6 Space Trajectory Optimization and L^1-Optimal Control Problems

I. Michael Ross

Department of Mechanical and Astronautical Engineering, Naval Postgraduate School, Monterey, CA 93943

Contents

6.1 Introduction

The engineering feasibility of a space mission is overwhelmingly dictated by the amount of propellant required to accomplish it. This requirement stems from the simple notion that if propellant consumption was not a prime driver, then amazing things are possible such as geostationary spacecraft in low-Earth, non-Keplerian orbits. The need for lowering the fuel consumption is so great for a space mission that optimal open-loop guidance is flown during the critical endo-atmospheric segment of launch (even for the manned Space-Shuttle) in preference to non-optimal feedback guidance. In fact, the Holy Grail of ascent guidance can be simply described as fuel-optimal feedback control [1].

The cost of fuel in space is exponentially larger than its terrestrial cost because space economics is currently driven by space transportation costs rather than the chemical composition of fuel. Recently, this simple point became more mundane when the U.S. Government was charged more than twice the peace-time market-value of gasoline due to the increased cost of transportation in a war zone [2]. That is, the cost of fuel is not just intrinsic; it is also driven by a routine of operations or the lack of it thereof. Given that space operations (access to space) are not yet routine, fuel in space continues to be extraordinarily expensive thereby dictating the feasibility of any proposed architecture.

It is worth noting that since current launch costs continue to be high, the economics of refueling an aging spacecraft need to be offset by the possibility of launching a

cheaper, advanced spacecraft. As a result of this economic fact, multiple spacecraft of undetermined number need to be refueled simply to break even [3]. Thus, in the absence of an economically viable strategy for refueling, minimum-fuel maneuvers will continue to dominate the design, guidance, control, and operations of a space system.

In principle, formulating the problem of designing minimum fuel trajectories is quite simple: the rocket equation provides the necessary physics, and the problem can be formulated either as a Mayer problem (maximizing the final mass) or as an equivalent Lagrange problem [4]. In these well-documented formulations, the mass-flow-rate equation is part of the dynamical system and one needs to explicitly account for the type of fuel used in terms of the specific impulse of the propellant. Including the coupling of the propulsion system with the mechanical system makes such a problem formulation undesirable during a preliminary phase of mission analysis as it is difficult to independently evaluate the merits of a trajectory or guidance algorithm that is intimately connected to a particular engine or propellant characteristic. Thus, mission analysts frequently use the normalizing concept of the characteristic velocity [4] that is sometimes simply referred to as the total "delta-V" requirement even when impulsive maneuvers are not employed. The most obvious way to compute these delta-Vs is to take Euclidean norms. In this chapter, we show that these Euclidean norms are part of a class of L^1 cost functions and not the popular quadratic costs. As noted in Ref. [5], this point is frequently misunderstood in the literature resulting in the design of poor guidance and control algorithms that incur fuel penalties as high as 50%. On the other hand, L^1 cost functions based on absolute values have been widely considered going back as far as the 1960s; see, for example, Ref. [6]. In the language introduced in this chapter, these early L^1 cost functions can be described as l^1-variants of the L^1 norm while the correct Euclidean-based cost functions are the l^2-variants of the L^1 norm.

In an effort to clarify the above points, this chapter begins with first principles. By considering various thruster configurations and the physics of the propulsion system, we motivate a definition of l^p-variants of the L^1 norm. That is, by considering the way the engines are mounted onto the spacecraft body we naturally arrive at l^p versions of the L^1 norm of the thrust. These class of L^1 norms of the thrust directly measure fuel consumption. By extending this definition to thrust acceleration, the resulting mathematics shows a proper way to decouple the propulsion system's performance from that of the trajectory so that a correct analysis can be carried out. Although these physics-based formulations are somewhat formal, it creates apparent problems in theory and computation because the cost function is nonsmooth (i.e., the integrand is non-differentiable). Rather than employ formal nonsmooth analysis, [7, 8] we develop an alternative approach that transforms the nonsmooth problems to problems with smooth functions while maintaining the nonsmooth geometric structure. The price we pay for this approach is an increase in the number of variables and constraints. Such transformation techniques are quite rampant in analysis; that is, the exchange of an undesirable effect to a desirable one by paying an affordable price. A well-known example of this barter in spacecraft dynamics is the parameterization of $SO(3)$: a 4-vector "quaternion" in S^3 is frequently preferred over a singularity-prone employment of three Eulerian angles.

In order to demonstrate the merits of solving the apparently more difficult nonsmooth L^1 optimal control problem, we use a double-integrator example to highlight the issues,

and motivate the practical importance of a Sobolev space perspective for optimal control. Case studies for the nonlinear problem of orbit transfer demonstrate the theory and computation of solving practical problems. Lest it be misconstrued that practical problems are essentially smooth, or that the nonsmooth effects can be smoothed, we briefly digress to illustrate points to the contrary. To this end, consider a modern electric-propulsion system. When the electric power to the engine, P_e, is zero, the thrust force, T, is zero. Thus, $(P_e, T) = (0, 0)$ is a feasible point in the power-thrust space; see Figure 6.1. As P_e is continuously increased, T remains zero until P_e achieves a threshold value, $P_{e,0}$. At $P_e = P_{e,0}$, the engine generates a thrust of $T = T_0 > 0$ as shown in Figure 6.1. This is the minimum non-zero value of thrust the engine can generate. Thus, the feasible values of thrust for a practical electric engine is given by the union of two disjoint sets,

$$T \in [T_0,\ T_{\max}(P_{\max})] \cup \{0\}\,, \tag{6.1}$$

where $T_{\max}(P_{\max})$ is the power-dependent maximum available thrust, and P_{max} is the maximum available power which may be less than the engine power capacity, $P_{e,max}$, due to housekeeping power requirements, available solar energy and a host of other real-world factors. Note that $T_{max}(P_{max}) \leq T_{sup}$ where T_{sup} is the maximum possible thrust. Thus, the practical control variable for such engines is electrical power and not thrust. In this case, the thrust force becomes part of the controlled vector field in the dynamical equations governing the spacecraft motion. Consequently, the real-world problem data is truly nonsmooth. Smoothing the data (e.g., by curve fitting) generates infeasible values of thrust [10] at worst and non-optimal controls at best—both of which are truly undesirable as already noted. Clearly, in accounting for the stringent fuel requirements of practical space missions, nonsmooth phenomena are inescapable. Thus, contrary to conventional wisdom, the more practical the problem, the more the required mathematics.

Throughout this chapter, we use the words propellant and fuel interchangeably since the differences between them are relatively irrelevant for the discussions that follow.

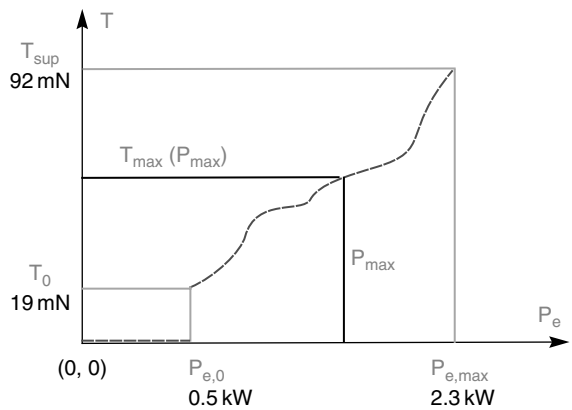


Fig. 6.1. Feasible region for a practical electric powered space propulsion system; indicated numerical values are for the NSTAR engine [9].

6.2 Geometry and the mass flow equations

Suppose that we have a single thruster that steers the spacecraft by gimbaling (see Figure 6.2 (a)). Let (x, y, z) be orthogonal body-fixed axes and $\mathbf{T} = (T_x, T_y, T_z) \in \mathbb{R}^3$ be the thrust force acting on a spacecraft. Then, the rocket equation is given by,

$$\dot{m} = -\frac{\sqrt{T_x^2 + T_y^2 + T_z^2}}{v_e} = -\frac{\|\mathbf{T}\|_2}{v_e} \tag{6.2}$$

where v_e is the exhaust speed, $\dot{m}$ is the mass-flow rate, and

$$\|\mathbf{T}\|_p := (|T_x|^p + |T_y|^p + |T_z|^p)^{1/p}$$

is the l^p-norm [11] of the thrust vector. If thrusting is achieved by six (ungimbaled) identical engines (see Figure 6.2 (b)) rigidly mounted to the body axes, then we have,

$$\dot{m} = -\frac{|T_x| + |T_y| + |T_z|}{v_e} = -\frac{\|\mathbf{T}\|_1}{v_e}. \tag{6.3}$$

If we have one main engine to perform the guidance while vernier engines are used to steer the thrust vector (as in launch vehicles, for example; see also Figure 6.2 (c)), we can write,

$$\dot{m} \simeq -\frac{\|\mathbf{T}\|_\infty}{v_e}, \tag{6.4}$$

where the approximation implies that we are ignoring the fuel consumption arising from the use of the vernier engines. Thus, the rocket equation can be unified as,

$$\dot{m} = -\frac{\|\mathbf{T}\|_p}{v_e} \qquad p = 1,\ 2 \text{ or } \infty, \tag{6.5}$$

where we have ignored the fact that this equation is an approximation for $p = \infty$. Note that p is now a design option (i.e., gimbaled single engine or multiple ungimbaled engines).

The unified rocket equation holds in other situations as well. For example, in cases when it is inconvenient to use spacecraft body axes, Eq. (6.2) can be used if (T_x, T_y, T_z) are any orthogonal components of $\mathbf{T}$. In such cases, steering must be interpreted to be

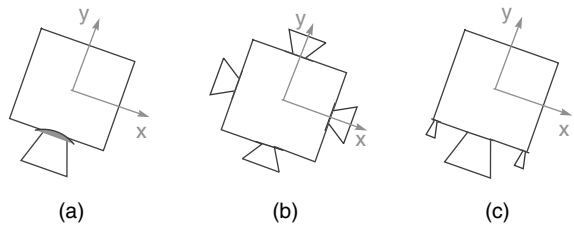


Fig. 6.2. Space vehicle thruster configurations: (a) l^2, (b) l^1, and (c) l^∞ mass flow rates; additional thrusters not shown.

provided by attitude control (with a transfer function of unity). Similarly, Eq. (6.3) can be used even when the axes are neither orthogonal nor body-fixed. The versatility of such formulations has been used quite extensively elsewhere [10, 12–15]. Finally, note that Eq. (6.5) applies whether or not the thrust region is continuous, discrete (e.g., on-off thrusters), or even disjoint as in Eq. (6.1).

In regarding T as control variable, we note that physics bounds its control authority; hence, we have $\mathbf{T} \in \mathbb{U} \subset \mathbb{R}^3$ where $\mathbb{U}$ is the control space, a compact set. Suppose that $\mathbf{T}$ can be varied continuously (i.e., $\mathbb{U}$ is a continuous set). In the l^2 mass-flow-rate configuration, a bound on the thrust implies a bound on the l^2 norm; hence the control space for this configuration is a Euclidean ball, indicated as $\mathbb{U}^2$ in Figure 6.3. On the other hand, in the l^1 mass-flow-rate configuration, bounds on the thrust generated by each thruster implies a bound on the l^∞-norm of $\mathbf{T}$. Thus, for identical engines, the control space for the l^1-configuration is the "l^∞ ball," a solid cube, denoted as $\mathbb{U}^1$, in Figure 6.3 (cutaway view).

