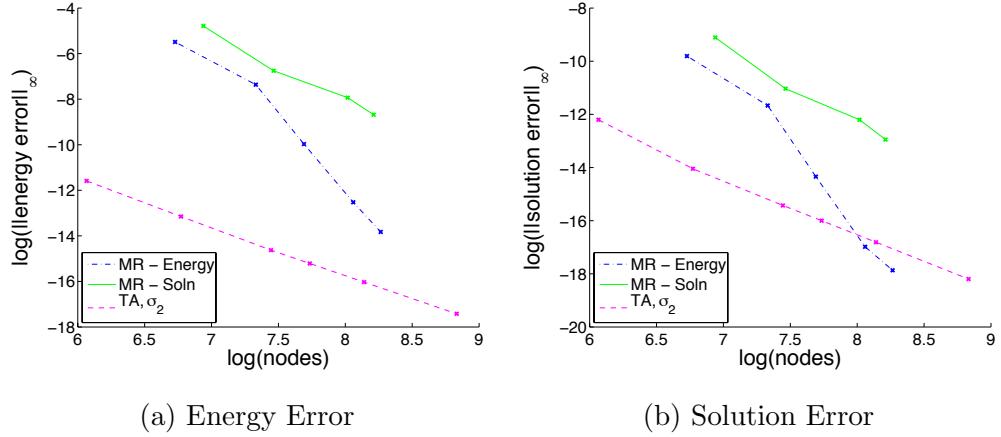


Figure 4.13: Shoot the Moon convergence rate of DMOC optimized solution when considering the (a) energy error and (b) solution error. The log of the maximum norm of the energy error versus the log of the total number of nodes.

Table 4.1: Convergence Rates

	Energy Error	Solution Error
Elliptical orbit transfer, Constant step size	2.9	2.9
Elliptical orbit transfer, Energy MR	2.9	3.7
Elliptical orbit transfer, Soln MR	2.8	3.3
Elliptical orbit transfer, TA σ_1	2.1	1.2
Elliptical orbit transfer, TA σ_2	2.6	1.7
Shoot the Moon, Energy MR	5.3	5.1
Shoot the Moon, Soln MR	2.9	2.9
Shoot the Moon, TA σ_2	2.2	2.3

a time adaptive form of DMOC can be derived using similar strategies. Such an optimizer should efficiently produce optimal solutions with excellent energy behavior.

Chapter 5

Time Adaptive DMOC

5.1 Introduction

It is impractical to optimize nonlinear problems, particularly those in trajectory design, using DMOC with a constant step size. Different strategies can be employed to circumvent this issue such as using sections of constant step size, as described in Chapter 3, or using mesh refinement to design the step size profile, described in Chapter 4. However, it is desirable to develop a form of DMOC that allows for variable step size while maintaining the convergence and energy properties expected for DMOC. Furthermore, full time adaption should allow for the step size, determined by the dynamics, to be updated during the optimization. Time adaptive DMOC builds on the time adaption strategy developed for variational integrators described by Kharevych in [24]. However, the transition from time adaptive variational integrators to time adaptive DMOC is not as obvious as it may initially seem.

This chapter begins by describing Lagrangian mechanics with time adaption, setting the stage for a clear derivation and analysis of time adaptive variational integrators. The most obvious, and incorrect, attempt at translating time adaptive variational integrators to DMOC is presented to demonstrate why time adaptive DMOC requires different considerations than variational integrators. Next, a correct method for approaching time adaption for the optimal control problem is

described. The method is validated with a simple example before proceeding with more interesting examples, including the elliptical orbit transfer problem presented in Chapter 4 and the reconfiguration of a cubesat.

5.2 Lagrangian Mechanics with Time Adaption

Before the derivation of time adaptive DMOC is presented, it is necessary to fully understand the derivation of variational integrators with time adaption. First consider a continuous system with configuration variables and time as functions of the parameter τ . This idea originates with the development of variational integrators for collision by Fetecau, Marsden, Ortiz, and West [15].

Following their notation, it is necessary to present some of their definitions. Consider a configuration manifold Q , and let the path space be defined as

$$\mathcal{M} = \mathcal{T} \times \mathcal{Q}([0, \tau_F], Q),$$

where

$$\begin{aligned}\mathcal{T} &= \{c_t \in C^\infty([0, \tau_F], \mathbb{R}) | c'_t > 0 \text{ in } [0, \tau_F]\}, \\ \mathcal{Q}([0, \tau_F], Q) &= \{c_q : [0, \tau_F] \rightarrow Q | c_q \text{ is a } C^2 \text{ curve}\}.\end{aligned}$$

A path $c \in \mathcal{M}$ is a pair $c = (c_t, c_q)$. Thus, given a path defined in this way, the associated path $q : [c_t(0), c_t(\tau_F)] \rightarrow Q$ is given by

$$q(t) = c_q(c_t^{-1}(t)). \quad (5.1)$$

Equivalently, $c_q(\tau) = q(t)$, where τ is a time parameter. It is useful to note that c'_q denotes derivatives of c_q with respect to τ and \dot{q} denotes derivatives of q with

respect to t . With this in mind,

$$c'_q = \frac{dq}{d\tau}, \quad (5.2)$$

$$c'_t = \frac{dt}{d\tau}, \quad (5.3)$$

$$\dot{q} = \frac{c'_q}{c'_t}. \quad (5.4)$$

The action map $\mathfrak{G} : \mathcal{M} \rightarrow \mathbb{R}$ for the Lagrangian system in this new setting is given by

$$\mathfrak{G}(c_t, c_q) = \int_0^{\tau_f} L \left(c_q(\tau), \frac{c'_q(\tau)}{c'_t(\tau)} \right) c'_t(\tau) d\tau. \quad (5.5)$$

The action map for the associated curve q may be written

$$\mathfrak{G}(q) = \int_{c_t(0)}^{c_t(\tau_f)} L(q(s), \dot{q}(s)) ds, \quad (5.6)$$

where $s = c_t(\tau)$ is the change of coordinates.

5.2.1 Continuous System with Time Adaption

Now, consider a Lagrangian system with time adaption. From §2.2.3, a time adapted Lagrangian is given by

$$\tilde{L}(q(t), \dot{q}(t)) = \sigma(q) (L(q(t), \dot{q}(t)) + H_0), \quad (5.7)$$

where the time adaption will be enforced such that

$$c'_t = \frac{dt}{d\tau} = \sigma(q). \quad (5.8)$$

\tilde{L} may be transformed into τ coordinates by

$$\tilde{L} \left(c_q(\tau), \frac{c'_q(\tau)}{c'_t(\tau)} \right) = \sigma(c_q(\tau)) \left(L \left(c_q(\tau), \frac{c'_q(\tau)}{c'_t(\tau)} \right) + H_0 \right). \quad (5.9)$$

For ease of notation, (τ) will not be included henceforth, but it is implied. Equation (5.9) may equivalently be written

$$\tilde{L} \left(c_q, \frac{c'_q}{c'_t} \right) = \left(L \left(c_q, \frac{c'_q}{c'_t} \right) + H_0 \right) \cdot c'_t + c_\lambda \cdot (c'_t - \sigma(c_q)), \quad (5.10)$$

where $c_\lambda(\tau) = \lambda(t)$ is a Lagrange multiplier that enforces the time adaption constraint, $c'_t = \sigma(c_q)$. Therefore, the action map of the time adapted system is given by

$$\mathfrak{G}(c_t, c_q, c_\lambda) = \int_0^{\tau_f} \left(L \left(c_q(\tau), \frac{c'_q(\tau)}{c'_t(\tau)} \right) + H_0 \right) \cdot c'_t + c_\lambda \cdot (c'_t - \sigma(c_q)) d\tau, \quad (5.11)$$

where the path c is now represented by $c = (c_t, c_q, c_\lambda)$. Variations of the action map with respect to the path gives

$$\begin{aligned} \delta \int_0^{\tau_f} & \left(L \left(c_q(\tau), \frac{c'_q(\tau)}{c'_t(\tau)} \right) + H_0 \right) \cdot c'_t + c_\lambda \cdot (c'_t - \sigma(c_q)) d\tau = \\ & \int_0^{\tau_f} \left(\left[\frac{\partial L}{\partial q} \cdot \delta c_q + \frac{\partial L}{\partial \dot{q}} \left(\frac{\delta c'_q}{c'_t} - \frac{c'_q \delta c'_t}{(c'_t)^2} \right) \right] c'_t + c_\lambda \left(\delta c'_t - \frac{\partial \sigma}{\partial q} \cdot \delta c_q \right) \right. \\ & \left. + (L + H_0) \delta c'_t + (c'_t - \sigma(c_q)) \delta c_\lambda \right) d\tau. \end{aligned}$$

Multiple applications of integration by parts and the requirement that variations vanish on the endpoints generates the equations of motion,

$$\frac{d}{d\tau} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} c'_t + c_\lambda \frac{\partial \sigma}{\partial q} = 0 \quad (5.12)$$

$$\frac{d}{d\tau} \left(\frac{\partial L}{\partial \dot{q}} \frac{c'_q}{c'_t} - L - H_0 - c_\lambda \right) = 0 \quad (5.13)$$

$$c'_t - \sigma(c_q) = 0. \quad (5.14)$$

Incorporating equation (5.14) into equation (5.12), recognizing that $\frac{\partial L}{\partial \dot{q}} \frac{c'_q}{c'_t} - L = E$ and $H_0 = E_0$, the initial energy, and transforming to t coordinates gives

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} = 0, \quad (5.15a)$$

$$\frac{d}{dt} (E - E_0 - \lambda) = 0. \quad (5.15b)$$

5.2.2 Continuous System with Time Adaption and Forces

The force term for the associated curve q ,

$$\int_0^T f(q(t), \dot{q}(t), u(t)) \delta q(t) dt, \quad (5.16)$$

may be rewritten considering the transformation $\delta q(t) = \delta c_q - \frac{c'_q}{c'_t} \delta c_t$,

$$\begin{aligned} \int_0^T f(q(t), \dot{q}(t), u(t)) \delta q(t) dt &= \\ \int_0^{\tau_f} \left(f \left(c_q, \frac{c'_q}{c'_t}, c_u \right) c'_t \cdot \delta c_q - f \left(c_q, \frac{c'_q}{c'_t}, c_u \right) c'_q \cdot \delta c_t \right) d\tau, \end{aligned} \quad (5.17)$$

where $dt = c'_t d\tau$, and c_u is the control parameter in τ coordinates. The Lagrange-d'Alembert principle requires that

$$\begin{aligned} \delta \int_0^{\tau_f} \left(L \left(c_q, \frac{c'_q}{c'_t} \right) + H_0 \right) \cdot c'_t + c_\lambda \cdot (c'_t - \sigma(c_q)) d\tau + \\ \int_0^{\tau_f} \left(f \left(c_q, \frac{c'_q}{c'_t}, c_u \right) c'_t \cdot \delta c_q - f \left(c_q, \frac{c'_q}{c'_t}, c_u \right) c'_q \cdot \delta c_t \right) d\tau = 0, \end{aligned} \quad (5.18)$$

for all variations δc_q , δc_t , and δc_λ , with $\delta c_q(0) = \delta c_q(\tau_f) = 0$, and $\delta c_t(0) = \delta c_t(\tau_f) = 0$, and $\delta c_\lambda(0) = \delta c_\lambda(\tau_f) = 0$. This principle gives the forced equations of motion, written in t coordinates, with H_0 replaced by E_0 ,

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} = f, \quad (5.19a)$$

$$\frac{d}{dt} (E - E_0 - \lambda) = f \dot{q}. \quad (5.19b)$$

5.2.3 Correspondence Between Original System and Time Adapted System

If (q, \dot{q}) is a solution of the regular Euler-Lagrange equations,

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0, \quad (5.20a)$$

$$\frac{dE}{dt} = 0, \quad (5.20b)$$

then (q, \dot{q}, λ) is a solution of the time adapted Euler-Lagrange equations,

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} = 0, \quad (5.21a)$$

$$\frac{d}{dt} (E - E_0 - \lambda) = 0, \quad (5.21b)$$

if $\lambda = 0$.

