

Fuel-Optimal Rocket Landing with Aerodynamic Controls

Xinfu Liu*

Beijing Institute of Technology, 100081 Beijing, People's Republic of China

DOI: 10.2514/1.G003537

Aerodynamic forces are not negligible for a reusable rocket returning back to Earth. How the aerodynamic controls and propulsion should be coordinated to realize fuel-optimal precise landing is addressed in this paper. To this end, a model-based optimal control problem is formulated with the rocket's angle of attack and thrust as control inputs, and constraints on the controls are included to reflect the capabilities of the vehicle. Precise landing requires the (highly nonlinear and nonconvex) problem to be solved onboard in real time. This ability of online computation is becoming increasingly desired in aerospace guidance and control for autonomous missions. Hence, this paper presents how to solve the rocket landing problem via convex optimization that has guaranteed polynomial-time complexity. Specifically, a novel methodology of handling the rocket nonlinear dynamics is introduced, and a relaxation technique used to convexify nonconvex constraints is theoretically proved to be valid. High efficiency of the proposed method, with potential for online computation, is demonstrated by numerical examples and comparisons with other methods.

I. Introduction

MAKING rockets practically reusable has gained much attention worldwide in the past few years and will continue acting as a focus of the space industry. Its purpose is to significantly reduce the cost of space exploration and realize easier and faster access to space. Private companies such as SpaceX and Blue Origin have been making remarkable progress in turning rocket reusability into reality. One of the key technologies to achieve this historic breakthrough is called pinpoint (or precise) landing, which generally requires the rocket to be able to autonomously perform onboard real-time trajectory planning. Though reusable launch vehicles have been envisioned and discussed since the 1990s [1,2] and returning reusable rockets safely back to Earth, still in its early stage, has been practically demonstrated to be achievable in recent years, there still exist very limited literature discussing the trajectory planning and optimization problem for rocket vertical landing with full consideration on the aerodynamic forces.

Two methods are currently available to recover a rocket (or a booster), which are return-to-launch-site and downrange landing. The former consists of a pitch-over maneuver to turn the velocity vector around toward the launch site [3,4]. The latter recovers a rocket in the downrange direction, such as landing on a drone ship at sea. To safely land a rocket, it is critical to control the thrust vector so that it can take a soft landing at a specified landing pad and have the right attitude at touchdown to avoid tipping over. In Ref. [2], the magnitude of the thrust acceleration profile in each axis of an inertial coordinate system was presumed to be a function of three (preselected) linearly independent polynomials with three unknown coefficients to be determined by the end conditions. An alternative is to design the thrust acceleration vector as a linear function of time, whose vector coefficients are determined to meet the end conditions [5]. In these approaches, the acceleration profiles were not optimal. Additionally, the aerodynamic forces were simply ignored. Nevertheless, they are not negligible unless the rocket is close, and in low speed, to the landing pad. In a recent work in Ref. [6], aerodynamic drag was incorporated in the system dynamics to design fuel-optimal thrust profiles, which was an improvement in generating more accurate landing trajectories.

Presented as Paper 2017-1732 at the AIAA Guidance, Navigation, and Control Conference, Grapevine, TX, 9–13 January 2017; received 25 December 2017; revision received 2 June 2018; accepted for publication 18 August 2018; published online 12 November 2018. Copyright © 2018 by Xinfu Liu. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 0731-5090 (print) or 1533-3884 (online) to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

*Associate Professor, School of Aerospace Engineering; lau.xinfu@gmail.com. Member AIAA.