It is instructive to look at the mass-flow rate as a region in $\mathbb{R}^3$ by associating to $|\dot{m}|$ the same direction as the net thrust force. Thus the set,

$$\mathbb{F}^p := \left\{ \mathbf{F}^p \in \mathbb{R}^3 : \mathbf{F}^p = \|\mathbf{T}\|_p \left(\mathbf{T} / \|\mathbf{T}\|_2 \right), \ \mathbf{T} \in \mathbb{U}^p \right\} \tag{6.6}$$

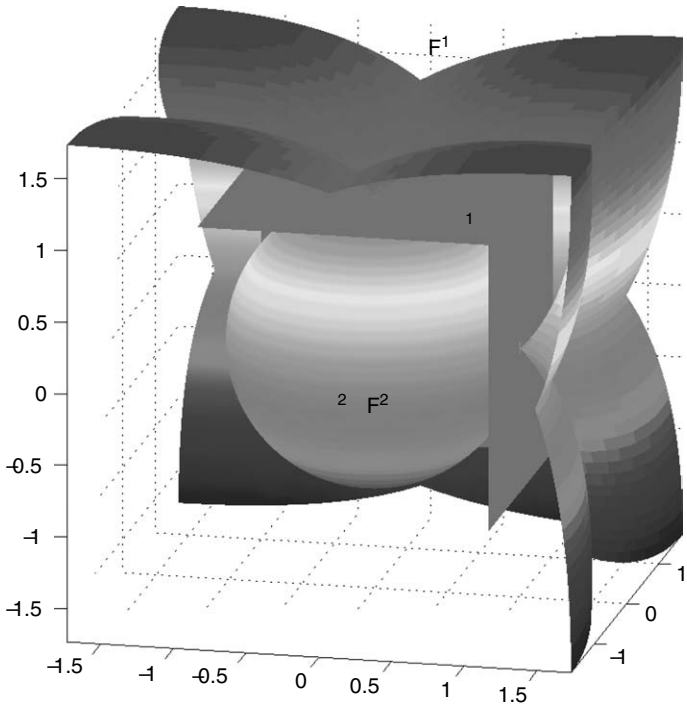


Fig. 6.3. Cutaway views of the geometries of the control space and their corresponding mass-flow rates. (see Color plate 5)

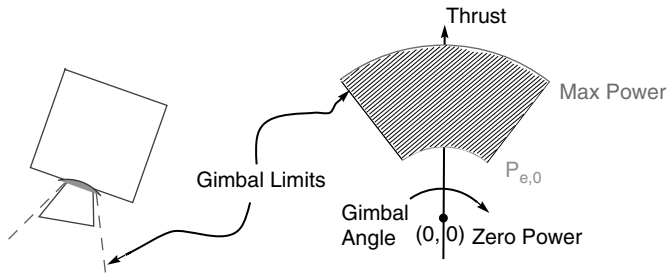


Fig. 6.4. A two-dimensional illustration of a practical control space, $\mathbb{U}$, in low-thrust trajectory optimization.

associates every net thrust firing, $\mathbf{T}$, a vector $\mathbf{F}^p$ whose Euclidean norm is the absolute value of the mass-flow-rate scaled by $-1/v_e$ (Eq. (6.5)). Clearly, we have, $\mathbb{F}^2 = \mathbb{U}^2$. On the other hand, $\mathbb{F}^1 \neq \mathbb{U}^1$. A cutaway view of the space $\mathbb{F}^1$ is the petal-shaped region shown in Figure 6.3. The mismatch between the geometries of the mass-flow-rate and the control space can generate some apparently peculiar control programs and fuel consumptions. Although not articulated in terms of geometric mismatches, it was Bilimoria and Wie [16] who first showed that a mismatch between the inertia ellipsoid (a sphere in their example), and the control space (an l^∞ ball) generates counter-intuitive time-optimal maneuvers in the sense that the rigid-body rotations are almost always not about the eigenaxis. This phenomenon was re-discovered in Ref. [17].

In practical applications, the control space, $\mathbb{U}$, can be quite different from the sets discussed above, and these characteristics can lead to quite interesting controllers. For example, in the l^2 mass-flow-rate configuration (see Figure 6.2 (a)), if the engine gimbals are limited and the propulsion is electric, then $\mathbb{U}$ is a non-convex disjoint set as illustrated in Figure 6.4 (see also Eq. (6.1)). Thus, solving practical problems requires a more careful modeling of the control space, and quite often, $\mathbb{U}$ has a complex geometric structure arising from systems' engineering considerations such as the placement of the thrusters, cant angles, and so on. Our intent here is not to document these issues but to simply note that as a result of the structure of $\mathbb{U}$, practical optimal trajectories [10, 15] can differ substantially from textbook cases [4, 18].

Although our focus here is largely thrusting space vehicles, we note that all of the preceding notions apply to air vehicles as well. This is because, for air vehicles, the mass-flow equations are the same except that one uses $c = 1/v_e$ as the thrust-specific fuel consumption parameter.

6.3 Cost functions and Lebesgue norms

Propellant consumption is simply the change in mass of the spacecraft. If v_e is a constant, then from Eq. (6.5) we have

$$m(t_0) - m(t_f) = -\int_{t_0}^{t_f} \dot{m}\,\mathrm{d}t = \frac{1}{v_e}\int_{t_0}^{t_f} \|\mathbf{T}(t)\|_p\,\mathrm{d}t, \tag{6.7}$$

where $\mathbf{T}(\cdot) : [t_0, t_f] \to \mathbf{T} \in \mathbb{R}^3$ is the thrust vector function of time. Thus, we can say that the L^1-norm of the scalar function, $[t_0, t_f] \mapsto \|\mathbf{T}\|_p \in \mathbb{R}$, is a measure of the fuel consumption, and is, in fact, equal to the propellant consumption with a proportionality factor, $1/v_e$. If v_e is not a constant, then of course $1/v_e$ must be inside the integral in Eq. (6.7) and takes the role of a weight function. Thus, in performing minimum-fuel analysis independent of the propulsion system, it is obvious from Eq. (6.7), that the proper family of cost functions is indexed by p and can be defined as,

$$J[\mathbf{T}(\cdot)] := \int_{t_0}^{t_f} \|\mathbf{T}(t)\|_p \, \mathrm{d}t, \tag{6.8}$$

where J is the functional, $\mathbf{T}(\cdot) \mapsto \mathbb{R}$. In solving optimal control problems, it is useful to be cognizant of the space, $\mathcal{U}$, of admissible controls so that the problem formulation can be changed to search for controls in a more desirable space should the solution in a particular formulation turn out to be less than desirable. As Pontryagin et al. [19] note, $\mathcal{U}$ is frequently taken to be the (incomplete) space of piecewise continuous bounded functions for engineering applications but expanded to the space of measurable bounded functions for rigorous mathematical proofs. Deferring the implications of this observation, we simply note that $\mathbf{T}(\cdot) \in \mathcal{U}$, so that the functional J in Eq. (6.8) is understood to mean, $J : \mathcal{U} \to \mathbb{R}$. In subsequent sections, we will evaluate J from a larger space, $\mathcal{X} \times \mathcal{U} \times \mathbb{R}^n$, where $\mathcal{X}$ is the function space corresponding to the state variable so that the functional J is understood to mean, $J : \mathcal{X} \times \mathcal{U} \times \mathbb{R}^n \to \mathbb{R}$. It will be apparent later that the proper space for $\mathcal{X}$ (and $\mathcal{U}$) is a Sobolev space [20], as it forms the most natural space for both theoretical [21] and computational considerations [22].

6.3.1 Quadratic cost is not $p = 2$

By a minor abuse of notation, we denote by J_2 the cost function for $p = 2$; thus, by setting $p = 2$ in Eq. (6.8) we have,

$$(J_2)^2 = \left(\int_{t_0}^{t_f} \|\mathbf{T}(t)\|_2 \, \mathrm{d}t \right)^2. \tag{6.9}$$

Similarly, let J_Q denote the standard quadratic cost function; then, we have,

$$\begin{aligned} J_Q &:= \int_{t_0}^{t_f} (T_x^2(t) + T_y^2(t) + T_z^2(t)) \, \mathrm{d}t \\ &= \int_{t_0}^{t_f} \|\mathbf{T}(t)\|_2^2 \, \mathrm{d}t \\ &\neq \left(\int_{t_0}^{t_f} \|\mathbf{T}(t)\|_2 \, \mathrm{d}t \right)^2. \end{aligned}$$

Thus,

$$(J_2)^2 \neq J_Q.$$

The importance of this observation is that integration does not commute with the operation of taking powers. Thus, the oft-used argument that minimizing a quantity is the same as minimizing its square applies to J_2^2, which measures fuel consumption, but minimizing J_2 is not the same as minimizing J_Q. In physical terms, this is equivalent to noting that $v_e^2\left(m(t_0)-m(t_f)\right)^2 \neq J_Q$; see Eq. (6.7).

6.3.2 Fuel expenditures are measured by L^1 norms

For a scalar-valued function, $f:\mathbb{R}\supseteq\Omega\to\mathbb{R}$, the L^p-norm of f, denoted by $\|f\|_{L^p}$ $(<\infty)$ is defined by [11],

$$\|f\|_{L^p} := \left(\int_\Omega |f(t)|^p \,\mathrm{d}t\right)^{1/p}, \tag{6.10}$$

where $|\cdot|$ denotes the absolute value. For vector-valued functions, $\mathbf{f}:\mathbb{R}\supseteq\Omega\to\mathbb{R}^n$, $n>1$, the L^p-norm, $\|\mathbf{f}\|_{L^p}$ is frequently defined to be derived from Eq. (6.10) with $|\cdot|$ replaced by the Euclidean norm in $\mathbb{R}^n$. Thus, for example, if $n=2$ so that $\mathbf{f}(t)=(f_1(t),f_2(t))$, then, by this definition of a norm, $\|\mathbf{f}\|_{L^p}$ is given by,

$$\|\mathbf{f}\|_{L^p} := \left(\int_\Omega \left(\sqrt{f_1^2(t)+f_2^2(t)}\right)^p \mathrm{d}t\right)^{1/p}.$$

Applying this definition for the function, $\mathbf{T}(\cdot)$, we get,

$$\|\mathbf{T}(\cdot)\|_{L^1} = \int_{t_0}^{t_f} \sqrt{T_x^2(t)+T_y^2(t)+T_z^2(t)}\mathrm{d}t$$

$$\|\mathbf{T}(\cdot)\|_{L^2}^2 = \int_{t_0}^{t_f} (T_x^2(t)+T_y^2(t)+T_z^2(t))\mathrm{d}t.$$

Clearly, $J_Q=\|\mathbf{T}(\cdot)\|_{L^2}^2$, the L^2-norm of $\mathbf{T}(\cdot)$ and as shown in the previous subsection does not measure fuel. On the other hand, $\|\mathbf{T}(\cdot)\|_{L^1}$, does indeed measure fuel consumption and follows from Eq. (6.8) with $p=2$.

Since finite-dimensional norms are equivalent, we can also define the L^p-norm of a vector-valued function, $\mathbf{f}$, in Eq. (6.10) with $|\cdot|$ replaced by the l^1 norm in $\mathbb{R}^n$. Thus, for $\mathbf{f}(t)=(f_1(t),f_2(t))$, we can define, $\|\mathbf{f}\|_{L^p}$, as

$$\|\mathbf{f}\|_{L^p} := \left(\int_\Omega \big(\,|f_1(t)|+|f_2(t)|\,\big)^p \mathrm{d}t\right)^{1/p}.$$

Using this definition, we get,

$$\|\mathbf{T}(\cdot)\|_{L^1} = \int_{t_0}^{t_f} \big(|T_x(t)|+|T_y(t)|+|T_z(t)|\big)\mathrm{d}t$$

$$\|\mathbf{T}(\cdot)\|_{L^2}^2 = \int_{t_0}^{t_f} \big(|T_x(t)|+|T_y(t)|+|T_z(t)|\big)^2 \,\mathrm{d}t.$$

From Eq. (6.8), by substituting $p=1$, it follows that $\|\mathbf{T}(\cdot)\|_{L^1}$ is indeed a measure of the fuel consumption while $\|\mathbf{T}(\cdot)\|_{L^2}$ once again fails the test.

6.3.3 L^1 cost and l^p geometry

In addition to performing minimum-fuel analysis independent of the the propulsion system, one sometimes prefers to ignore the change in mass, particularly if the burn time is small and/or the specific impulse is high. In this case, the control may be taken as the thrust acceleration, $\mathbf{u} = \mathbf{T}/m$. By using the same arguments leading to Eq. (6.8), we can now state a fundamental result: the cost functions for minimum fuel control are a family of L^1-norms of the control function, $t \mapsto \mathbf{u}$. Specifically, the minimum fuel cost (see Eq. (6.8)) is the L^1-norm of the l^p-norm function $[t_0, t_f] \mapsto \|\mathbf{u}\|_p \in \mathbb{R}$

$$J[\mathbf{u}(\cdot)] = \int_{t_0}^{t_f} \|\mathbf{u}(t)\|_p \, \mathrm{d}t, \tag{6.11}$$

where we may use $\mathbf{u}$ to be either the thrust or the acceleration with the latter form of the control accompanied by the caveat mentioned above. Among others, one possible reason why this "l^p-variant" of the L^1-norm is not used as a cost function is that the running cost, i.e., the integrand in Eq. (6.11), is not differentiable with respect to the parameter $\mathbf{u}$. Deferring the details of the implications of this non-differentiability, we note that the Pontryagin version [19] of the Minimum (Maximum) Principle does not require differentiability of the integrand with respect to the control parameter; only differentiability with respect to the states is required. Nonetheless, it is worth noting that new versions of the Minimum Principle [8, 21, 23] do not even require differentiability with respect to the states: thanks to the era of nonsmooth analysis pioneered by Clarke, Sussmann and others [8, 23–25].

6.3.4 Penalty for not using the L^1 cost

The penalty in propellant consumption for designing trajectories not based on the proper family of L^1 cost functions can be summarized by the following fundamental fact.

Proposition 6.3.1. *Given two optimal control problems, F and G, that only differ in the optimality criteria, the F-cost of the G-optimal solution can never improve the F-cost of the F-optimal solution. For minimization problems, we have,*

$$J_F[\mathbf{x}_F(\cdot), \mathbf{u}_F(\cdot)] \leq J_F[\mathbf{x}_G(\cdot), \mathbf{u}_G(\cdot)].$$

The proof of this proposition is elementary; see Ref. [5].

If we now let the functional J_F be the L^1 cost and J_G be any other cost functional (such as a quadratic cost), it is clear that the system trajectory for Problem G cannot yield better fuel performance than the L^1 cost.

In addition to penalties in fuel consumption, additional penalties may arise in the design of the control system itself. For example, the thrust force (or acceleration) appears linearly in a Newtonian dynamical system: this is a direct consequence of Newton's Laws of motion and not a simplification from linearization. In minimizing such control-affine systems, barring the possibility of singular arcs, the L^1-optimal controller has a

bang-off-bang structure. On the other hand, quadratic-cost-optimal controllers are continuous controllers. Continuous thrusting is frequently not desirable for spacecraft guidance and control since these controllers typically create undesirable effects on the payload. For example, thrusting increases the microgravity environment on the space station or induces undesirable effects on precision pointing payloads. Hence it is preferable to do much of the science during the "off periods". Thus, it is important to be cognizant of not creating new systems-engineering problems that were non-existent prior to active control considerations. The double integrator example in the next section illustrates all the main points including a quantification of the fuel penalty incurred in not using the L^1 cost.