Proof Plug equation (5.20a) and $\lambda = 0$ into the right-hand side of equation (5.21a), verifying that $(q, \dot{q}, \lambda = 0)$ is a solution of equation (5.21a). Equation (5.21b) also holds because $\frac{d}{dt} E_0 = 0$ since E_0 is a constant, $\frac{d}{dt} \lambda = 0$ by definition of λ , and $\frac{d}{dt} E = 0$ according to equation (5.20b). \square

Conversely, if (q, \dot{q}, λ) is a solution of the time adapted Euler-Lagrange equations, equations (5.21), restricted to the energy surface $E = E_0$, then (q, \dot{q}) is also a solution of the regular Euler-Lagrange equations if $\lambda(0) = 0$.

Proof Equation (5.21b) gives that $\frac{d}{dt} (E - E_0) = \frac{d}{dt} \lambda$. Since $E = E_0$, then $\frac{d}{dt} \lambda = 0$, and $\lambda = 0$ because $\lambda(0) = 0$. With $\lambda = 0$, equations (5.21) are equivalent to equations (5.21b). \square

Numerically, λ converges to zero with second-order convergence; thus the time adapted system converges to the original system in the limit as the step size converges to zero. Note that λ also converges to zero when the system includes forces.

5.2.4 Discrete System with Time Adaption

Before defining the discrete version of relevant integrals, it is necessary to define the discrete step sizes for both t and τ ,

$$d\tau = \tau_{k+1} - \tau_k = h, \quad (5.22)$$

$$dt = t_{k+1} - t_k = h_k. \quad (5.23)$$

The action integral may be approximated according to the following quadrature rules,

$$\int_0^{\tau_f} L \left(c_q, \frac{c_q'}{c_t'} \right) \cdot c_t'(\tau) d\tau \approx \sum_{k=0}^{N-1} \bar{L}_d(q_k, q_{k+1}, h, h_k) \frac{h_k}{h}, \quad (5.24)$$

$$\int_0^T L(q(t), \dot{q}(t)) dt \approx \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}, h_k), \quad (5.25)$$

where

$$\bar{L}_d = h L \left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h_k} \right), \quad (5.26)$$

$$L_d = h_k L \left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h_k} \right). \quad (5.27)$$

Based on the definitions of \bar{L}_d and L_d , the right-hand sides of equation (5.24) and equation (5.25) are equivalent.

Similarly,

$$\begin{aligned} \int_0^{\tau_f} c_{\lambda_k} (c_t' - \sigma(c_q)) d\tau &\approx \sum_{k=0}^{N-1} h \lambda_k \left(\frac{h_k}{h} - \sigma(q_k, q_{k+1}) \right) \\ &= \sum_{k=0}^{N-1} \lambda_k (h_k - h \sigma(q_k, q_{k+1})), \end{aligned} \quad (5.28)$$

and

$$\begin{aligned}
\int_{kh}^{(k+1)h} f \left(c_q, \frac{c_q'}{c_t'}, c_u \right) c_t' \delta c_q \, d\tau &\approx \sum_{k=0}^{N-1} [f_k^-(q_k, q_{k+1}, u_k) \delta q_k + f_k^+(q_k, q_{k+1}, u_k) \delta q_{k+1}] \\
&= \sum_{k=0}^{N-1} [f_k^-(q_k, q_{k+1}, u_k) \delta q_k + f_k^+(q_k, q_{k+1}, u_k) \delta q_{k+1}],
\end{aligned} \tag{5.29}$$

where $f_k^- = f_k^+ = \frac{h_k}{2} f_k$. The next part of the force integral may be approximated by

$$\begin{aligned}
\int_{kh}^{(k+1)h} f \left(c_q, \frac{c_q'}{c_t'}, c_u \right) c_q' \delta c_t \, d\tau &\approx \\
&\sum_{k=0}^{N-1} \left[f_k^-(q_k, q_{k+1}, u_k) \frac{q_{k+1} - q_k}{h_k} \delta t_k + f_k^+(q_k, q_{k+1}, u_k) \frac{q_{k+1} - q_k}{h_k} \delta t_{k+1} \right] \\
&= \sum_{k=0}^{N-1} \frac{h_k}{h} \left[f_k^-(q_k, q_{k+1}, u_k) \left(\frac{q_{k+1} - q_k}{h_k} \right) \delta t_k + f_k^+(q_k, q_{k+1}, u_k) \left(\frac{q_{k+1} - q_k}{h_k} \right) \delta t_{k+1} \right].
\end{aligned} \tag{5.30}$$

Based on these approximations, the discrete action principle may be written

$$\begin{aligned}
&\delta \sum_{k=0}^{N-1} [L_d(q_k, q_{k+1}, h_k) + h_k H_0 + \lambda_k(h_k - h\sigma(q_k, q_{k+1}))] \\
&+ \sum_{k=0}^{N-1} \left[f_k^-(q_k, q_{k+1}, u_k) \cdot (\delta q_k - \frac{q_{k+1} - q_k}{h_k} \delta t_k) \right] \\
&+ \sum_{k=0}^{N-1} \left[f_k^+(q_k, q_{k+1}, u_k) \cdot (\delta q_{k+1} - \frac{q_{k+1} - q_k}{h_k} \delta t_{k+1}) \right] = 0. \tag{5.31}
\end{aligned}$$

Variations with respect to q_k , λ_k , and t_k generate the discrete Euler-Lagrange equations as well as equations enforcing the time adaption and energy dissipation,

$$\begin{aligned} & D_1 L_d(q_k, q_{k+1}, h_k) + D_2 L_d(q_{k-1}, q_k, h_{k-1}) \\ & - h \lambda_k \frac{\partial \sigma(q_k, q_{k+1})}{\partial q_k} - h \lambda_{k-1} \frac{\partial \sigma(q_{k-1}, q_k)}{\partial q_k} + f_k^- + f_{k-1}^+ = 0, \end{aligned} \quad (5.32a)$$

$$t_{k+1} = t_k + h \sigma(q_k, q_{k+1}), \quad (5.32b)$$

$$\lambda_k = \lambda_{k-1} + E_{k+1} - E_k - f_k^- \left(\frac{q_{k+1} - q_k}{h_k} \right) - f_{k-1}^+ \left(\frac{q_k - q_{k-1}}{h_{k-1}} \right), \quad (5.32c)$$

where

$$E_{k+1} = -D_3 L_d(q_k, q_{k+1}, t_{k+1} - t_k). \quad (5.33)$$

Equations (5.32) are exactly the variational integrator equations with time adaption presented by [24]. However, note that the notation used here differs slightly from the notation used in [24], particularly for the external forces.

Preservation Properties

Since the usual Euler-Lagrange equations include energy preservation, it is useful to analyze the time adapted system to determine what quantities, if any, are conserved. Consider the system of equations for the time adapted continuos system given by equation (5.15). Manipulation of equation (5.15b) gives

$$\frac{d}{dt} \lambda = \frac{d}{dt} E \quad (5.34a)$$

$$\dot{\lambda} = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \dot{q} - L \right) \quad (5.34b)$$

$$\dot{\lambda} = \left(\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} \right) \dot{q}. \quad (5.34c)$$

Using the relation from equation (5.15a),

$$\dot{\lambda} + \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} \dot{q} = 0 \quad (5.35a)$$

$$\dot{\lambda}\sigma + \lambda \frac{\partial \sigma}{\partial q} \dot{q} = 0 \quad (5.35b)$$

$$\frac{d}{dt} (\lambda \cdot \sigma) = 0. \quad (5.35c)$$

Hence, $\lambda \cdot \sigma$ is a conserved quantity. Since $\dot{\lambda} = \dot{E}$, it follows that

$$\lambda(t) = \lambda_0 + E(t) - E(0) = E(t) - E(0), \quad (5.36)$$

because $\lambda_0 = 0$ by definition. Therefore, the time adapted energy being preserved is

$$\hat{E}(t) = \lambda(t)\sigma = (E(t) - E(0)) \cdot \sigma(q). \quad (5.37)$$

Rearranging this equation, it is clear that

$$E(t) - E(0) = \frac{\hat{E}(t)}{\sigma(q)}, \quad (5.38)$$

and if σ is bounded from below, it gives a bound on the energy drift [24].

The new time adapted continuous system may be written

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} = 0, \quad (5.39)$$

$$\frac{d}{dt} (\lambda \cdot \sigma) = \frac{d}{dt} (\hat{E}) = 0, \quad (5.40)$$

where \hat{E} is the quantity being preserved.

Analyzing the system from a discrete perspective, consider variations of the discrete action

$$\delta \sum_{k=0}^{N-1} [L_d(q_k, q_{k+1}, h_k) + h_k H_0 + \lambda_k (h_k - (\tau_{k+1} - \tau_k) \sigma(q_k, q_{k+1}))] = 0, \quad (5.41)$$

with respect to τ_k for a modified mechanical system with constant step size $h =$

$\tau_{k+1} - \tau_k$. This generates the difference in discrete energy,

$$\widehat{E}_{k+1} - \widehat{E}_k = \lambda_k \sigma(q_k, q_{k+1}) - \lambda_{k-1} \sigma(q_{k-1}, q_k), \quad (5.42)$$

for the modified system. Applying recursion relationships and $\lambda_0 = 0$, this may be written as

$$\widehat{E}_{k+1} - \widehat{E}_1 = \lambda_k \sigma(q_k, q_{k+1}) = (E_{k+1} - E_1) \sigma(q_k, q_{k+1}). \quad (5.43)$$

Since $\widehat{E}_{k+1} - \widehat{E}_1$ defines the energy drift for a variational integrator with constant step size, h , the modified discrete system inherits the usual energy preservation properties. Specifically, the energy drift is bounded such that $|\widehat{E}_{k+1} - \widehat{E}_1| = \mathcal{O}(h^2)$. This relationship can be used to bound the energy of the time adapted system. In particular,

$$|E_{k+1} - E_1| = \left| \frac{\widehat{E}_{k+1} - \widehat{E}_1}{\sigma(q_k, q_{k+1})} \right| = \mathcal{O} \left(\frac{h^2}{\sigma_{min}} \right), \quad (5.44)$$

because σ is bounded from below by σ_{min} . Therefore, even though the discrete energy may drift further from the initial value than for integration with constant time steps, the drift is still bounded with no error accumulation.

Preservation Properties with Forces

For a system with forces, it is important to see how the forces affect the energy evolution. The analysis proceeds as before, beginning by rewriting equation (5.19b) as

$$\begin{aligned} \frac{d}{dt} E - \frac{d}{dt} \lambda &= f \dot{q} \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \dot{q} - \frac{\partial L}{\partial q} \dot{q} - \frac{d}{dt} \lambda &= f \dot{q}. \end{aligned} \quad (5.45)$$

Replace f with the left-hand side of equation (5.19a),

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \dot{q} - \frac{\partial L}{\partial q} \dot{q} - \frac{d}{dt} \lambda &= - \left(\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} \right) \dot{q} \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \dot{q} - \frac{\partial L}{\partial q} \dot{q} - \frac{d}{dt} \lambda &= \left(\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} + \frac{\lambda}{\sigma} \frac{\partial \sigma}{\partial q} \right) \dot{q} \end{aligned} \quad (5.46)$$

$$\begin{aligned} \frac{d}{dt} \lambda \sigma + \lambda \frac{\partial \sigma}{\partial q} \dot{q} &= 0 \\ \frac{d}{dt} (\lambda \cdot \sigma) &= 0, \end{aligned} \quad (5.47)$$

so $\lambda \cdot \sigma$ is a conserved quantity, as before. From equation (5.19b),

$$\dot{\lambda} = \frac{dE}{dt} - f \dot{q}, \quad (5.48)$$

$$\lambda(t) = E(t) - E(0) - \int_0^t f(q(s), \dot{q}(s), u(s)) \dot{q}(s) ds. \quad (5.49)$$

The time adapted energy being preserved is

$$\hat{E} = \lambda \cdot \sigma = \left(E(t) - E(0) - \int_0^t f(q(s), \dot{q}(s), u(s)) \dot{q}(s) ds \right) \sigma. \quad (5.50)$$

Rearranging this equation,

$$\frac{\hat{E}(t)}{\sigma(q)} = E(t) - E(0) - \int_0^t f(q(s), \dot{q}(s), u(s)) \dot{q}(s) ds. \quad (5.51)$$

The energy should evolve according to the integral of the applied forces, and since $\hat{E}(t)$ is preserved and σ is bounded from below, there is a bound on the drift in true energy evolution.

Considering the discrete formulation, the energy drift is bounded by

$$|E_{k+1} - E_1 - \sum_{i=1}^k f_{i-1}^+(q_j - q_{j-1}) + f_i^-(q_{j+1} - q_j)| = \mathcal{O} \left(\frac{h^2}{\sigma_{min}} \right). \quad (5.52)$$

5.3 Naive Time Adaption for DMOC

Since the regular form of DMOC is directly related to variational integrators, it appears that the time adapted form of DMOC should also be related to time adapted variational integrators. However, this assumption is incorrect as demonstrated with a simple optimal control example. Consider the simple system with Lagrangian, $L = \frac{1}{2}\dot{q}^2$. The controlled equations of motion are

$$\ddot{q} = u, \quad (5.53)$$

where u is the control force. The goal is to move the system from some initial condition (q_0, \dot{q}_0) to the final condition (q_N, \dot{q}_N) while minimizing the control effort; therefore, the cost function is $C = \frac{1}{2}u^2$. The analytical solution to this optimal control problem is given by

$$q(t) = c_1 + c_2t + \frac{c_3}{2}t^2 + \frac{c_4}{6}t^3, \quad (5.54a)$$

$$u(t) = c_4t + c_3, \quad (5.54b)$$

where the constants c_1 , c_2 , c_3 , and c_4 are determined by the boundary conditions,

$$c_1 = q_0,$$

$$c_2 = \dot{q}_0,$$

$$c_3 = -\frac{2}{t_N^2} ((2\dot{q}_0 + \dot{q}_N)t_N + 3(q_0 - q_N)),$$

$$c_4 = \frac{6}{t_N^3} ((\dot{q}_0 + \dot{q}_N)t_N + 2(q_0 - q_N)).$$

Now consider a specific example on the time interval $[0, 10]$ with boundary conditions $q_0 = 1$, $\dot{q}_0 = 1$, $q_N = 11$, and $\dot{q}_N = 0$. A time adapted initial guess is created using equation (5.32) with $f = u = 0$ and $\sigma = \frac{q}{2}$. This initial guess is optimized using the time adapted equations, equations (5.32), as optimization constraints and with control force $f = u$ an optimization variable. As before, the

discrete objective function is given by

$$J_d = \sum_{k=0}^{N-1} \frac{h}{2} u_k^2.$$

Successful optimization generates the trajectory and control profile shown in Figure 5.1. The analytical solutions for the trajectory $q(t)$ and control $u(t)$, based on equation (5.54) are included. It is obvious from the optimized control profile shown in Figure 5.1(b), that the optimizer converges to a different optimal solution. Even if the true solution is used as the initial guess, the incorrect solution is still generated. This result indicates that the equations that work for time adapted variational integrators do not directly translate to a time adapted form of DMOC. The effect of time adaption on control forces and the optimal control problem must be considered.

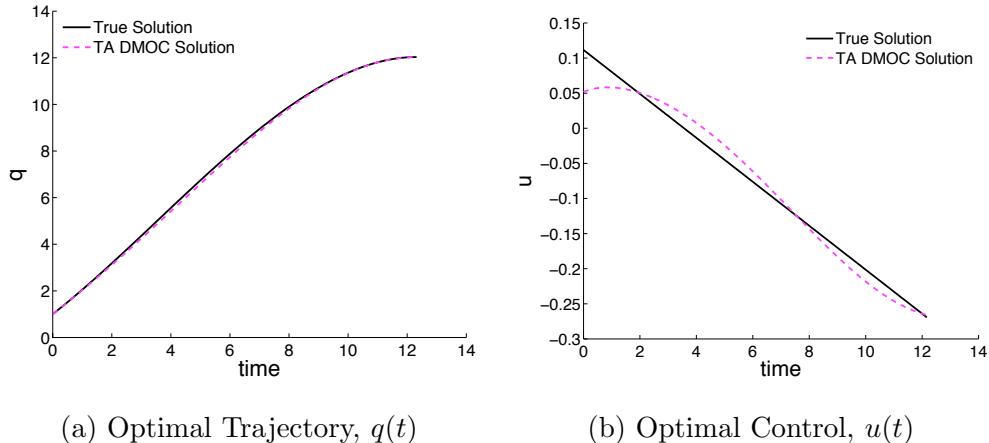


Figure 5.1: Naive formulation of time adapted DMOC leads to incorrect optimal solution both for the (a) optimal trajectory and (b) optimal control force.

5.4 Time Adaption for Optimal Control Problem

To understand how to properly employ time adaption with DMOC, it is necessary to begin by determining how the optimal control Lagrangian \mathcal{L} is computed using the optimal control Hamiltonian \mathcal{H} . Then, this relationship may be ex-

ploited to formulate the time adapted optimal control Lagrangian $\tilde{\mathcal{L}}$ based on $\tilde{\mathcal{H}}$. Euler-Lagrange equations derived from \mathcal{L} provide necessary optimality conditions, assuming sufficient smoothness of the solution. Considering the time adapted optimal control Lagrangian, $\tilde{\mathcal{L}}$, new Euler-Lagrange equations can be derived that provide a set of necessary optimality conditions for the time adapted system.

5.4.1 Transformation from Optimal Control Hamiltonian to Lagrangian

Recalling that $L = \frac{\partial H}{\partial p} \cdot p - H$, the optimal control Lagrangian may be written

$$\mathcal{L} = \frac{\partial \mathcal{H}}{\partial p_{o.c.}} \cdot p_{o.c.} - \mathcal{H}, \quad (5.55)$$

where $p_{o.c.}$ is the momentum of the optimal control problem. Ordinarily, the momentum for a Hamiltonian system may be computed according to the equation

$$p = \frac{\partial L}{\partial \dot{q}}, \quad (5.56)$$

where q represents the state. For the optimal control problem, the state is augmented with the adjoint variable μ ; therefore, denote the state and its derivative by

$$x = (q, \mu), \quad (5.57a)$$

$$\dot{x} = (\dot{q}, \dot{\mu}). \quad (5.57b)$$

Thus, the optimal control momentum is given by

$$p_{o.c.} = \frac{\partial \mathcal{L}}{\partial \dot{x}} = \left(\frac{\partial \mathcal{L}}{\partial \dot{q}}, \frac{\partial \mathcal{L}}{\partial \dot{\mu}} \right). \quad (5.58)$$

Consider the optimal control Lagrangian for the simple example in §5.3,

$$\mathcal{L} = \left(-\frac{1}{2}\mu^2 + \nu(\dot{q} - \ddot{q}) + \mu\ddot{q} \right), \quad (5.59)$$

where ν and μ are the adjoint variables. Since variations of the action of the optimal control Lagrangian are important, consider the action of the optimal control Lagrangian,

$$\mathfrak{G}(q) = \int_0^T \left(-\frac{1}{2}\mu^2 + \nu(\dot{q} - \ddot{q}) + \mu\ddot{q} \right) dt. \quad (5.60)$$

Integrating the term containing \ddot{q} by parts and neglecting the boundary term (it will disappear when considering variations), the optimal control Lagrangian may be written,

$$\mathcal{L} = -\frac{1}{2}\mu^2 - \dot{\mu}\dot{q}. \quad (5.61)$$

Application of equation (5.58) gives the momentum

$$p_{o.c.} = (-\dot{\mu}, -\dot{q}), \quad (5.62)$$

and when applied in equation (5.55) generates the expression

$$\mathcal{L} = -\frac{\partial \mathcal{H}}{\partial(-\dot{q})}\dot{q} - \frac{\partial \mathcal{H}}{\partial(-\dot{\mu})}\dot{\mu} - \mathcal{H}, \quad (5.63)$$

where

$$\mathcal{H} = \frac{1}{2}\mu^2 + \nu\dot{q}. \quad (5.64)$$

Examination of the differential equations for the adjoint variables reveals that $\nu = -\dot{\mu}$ for this system, giving

$$\mathcal{H} = \frac{1}{2}\mu^2 - \dot{\mu}\dot{q}. \quad (5.65)$$

Application of equation (5.63) returns the expected expression for \mathcal{L} given in equation (5.61).

For a more general optimal control problem with dynamics given by

$$\ddot{q} = F(q, \dot{q}) + G(q)u,$$

the optimal control Hamiltonian is given by

$$\mathcal{H} = -\frac{1}{2}u^2 + \nu\dot{q} + \mu(F(q, \dot{q}) + G(q)u). \quad (5.66)$$

Variations of this Hamiltonian with respect to u gives the expression for the optimal control

$$u = \mu G(q). \quad (5.67)$$

Therefore, the optimal control Lagrangian can be written

$$\mathcal{L} = -\frac{1}{2}G(q)^2\mu^2 - \mu F(q, \dot{q}) - \dot{\mu}\dot{q}. \quad (5.68)$$

It is assumed that $\frac{\partial^2 F}{\partial \dot{q}^2} = 0$. This assumption is valid for all problems discussed in this thesis. Based on this more general optimal control Lagrangian, the optimal control momentum is

$$p_{o.c.} = \left(-\dot{\mu} - \mu \frac{\partial F}{\partial \dot{q}}, -\dot{q} \right). \quad (5.69)$$

Note that for a mechanical system, the Lagrangian depends on q and \dot{q} , and it is written $L(q, \dot{q})$. For the optimal control problem, the optimal control Lagrangian depends on q , μ , \dot{q} , and $\dot{\mu}$, written $\mathcal{L}(q, \mu, \dot{q}, \dot{\mu})$.

5.4.2 Transformation of Time Adapted Optimal Control Hamiltonian and Lagrangian

Recall that the time adapted Hamiltonian is given by $\tilde{H} = \sigma(q)(H - H_0)$. Therefore, the time adapted optimal control Hamiltonian is

$$\tilde{\mathcal{H}} = \sigma(q)(\mathcal{H} - \mathcal{H}_0), \quad (5.70)$$

where \mathcal{H}_0 replaces H_0 and is used to denote the initial value of the optimal control Hamiltonian, representing an energy of the optimal control problem. Consequently,

the time adapted optimal control Lagrangian can be written

$$\tilde{\mathcal{L}} = \frac{\partial \tilde{\mathcal{H}}}{\partial p_{o.c.}} \cdot p_{c.o} - \tilde{\mathcal{H}} = \sigma(q) \left(\frac{\partial \mathcal{H}}{\partial p_{o.c.}} \cdot p_{c.o} - \mathcal{H} + \mathcal{H}_0 \right) = \sigma(q) (\mathcal{L} + \mathcal{H}_0), \quad (5.71)$$

where the simplification on the right-hand side is possible under the assumption that $\sigma(q)$ is not a function of the optimal control problem momentum, $p_{o.c.}$. Using the same representation as for equation (5.10),

$$\tilde{\mathcal{L}}(\tau) = c'_t (\mathcal{L} + \mathcal{H}_0) + c_\lambda (c'_t - \sigma(c_q)). \quad (5.72)$$

The variation of the action is

$$\delta \tilde{\mathcal{L}}(\tau) = \delta \int_0^{\tau_f} [c'_t (\mathcal{L}(\tau) + \mathcal{H}_0) + c_\lambda (c'_t - \sigma(c_q))] d\tau = 0, \quad (5.73)$$

with variations vanishing on the endpoints.