6.3.5 A note on global optimality

Obviously, zero propellant is the absolute lowest possible cost. This fact can be mathematically stated as,

$$\inf_{\mathbf{u}(\cdot)\in\mathcal{U}} \left(\int_{t_0}^{t_f} \|\mathbf{u}(t)\|_p \, \mathrm{d}t \right) = 0,$$

where $p = 1, 2$ or ∞ as before. Thus, if the L^1 cost is zero, it is apparent that we have a globally fuel-optimal solution. In other words, there is no need to prove necessary or sufficient conditions for global optimality if the L^1 cost is zero. Such globally optimal solutions are extremely useful in the design of spacecraft formations, and are further discussed in Refs. [12–14]. An interesting consequence of the existence of such solutions is that there may be several global minima. A simple approach to finding these solutions is to design cost functionals, $J_G[\mathbf{x}(\cdot), \mathbf{u}(\cdot)]$, that are not necessarily the L^1 cost, but are such that

$$\int_{t_0}^{t_f} \|\mathbf{u}_G^*(t)\|_p \, \mathrm{d}t = 0,$$

where $\mathbf{u}_G^*$ is the control solution to some Problem G. The advantage of such problem formulations from both a theoretical and computational perspective is that the optimal system trajectories can be different from one problem formulation to another while yielding the same zero fuel cost. Thus, for example, if we were to solve a quadratic cost problem (as Problem G) and the system trajectory generated a solution such that the control was zero, then it is also a zero propellant solution. Since there is no guarantee that the state trajectory converges (theoretically and computationally) to the same trajectory as the L^1 solution, this seemingly undesirable property can be exploited to seek alternative global minimums. Such a strategy is used in Refs. [12–14] to design various spacecraft formations.

6.4 Double integrator example

The second-order control system, $\ddot{x} = u$, is widely studied [26] due to the simple reason that it is a quintessential Newtonian system: any information gleaned from a study

of double-integrators has broad implications. In this spirit, we formulate an L^1 optimal control problem as,

$$\mathbf{x}^T := [x, v] \quad \mathbf{u} := [u] \quad \mathbb{U} := \{u : |u| \leq 6\}$$

$$(L^1P) \begin{cases} \text{Minimize} & J_1[\mathbf{x}(\cdot), \mathbf{u}(\cdot)] = \int_0^1 |u(t)| \, \mathrm{d}t \\ \text{Subject to} & \dot{x} = v \\ & \dot{v} = u \\ & (x_0, v_0) = (0, 0) \\ & (x_f, v_f) = (1, 0) \end{cases}$$

Although the absolute value function, $u \mapsto |u|$, is not differentiable, the Pontryagin version of the Minimum Principle is still applicable as noted earlier. It is straightforward to show that the solution to Problem L^1P is given by,

$$u_1(t) = \begin{cases} 6 & t \in \Delta_1 \\ 0 & t \in \Delta_2 \\ -6 & t \in \Delta_3 \end{cases}$$

$$x_1(t) = \begin{cases} 3t^2 & t \in \Delta_1 \\ 3\Delta(2t - \Delta) & t \in \Delta_2 \\ 6(t + \Delta - \Delta^2) - 3(1 + t^2) & t \in \Delta_3 \end{cases}$$

$$v_1(t) = \begin{cases} 6t & t \in \Delta_1 \\ 6\Delta & t \in \Delta_2 \\ 6(1 - t) & t \in \Delta_3 \end{cases}$$

$$\lambda_{x_1}(t) = \frac{2}{2\Delta - 1}$$

$$\lambda_{v_1}(t) = \frac{1 - 2t}{2\Delta - 1},$$

where Δ_i, $i = 1, 2, 3$ are three subintervals of $[0, 1]$ defined by,

$$\Delta_1 = [0, \Delta], \quad \Delta_2 = [\Delta, 1 - \Delta], \quad \Delta_3 = [1 - \Delta, 1]$$

and

$$\Delta = \frac{1}{2} - \sqrt{\frac{1}{12}} \simeq 0.211.$$

In addition, we have,

$$J_1[\mathbf{x}_1(\cdot), \mathbf{u}_1(\cdot)] = \int_0^1 |u_1(t)| \, \mathrm{d}t = 12\Delta \simeq 2.536. \tag{6.12}$$

Now suppose that we change the cost function in Problem L^1P to a quadratic cost while keeping everything else identical; then, we can write,

$$\text{(LQP)} \begin{cases} \text{Minimize} & J_Q[\mathbf{x}(\cdot), \mathbf{u}(\cdot)] = \int_0^1 u^2(t)\,\mathrm{d}t \\ \text{Subject to} & \dot{x} = v \\ & \dot{v} = u \\ & (x_0, v_0) = (0, 0) \\ & (x_f, v_f) = (1, 0) \end{cases}$$

The optimal solution is given by,

$$\begin{aligned} u_Q(t) &= 6 - 12t \\ x_Q(t) &= t^2(3 - 2t) \\ v_Q(t) &= 6t(1 - t) \\ \lambda_{x_Q}(t) &= -12 \\ \lambda_{v_Q}(t) &= 12t - 6 \end{aligned}$$

and

$$J_Q[\mathbf{x}_Q(\cdot), \mathbf{u}_Q(\cdot)] = \int_0^1 u_Q^2(t)\,\mathrm{d}t = 12 \tag{6.13}$$

That $\max_{t\in[0,1]} |u_Q(t)| = 6$ explains why the control space in Problem L^1P was bounded accordingly.

6.4.1 LQP v/s L^1P

In comparing the performance of the two controllers, it is quite a simple matter to evaluate the L^1-cost of the quadratic control as,

$$J_1[\mathbf{x}_Q(\cdot), \mathbf{u}_Q(\cdot)] = \int_0^1 \left|u_Q(t)\right|\,\mathrm{d}t = 3.0 \tag{6.14}$$

Comparing this result with Eq. (6.12), we find that the quadratic controller is 18.3% more expensive (in fuel) than the L^1-optimal controller; obviously, a substantial margin. Further differences between the controllers are more evident in Figure 6.5. In comparing the two controllers, it is quite obvious that the L^1-controller is more desirable than the quadratic controller due to all the reasons outlined in Section 6.3.4. Quantitatively we note that we have a preferred zero-control action for approximately 58% of the time interval.

Despite the large differences between the two optimal controls, Figure 6.6 appears to indicate that there is little difference between the state trajectories. This apparently small difference comes about because plots such as Figure 6.6 do not adequately capture the true distance between two functions in the correct topology. The proper space to view

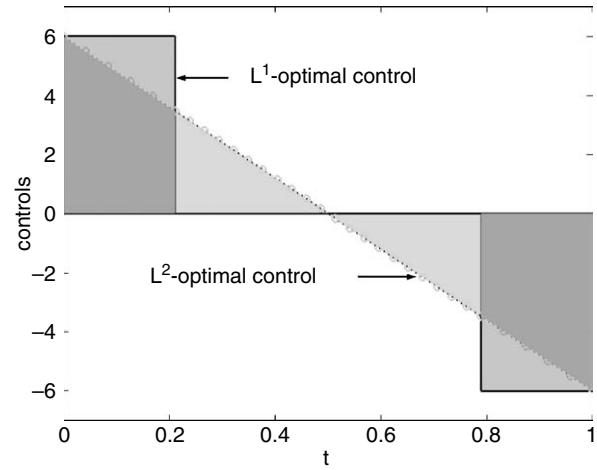


Fig. 6.5. Control plots for the quadratic and L^1-optimal control problems.

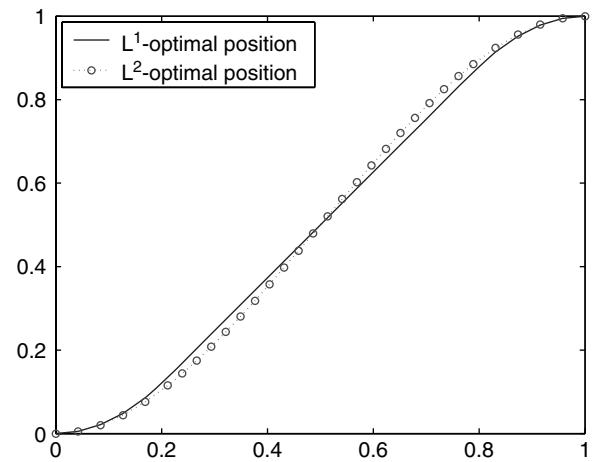


Fig. 6.6. Position plots for the quadratic and L^1-optimal control problems.

functions in control theory is a Sobolev space [20–22]. This space, denoted as, $W^{m,p}$, consists of all functions, $f : \mathbb{R} \supseteq \Omega \to \mathbb{R}$ whose jth-derivative is in L^p (see Eq. (6.10)) for all $0 \leq j \leq m$. The Sobolev norm of f is defined as,

$$\|f\|_{W^{m,p}} := \sum_{j=0}^{m} \left\|f^{(j)}\right\|_{L^p} . \tag{6.15}$$

Thus, in observing the plots in Figure 6.6 as being close to one another, we are implicitly viewing them in some L^p-norm. For example,

$$\left\| x_1(\cdot) - x_Q(\cdot) \right\|_{L^\infty} := ess \sup_{t\in[0,1]} \left| x_1(t) - x_Q(t) \right| = \max_{t\in[0,1]} \left| x_1(t) - x_Q(t) \right| \simeq 0.03.$$

When we observe the same functions in a Sobolev norm, say, $W^{1,\infty}$, then we have,

$$\left\| x_1(\cdot) - x_Q(\cdot) \right\|_{W^{1,\infty}} = \max_{t\in[0,1]} \left| x_1(t) - x_Q(t) \right| + \max_{t\in[0,1]} \left| \dot{x}_1(t) - \dot{x}_Q(t) \right| \simeq 0.30,$$

where we have replaced *ess* sup by max as before. Thus, the functions plotted in Figure 6.6 are ten times further apart in the Sobolev norm when compared to the corresponding Lebesgue norm. Since $\dot{x} = v$, the velocity plot shown in Figure 6.7 is more representative of the distance between the position functions.

The above arguments are essentially primal in flavor. A dual space perspective provides a more complete picture as covector spaces are fundamental to optimization. In this perspective [27], the position plot (Figure 6.8) must be jointly considered with the position costates. This view is quite justified by the large separation between the costates as evident in Figure 6.8. Thus, costates serve very important purposes in computational optimal control theory and are further illustrated in Section 6.7.

It is worth noting that if the control space, $\mathbb{U} = \{u : |u| \leq 6\}$, is changed to $\mathbb{U} = \mathbb{R}$, the solution to Problem LQP remains unaltered while a solution to Problem L^1P does not exist. In order to contemplate a solution to Problem L^1P for $\mathbb{U} = \mathbb{R}$, the space of admissible controls must be expanded from Lebesgue measurable functions (the assumption in the Pontryagin version of the Minimum Principle) to the space of generalized functions [28].

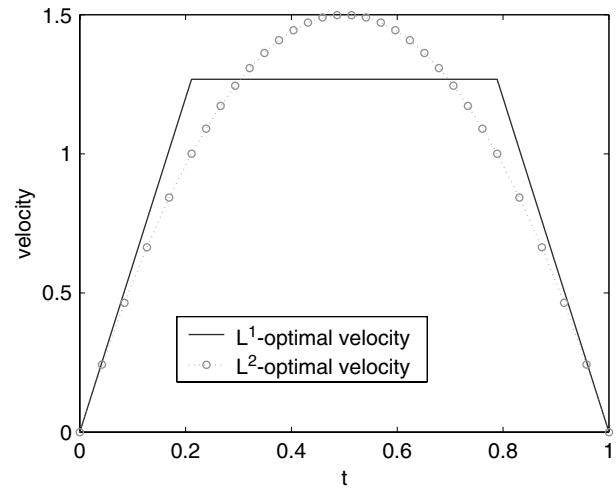


Fig. 6.7. Velocity plots for the quadratic and L^1-optimal control problems.

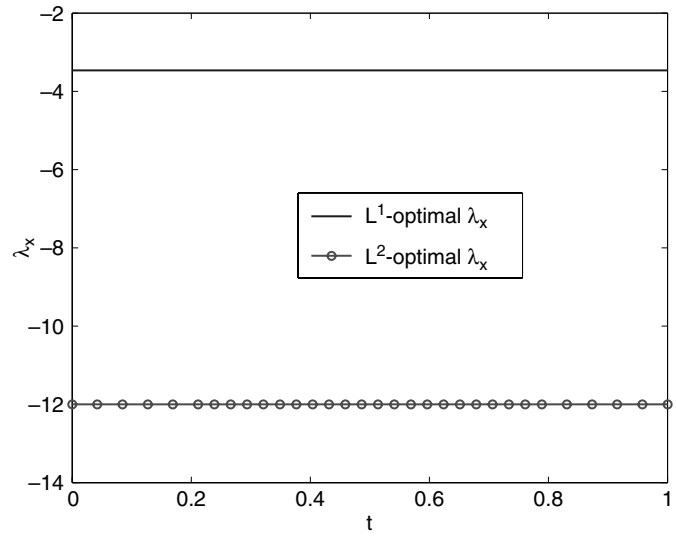


Fig. 6.8. Dual position trajectories for the quadratic and L^1-optimal control problems.