5.4.3 Time Adapted DMOC: Discrete Time Adapted Euler La-grange Equations

The discrete time adapted action for the optimal control problem is given by

$$\hat{\mathcal{S}}_0^N = \sum_{k=0}^{N-1} [\mathcal{L}_d(q_k, q_{k+1}, \mu_k, \mu_{k+1}, h_k) + h_k \mathcal{H}_0 + \lambda_k (h_k - h \sigma(q_k, q_{k+1}))] \quad (5.74)$$

where

$$\begin{aligned} \mathcal{L}_d(q_k, q_{k+1}, \mu_k, \mu_{k+1}, h_k) &= h_k \left[-\frac{1}{2} G \left(\frac{q_k + q_{k+1}}{2} \right)^2 \left(\frac{\mu_k + \mu_{k+1}}{2} \right)^2 \right. \\ &\quad \left. - \left(\frac{\mu_k + \mu_{k+1}}{2} \right) F \left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h_k} \right) - \left(\frac{\mu_{k+1} - \mu_k}{h_k} \right) \left(\frac{q_{k+1} - q_k}{h_k} \right) \right], \end{aligned} \quad (5.75)$$

and $h_k = t_{k+1} - t_k$ and $h = \tau_{k+1} - \tau_k$ is a constant. Then, variations of the discrete action for the optimal control problem with respect to q_k , μ_k , t_k , and λ_k ,

$$\delta \sum_{k=0}^{N-1} [\mathcal{L}_d(q_k, q_{k+1}, \mu_k, \mu_{k+1}, h_k) + h_k \mathcal{H}_0 + \lambda_k (h_k - h \sigma(q_k, q_{k+1}))] = 0, \quad (5.76)$$

generate the discrete time adapted Euler-Lagrange equations for the optimal control problem

$$\frac{\partial}{\partial q_k} \mathcal{L}_{k-1,k} + \frac{\partial}{\partial q_k} \mathcal{L}_{k,k+1} - h\lambda_k \frac{\partial \sigma(q_k, q_{k+1})}{\partial q_k} - h\lambda_{k-1} \frac{\partial \sigma(q_{k-1}, q_k)}{\partial q_k} = 0, \quad (5.77a)$$

$$\frac{\partial}{\partial \mu_k} \mathcal{L}_{k-1,k} + \frac{\partial}{\partial \mu_k} \mathcal{L}_{k,k+1} = 0, \quad (5.77b)$$

$$\lambda_{k-1} - \lambda_k + \frac{\partial}{\partial t_k} \mathcal{L}_{k-1,k} - \frac{\partial}{\partial t_k} \mathcal{L}_{k,k+1} = 0, \quad (5.77c)$$

$$t_{k+1} - t_k - h\sigma(q_k, q_{k+1}) = 0, \quad (5.77d)$$

where $\mathcal{L}_{k-1,k} = \mathcal{L}_d(q_{k-1}, q_k, \mu_{k-1}, \mu_k, h_{k-1})$ and $\mathcal{L}_{k,k+1} = \mathcal{L}_d(q_k, q_{k+1}, \mu_k, \mu_{k+1}, h_k)$.

Note that all variations of \mathcal{H}_0 vanish since it is a constant.

Equation (5.77a) are constraints equations for the adjoint variables, equation (5.77b) are equivalent to the usual discrete Euler-Lagrange equations, equation (5.77c) enforces preservation of the optimal control Hamiltonian function, and equation (5.77d) enforces the time adaption. Equations (5.77) serve as constraints that enforce the dynamics. Since the cost function is built into the Lagrangian, it is not necessary to enforce the cost function separately.

Boundary Conditions

The boundary conditions for configuration variable q are the same as for regular DMO. That is, $q(0) = q_0$ and $q(T) = q_N$, as before. The momentum boundary conditions require more care. Recall that for the optimal control problem, there is an augmented state consisting of (q, μ) . Consequently, there are discrete momentum variables p_q and p_μ computed according to the discrete Legendre transform,

$$p_{q_0} = -\frac{\partial}{\partial q_0} \mathcal{L}_d(q_0, q_1, \mu_0, \mu_1, h_0) + h\lambda_0 \frac{\partial \sigma(q_0, q_1)}{\partial q_0}, \quad (5.78a)$$

$$p_{\mu_0} = -\frac{\partial}{\partial \mu_0} \mathcal{L}_d(q_0, q_1, \mu_0, \mu_1, h_0), \quad (5.78b)$$

$$p_{q_N} = \frac{\partial}{\partial q_N} \mathcal{L}_d(q_{N-1}, q_N, \mu_{N-1}, \mu_N, h_{N-1}) - h\lambda_N \frac{\partial \sigma(q_{N-1}, q_N)}{\partial q_N}, \quad (5.78c)$$

$$p_{\mu_N} = \frac{\partial}{\partial \mu_N} \mathcal{L}_d(q_{N-1}, q_N, \mu_{N-1}, \mu_N, h_{N-1}). \quad (5.78d)$$

The continuous momentum boundary values are determined via the continuous Legendre transform, and the boundary conditions are given by

$$\frac{\partial \mathcal{L}(q_0, \dot{q}_0)}{\partial \dot{q}_0} - p_{q_0} = 0, \quad (5.79a)$$

$$\frac{\partial \mathcal{L}(q_0, \dot{q}_0)}{\partial \dot{\mu}_0} - p_{\mu_0} = 0, \quad (5.79b)$$

$$\frac{\partial \mathcal{L}(q_N, \dot{q}_N)}{\partial \dot{q}_N} - p_{q_N} = 0, \quad (5.79c)$$

$$\frac{\partial \mathcal{L}(q_N, \dot{q}_N)}{\partial \dot{\mu}_N} - p_{\mu_N} = 0. \quad (5.79d)$$

External Forces

This formulation is valid even for systems with external forces in addition to control forces. The external forces are included in $F(q, \dot{q})$ as part of the dynamics. If the system is subject to a time-dependent external force, the dynamics are given by

$$\ddot{q} = F(q, \dot{q}) + G(q)u + F_t(q, t),$$

where $F_t(q, t)$ represents the time-dependent external force. Then, the optimal control Lagrangian is

$$\mathcal{L} = -\frac{1}{2}G(q)^2\mu^2 - \mu F(q, \dot{q}) - \dot{\mu}\dot{q} - \mu F_t(q, t). \quad (5.81)$$

The discrete version of this forced optimal control Lagrangian replaces \mathcal{L}_d in equation (5.74), and variations of this new time adapted Lagrangian should be zero, leading to new Euler-Lagrange equations including both control forces and external forces. Furthermore, the momentum boundary conditions are still given by equations (5.79).

5.4.4 Time Adaptive DMOC: an Indirect Method

Even though DMOC is a direct method for optimal control, formulation of time adapted DMOC as described in §5.4.3 actually results in an indirect method for

solving the optimal control problem. Equations (5.77) combined with the boundary conditions describe a boundary value problem, which can be solved with any BVP solver. For the solution of all examples in this chapter, the implementation is nearly identical to regular DMOC with the Euler-Lagrange equations and boundary conditions enforced as constraints and with cost function set to one. Then the SQP solver SNOPT determines the feasible solution, which in this case is the locally optimal solution.

Table 5.1 demonstrates the parallels between the Lagrangian of the mechanical system, L , and the optimal control Lagrangian, \mathcal{L} , for continuous and discrete settings. Variations of the action of the Lagrangian of the mechanical system lead to the Euler-Lagrange (EL) equations of motion. Variations of the action of the time adapted Lagrangian, \tilde{L} , lead to the time adapted (TA) Euler-Lagrange equations of motion. Variations of the action of the optimal control Lagrangian lead to necessary optimality conditions (nec. opt. cond.), and time adapted necessary optimality conditions (TA nec. opt. cond.) result for the time adapted optimal control Lagrangian. The discrete versions are denoted by D.

Table 5.1: Time Adaption Comparison

Continuous		Discrete	
L	\tilde{L}	L_d	\tilde{L}_d
\Downarrow	\Downarrow	\Downarrow	\Downarrow
EL equations	TA EL equations	DEL equations	TA DEL equations
\mathcal{L}	$\tilde{\mathcal{L}}$	\mathcal{L}_d	$\tilde{\mathcal{L}}_d$
\Downarrow	\Downarrow	\Downarrow	\Downarrow
nec. opt. cond.	TA nec. opt. cond.	D nec. opt. cond.	DTA nec. opt. cond.

5.4.5 Results for Simple Example

Consider again the simple example with $L = \frac{1}{2}\dot{q}^2$, dynamics $\ddot{q} = u$, and time adapted according to $\sigma = \frac{q}{2}$. Taking variations of \mathcal{H} with respect to u gives that

$u = \mu$, and the optimal control Lagrangian is

$$\mathcal{L} = -\frac{1}{2}\mu^2 - \dot{\mu}\dot{q}. \quad (5.82)$$

The discrete action principle is

$$\begin{aligned} \delta \sum_{k=0}^{N-1} \left[-\frac{h_k}{2} \left(\frac{\mu_k + \mu_{k+1}}{2} \right)^2 - h_k \left(\frac{\mu_{k+1} - \mu_k}{h_k} \right) \left(\frac{q_{k+1} - q_k}{h_k} \right) + h_k \mathcal{H}_0 \right. \\ \left. + \lambda_k \left(h_k - h \left(\frac{q_k + q_{k+1}}{4} \right) \right) \right] = 0. \end{aligned} \quad (5.83)$$

Variations of the discrete action principle with respect to q_k , μ_k , t_k , and λ_k generate the discrete time adapted Euler-Lagrange equations for the optimal control problem,

$$\left(\frac{\mu_{k+1} - \mu_k}{h_k} \right) - \left(\frac{\mu_k - \mu_{k-1}}{h_{k-1}} \right) - \frac{h}{4} \lambda_k - \frac{h}{4} \lambda_{k-1} = 0, \quad (5.84a)$$

$$\begin{aligned} \left(\frac{q_{k+1} - q_k}{h_k} \right) - \left(\frac{q_k - q_{k-1}}{h_{k-1}} \right) - \frac{h_k}{2} \left(\frac{\mu_k + \mu_{k+1}}{2} \right) \\ - \frac{h_{k-1}}{2} \left(\frac{\mu_{k-1} + \mu_k}{2} \right) = 0, \end{aligned} \quad (5.84b)$$

$$\begin{aligned} \lambda_{k-1} - \lambda_k + \left(\frac{q_k - q_{k-1}}{h_{k-1}} \right) \left(\frac{\mu_k - \mu_{k-1}}{h_{k-1}} \right) - \frac{1}{2} \left(\frac{\mu_{k-1} + \mu_k}{2} \right) \\ - \left(\frac{q_{k+1} - q_k}{h_k} \right) \left(\frac{\mu_{k+1} - \mu_k}{h_k} \right) + \frac{1}{2} \left(\frac{\mu_k + \mu_{k+1}}{2} \right) = 0, \end{aligned} \quad (5.84c)$$

$$t_{k+1} - t_k - h \left(\frac{q_k + q_{k+1}}{4} \right) = 0. \quad (5.84d)$$

Recall that the momentum for this example is $p_{o.c.} = (-\dot{\mu}, -\dot{q})$. Therefore, the momentum boundary conditions given in equation (5.79) can be written as

$$-\dot{\mu}_0 - p_{q_0} = 0, \quad -\dot{q}_0 - p_{\mu_0} = 0,$$

$$-\dot{\mu}_N - p_{q_N} = 0, \quad -\dot{q}_N - p_{\mu_N} = 0.$$

For this example, the initial and final discrete configurations must equal the

continuous ones:

$$q(0) = q_0, \quad (5.85a)$$

$$q(T) = q_N. \quad (5.85b)$$

Next, the initial and final velocity values should be enforced; consequently, the boundary conditions including \dot{q}_0 and \dot{q}_N should also be enforced,

$$-\dot{q}_0 - p_{\mu_0} = 0, \quad (5.86a)$$

$$-\dot{q}_N - p_{\mu_N} = 0. \quad (5.86b)$$

Furthermore, initial conditions for time and λ are included such that $t_0 = 0$ and $\lambda_0 = 0$. These boundary conditions are sufficient for a well-posed boundary value problem, so the boundary conditions for p_q need not be enforced.

Examining equation (5.86a), and since $\lambda_0 = 0$ by definition, this constraint simplifies to

$$-\dot{q}_0 + \left(\frac{q_1 - q_0}{h_0} \right) + \frac{h_0}{2} \left(\frac{\mu_0 + \mu_1}{2} \right) = 0, \quad (5.87)$$

which looks very similar to the usual momentum boundary condition with $\left(\frac{\mu_0 + \mu_1}{2} \right) = u_0$.

Using the simple initial guess described in §5.3, this time adapted form of DMOC successfully produces the correct optimal solution. If the final time is held fixed with time adapted DMOC, the problem is over-constrained. Allowing the final time to vary, time adapted DMOC finds an optimal solution with a slightly different final time than the initial guess. A different final time means that the boundary conditions are slightly different, and therefore, so is the optimal solution. To verify that time adapted DMOC generates the correct optimal solution, the optimal solution is used as an initial guess for regular DMOC. In this way, the optimal solutions from regular DMOC and time adapted DMOC can be compared because they share the same time grid. Figure 5.3 compares the time adapted DMOC optimal solution with the regular DMOC optimal solution for both the

optimal trajectory and optimal control. As shown in the figure, the solutions match, confirming that time adapted DMOC converges to the correct optimal solution.

Two different energy metrics are examined to compare the DMOC and time adapted DMOC solutions. First, the discrete energy drift,

$$E_d = E_{k+1} - E_1 - \sum_{i=1}^k f_{i-1}^+(q_j - q_{j-1}) + f_i^-(q_{j+1} - q_j), \quad (5.88)$$

for regular DMOC, and

$$E_d = E_{k+1} - E_1, \quad (5.89)$$

for time adapted DMOC, where

$$E_{k+1} = -D_3 L_d(q_k, q_{k+1}, t_{k+1} - t_k) \text{ for regular DMOC,}$$

$$E_{k+1} = -D_5 \mathcal{L}_d(q_k, q_{k+1}, \mu_k, \mu_{k+1}, t_{k+1} - t_k) \text{ for time adapted DMOC,}$$

should converge to zero with second-order convergence. Since the expressions for discrete energy drift are different for DMOC and time adapted DMOC, it is also useful to consider the discrete version of

$$\Delta E_c = E_c(t) - E_c(0), \quad (5.90)$$

where

$$E_c(t) = E(t) - \int_0^t f(q(s), \dot{q}(s), u(s)) \dot{q}(s) ds, \quad (5.91)$$

and $E(t)$ represents the total energy at each time. The integral term represents the energy injected into the system by the control forces. The discrete Legendre transform is employed to compute the momenta and corresponding velocities at each node, which are then used to compute the discrete version of equation (5.90).

Figure 5.5 shows the convergence for errors in position and control for regular DMOC with constant step size, regular DMOC with time adapted initial guess (time adapted variational integrators generate an initial guess with variable time

grid; this time grid is held fixed), and time adapted DMOC. Both plots display the expected second-order convergence. Notice that the errors are slightly smaller for regular DMOC with constant step size. This is not too surprising since the time adaption is arbitrary.

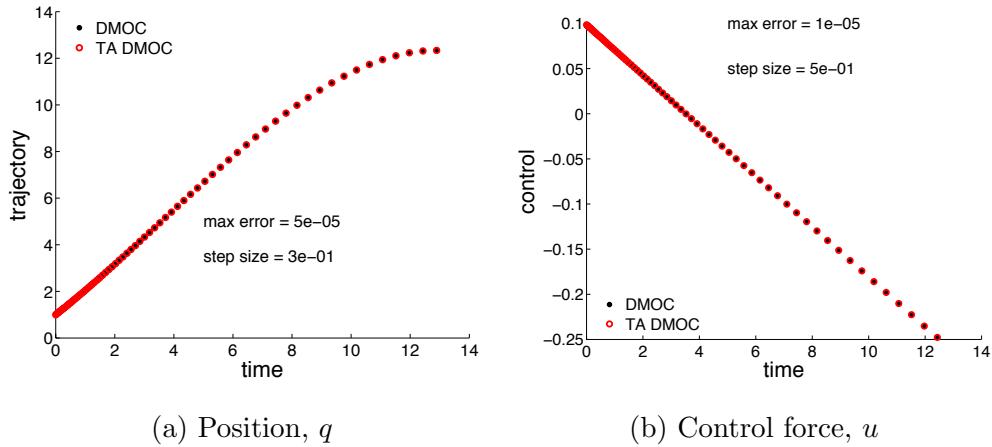


Figure 5.2: Simple example: regular DMOC and time adapted DMOC generate the same optimal (a) trajectory and (b) control

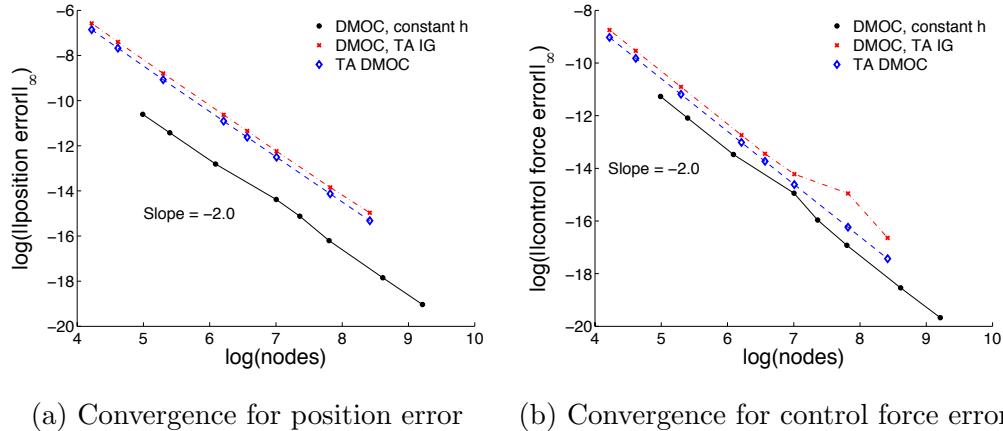


Figure 5.3: Simple example: comparison of solution error for regular DMOC with constant step size, regular DMOC with time adapted initial guess, and time adapted DMOC. The error in (a) position and (b) control force converges to zero with a slope of -2.

The energy metrics, E_d and ΔE_c , are compared for regular DMOC and time adapted DMOC in Figures 5.4(a) and (b), respectively. Note that the energy met-

ric is better for time adapted DMOC in both instances. Figure 5.5(a) shows the rate of convergence for λ , which converges to zero with second-order convergence as predicted. Figure 5.5(b) displays the log of minimum step size versus the log of CPU time in seconds. For most minimum step sizes, time adapted DMOC converges faster than regular DMOC with constant step size and regular DMOC with time adapted initial guess. Also, it should be noted that as the minimum step size decreases, regular DMOC with time adapted initial guess starts having convergence problems. In comparison, time adapted DMOC converges to the optimal solution every time with stringent tolerances. It is interesting to note that time adapted DMOC includes optimization variables q_k , μ_k , t_k , and λ_k compared to just q_k and u_k for regular DMOC. Even with twice as many optimization variables, time adapted DMOC still converges faster.

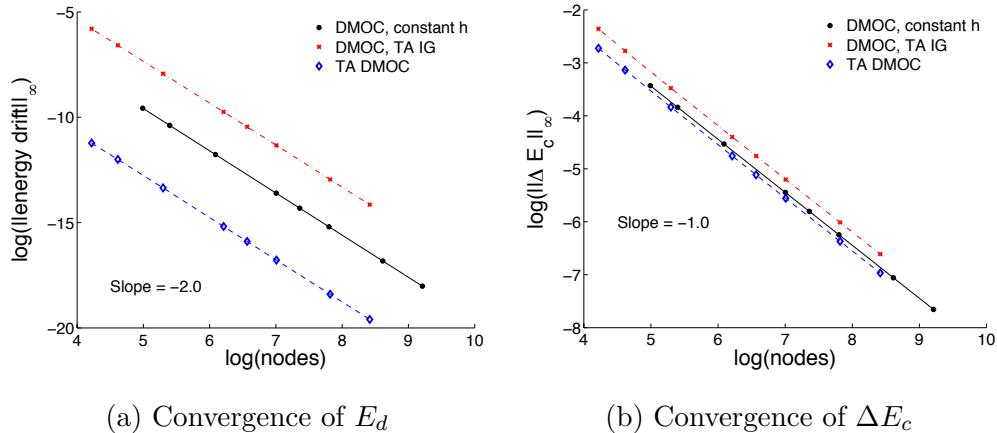


Figure 5.4: Simple example: comparison of energy behavior with regular DMOC with constant step size or time adapted initial guess and time adapted DMOC. Convergence of (a) discrete energy drift, E_d , and (b) ΔE_c .

5.5 Examples

Several different examples are presented that demonstrate different aspects of DMOC with time adaption. First, the elliptical orbit transfer problem, first presented in Chapter 4, is solved using time adaptive DMOC. For this problem, the

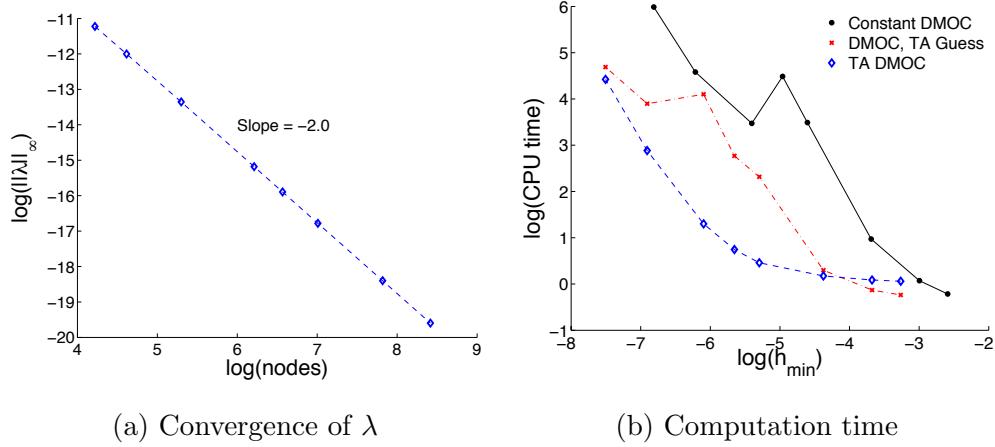


Figure 5.5: Simple example: convergence of (a) λ , and (b) log-log plot of minimum step size versus CPU time in seconds. In most cases, time Adapted DMOC converges fastest.

control force is defined by $f = ru$, where r is a configuration variable. Therefore, in contrast to the simple example, $g(q) \neq 1$. Also, $f(q, \dot{q})$ is nonzero. Next, the problem of reconfiguring a cubesat is presented, demonstrating another potential application for time adapted DMOC.