Circumventing these technicalities by using a continuation method of Lawden [29], it is straightforward to show [5] that the optimal control is given by

$$u_\delta(t) = \delta(t) - \delta(1-t)$$

where δ is the Dirac delta function. The states are then given by,

$$x_\delta(t) = t$$

$$v_\delta(t) = \begin{cases} [0,1] & t = 0 \\ 1 & t \in (0,1), \\ [0,1] & t = 1 \end{cases}$$

where $v_\delta(t)$ is expressed in a set-valued form consistent with nonsmooth calculus; [8] see also Ref. [30] for a practical demonstration of nonsmooth concepts. In this spirit, the L^1-cost of impulse control is given by,

$$J_1[\mathbf{x}_\delta(\cdot), \mathbf{u}_\delta(\cdot)] = 2 \tag{6.16}$$

Thus, the quadratic controller (see Eq. (6.14)) is 50% more expensive than the impulse controller. Note however that the impulse cost is only a mathematical phenomenon whereas the cost obtained by solving the $L1P$ is indeed the true cost of fuel. Furthermore, these differences in cost have nothing to do with "gravity-" or "drag-loss," terminology that is quite common in orbital mechanics to describe other phenomena.

6.5 Issues in solving nonlinear L^1-optimal control problems

While the previous sections illuminated the core principles in formulating the nonsmooth L^1 problems and the penalties incurred in solving "simpler" smooth problems, the approach used in Section 6.4 is not portable to solving astrodynamical systems because closed-form solutions to even simple optimal control problems are unknown. In order to frame the key issues in computing L^1-optimal controls for general dynamical systems, we summarize the problem statement as,

$$\text{(B)} \begin{cases} \text{Minimize} & J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_0, t_f] = \displaystyle\int_{t_0}^{t_f} \|\mathbf{u}(t)\|_p \, \mathrm{d}t \\ \text{Subject to} & \dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ & \mathbf{u}(t) \in \mathbb{U} \\ & (\mathbf{x}_0, \mathbf{x}_f, t_0, t_f) \in \mathbb{E} \end{cases},$$

where $\mathbb{E} \subset \mathbb{R}^{N_x} \times \mathbb{R}^{N_x} \times \mathbb{R} \times \mathbb{R}$ is some given endpoint set and $\mathbb{U}$ is a compact set as before. State constraints of the form, $\mathbf{x}(t) \in \mathbb{X}$ can also be added to the problem, but we focus on Problem B as formulated above to only limit the scope of the discussion; the ideas extend to these situations as well. The functional J is the map, $\mathcal{X} \times \mathcal{U} \times \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$. As indicated earlier, although we typically take $\mathcal{X} = W^{1,1}$ for theoretical purposes, we limit $\mathcal{X}$ to the space $W^{1,\infty}$ for computation. Summarizing the result of Section 6.3.5, we have,

Proposition 6.5.2. *Any tuple,* $(\mathbf{x}^*(\cdot), \mathbf{u}^*(\cdot), t_0^*, t_f^*)$*, for which* $J[\mathbf{x}^*(\cdot), \mathbf{u}^*(\cdot), t_0^*, t_f^*] = 0$ *is a globally optimal solution to Problem B.*

There are essentially three methods for solving optimal control problems [31], all of which require a careful analysis of the Hamiltonian Minimization Condition [23] (HMC).

6.5.1 The Hamiltonian minimization condition

At each instant of time, t, the HMC is a nonsmooth static optimization problem,

$$\text{(HMC)} \begin{cases} \underset{\mathbf{u}}{\text{Minimize}} & H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}) = \|\mathbf{u}\|_p + \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \text{Subject to} & \mathbf{u} \in \mathbb{U} \end{cases},$$

where H is the *control* Hamiltonian [23]. In the framework of the Minimum Principle, $\boldsymbol{\lambda} \in \mathbb{R}^{N_x}$ is the costate where $t \mapsto \boldsymbol{\lambda}$ satisfies the adjoint equation while in Bellman's dynamic programming framework, $\boldsymbol{\lambda} = \partial\varphi/\partial\mathbf{x}$ where, $\varphi : \mathbb{R} \times \mathbb{R}^{N_x} \to \mathbb{R}$, is a function that satisfies the Hamilton–Jacobi–Bellman (HJB) partial differential equation [8],

$$\mathcal{H}\left(\varphi_{\mathbf{x}}(t, \mathbf{x}), \mathbf{x}\right) + \varphi_t(t, \mathbf{x}) = 0, \tag{6.17}$$

where $\mathcal{H} : \mathbb{R}^{N_x} \times \mathbb{R}^{N_x} \to \mathbb{R}$ is the *lower* Hamiltonian [8] defined as,

$$\mathcal{H}(\boldsymbol{\lambda}, \mathbf{x}) := \min_{\mathbf{u} \in \mathbb{U}} H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}). \tag{6.18}$$

In recognizing that Problem HMC is fundamental to solving optimal control problems, we discuss some key issues pertaining to this problem.

6.5.2 Issues in solving Problem HMC

In Section 6.4, the control variable was one-dimensional ($N_u = 1$). This facilitated solving Problem HMC simply by inspection without resorting to nonsmooth calculus [8]. To solve problems in higher dimensional spaces, we need a more systematic procedure. Rather than resort to formal nonsmooth analysis, a procedure that is tenable to both analysis and computation is to convert the nonsmooth HMC to an equivalent problem where the functions describing the objective function and the constraint set are smooth. Such conversion techniques, well-known in nonlinear programming, can be achieved by exchanging the complication of nonsmoothness in a lower dimensional space to a simpler problem in higher dimensions. As noted in Section 6.1, similar trades are rampant in engineering analysis. In order to focus our attention to the conversion issue, we demonstrate this procedure for the HMC by limiting the scope of the problem to the case when $\mathbf{f}$ is differentiable with respect to $\mathbf{u}$, and $\mathbb{U}$ is given in terms of inequalities as follows,

$$\mathbb{U} := \left\{ \mathbf{u} \in \mathbb{R}^{N_u} : \mathbf{h}^L \leq \mathbf{h}(\mathbf{u}) \leq \mathbf{h}^U \right\},$$

where $\mathbf{h} : \mathbb{R}^{N_u} \to \mathbb{R}^{N_h}$ is a differentiable function and $\mathbf{h}^L, \mathbf{h}^U \in \mathbb{R}^{N_h}$ are the lower and upper bounds on the values of the function $\mathbf{h}$ respectively. Much of the analysis to follow easily extends to more complex situations (see for example, Section 6.8 of this chapter and Ref. [32]), but our intent here is not an enumeration of these situations but to demonstrate a methodology. Hence, we choose to illustrate the concepts for one of the most prevalent cases in engineering applications.

As noted previously, the function, $\mathbf{u} \mapsto H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u})$, is nonsmooth because $\|\mathbf{u}\|_p$ is nonsmooth. For example, when $p = 2$ and $N_u = 3$,

$$\|\mathbf{u}\|_2 = \sqrt{u_1^2 + u_2^2 + u_3^2}.$$

The function, $\mathbf{u} \mapsto \|\mathbf{u}\|_2$, is not differentiable at the origin $(0, 0, 0)$. This is illustrated in Figure 6.9 for $\mathbf{u} \in \mathbb{R}^2$. As Betts [33, 34] notes, this single point can cause major problems in computation. The singular point cannot be ignored even in a theoretical framework as it is the most desirable point: as evident in Section 6.4, it occurs for about 58% of the time interval in the solution to Problem L^1P. That is, one point in the solution to Problem HMC can easily get smeared over a substantial time interval. In mathematical terms, this is simply an effect of the chain rule in evaluating the derivative of $t \mapsto \|\mathbf{u}\|_2$ by way of the gradient of $\mathbf{u} \mapsto \|\mathbf{u}\|_2$. Noting that the function $\mathbf{u} \mapsto \|\mathbf{u}\|_2^2$ is differentiable,

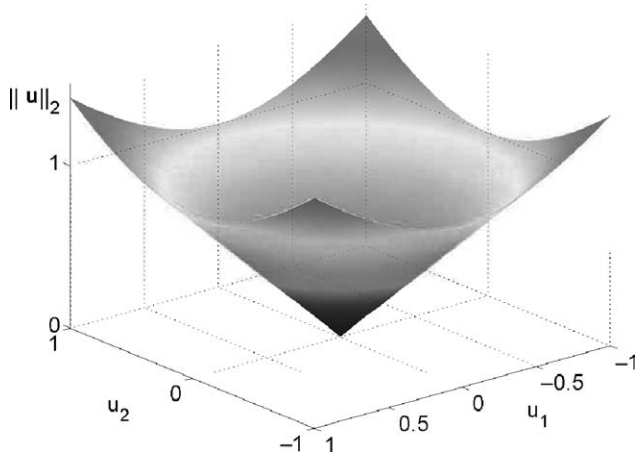


Fig. 6.9. Illustrating the nonsmooth structure of $\mathbf{u} \mapsto \|\mathbf{u}\|_2$. (see Color plate 6)

the nonsmooth HMC for $p = 2$ can be converted to a smooth one by introducing a pseudo-control variable $u_4 := \|\mathbf{u}\|_2$. That is, Problem HMC for $p = 2$ (and $N_u = 3$) can be transformed to a smooth nonlinear programming (Nl^2P) problem in an augmented control variable, $\mathbf{u}_a \in \mathbb{R}^{N_u+1}$, as,

$$\mathbf{u}_a^T := [\mathbf{u}^T, u_4] \equiv [u_1, u_2, u_3, u_4]$$

$$(\mathrm{N}l^2\mathrm{P}) \begin{cases} \underset{\mathbf{u}_a}{\text{Minimize}} & H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}_a) = u_4 + \boldsymbol{\lambda}^T \mathbf{f}(\mathbf{x}, \mathbf{u}) \\ \text{Subject to} & \mathbf{u} \in \mathbb{U} \\ & \|\mathbf{u}\|_2^2 - u_4^2 = 0 \\ & u_4 \geq 0 \end{cases},$$

where we have retained the use of the symbol H for the new Hamiltonian by a minor abuse of notation. Since the original problem was nonsmooth, the inequality, $u_4 \geq 0$, essentially retains the nonsmooth geometric structure of the problem although the function used in the inequality is now differentiable. Thus, the standard Karush–Kuhn–Tucker (KKT) conditions for Problem Nl^2P can be applied. The minimum-fuel orbit transfer example discussed in Section 6.7 further discusses the KKT conditions in conjunction with the larger problem of actually solving the optimal control problem.

The situation for $p = 1$ is similar, except that it requires the introduction of many more control variables. This is because the function,

$$\mathbf{u} \mapsto \|\mathbf{u}\|_1 = |u_1| + |u_2| + |u_3|$$

is non-differentiable at the origin, $(0, 0, 0)$, as well as all other points where $u_i = 0$, $i = 1, 2, 3$ (see Figure 6.10). By introducing variables, $v_i \geq 0$, $w_i \geq 0$, $i = 1, 2, 3$, the

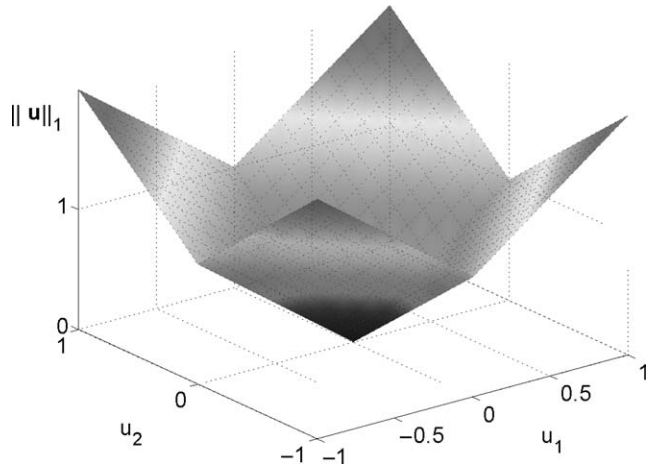


Fig. 6.10. Illustrating the nonsmooth structure of $\mathbf{u} \mapsto \|\mathbf{u}\|_1$. (see Color plate 7)

nonsmooth HMC problem for $p = 1$ can be transformed to a smooth nonlinear programming (Nl^1P) problem in an augmented control variable, $\mathbf{u}_a \in \mathbb{R}^{2N_u}$, as,

$$\mathbf{u}_a := [\mathbf{v}, \mathbf{w}]$$

$$(\mathrm{N}l^1\mathrm{P}) \begin{cases} \underset{\mathbf{u}_a}{\text{Minimize}} & H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}_a) = \mathbf{1}^T \mathbf{u}_a + \boldsymbol{\lambda}^T \widetilde{\mathbf{f}}(\mathbf{x}, \mathbf{u}_a) \\ \text{Subject to} & \widetilde{\mathbf{h}}^L \le \widetilde{\mathbf{h}}(\mathbf{u}_a) \le \widetilde{\mathbf{h}}^U \\ & \mathbf{v} \ge \mathbf{0} \\ & \mathbf{w} \ge \mathbf{0} \end{cases},$$

where $\mathbf{1}$ is an $\mathbb{R}^{2N_u}$-vector of ones and tildes over the symbols implies transformed functions and variables when $\mathbf{u}$ is transformed to $\mathbf{u}_a$. For example, when $\mathbf{f}$ is control-affine,

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{a}(\mathbf{x}) + \mathbf{B}(\mathbf{x})\mathbf{u},$$

where $\mathbf{a} : \mathbb{R}^{N_x} \to \mathbb{R}^{N_x}$ and $\mathbf{B} : \mathbb{R}^{N_x} \to \mathbb{R}^{N_x \times N_u}$, then $\widetilde{\mathbf{f}}$ is given by,

$$\widetilde{\mathbf{f}}(\mathbf{x}, [\mathbf{v}, \mathbf{w}]) = \mathbf{a}(\mathbf{x}) + \mathbf{B}(\mathbf{x})[\mathbf{v} - \mathbf{w}].$$

Once Problem HMC is converted to an NLP with smooth functions, the KKT conditions then describe the necessary conditions for a putative optimal controller.