5.5.1 Elliptical Orbit Transfer

In contrast to how this problem is approached in Chapter 4, the elliptical orbit transfer is now presented in 2d-polar coordinates, $q = (r, \varphi)$. From before, a spacecraft orbits a body in an elliptical orbit such that after one full orbit, it enters a slightly different orbit with a larger apogee radius. The Lagrangian for this system is

$$L(q, \dot{q}) = \frac{1}{2}m(\dot{r}^2 + r^2\dot{\varphi}^2) + \frac{GMm}{r}, \quad (5.92)$$

where G is the universal constant of gravitation, M is the mass of the primary body, and m is the mass of the satellite. To best illustrate the effects of time adaption, the problem is scaled such that $m = 1$ and $GM = 1$. Configuration variables r and φ represent the radial distance of the spacecraft from the center of the primary body and the angular position of the spacecraft with respect to the line through the primary body and the perigee of the elliptical orbit, respectively.

The controlled dynamics of the system are

$$\ddot{r} = r\dot{\varphi}^2 - \frac{GM}{r^2}, \quad (5.93a)$$

$$\ddot{\varphi} = -2\frac{\dot{r}}{r}\dot{\varphi} + \frac{u}{rm}. \quad (5.93b)$$

Aiming to minimize the control effort, the optimal control Hamiltonian is

$$\mathcal{H} = -\frac{1}{2}u^2 + \nu_r \dot{r} + \nu_\varphi \dot{\varphi} + \mu_r \left(r\dot{\varphi}^2 - \frac{GM}{r^2} \right) + \mu_\varphi, \left(-2\frac{\dot{r}}{r}\dot{\varphi} + \frac{u}{rm} \right), \quad (5.94)$$

and $\frac{\partial \mathcal{H}}{\partial u} = 0$ requires that $u = \frac{\mu_\varphi}{rm}$. Thus, the optimal control Lagrangian is

$$\mathcal{L} = -\frac{1}{2} \left(\frac{\mu_\varphi}{rm} \right)^2 - \mu_r \left(r\dot{\varphi}^2 - \frac{GM}{r^2} \right) + \mu_\varphi \left(2\frac{\dot{r}}{r}\dot{\varphi} \right) - \dot{\mu}_r \dot{r} - \dot{\mu}_\varphi \dot{\varphi}. \quad (5.95)$$

Using the discrete version of this Lagrangian in equation (5.76) generates the discrete Euler-Lagrange equations to be enforced as constraints. The spacecraft begins in an elliptical orbit with $r_{p1} = 1$ and $r_{a1} = 2$. The spacecraft ends in an elliptical orbit with the same perigee and $r_{a2} = 4$. The boundary conditions to be enforced include $r_0 = 1$, $r_N = 4$, $\varphi_0 = 0$, $\varphi_N = \pi$, $\dot{r}_0 = \dot{r}_N = 0$, $\dot{\varphi}_0 = \frac{1}{r_{p1}} \sqrt{GM(\frac{2}{r_{p1}} - \frac{1}{a_1})}$, and $\dot{\varphi}_N = \frac{1}{r_{a2}} \sqrt{GM(\frac{2}{r_{a2}} - \frac{1}{a_2})}$, where $a = \frac{1}{2}(r_p + r_a)$ is the semi-major axis of the ellipse.

Several time adaption strategies are tested, given by

$$\sigma_1 = \frac{1}{\sqrt{E_0 - W\left(\frac{q_k+q_{k+1}}{2}\right) + \nu}}, \quad (5.96)$$

$$\sigma_2 = \frac{1}{\sqrt{E_0 - W\left(\frac{q+k+q_{k+1}}{2}\right) + \left\| \nabla W\left(\frac{q+k+q_{k+1}}{2}\right) \right\|^2 + \nu}}, \quad (5.97)$$

$$\sigma_3 = \frac{1}{\left\| \nabla W(q_k) + \nabla W(q_{k+1}) + \nu \right\|}, \quad (5.98)$$

where $W = \frac{GM}{r}$ is the potential energy, E_0 is the initial energy, $\|\cdot\|$ denotes the 2-norm, and ν is a small constant.

Figure 5.6 compares the optimal trajectories for time adapted DMOC and regular DMOC using the time adapted solution as initial guess. Figure 5.7 compares the optimal control solutions. As shown in the both figures, the optimal solutions from time adapted DMOC and regular DMOC match for all three time adaption strategies.

Figure 5.8 compares the energy metrics, E_d and ΔE_c . As shown in Figure 5.8(a), the discrete energy drift for solutions generated with time adapted DMOC is smaller than the discrete energy drift for regular DMOC, even with time adapted initial guess. For ΔE_c , shown in Figure 5.8(b), time adapted DMOC produces slightly better results than regular DMOC.

Figure 5.9(a) displays the convergence of λ for all three time adaption strategies. As expected, λ approaches zero with second order convergence. Figure 5.9(b) exhibits the log of minimum step size versus log of the computation time. Time adapted DMOC converges faster than regular DMOC with constant step size or time adapted initial guess. As the minimum step size decreases, convergence with regular DMOC becomes less dependable, but time adapted DMOC continues to converge very well. Since an analytical solution to this optimal control problem does not exist, convergence plots of the error in configuration or control are not included.

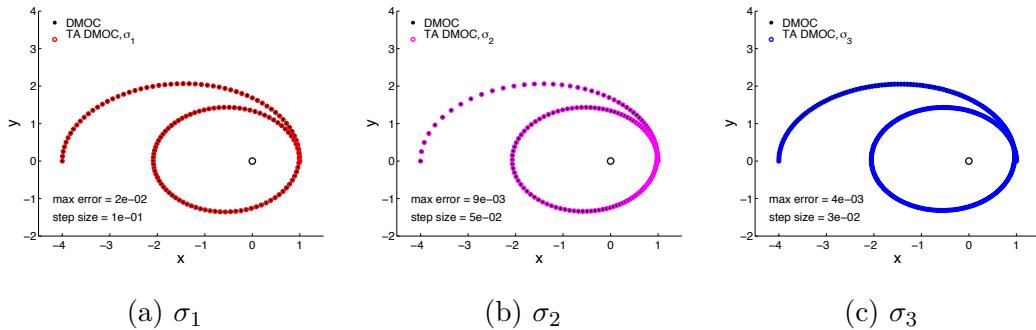


Figure 5.6: Elliptical orbit transfer: optimal trajectory for regular DMOC and time adapted DMOC with (a) σ_1 , (b) σ_2 , and (c) σ_3 . The same optimal solution is achieved using DMOC and time adapted DMOC.

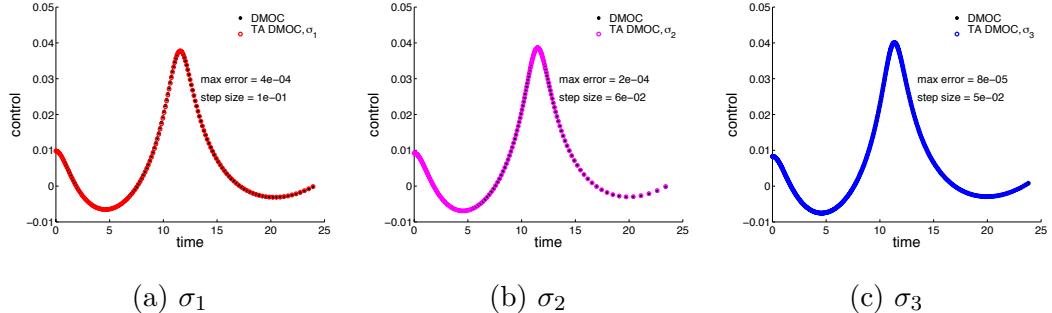


Figure 5.7: Elliptical orbit transfer: optimal control for regular DMOC and time adapted DMOC with (a) σ_1 , (b) σ_2 , and (c) σ_3 . The same optimal solution is achieved using DMOC and time adapted DMOC.

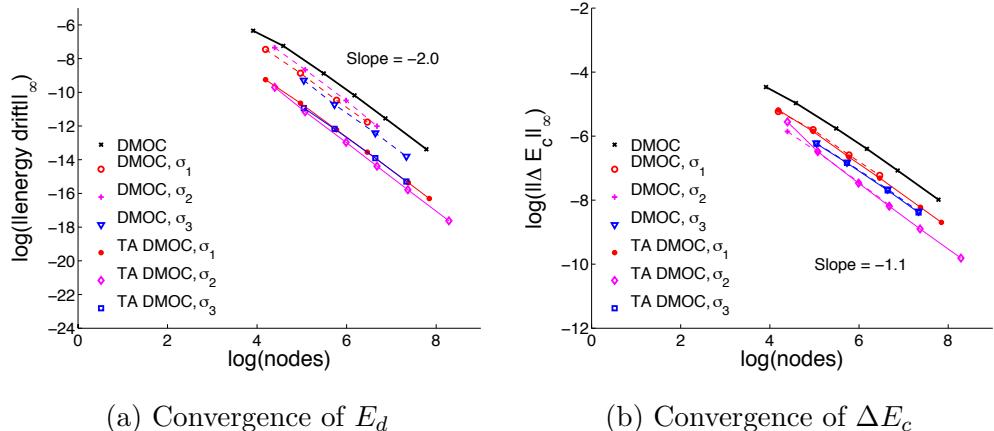


Figure 5.8: Elliptical orbit transfer: comparison of energy behavior. Convergence of (a) discrete energy drift, E_d , and (b) ΔE_c .

5.5.2 Cubesat Reconfiguration

This example is modeled on the hovercraft reconfiguration example presented by [23]. Consider a cubesat with configuration described by position, (x, y) , and orientation, θ . The cubesat is to be moved from some initial configuration (x_0, y_0, θ_0) to a final configuration (x_N, y_N, θ_N) using optimal control. It is controlled by two control forces, f_1 and f_2 , applied at a distance r from the center of mass such that f_1 acts in the direction of motion, and f_2 acts perpendicular to the motion. The

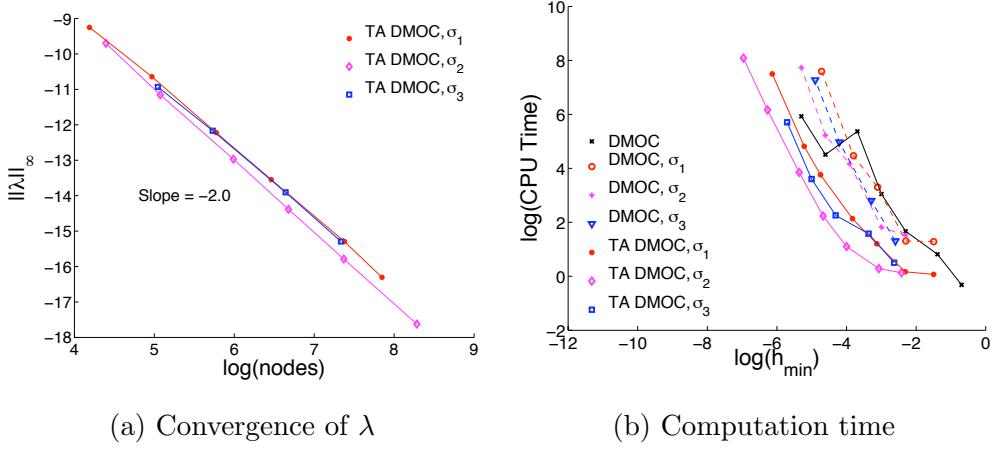


Figure 5.9: Elliptical orbit transfer: convergence of (a) λ . (b) Log of computation time versus log of minimum step size shows that the time adapted solutions converge fastest.

Lagrangian of this system describes the kinetic energy of the cubesat,

$$L(q, \dot{q}) = \frac{1}{2}(m\dot{x}^2 + m\dot{y}^2 + J\dot{\theta}^2), \quad (5.99)$$

where m is the mass and J is the moment of inertia. For this example, m and J both equal one. The controlled equations of motion are given by

$$\ddot{x} = f_1 \cos(\theta) - f_2 \sin(\theta) \quad (5.100a)$$

$$\ddot{y} = f_1 \sin(\theta) + f_2 \cos(\theta) \quad (5.100b)$$

$$\ddot{\theta} = -rf_2. \quad (5.100c)$$

Aiming to minimize control effort, the optimal control Lagrangian, in terms of the state and adjoint variables, is

$$\mathcal{L} = -\frac{1}{2} (\mu_x^2 + \mu_y^2 + r^2 \mu_\theta^2 + 2r\mu_\theta (\mu_x \sin(\theta) - \mu_y \cos(\theta))) - \dot{\mu}_x \dot{x} - \dot{\mu}_y \dot{y} - \dot{\mu}_\theta \dot{\theta}. \quad (5.101)$$

Time is adapted according to

$$\sigma = x^2 + y^2, \quad (5.102)$$

generating smaller time steps when the cubesat moves closer to its target location, located near the origin. The square of the distance from the origin is used for simplicity when deriving the constraint equations.

An initial guess is optimized first using time adapted DMOC. This optimal solution is then used as an initial guess for regular DMOC to verify that both methods converge to the same optimal solution. Figure 5.10 demonstrates that regular DMOC and time adapted DMOC generate the same optimal solution.

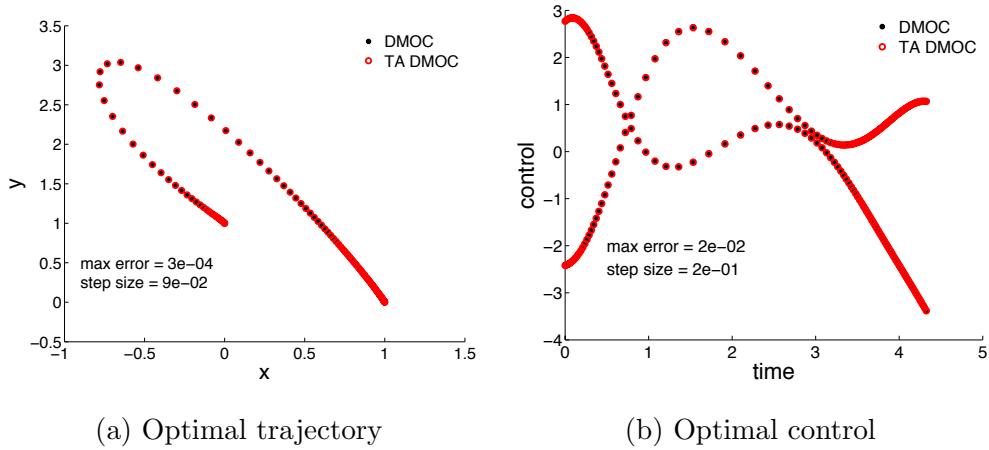


Figure 5.10: Cubesat reconfiguration: regular DMOC and time adapted DMOC generate the same optimal solution for the (a) trajectory and (b) control forces f_1 and f_2 .

Figure 5.11 compares the energy metrics, E_d and ΔE_c . As shown in the plots, time adapted DMOC produces smaller values for both the discrete energy drift and ΔE_c . Figure 5.12(a) shows that λ converges to zero with second order convergence as expected. Figure 5.12(b) compares the computation time, and in contrast to the other examples, time adapted DMOC is slower than regular DMOC for this example because σ is not a function of the dynamics.

5.6 Conclusion

The process used to derive time adapted variational integrators can be applied to the optimal control problem, leading to a time adaptive form of DMOC. Time adapted DMOC is now an indirect approach to solving the optimal control problem

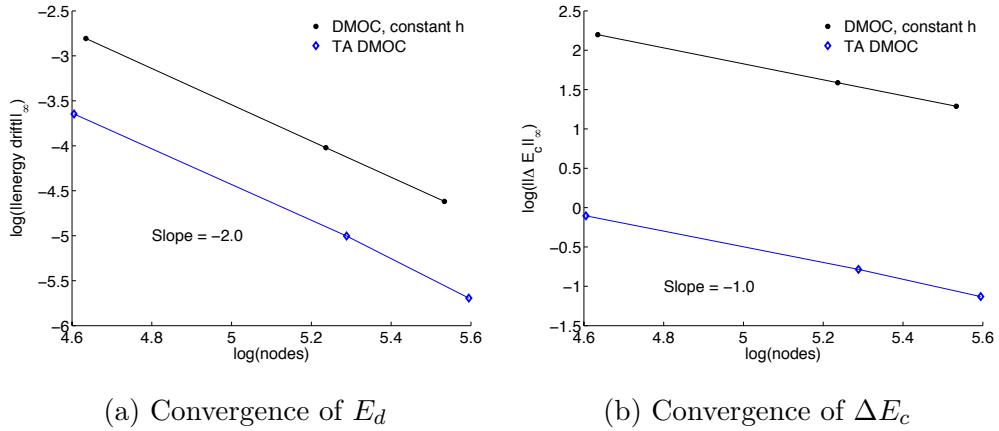


Figure 5.11: Cubesat reconfiguration: energy comparison. (a) Discrete energy, E_d , converges to zero with a slope of -2, as expected. (b) ΔE_c is smaller for time adapted DMOC. Time adapted DMOC produces solutions with smaller errors for both energy metrics.

even though regular DMOC is a direct optimization method. Variations of the discrete action of the time adapted optimal control Lagrangian with respect to the state, time, and adjoint variables lead to discrete Euler-Lagrange equations that serve as constraints. The problem is now a boundary value problem, and it is sufficient to set the cost function equal to one. The problem may be solved using SQP as before, but it may also be solved using another BVP solver.

The method is first tested on a very simple example with an analytical optimal control solution to verify that the method produces correct optimal control solutions. Then, it is tested on more relevant examples including the elliptical orbit transfer and the reconfiguration of a cubesat. It should be noted that since time is an optimization variable that changes throughout the optimization, the optimal solutions are slightly different than those achieved with regular DMOC. This is due to the difference in final time. The time adapted optimal solutions are verified by using them as initial guesses for regular DMOC, which then produces the same optimal solution.

While it is desirable to enforce a constraint on the final time, it appears that such a constraint over-constrains the problem. While a variable final time is fine for many problems, some problems may require a fixed final time, so this issue

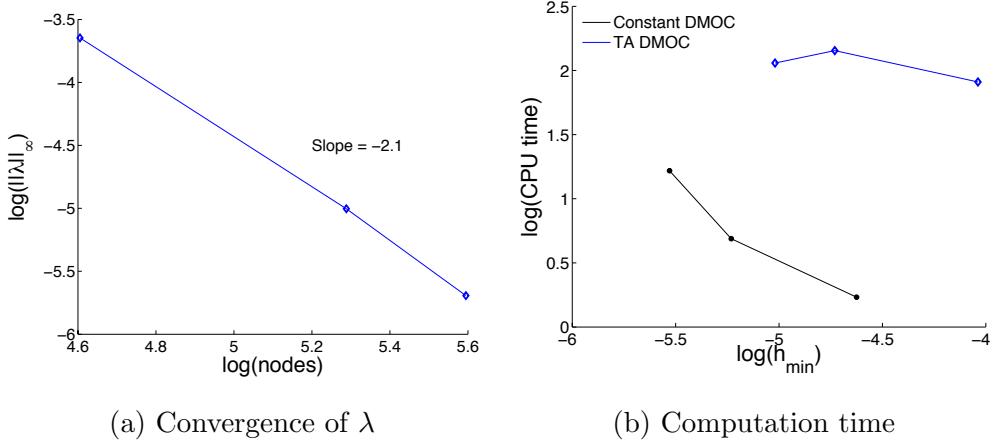


Figure 5.12: Cubesat reconfiguration: convergence of (a) λ , and (b) log of computation time versus log of minimum step size shows that the time adapted solutions general converge slower than regular DMOC because σ is not a function of the dynamics.

warrants further exploration. It is notable that while regular DMOC consists of optimization variables q and u , and time adapted DMOC has twice as many optimization variables, q , μ , t , and λ , time adapted DMOC converges faster than regular DMOC in most cases for which σ is a function of the dynamics. Also, as shown in the examples, time adapted DMOC displays the same energy convergence rate as regular DMOC, verifying that the energy drift is bounded for time adapted DMOC, just as it is for regular DMOC. Furthermore, as predicted λ converges to zero, verifying that the time adapted system converges to the regular system. Overall, time adapted DMOC provides a great optimization method for highly nonlinear problems for which variable step size is absolutely necessary.

Chapter 6

Conclusions and Future Work

This thesis demonstrates how the optimal control algorithm DMOC can be used for the design of spacecraft trajectories and how to better adapt it for such nonlinear problems. First, when combined with design techniques based on invariant manifolds of the 3-body problem, DMOC successfully optimizes an initial guess in the 4-body problem, locating a natural transfer from the Earth to the Moon requiring no mid-course ΔV . Earlier designs of this trajectory, presented by Koon, Lo, Marsden, and Ross [27, 28], utilized differential correction to generate a trajectory with a small, mid-course ΔV of 34 m/s valid for the real dynamics of the solar system described by the JPL ephemeris. DMOC could certainly optimize the problem using a more realistic model as well. In addition, it is shown that DMOC is effective for low thrust design, producing a trajectory that ends in an elliptical orbit at the Moon, instead of ballistic capture. The DMOC results for the Shoot the Moon problem are competitive with those in the literature, and the optimization process is fast. Optimizations with thousands of nodes converge in just minutes.

Furthermore, since DMOC is formulated for use with a constant step size, special considerations are necessary for use with nonlinear trajectory problems. For Shoot the Moon, the step size profile is simply divided into sections of constant step size. Chapter 4 examines the use of mesh refinement for automated step size design. Classical mesh refinement aims to design the step size profile in such

a way as to reduce solution errors caused by the discretization. A new mesh refinement technique is proposed based on errors in the energy evolution. The mesh refinement schemes are compared with trajectories generated using time adaptive variational integrators, with step size profiles that are continuous and dictated by the dynamics. For the elliptical orbit transfer, solution-based mesh refinement produces the best results, but the time adapted solutions proved to be very competitive and motivated the exploration of time adaptive DMOC. For the Shoot the Moon problem, mesh refinement based on errors in the energy derivative lead to better results than solution-based mesh refinement, but the time adapted scheme performs the best. Overall, mesh refinement in which the mesh is iteratively designed based on errors in the solution or energy is cumbersome. Trajectories with time adapted step size profiles are much easier to produce and converge faster during optimization.

The thesis concludes with the development of a fully time adapted version of DMOC. Proper application of time adaption requires that Hamilton's principle be applied to the time adapted Lagrangian of the optimal control problem, instead of to the Lagrangian of the mechanical system. Therefore, instead of discretizing the Lagrange-d'Alembert principle, discretization of Hamilton's principle leads to discrete Euler-Lagrange equations that serve as constraints for a boundary value problem. This problem can be solved in the same way as regular DMOC using SQP, but with the cost function set to one. It should be noted that this formulation of time adapted DMOC is an indirect optimization method even though regular DMOC is a direct method. Optimization employing time adapted DMOC is demonstrated for the elliptical orbit transfer problem and the reconfiguration of a cubesat. Time adaptive DMOC proves to be efficient and accurate, preserving the energy and convergence properties of regular DMOC.

There are many possibilities for future work focusing on the application of DMOC to mission design and time adaptive DMOC.