6.5.3 HMC on HJB

A cursory examination of Problems Nl^1P and Nl^2P reveal that it may be quite difficult to obtain a closed-form solution. An examination of the KKT conditions for these problems strengthen this observation which has far-reaching consequences.

In the absence of a closed-form solution to Problem HMC, an explicit expression for the map, $(\boldsymbol{\lambda}, \mathbf{x}) \mapsto \mathbf{u}$, cannot be obtained. This means that the lower Hamiltonian (Eq. (6.18)) cannot be constructed explicitly. That is, it would be impossible to even write down explicitly the HJB partial differential equation. This elementary observation almost immediately eliminates the HJB as a practical means for solving problems beyond academic ones.

In cases where the controls can be eliminated, the HJB suffers from at least two additional well-known problems [8, 21, 26] that merit recounting. As is the case for a large number of problems, a differentiable solution to the HJB does not exist for the L^1-optimal control problem; however, if the notion of differentiability is expanded along the lines of nonsmooth analysis, then, according to the celebrated result of Crandall and Lions [8, 21], the Bellman value function is a unique viscosity solution to the HJB. This theoretical breakthrough has not yet translated to practical problem solving, as even smooth partial differential equations continue to be challenging problems.

The third problem associated with Eq. (6.17) is the absence of good computational techniques for solving partial differential equations involving more than three independent variables. Even for a coplanar orbit transfer problem (discussed further in Section 6.7), $N_x = 4$. For practical three-dimensional space models, $N_x = 6$; hence, the number of independent variables in φ is seven. Given that the vast majority of computational techniques for solving partial differential equations is limited to two independent variables, it is clear that solving the HJB for a practical problem is far from feasible.

It is worth noting at this stage that even if the HJB can be solved numerically, it loses one of its major attractions: the ability to generate feedback solutions in closed form. This is simply because, a numerical solution to the HJB implies a table lookup data for feedback control or an approximation at best for a closed-form solution by way of a surface fit for Bellman's value function. Thus, although the Hamilton–Jacobi framework is quite elegant, the absence of a viable methodology that overcomes the major technical hurdles to solve a generic problem limits its utility to low dimensional academic problems; hence, we eliminate this approach from further consideration.

6.5.4 HMC on the Minimum Principle

Unlike the HJB framework, the Minimum Principle does not require an explicit solution to Problem HMC. This first step immediately trumps the HJB from a solvability perspective; however, an application of the Minimum Principle results in a nonlinear, differential-algebraic boundary value problem (BVP). Given that even linear differential BVPs do not have closed-form solutions, finding analytical solutions to optimal controls does appear to be quite daunting. This task is quite formidable even from a numerical perspective as the Hamiltonian BVP has a fundamental sensitivity problem that results from its symplectic structure [26]. As discussed by Bryson and Ho [26], when a shooting-type method is applied to solve the Hamiltonian BVP, the sensitivity of the initial conditions with respect to the final conditions is so large that the values of the intervening variables often exceeds the numerical range of a computer. While multiple-shooting algorithms alleviate this particular issue, the vast number of other problems associated with shooting

methods as detailed by Betts [33, 34] makes them fundamentally unsuitable for computing optimal controls.

From a modern perspective [22, 46], a BVP is essentially a problem of solving a generalized equation of the form, $0 \in \mathcal{F}(x)$, where $\mathcal{F}$ is a set-valued map. By resisting the temptation to use shooting methods, generalized equations can be solved more robustly by a combination of operator methods that retain the structure of $\mathcal{F}$ and nonlinear programming techniques [24]. Details of this approach are well documented by Betts [34] and Hager [22].

6.6 Solving nonlinear L^1-optimal control problems

As a result of the observations of the preceding paragraphs, solving optimal control problems, L^1 or otherwise, are widely perceived as difficult problems. Over the last decade, as a result of major advancements in approximation theory and optimization techniques, solving optimal control problems, particularly smooth problems, are no longer considered to be difficult. This is evident by the broad class of complex optimal control problems that have been solved with relative ease [1, 12–14, 30, 33–36]. This approach is essentially a modification and modernization of Euler's abandoned idea in solving calculus-of-variations problems combined with Lagrange's multiplier theory [37]. An early version of this approach is due to Bernoulli. This neo-Bernoulli–Euler–Lagrange approach, is encapsulated as the Covector Mapping Principle (CMP) and represents a triad of ideas for solving optimal problems [22, 31, 37]. When infused with modern computational power, the CMP facilitates real-time computation of optimal controls [38–40] thus enabling a neo-classical approach to feedback guidance and control.

6.6.1 A brief history of the covector mapping principle

According to Mordukhovich [25], Euler discovered the Euler–Lagrange equations by discretizing the fundamental problem of the calculus of variations, and then passing to the limit. Upon receiving Lagrange's letter containing ". . . a beautiful and revolutionary idea . . . Euler dropped his own method, espoused that of Lagrange, and renamed the subject the calculus of variations" [41]. Thus, the invention of direct methods [42] of the 1960s are, conceptually, Euler's abandoned idea before the limiting process [31]. Of course, modern direct methods [33, 34] typically discretize the problem using higher-order methods although Eulerian approximations continue to be widely used. A key point in modernizing Euler's original idea is the absence of the limiting process by solving the problem on a digital computer for a sufficiently fine grid, much the same way as we solve initial value problems by a Runge-Kutta method for some non-zero step size. We therefore expect the discrete solution to satisfy point wise the Euler–Lagrange equations. That this expectation does not necessarily bear fruit is one of the many reasons why indirect methods were popular (until the early 1990s) despite their well-known problems in solving symplectic (Hamiltonian) boundary-value problems [26, 31, 34].

Unlike the Euler-Lagrange approach with its distinct primal "flavor," had Euler combined his discretization approach with Lagrange's multiplier theory, the history of the

calculus of variations might have taken on an early dual flavor. This combination did not take place until 200 years later, after the discovery of Minimum Principle [25]. What is remarkable about this combination is that the discrete problem does not generally satisfy the discrete Minimum Principle without an additional assumption of convexity. Since no convexity assumptions are required for the validity of the continuous-time Minimum Principle, discrete solutions are viewed with great suspicion. Significant fodder for this suspicion is provided by higher-order methods. That is, rather than improve the quality of the solution, a higher-order discretization can lead to a completely disastrous solution [22, 35, 43]. These experiments of the late 1990s paved a way for a deeper understanding of optimal control theory by connecting the first principles to approximation theory and computation in Sobolev spaces [22, 35]. In other words, convergence of the approximation takes center stage for both theory and practice [25, 35].

6.6.2 Convergence and the covector mapping principle

The emerging issues in the neo-Bernoulli–Euler–Lagrange approach can be effectively visualized by the diagram shown in Figure 6.11. Here, Problem B is not necessarily limited to the problem discussed in Section 6.5 although our major focus continues to be the L^1-optimal control problem. The bottom of Figure 6.11 represents a generalization of Euler's initial idea of discretizing Problem B to Problem B^N where N denotes the number of discrete points. These are the classical direct methods. If convergence can be proved, then passing to the limit, $N \to \infty$, solves the original continuous problem in the limit (see the bottom convergence arrow in Figure 6.11). A convergence theorem is also a practical necessity since it ensures us that we can obtain solutions to an arbitrary precision (within the limits of digital precision). Note that Euler assumed convergence

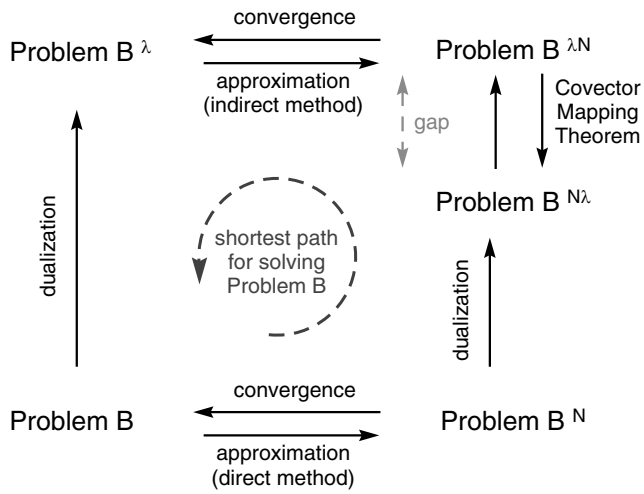


Fig. 6.11. The Covector Mapping Principle.

which can be shown to be valid for the simplest problem (general problem during Euler's days) of the calculus of variations (for Euler discretizations) but are generally invalid in the context of the Minimum Principle as indicated earlier.

When Problem B is a modern optimal control problem, Problem B^N is a nonlinear programming (NLP) problem if $\mathbb{U}$ is a continuous set; in general, it is a mixed-variable programming problem [22, 32]. Hence, Problem $B^{N\lambda}$ refers to the set of necessary conditions obtained by applying the Karush–Kuhn–Tucker (KKT) theorem. On the other hand, Problem $B^{\lambda N}$ refers to the discretization of the continuous differential-algebraic BVP obtained by applying the Minimum Principle. As indicated earlier, Problems $B^{\lambda N}$ and $B^{N\lambda}$ do not necessarily generate the same solution. That is, dualization and discretization are not commutative operations. Recent research by Hager [35] and Betts et al. [44] provide additional fodder to this concept. While Hager has shown that convergent Runge-Kutta methods fail to converge Betts et al. have shown that non-convergent methods converge. What has become clear is that a new theoretical framework is quite essential to understand seemingly contradictory results.

6.6.3 Linking theory, practice and computation

For the solution of the approximate problem (B^N) to be indistinguishable in a practical sense from some unknown exact theoretical solution to Problem B, we need to solve Problem B^N for a "sufficiently large grid". Thanks to the exponential convergence property of pseudospectral methods [45], these grids can be remarkably small. When combined with sparse efficient methods for solving NLPs, solutions to Problem B^N can be rendered virtually indistinguishable from theoretical solutions to Problem B if convergent methods (in the sense of discretization [46]) are adopted. Convergence of the discretization is sharply distinguished from the convergence of the NLP. A proper design of a computational method requires convergence analysis. The new ideas on convergence require set-valued analysis [22] and connections to the symplectic structure of Hamiltonian systems [31]. The absence of these connections lead to difficult problems or disastrous results even with convergent NLPs [35, 43]. Exploiting the global convergence properties of modern NLP algorithms [47] with relaxation techniques in discretization [31] implies that optimal control problems can be solved routinely.

The statements of the preceding paragraph are deeply theoretical since modern computational methods facilitate a practical demonstration of "epsilons, deltas, limits and sequences," the hallmark of functional analysis. Thus the practice of optimal control today is more firmly rooted and integrated with theory than ever before. This point is better understood by way of Figure 6.11; here, Problem B^λ is the Hamiltonian BVP discussed earlier and Problem $B^{\lambda N}$ represents the approximation (recall that any numerical method requiring a digital computer is an approximation). The reason certain well-known discretization methods (like a class of Runge-Kutta methods [35, 43]) fail for optimal control problems is that dualization and discretization are non-commutative operations indicated by the commutation gap shown in Figure 6.11. A zero gap does not guarantee convergence while convergence does not guarantee zero gap (except in the limit). In principle, this gap can be closed for finite N if there exist an order-preserving map

between the duals [35, 46, 48]. Such maps have been obtained (i.e., Covector Mapping Theorems) for a special class of symplectic Runge-Kutta [35] methods and modifications [48] to pseudospectral methods [49]. Thus the Covector Mapping Principle essentially encapsulates the approximation issues that started with the work of Bernoulli, Euler and Lagrange [37]. It is thus apparent that the oft mentioned difficulties in solving optimal control problems can be completely circumvented today by modernizing and extending Euler's original ideas as depicted in Figure 6.11. This essentially implies that a robust general procedure that is tenable for solving practical problems is a practical combination of functional analysis with approximation theory. Indeed, in recent years, a broad class of complex optimal control problems have been solved under this framework with relative ease [1, 12–14, 30, 33–35, 48, 49]. Additional details on these ensemble of topics along with extensive references are discussed in Refs. [31] and [37].

6.6.4 Feedback guidance and control

Suppose that Problem B can be solved in real time. This means that for any $(t_0, \mathbf{x}_0)$, we can solve the optimal control problem in negligible time. Then, replacing the initial conditions by current conditions, $(t, \mathbf{x})$, it is apparent that we have a feedback map, $(t, \mathbf{x}) \mapsto \mathbf{u}$. In other words, real-time computation implies feedback control. Theoretically, real-time computation implies zero computation time; in practice, the real issue is the measurable effect, if any, of a non-zero computation time. Stated differently, a key issue in feedback control is the required minimum computational speed for feedback implementation rather than the imposition of the theoretical real-time computation of optimal controls. If we had perfect models and a deterministic system, feedback would be unnecessary provided the perfect model was used in the computation of the control. In other words, the higher the fidelity of the models used in the computation of control, the less the demand on real-time computation. Further, the need for computational speed is less if the time constant of the system is larger. Thus, if the system time constant is large and reasonably high fidelity models are chosen for the computation of control, implementing feedback controls by way of online optimization is not a difficult problem. These are precisely the conditions for orbit control: the time constant of a low Earth orbit (LEO) is the orbital period of about 90 minutes and nonlinear models of relatively high accuracy are available. Hence, if recomputed optimal thrusting programs were to be available every minute for LEO spacecraft, then it is possible to implement a sampled-data feedback control with 90 samples per orbit. As demonstrated in the next section and elsewhere [5, 10, 15], minimum-fuel orbit transfer problems can be solved on Pentium 4 computers in under 30 seconds (thus implying the possibility of 180 samples for LEO). Faster computational speeds are easily possible [38] with optimized code and/or by removing the overhead associated with the operating system (Windows) and the problem solving environment (MATLAB). For example, in Ref. [38], the optimal solution to a flexible robot arm was obtained in 0.03 seconds (thus making avail the possibility of a 30 Hz sampling frequency). Applications of such feedback solutions to other problems are extensively discussed elsewhere [38–40]. Thus, optimal feedback orbit control via real-time optimization is a clear modern-day reality.

6.7 L^1-Formulation of the minimum-fuel orbit transfer problem

We will now illustrate some of the ideas described in Sections 6.5 and 6.6 by solving a new formulation of the minimum-fuel orbit transfer problem. As noted earlier, the minimum-fuel orbit transfer problem is a central problem in orbit control. This problem can be easily formulated by posing it as a problem of maximizing the final mass; however, in this formulation, the astrodynamics of the problem is coupled to the propulsion system of the spacecraft by way of the specific impulse of the propellant (Eq. (6.7)). As noted in Section 6.1, it is frequently desirable to decouple the propulsion system performance from the astrodynamics of the problem by comparing the cost of a maneuver in terms of the characteristic velocity, i.e., the velocity change attributable to a generic propulsion system. This translates to using the I^2-variant of the L^1-cost. The following coplanar orbit transfer problem defines this formulation:

$$\mathbf{x}^T := [r, \theta, v_r, v_t] \quad \mathbf{u}^T := [u_r, u_t] \quad \mathbf{u} \in \mathbb{U}$$
$$\mathbb{U} := \left\{\mathbf{u} \in \mathbb{R}^2 : \ \|\mathbf{u}\|_2 \leq u_{max}\right\}$$

$$\text{(O)} \begin{cases} \text{Minimize} & J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_f] = \displaystyle\int_{t_0}^{t_f} \|\mathbf{u}(t)\|_2 \, \mathrm{d}t \\ \text{Subject to} & \dot{r} = v_r \\ & \dot{\theta} = \dfrac{v_t}{r} \\ & \dot{v}_r = \dfrac{v_t^2}{r} - \dfrac{1}{r^2} + u_r \\ & \dot{v}_t = -\dfrac{v_r v_t}{r} + u_t \\ & \mathbf{e}_0(t_0, \mathbf{x}_0) = \mathbf{0} \\ & \mathbf{e}_f(\mathbf{x}_f) = \mathbf{0} \end{cases} .$$

The functions for the endpoint conditions,

$$\mathbf{e}_0(t, \mathbf{x}) := \begin{pmatrix} t \\ a_0[(v_r^2 + v_t^2)r - 2] + r \\ r[1 + e_0 \cos(\theta - \omega_0)] - (v_t r)^2 \\ v_r[1 + e_0 \cos(\theta - \omega_0)] - e v_t \sin(\theta - \omega_0) \end{pmatrix}$$

$$\mathbf{e}_f(\mathbf{x}) := \begin{pmatrix} a_f[(v_r^2 + v_t^2)r - 2] + r \\ r[1 + e_f \cos(\theta - \omega_f)] - (v_t r)^2 \\ v_r[1 + e_f \cos(\theta - \omega_f)] - e v_t \sin(\theta - \omega_f) \end{pmatrix}$$

describe the initial and final manifolds in Problem O in terms of the initial and final orbits respectively, where (a_0, e_0, ω_0) and (a_f, e_f, ω_f) are the standard orbital elements. Except for its resemblance to the dynamical model, this problem formulation is different in every respect when compared to the continuous-thrust problem posed by Moyer and Pinkham [50] and discussed in the texts by Bryson and Ho [26] and Bryson [18].

Let $\boldsymbol{\lambda}^T := [\lambda_r, \lambda_\theta, \lambda_{v_r}, \lambda_{v_t}]$ and $\mathbf{u}_a^T := [u_r, u_t, u]$, where $u = \|\mathbf{u}\|_2$; then, the Hamiltonian Minimization Condition (see Problem Nl^2P discussed earlier), simplifies to,

$$\begin{aligned} \underset{\mathbf{u}_a}{\text{Minimize}} \quad & H(\boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}_a) = u + \lambda_{v_r} u_r + \lambda_{v_t} u_t + H_0(\boldsymbol{\lambda}, \mathbf{x}) \\ \text{Subject to} \quad & u_r^2 + u_t^2 - u^2 = 0 \\ & 0 \le u \le u_{max} \end{aligned}$$

where $H_0(\boldsymbol{\lambda}, \mathbf{x})$ denotes terms in the Hamiltonian that do not depend upon the controls. Note that the control space is a nonconvex cone in $\mathbb{R}^3$ (see Figure 6.9). The KKT conditions for this problem can be obtained by forming the Lagrangian of the Hamiltonian ($\overline{H}$),

$$\overline{H}(\boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{x}, \mathbf{u}_a) = u + \lambda_{v_r} u_r + \lambda_{v_t} u_t + H_0(\boldsymbol{\lambda}, \mathbf{x}) + \mu_1 u + \mu_2 (u_r^2 + u_t^2 - u^2),$$

where μ_1 and μ_2 are the KKT (Lagrange) multipliers associated with the control constraints with μ_1 satisfying the complementarity condition,

$$\begin{array}{ccc} u = 0 & & \mu_1 \le 0 \\ 0 < u < u_{max} & \Leftrightarrow & \mu_1 = 0 \\ u = u_{max} & & \mu_1 \ge 0, \end{array} \tag{6.19}$$

while μ_2 is unrestricted in sign. Thus, the function, $t \mapsto \mu_1$, supplies the switching information. The vanishing of the gradient of the Lagrangian of the Hamiltonian, $\partial \overline{H} / \partial \mathbf{u}_a$, provides three additional necessary conditions,

$$\lambda_{v_r} + 2\mu_2 u_r = 0 \tag{6.20}$$

$$\lambda_{v_t} + 2\mu_2 u_t = 0 \tag{6.21}$$

$$1 + \mu_1 - 2\mu_2 u = 0. \tag{6.22}$$

From Eqs. (6.19) and (6.22) it follows that

$$u = 0 \Rightarrow \mu_1 = -1.$$

This result is quite interesting. If the optimal control program is not identically equal to zero (i.e., zero cost), the function $t \mapsto \mu_1$ must jump at the points where $t \mapsto u = \|\mathbf{u}\|_2$ goes from zero to some non-zero value either via singular (i.e., $\mathbf{u} \in int\, \mathbb{U}$) or bang-bang (i.e., $\mathbf{u} \in bdry\, \mathbb{U}$) thrusting. That this phenomenon does indeed occur is shown in Figure 6.12 for a sample solution corresponding to the following case:

$$a_0 = 1,\ a_f = 2,\ e_0 = 0.1,\ e_f = 0.2,\ \omega_0 = 1,\ \omega_f = 2,\ u_{max} = 0.05.$$

The plot shown in Figure 6.12 was *not* obtained by solving the "difficult" Hamiltonian BVP (i.e., an "indirect method" indicated in Figure 6.11), rather, it was obtained quite readily by an application of the CMP to the Legendre pseudospectral method [48]. In fact, Problem *O* was easily solved by way of the software package DIDO [51]. DIDO is a minimalist's approach to solving optimal control problems: only the problem formulation is required, and in a form that is almost identical to writing it on a piece of paper and

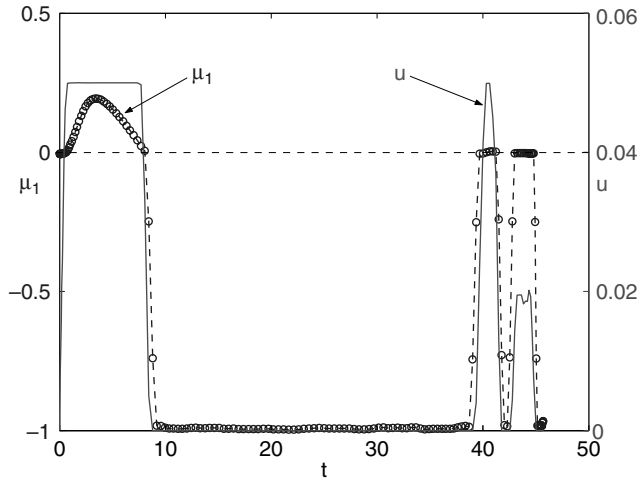


Fig. 6.12. Demonstrating the Hamiltonian Minimization Condition for Problem *O*; note the singular control and the vanishing of the switching function.

pencil. The latter is facilitated by the use of object-oriented programming readily available within the MATLAB problem solving environment.

A number of features are noteworthy in Figure 6.12. Observe the excellent correlation between the switching function, $t \mapsto \mu_1$, and the control trajectory, $t \mapsto u$, in conformance with the equations resulting from the Hamiltonian Minimization Condition (i.e., Eq. (6.19)). The last burn appears to be a singular arc (with u taking values near 0.02) as evident by $\mu_1 = 0$ (within numerical precision). The second burn appears to be a touch point case with μ_1 near zero but its slight uptick drives u towards its maximum value of 0.05.

The optimal trajectory along with the vectoring program is shown in Figure 6.13. Strictly speaking we do not know if the computed trajectory is optimal; however, we can conclude that it is at least an extremal by verifying the necessary conditions for optimality. Thus, one of the indicators of optimality is the agreement of the switching function with the control program shown in Figure 6.12. Many other indicators of optimality can be derived by an application of the Minimum Principle. For the purposes of brevity, we do not discuss them here; extensive examples of such ideas are presented elsewhere [13–15, 31, 36, 48, 51].

6.8 A simple extension to distributed space systems

A distributed space system (DSS) is a multi-agent control system that has long been recognized [52, 53] as a key technology area to enhance the scope of both military [52] and civilian [53] space applications. While much of the challenges are in distributing the functionality of a remote sensing problem, the difficulties in the design, control and

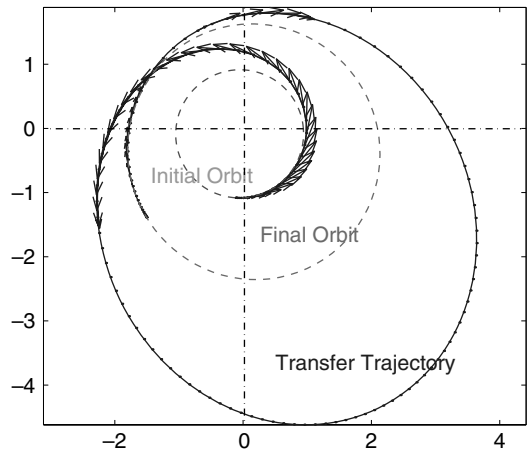


Fig. 6.13. A benchmark optimal low-thrust orbit transfer.

operations of the DSS arises chiefly as a result of managing the complexity of multiple agents. From the perspective of examining each agent separately, the problem is indeed quite formidable; however, a systems' approach to a DSS dramatically reduces some of the major problems in multi-agent control in much the same way as matrix analysis simplifies solving a system of linear equations by taking the view that a collection of linear equations is essentially one matrix equation. In order to appreciate how this perspective dramatically simplifies multi-agent control, consider a collection of $N_s \in \mathbb{N}$ spacecraft that constitute a DSS. Let $\mathbf{x}^i(t) \in \mathbb{R}^{N_{x^i}}$, $\mathbf{u}^i(t) \in \mathbb{R}^{N_{u^i}}$ denote the state and control vectors of the i^{th} spacecraft at time t. Then, the fuel consumption for any one spacecraft, i, is given by,

$$J_s[\mathbf{x}^i(\cdot), \mathbf{u}^i(\cdot), t_0, t_f] = \int_{t_0}^{t_f} \left\| \mathbf{u}^i(t) \right\|_p \mathrm{d}t \tag{6.23}$$