DMOC and Mission Design

Haapala and Howell [20] examines the use of periapse Poincaré maps in a 3-body system for the design of transit trajectories and heteroclinic connections. A periapse Poincaré map records the location of subsequent periapsis passages around the second primary body. Using the method described, it is possible to create trajectories with particular behavior around m_2 before traveling towards m_1 or the exterior region. If two such trajectories intersect in position space, they may be joined and used as an initial condition for DMOC optimization. Optimizing with DMOC would produce a viable trajectory with optimized control for such a transfer. For example, this idea could be very useful in the Shoot the Moon problem. In the Earth-Moon 3-body system, several trajectories may be created using the periapse Poincaré map method that orbit the Moon a specified number of times. Then, if an intersection exists between an Earth-Moon trajectory and an Earth-bound trajectory in the Sun-Earth 3-body system, DMOC may be used to remove the energy discontinuity using optimal control.

Furthermore, Haapala uses her method to locate initial guesses for heteroclinic connections before refining them with a corrections algorithm. DMOC could also be used in this case, employing optimal control to smooth any energy discontinuities in the initial guess. Enforcing boundary conditions on the periodic orbits around L_1 and L_2 , it is reasonable to assume that the DMOC optimization process will locate the natural heteroclinic connection.

Continuing with the theme of heteroclinic connections, Davis et al. [12] proposes a method for constructing transfers between periodic orbits of different energies. An unstable manifold trajectory from the first periodic orbit is connected with a stable manifold trajectory of the second periodic orbit using two deterministic maneuvers. These maneuvers are then refined using a genetic algorithm. Conversely, DMOC may be used to determine the optimal control necessary for such a connection.

Authors such as Ross and Scheeres [44] and Bosanac et al. [9] have shown the success of combining invariant manifolds and resonant gravity assists to create

interesting transfers in the Jovian and Saturian moon system. DMOC may prove very useful in such problems as well. For example, Bosanac et al. [9] use repeated gravity assists to reduce the semi-major axis of an elliptical orbit about Saturn and Titan targeting a stable invariant manifold leading to Titan. DMOC could be useful for optimizing the control required along each orbit in the resonance, the control necessary to intersect with the manifold at the correct energy, and finally the control to enter a permanent orbit at Titan.

In addition, as shown in §3.3, unusual initial guesses for Earth-to-Moon transfers may be located by integrating initial conditions on the invariant manifolds using 4-body dynamics. Even though invariant manifolds only exist in the 3-body problem, their effects can still be felt in the 4-body problem. Therefore, considering the Sun-Earth and Earth-Moon 3-body systems, how do their manifolds behave if integrated from the periodic orbits using 4-body dynamics? Locating intersecting trajectories with desired behavior at the Earth and the Moon and then optimizing via DMOC may produce different transfers than those already proposed.

Time Adaptive DMOC

The version of time adapted DMOC proposed here requires that the final time be unconstrained. This could be undesirable for some problems, so a method that allows the final time to be fixed should be explored. In addition, since time adaption leads to an indirect optimization method, a different formulation for time adapted DMOC that preserves its status as a direct method should be examined. Also, it would be interesting to compare time adapted DMOC with regular DMOC using initial guesses employing Bett's mesh refinement strategy to design the time grid. Which strategy generates the most accurate optimal solutions? Furthermore, it is unclear whether time adaption is possible with $\sigma(q, t)$ instead of $\sigma(q)$. For example, a problem like Shoot the Moon, for which the potential forces from the Moon are time dependent, would require $\sigma(q, t)$ to ensure finer time stepping near the Moon. However, defining the time grid based on time is rather circular, so this problem should be handled with care.

Bibliography

- [1] R. Abraham and J. E. Marsden. *Foundations of Mechanics*. Addison-Wesley, New York, 2nd edition, 1978.
- [2] B. T. Barden, K. C. Howell, and M. W. Lo. Applications of dynamical systems theory to trajectory design for a libration point mission. *Journal of the Astronautical Sciences*, 45(2):161–178, 1997.
- [3] R. R. Bate, D. D. Mueller, and J. E. White. *Fundamentals of Astrodynamics*. Dover, New York, 1971.
- [4] E. A. Belbruno and J. K. Miller. Sun-perturbed earth-to-moon transfers with ballistic capture. *Journal of Guidance, Control, and Dynamics*, 16(4):770–775, 1993.
- [5] J. T. Betts. Survey of numerical methods for trajectory optimization. *AIAA J. Guidance, Control, and Dynamics*, 21(2):193–207, 1998.
- [6] J. T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming*. SIAM, 2001.
- [7] J. T. Betts and W. P. Huffman. Mesh refinement in direct transcription methods for optimal control. *Optimal Control Applications and Methods*, 19:1–21, 1998.
- [8] T. Binder, L. Blank, H. G. Bock, R. Bulirsch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J. P. Schlöder, and O. von Stryk. Introduction to model based optimization of chemical processes on moving horizons. In M.

- Grötschel, S. O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems: State of the Art*, pages 295–340. Springer, Berlin, 2001.
- [9] N. Bosanac, J. E. Marsden, A. Moore, and S. Campagnola. Titan trajectory design using invariant manifolds and resonant gravity assists. In *AAS/AIAA Space Flight Mechanics Meeting, AAS 10-170*, 2010.
 - [10] O. Camino, M. Alonso, R. Blake, D. Milligan, J. de Bruin, and S. Ricken. Smart-1: Europe’s lunar mission paving the way for new cost effective ground operations (rcsgso). In *6th International Symposium “Reducing the Costs of Spacecraft Ground Systems and Operations (RCSGSO)”,* ESA/ESOC, Darmstadt, Germany, 2005.
 - [11] C. Conley. Low energy transit orbits in the restricted three-body problem. *SIAM Journal of Applied Mathematics*, 16:732–746, 1968.
 - [12] K. E. Davis, R. L. Anderson, D. J. Scheeres, and G. H. Born. Connecting libration point orbits of different energies using invariant manifolds. In *AAS Space Flight Mechanics Meeting*, Savannah, Georgia, 2009.
 - [13] M. Dellnitz, O. Junge, M. Post, and B. Thiere. On target for Venus—set oriented computation of energy efficient low thrust trajectories. *Celestial Mechanics & Dynamical Astronomy*, 95(1–4):357–370, 2006.
 - [14] M. Dellnitz, S. Ober-Blöbaum, M. Post, O. Schütze, and B. Thiere. A multi-objective approach to the design of low thrust space trajectories using optimal control. Submitted to *Celestial Mechanics and Dynamical Astronomy*, 2008.
 - [15] R. C. Fetecau, J. E. Marsden, M. Ortiz, and M. West. Nonsmooth lagrangian mechanics and variational collision integrators. *SIAM Journal of Applied Dynamical Systems*, 2(3):381–416, 2003.
 - [16] E. S. Gawlik, J. E. Marsden, S. Campagnola, and A. Moore. Invariant manifolds, discrete mechanics, and trajectory design for a mission to titan. In *AAS/AIAA Space Flight Mechanics Meeting, AAS 09-226*, 2009.

- [17] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *Report NA 97-2*, Department of Mathematics, University of California, San Diego, CA, USA, 1997.
- [18] G. Gómez, A. Jorba, J. Masdemont, and C. Simó. Study of the transfer from the earth to a halo orbit around the equilibrium point l_1 . *Celestial Mechanics and Dynamical Astronomy*, 56:541–562, 1993.
- [19] G. Gómez, W.S. Koon, M.W. Lo, J. E. Marsden, J. Masdemont, and S. D. Ross. Invariant manifolds, the spatial three-body problem and space mission design. In *Astrodynamicics Specialist Conference*. AAS/AIAA, 2001.
- [20] A. F. Haapala and K. C. Howell. Trajectory design using periapse poincaré maps and invariant manifolds. In *AAS/AIAA Space Flight Mechanics Meeting, AAS 11-131*, 2011.
- [21] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration*. Springer-Verlag, Berlin, 2002.
- [22] K. C. Howell and M. Kakoi. Transfer between th earth-moon and sun-earth systems using manifolds and transit orbits. *Acta Astronautica*, 59:367–380, 2006.
- [23] O. Junge, J. E. Marsden, and S. Ober-Blöbaum. Discrete mechanics and optimal control. In *Proceedings of the 16th IFAC World Congress*, 2005.
- [24] L. Kharevych. *Geometric Interpretation of Physical Systems for Improved Elasticity Simulations*. Ph.D. thesis, California Institute of Technology, 2010.
- [25] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross. The genesis trajectory and heteroclinic connections. In *Astrodynamicics*, volume 103, pages 2327–2343, 1999.
- [26] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross. Heteroclinic connections between periodic orbits and resonance transitions in celestial mechanics. *Chaos*, 10(2):427–469, 2000.

- [27] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross. Shoot the moon. *Spaceflight Mechanics*, 105(2):1017–1030, 2000.
- [28] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross. Low energy transfer to the moon. *Celestial Mechanics and Dynamical Astronomy*, 81(1–2):63–73, 2001.
- [29] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross. Resonance and capture of jupiter comets. *Celestial Mechanics and Dynamical Astronomy*, 81(1–2):27–38, 2001.
- [30] W. S. Koon, M. W. Lo, J. E. Marsden, and S. D. Ross. *Dynamical Systems, The Three-Body Problem, and Space Mission Design*. Marsden Books, 2008.
- [31] B. Leimkuhler and S. Reich. *Simulating Hamiltonian Dynamics*. Cambridge University Press, Cambridge, 2004.
- [32] M. Lo. The interplanetary superhighway and the origins program. *IEEE*, 2002.
- [33] M. W. Lo, R. .L. Anderson, G. Whiffen, and L. Romans. The role of invariant manifolds in low thrust trajectory design. In *AIAA/AAS Space Flight Mechanics Meeting, AAS 04-288*, 2004.
- [34] B. G. Marchand, K. C. Howell, and J. T. Betts. Discrete nonlinear optimal control of s/c formations near the l1 and l2 points of the sun-earth/moon system. In *AAS/AIAA Conference, AAS 05-341*, 2005.
- [35] J. E. Marsden and S. D. Ross. New methods in celestial mechanics and mission design. *American Mathematical Society*, 43(1):43–73, 2005.
- [36] J. E. Marsden and M. West. Discrete mechanics and variational integrators. *Acta Numerica*, 10:357–514, 2001.
- [37] R. McGehee. *Some Homoclinic Orbits for the Restricted Three-Body Problem*. Ph.D. thesis, University of Wisconsin, 1969.

- [38] G. Mingotti and F. Topputo. Ways to the moon: A survey. In *AAS/AIAA Space Flight Mechanics Meeting, AAS 11-283*, 2011.
- [39] G. Mingotti, F. Topputo, and F. Bernelli-Zazzera. Low-energy, low-thrust transfers to the moon. *Celestial Mechanics and Dynamical Astronomy*, 2009.
- [40] S. Ober-Blöbaum. *Discrete Mechanics and Optimal Control*. Ph.D. thesis, University of Paderborn, 2008.
- [41] S. Ober-Blöbaum, O. Junge, and J. E. Marsden. Discrete mechanics and optimal control: An analysis. *Control, Optimisation and Calculus of Variations*, 2010.
- [42] L. S. Pontryagin, V. G. Boltyanski, R. V. Gamkrelidze, and E. F. Miscenko. *The Mechanical Theory of Optimal Processes*. John Wiley & Sons, 1962.
- [43] S. D. Ross. *Cylindrical Manifolds and Tube Dynamics in the Restricted Three-Body Problem*. Ph.D. thesis, California Institute of Technology, 2004.
- [44] S. D. Ross and D. J. Scheeres. Multiple gravity assists, capture, and escape in the restricted three-body problem. *SIAM Journal of Applied Dynamical Systems*, 6(3):576–596, 2007.
- [45] V. Szebehely. *Theory of Orbits: The Restricted Problem of Three Bodies*. Academic, New York, 1967.