By defining the system state and control variables for the DSS as,

$$\mathbf{x} = (\mathbf{x}^1, \ldots, \mathbf{x}^{N_s}) \tag{6.24}$$

$$\mathbf{u} = (\mathbf{u}^1, \ldots, \mathbf{u}^{N_s}) \tag{6.25}$$

the total fuel consumption is quite simply given by,

$$J[\mathbf{x}(\cdot), \mathbf{u}(\cdot), t_0, t_f] = \sum_{i=1}^{N_s} J_s[\mathbf{x}^i(\cdot), \mathbf{u}^i(\cdot), t_0, t_f]. \tag{6.26}$$

Note that Eq. (6.26) is not an l^p variant of the L^1 norm of $\mathbf{u}(\cdot)$ except in the special case of its l^1 version. This is one of the many reasons why solving multi-agent problems becomes difficult when compared to agent-specific techniques. On the other hand, when viewed through the prism of an optimal control problem, Eq. (6.26) is yet another nonsmooth

cost functional. In certain applications, it may be necessary to require that each spacecraft in the DSS consume the same amount of propellant. This requirement can be stipulated by the constraints,

$$J_s[\mathbf{x}^i(\cdot), \mathbf{u}^i(\cdot), t_0, t_f] = J_s[\mathbf{x}^k(\cdot), \mathbf{u}^k(\cdot), t_0, t_f] \quad \forall\ i, k. \tag{6.27}$$

In generalizing Eq. (6.27) we write,

$$J_{ik}^L \leq J_s[\mathbf{x}^i(\cdot), \mathbf{u}^i(\cdot), t_0, t_f] - J_s[\mathbf{x}^k(\cdot), \mathbf{u}^k(\cdot), t_0, t_f] \leq J_{ik}^U, \tag{6.28}$$

where $J_{ik}^L, J_{ik}^U \in \mathbb{R}$ are part of the DSS requirements; for example, J_{ik}^L and J_{ik}^U may be non-zero numbers that facilitate a formulation of soft requirements on equal-fuel or rough numbers that facilitate budget allocation to the individual spacecraft. In any case, Eq. (6.28) is essentially a special case of a more general "mixed" state-control path constraint defined by,

$$\mathbf{h}^L \leq \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) \leq \mathbf{h}^U, \tag{6.29}$$

where $\mathbf{h} : \mathbb{R}^{N_x} \times \mathbb{R}^{N_u} \times \mathbb{R}^{N_p} \to \mathbb{R}^{N_h}$ and $\mathbf{h}^L, \mathbf{h}^U, \in \mathbb{R}^{N_h}$. The components of $\mathbf{h}, \mathbf{h}^L$ and $\mathbf{h}^U$ are given by Eq. (6.28) while the components of $\mathbf{p}$ are just t_0 and t_f. Such constraints are discussed in more detail in Refs. [12, 15] and [32]. It is clear that pure control constraints are naturally included in Eq. (6.29). Additional components of $\mathbf{h}$ come from topological considerations. For example, by using a generic metric (not necessarily Euclidean) to define distances between two spacecraft, the requirement that no two spacecraft collide can be written as,

$$d(\mathbf{x}^i(t), \mathbf{x}^j(t)) \geq b^{i,j} > 0 \qquad \forall\ t \text{ and } i \neq j$$

where $d(\mathbf{x}^i, \mathbf{x}^j) \in \mathbb{R}_+$ is the distance metric. Clearly, collision constraints fall within the framework of the construction of the function, $\mathbf{h}$ (and its lower and upper bounds). Many other DSS requirements can be included as components of $\mathbf{h}$; for example, a broad class of formations can be defined by the inequality,

$$c_l^{i,j} \leq d(\mathbf{x}^i(t), \mathbf{x}^j(t)) \leq c_u^{i,j} \qquad \forall\ t, i, j, \tag{6.30}$$

where $c_l^{i,j}$ and $c_u^{i,j}$ are formation design parameters that are specific to a given space mission [12, 14].

The construction of the system dynamics for a DSS is quite simple. Suppose that the dynamics of each spacecraft of the DSS is given by (see for example, Problem *O* discussed in Section 6.7),

$$\dot{\mathbf{x}}^i(t) = \mathbf{f}^i(\mathbf{x}^i(t), \mathbf{u}^i(t); \mathbf{p}^i) \quad i = 1 \ldots N_s, \tag{6.31}$$

where $\mathbf{f}^i : \mathbb{R}^{N_{x^i}} \times \mathbb{R}^{N_{u^i}} \times \mathbb{R}^{N_{p^i}} \to \mathbb{R}^{N_{x^i}}$ is a given function, $\mathbf{u}^i \in \mathbb{U}^i \subseteq \mathbb{R}^{N_{u^i}}$ and $\mathbf{p}^i \in \mathbb{R}^{N_{p^i}}$ is a vector of (constant) design parameters. By using Eqs. (6.24) and (6.25), the dynamics of the DSS may be represented quite succinctly as,

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t); \mathbf{p}) \quad \mathbf{u} \in \mathbb{U}, \tag{6.32}$$

where $\mathbb{U} = \mathbb{U}^1 \times \cdots \times \mathbb{U}^{N_s}$. Typically, the functions $\mathbf{f}^i$ are all the same so that $\mathbf{f}$ is simply N_s copies of $\mathbf{f}^1$. This fact can be exploited for real-time computation [38].

The optimal control system framework also facilitates a simplification of DSS management (design and operations) by exploiting the couplings between the dynamics, path constraints and the endpoint set, $\mathbb{E}$ (see Section 6.5). To observe this, consider a simple requirement of the form,

$$\mathbf{x}^i(t_f) \in \mathbb{E}^i \subset \mathbb{E}, \ \forall \ i. \tag{6.33}$$

In the framework of Problem *B*, it is sufficient to stipulate all the constraints of Eq. (6.33) as a single constraint,

$$\mathbf{x}^i(t_f) \in \mathbb{E}^i \subset \mathbb{E}, \quad \text{for} \ \ i = 1 \tag{6.34}$$

or any other index, i. This is because, the path constraints (Eq. (6.29)) will automatically enforce the remainder of the constraints in Eq. (6.33). As a matter of fact, if Eq. (6.33) is chosen over Eq. (6.34), the feasible set may be empty as a result of over-specification. Consequently, we may want to use Eq. (6.33) during design considerations to explore possible over-specifications but use Eq. (6.34) during flight operations. In the latter context, we may designate $i = 1$ as the leader, but it essentially reduces to semantics rather than a leader-follower architecture. In other words, in this framework, there is no leader or follower; rather a true system of multiple spacecraft, or a DSS. Note however that if there was a mission requirement to designate a particular spacecraft as a leader and the others as followers, it can be easily accomplished by picking out the particular index, i, representing the leader. Then, when the leader moves along some trajectory, $t \mapsto \mathbf{x}^i$, the distance constraints along with any additional path constraints, Eq. (6.29), dictate how the remainder of the spacecraft must follow certain trajectories to meet the configuration constraints. Thus, if any one spacecraft had an additional configuration constraint, it would automatically transfer in some fashion to the remainder of DSS by way of the couplings between the various equations.

Certain formation-type DSS missions are vaguely defined in terms of periodicity simply because the engineering requirements are vague [52]. A natural way to account for these requirements is to adapt Bohr's notion of almost periodic functions [54, 55]. That is, rather than impose strict periodicity, we specify,

$$\varepsilon_l^i \leq \mathbf{x}^i(t_0) - \mathbf{x}^i(t_f) \leq \varepsilon_u^i \quad \forall \ i \quad \text{or} \quad \text{for} \ \ i = 1, \tag{6.35}$$

where ε_l^i and ε_u^i are formation design parameters representing almost periodicity and Eq. (6.35) is to be taken within the context of Eqs. (6.33) and (6.34). Note that Eq. (6.35) is not the same as specifying standard boundary conditions because the values of $\mathbf{x}^i(t_0)$ and $\mathbf{x}^i(t_f)$ are unknown. In the same spirit, we can define relative periodicity by writing $\mathbf{x}^i(t_0) - \mathbf{x}^j(t_0) = \mathbf{x}^i(t_f) - \mathbf{x}^j(t_f)$ or relax the equality for almost relative periodicity. That is, the DSS collective can have an aperiodic configuration if its constituents have almost relative periodicity. This is one of the reasons why the proper way to view Eq. (6.35) is in terms of the endpoint map, $\mathbb{E}$.

What is clear from the preceding discussions is that by treating the DSS as yet another system in an optimal control framework, the design and control of a DSS can be fully

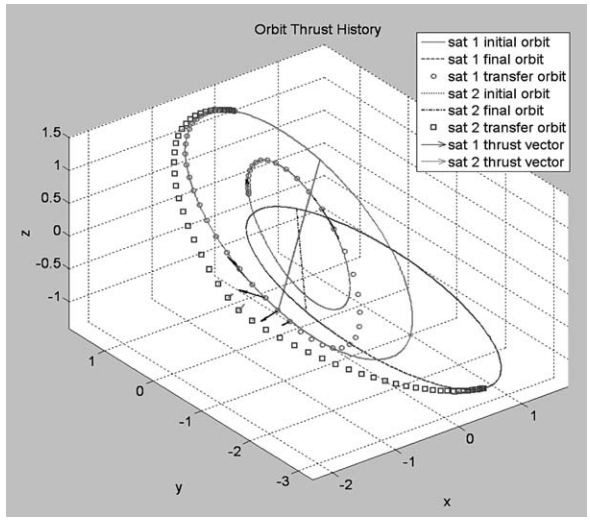


Fig. 6.14. A two-agent optimal space trajectory; from Ref. [15]. (see Color plate 8)

accounted for under the structure of Problem *B* discussed in Section 6.5. The missing details in Section 6.5 vis-à-vis the parameter **p** or the state constraints does not change the substance of the discussions as already alluded to earlier. Thus, by taking a systems' approach to the DSS formation problem, the mathematical problem can be framed under the constructs of Problem *B*. An application of this framework for formation design and control is discussed in Refs. [12, 13] and [14]. The same framework is used for configuration problems in Ref. [15]. A sample solution to a re-configuration problem is shown in Fig. 6.14. This plot [15] was obtained by casting the dynamics using the equinoctial element set for the state variables.

6.9 Conclusions

The L^1-optimal control problem forms a natural framework for formulating space trajectory optimization problems. Based on thruster configurations and the physics of the mass expulsion, several l^p variants of the L^1 norm of the thrust force can be articulated. Quadratic cost functions are inappropriate performance indices for space trajectory optimization problems. Nonsmooth issues dominate both theory and practice; in fact, practical problems are more likely to have non-convex, nonsmooth geometric structures. Transformation techniques can be applied to efficiently solve these problems. Real-time computation of the controls facilitates optimal feedback guidance and control. The same optimal control framework can be applied to design, control, and operate a distributed space system. These new possibilities are chiefly due to a confluence of two major tipping points that occurred in the late 1990s. The first advancement—and the most obvious one—was the widespread availability of extraordinary computing capabilities on ordinary

computers. The second advancement was in the first-principles integration of optimal control theory with approximation theory under the unifying perspective of computation in Sobolev spaces. This perspective obviates the sensitivity issues arising from the symplectic structure of Hamiltonian systems. In addition, while requiring differentiability was once a reflection on the inadequacy of the available tools for analysis, it is no longer a major problem in either theory or computation.

Acknowledgments

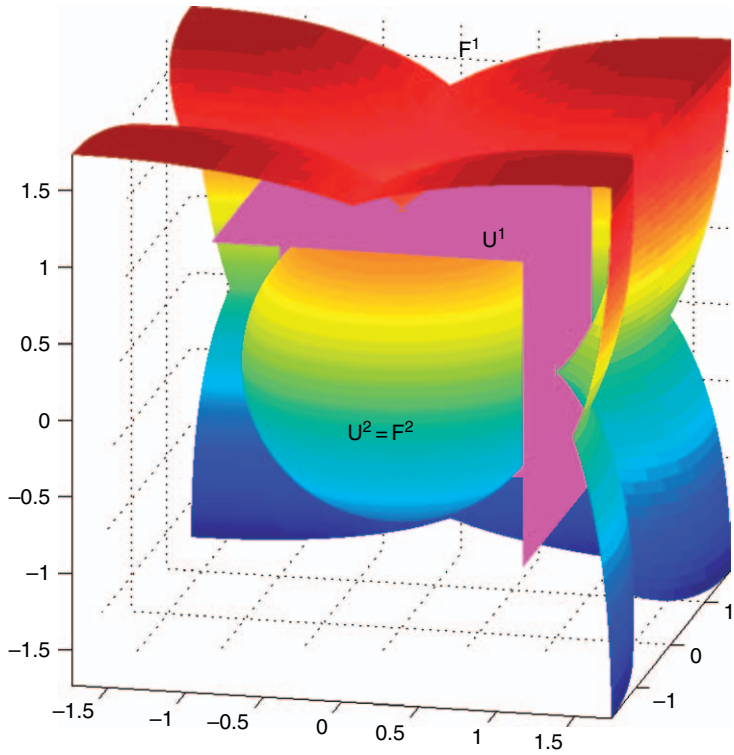
I gratefully acknowledge the generous funding for this research provided by NASA-GRC and the Secretary of the Air Force. In particular, I would like to thank John P. Riehl (NASA) and Stephen W. Paris (Boeing) for their enthusiastic support of pseudospectral methods for solving mission planning problems for NASA.

References

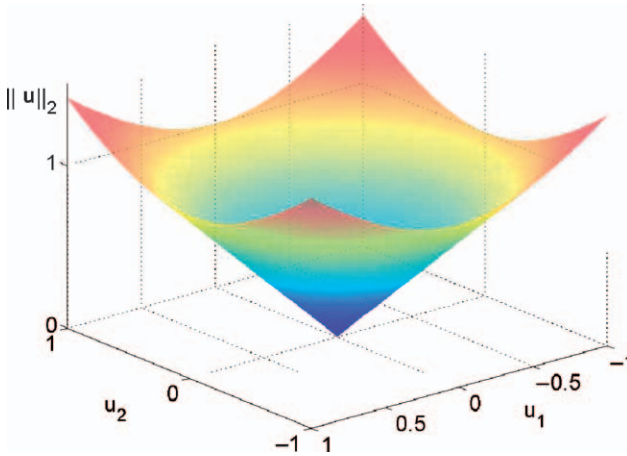
1. Lu, P., Sun, H. and Tsai, B. (2003). Closed-loop endoatmospheric ascent guidance. *Journal of Guidance, Control and Dynamics*. **26**(2), pp. 283–294.
2. Teather, D. (2004). Halliburton cleared in fuel row. *The Guardian*, January 7.
3. Hibbard, R.L. (1996). Satellite on-orbit refueling: A cost-effect analysis. M.S. Thesis in Systems Technology, Naval Postgraduate School, Monterey, CA.
4. Marec, J.-P. (1979). *Optimal Space Trajectories*, Elsevier, New York.
5. Ross, I.M. (2004). How to find minimum-fuel controllers. *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Providence, RI, August. AIAA Paper No. 2004-5346.
6. Neustadt, L.W. (1965). A general theory of minimum-fuel space trajectories. *SIAM Journal of Control*, Ser. A, **3**(2), pp. 317–356.
7. Clarke, F.H. (1990). *Optimization and Nonsmooth Analysis*, SIAM Publications, Philadelphia, PA.
8. Clarke, F.H., Ledyaev, Y.S., Stern, R.J. and Wolenski, P.R. (1998). *Nonsmooth Analysis and Control Theory,* Springer-Verlag, New York, NY.
9. Sovey, J.S., Rawlin, V.K. and Patterson, M.J. (2001). Ion propulsion development projects in U.S.: Space electric rocket test 1 to deep space 1. *Journal of Propulsion and Power*, **17**(3), pp. 517–526.
10. Croley, P.A. (2005). Reachable sets for multiple asteroid sample return missions. Astronautical Engineer's Degree Thesis, Naval Postgraduate School, Monterey, CA, June.
11. Kolmogorov, A.N. and Fomin, S.V. (1999). Elements of the Theory of Functions and Functional Analysis, **2**, Dover Publications, Mineola, NY.
12. Ross, I.M., King, J.T. and Fahroo, F. (2002). Designing optimal spacecraft formations. *Proceedings of the AIAA/AAS Astrodynamics Conference*, AIAA-2002-4635, Monterey, CA, August 5–8.
13. King, J.T. (2002). A Framework for designing optimal spacecraft formations. M.S. Thesis, Department of Aeronautical and Astronautical Engineering, Naval Postgraduate School, Monterey, CA, September.
14. Infeld, S.I., Josselyn, S.B., Murray, W. and Ross, I.M. (2004). Design and Control of Libration Point Spacecraft Formations. *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Providence, RI, August. AIAA Paper No. 2004-4786.
15. Mendy, P.B. (2004). Multiple satellite trajectory optimization. Astronautical Engineer Thesis, Department of Mechanical and Astronautical Engineering, Naval Postgraduate School, Monterey, CA, December.
16. Bilimoria, K.D. and Wie, B. (1993). Time-optimal three-axis reorientation of a rigid spacecraft. *Journal of Guidance, Control, and Dynamics*, **16**(3), pp. 446–452.
17. Chyba, M. (2003). Underwater vehicles: A surprising non time-optimal path. *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, December.
18. Bryson, A.E. (1999). Dynamic Optimization, Addison-Wesley Longman, Inc.

19. Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V. and Mischenko, E.F. (1962). The Mathematical Theory of Optimal Processes, Wiley-Interscience, New York, NY.
20. Adams, R.A. (1975). Sobolev Spaces, Academic Press, New York, NY.
21. Vinter, R.B. (2000). Optimal Control, Birkhäuser, Boston, MA.
22. Hager, W.W., Numerical analysis in optimal control, *International Series of Numererical Mathematics*, (K.-H. Hoffmann, I. Lasiecka, G. Leugering, J. Sprekels, and F. Troeltzsch, eds.), Birkhäuser, Basel, Switzerland, 2001, Vol. 139, pp. 83–93.
23. Sussmann, H.J. (2000). New theories of set-valued differentials and new versions of the maximum principle of optimal control theory. Nonlinear Control in the Year 2000, (A. Isidori, F. Lamnabhi-Lagarrigue, and Respondek, W., eds.), Springer-Verlag, London, pp. 487–526.
24. Rockafellar, R.T. (1993). Lagrange multipliers and optimality. *SIAM Review*, **35**, pp. 183–238.
25. Mordukhovich, B.S. (2005). *Variational Analysis and Generalized Differentiation, I: Basic Theory*, Vol. 330 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences] Series, Springer, Berlin.
26. Bryson, A.E. and Ho, Y.C. (1975). Applied Optimal Control, Hemisphere, New York.
27. Young, L.C. (1969). Lectures on the Calculus of Variations and Optimal Control Theory, Saunders, Philadelphia, PA.
28. Silva, G.N. and Vinter, R.B. (1997). Necessary conditions for optimal impulsive control problems. *SIAM Journal of Control and Optimization*, **35**(6), pp. 1829–1846, November.
29. Lawden, D.F. (1963). Optimal Trajectories for Space Navigation, Butterworths, London.
30. Ross, I.M. and Fahroo, F. (2004). Pseudospectral knotting methods for solving optimal control problems. *Journal of Guidance, Control and Dynamics*, **27**(3), pp. 397–405.
31. Ross, I.M. (2005) A roadmap for optimal control: The right way to commute. *Annals of the New York Academy of Sciences*, **1065**, pp. 210–231, December.
32. Ross, I.M. and D'Souza, C.D. (2005). Hybrid optimal control framework for mission planning. *Journal of Guidance, Control and Dynamics*, **28**(4), July–August, pp. 686–697.
33. Betts, J.T. (2001) Practical Methods for Optimal Control Using Nonlinear Programming, SIAM: Advances in Control and Design Series, Philadelphia, PA.
34. Betts, J.T. (1998). Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, **21**(2), pp. 193–207.
35. Hager, W.W. (2000). Runge-Kutta methods in optimal control and the transformed adjoint system. *Numerische Mathematik,* **87**, pp. 247–282.
36. Fahroo, F. and Ross, I.M. (2001). Costate estimation by a legendre pseudospectral method. *Journal of Guidance, Control and Dynamics*, **24**(2), pp. 270–277.
37. Ross, I.M. (2005). A historical introduction to the covector mapping principle. *AAS/AIAA Astrodynamics Specialist Conference*, Tahoe, NV, August 8–11, Paper AAS 05-332.
38. Ross, I.M. and Fahroo, F. (2006). Issues in the real-time computation of optimal control. *Mathematical and Computer Modelling*, An International Journal, **43**(9–10), pp. 1172–1188, May.
39. Sekhavat, P., Fleming A. and Ross, I.M. (2005). Time-Optimal Nonlinear Feedback Control for the NPSAT1 Spacecraft. *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, AIM 2005, July 24–28, Monterey, CA.
40. Yan, H., Lee, D.J., Ross, I.M. and Alfriend, K.T. (2005). Real-time outer and inner loop optimal control using DIDO. *AAS/AIAA Astrodynamics Specialist Conference*, Tahoe, NV, August 8–11, Paper AAS 05-353.
41. Goldstine, H.H. (1981). A History of the Calculus of Variations from the 17th to the 19th Century, Springer-Verlag, New York, NY, p. 110.
42. Cullum, J. (1969). Discrete approximations to continuous optimal control problems. *SIAM Journal of Control*, **7**(1), February, pp. 32–49.
43. Dontchev, A.L., Hager, W.W. and Veliov, V.M. (2000). Second-order Runge-Kutta approximations in control constrained optimal control problems. *SIAM Journal of Numerical Analysis*, **38**(1), pp. 202–226.
44. Betts, J.T., Biehn, N. and Campbell, S.L. (2000). Convergence of nonconvergent IRK discretizations of optimal control problems with state inequality constraints. *SIAM Journal of Scientific Computation* **23**(6), pp. 1981–2007.
45. Trefethen, L.N. (2000). *Spectral Methods in MATLAB,* SIAM, Philadelphia, PA.
46. Ross, I.M. and Fahroo, F. (2002). A Perspective on Methods for Trajectory Optimization. *Proceedings of the AIAA/AAS Astrodynamics Conference*, Monterey, CA, August. AIAA Paper No. 2002-4727.

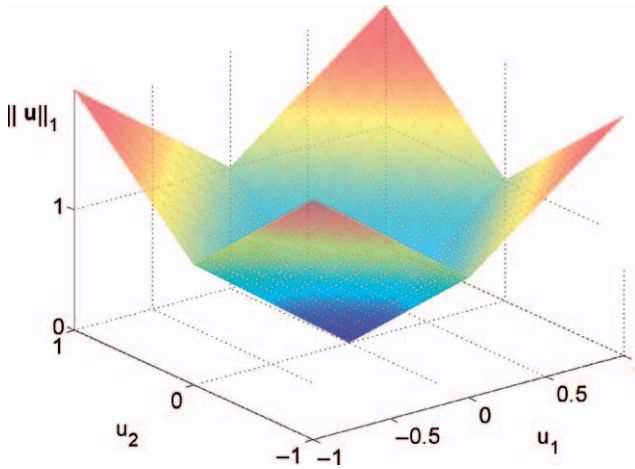
47. Boggs, P.T., Kearsley, A.J. and Tolle, J.W. (1999). A global convergence analysis of an algorithm for large-scale nonlinear optimization problems. *SIAM Journal of Optimization*, **9**(4), pp. 833–862.
48. Ross, I.M. and Fahroo, F. (2003). Legendre pseudospectral approximations of optimal control problems. *Lecture Notes in Control and Information Sciences*, **295**, Springer-Verlag, New York.
49. Elnagar, J., Kazemi, M.A. and Razzaghi, M. (1995). The pseudospectral legendre method for discretizing optimal control problems. *IEEE Transactions on Automatic Control*, **40**(10), pp. 1793–1796.
50. Moyer, H.G. and Pinkham, G. (1964). Several trajectory optimization techniques, Part II: Applications. Computing Methods in Optimization Problems, (A.V. Balakrishnan and L.W. Neustadt, eds.), New York, Academic Press.
51. Ross, I.M. (2004). User's manual for DIDO: A MATLAB application package for solving optimal control problems. Tomlab Optimization, Sweden, February.
52. New World Vistas (1995). Summary Volume, USAF Scientific Advisory Board, December.
53. Vincent, M.A. and Bender, P.L. (1987). The orbital mechanics of a space-borne gravitational wave experiment. Advances in the Astronautical Sciences, Astrodynamics, **65**, Part II, P. 1346. (J.K. Soldner, et al. ed.), Paper AAS 87-523. Full paper in AAS Microfiche Series, **55**.
54. Fischer, A. (1996). Structure of fourier exponents of almost periodic functions and periodicity of almost periodic functions. *Mathematica Bohemica*, **121**(3), pp. 249–262.
55. Corduneanu, C. (1968). Almost Periodic Functions, John Wiley, New York.



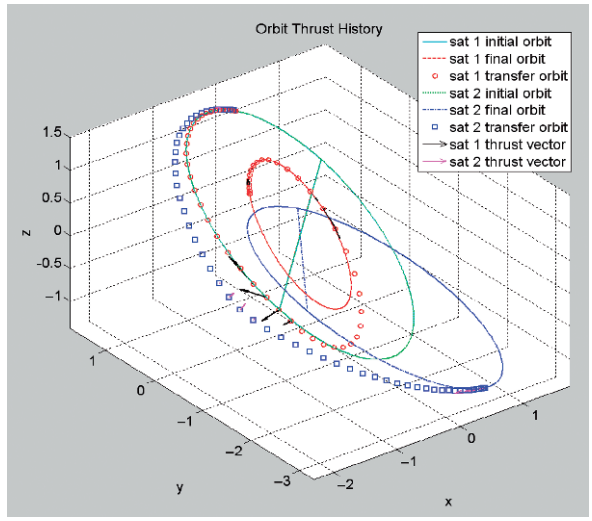
Color plate 5. Cutaway views of the geometries of the control space and their corresponding mass-flow rates. (see Fig. 6.3)



Color plate 6. Illustrating the nonsmooth structure of $\mathbf{u} \mapsto \|\mathbf{u}\|_2$. (see Fig. 6.9)



Color plate 7. Illustrating the nonsmooth structure of $\mathbf{u} \mapsto \|\mathbf{u}\|_1$. (see Fig. 6.10)



Color plate 8. A two-agent optimal space trajectory; from Ref. [15]. (see Fig. 6.14)