This paper addresses the fuel-optimal rocket landing problem with both aerodynamic forces and propulsion as control inputs in the system dynamics. Compared with existing work that mostly uses only the thrust magnitude and thrust direction as control inputs, the additional control freedom from the angle of attack (which affects the aerodynamic forces) will help generate more fuel-optimal trajectories. This is important in the sense that less fuel is required for landing and more payload may be delivered into space. Despite the advantages, the price to pay is that we need to solve the corresponding optimal control problem in real time to get a landing trajectory [7]. Blindly solving this nonconvex problem via brute force, such as simply by general nonlinear programming (NLP) algorithms, does not work because convergence and efficiency are not guaranteed. Alternatively, we may carefully re-formulate or analyze the problem so that specific high-efficient algorithms can be used to solve the problem. This idea is also exactly what is desired in an emerging trend called *computational guidance and control* [8,9], where it is advocated to put upfront investment in problem formulation, modeling, and analysis in order to achieve online computation. One convincing example is the rigorous work of convexifying the original nonconvex problems in planetary landing to convex problems and conducting analysis in guarantee of lossless convexification [10–13]. The investment is rewarded because the power of convex optimization algorithms [14], especially customized ones that are under development [15,16], can be exploited and thus onboard real-time trajectory optimization becomes reality in planetary landing, which will greatly improve the performance of future missions [17]. The work in Refs. [18,19] also emphasized the importance of putting efforts in converting certain classes of general nonconvex problems into convex problems. In recent years many other complex aerospace problems have been solved by convex optimization [20–29] (to name a few) and pseudospectral convex optimization [30,31]. Interested readers can also refer to a recent survey paper in Ref. [32] for more details.

In this paper for the rocket landing problem, our contribution lies in the significant efforts on transforming the original nonconvex problem into a convex optimization problem and guaranteeing the effectiveness of the relaxation technique used. Note that existing work of applying convex optimization mostly considers aerospace systems with either the thrust or the aerodynamic forces (not both) as control inputs, whereas the unique property for the rocket landing problem is that both the thrust and aerodynamic forces are present as control inputs. Appropriately converting the corresponding highly nonlinear dynamics into linear dynamics is the focus of this paper. In addition, this process will generally result in nonconvex control constraints. Though these constraints can be relaxed into convex constraints with an expanded feasible set, analyzing whether this relaxation technique is valid is the difficult part and in some cases such a technique is likely to be invalid in the sense that the solution of the relaxed problem is not even feasible to the constraints of the

original problem. Fortunately, theoretical analysis is given in this paper to guarantee the validity of the relaxation technique. These upfront investments in transformation, relaxation, and analysis is of paramount importance in enabling convex optimization algorithms (with polynomial-time complexity) to get the solution of the original problem by successively solving the convex optimization problems. Consequently, the proposed method makes it potential to achieve online computation of the landing trajectory and the associated control actions. Comparison with other solution methods will be provided to show the validity and high efficiency of the proposed method.

II. Problem Description

In this section, the fuel-optimal rocket landing problem is formulated as an optimal control problem. In addition to the thrust, the aerodynamic lift and drag forces (or equivalently the angle of attack) are considered as control inputs.

A. Dynamics and Constraints

In this paper, we consider a rocket's two-dimensional flight over a flat earth. Based on the geometry given in Fig. 1, the equations of motion of the rocket can be given as (the three-dimensional equations of motion can be found in Ref. [33])

$$\begin{aligned} \dot{r} &= V \sin \gamma \\ \dot{s} &= V \cos \gamma \\ \dot{V} &= \frac{-T \cos \epsilon - D}{m} - g \sin \gamma \\ \dot{\gamma} &= \frac{-T \sin \epsilon + L}{mV} - g \cos \gamma \\ \dot{m} &= -\frac{T}{g_0 I_{sp}} \end{aligned} \quad (1)$$

where r is the radial distance from the Earth's center to the rocket, s is the downrange, V is the velocity, γ is the flight-path angle measured from the horizon to the velocity vector (the γ shown in Fig. 1 has a negative value), g is the gravitational acceleration at r with g_0 corresponding to R_0 (the radius of the Earth), m is the rocket mass, T is the thrust magnitude, ϵ is the thrust direction defined as the angle between the thrust vector and the negative velocity vector (which is also called the thrust angle of attack in Ref. [33]), L is the lift force, D is the drag force, and I_{sp} is the specific impulse of the rocket engines.

To avoid possible numerical issues caused by large difference in the orders of magnitude of the variables in Eq. (1), we also give (with some abuse of notation for simplicity) the following dimensionless equations of motion, which are used in the numerical examples in this paper:

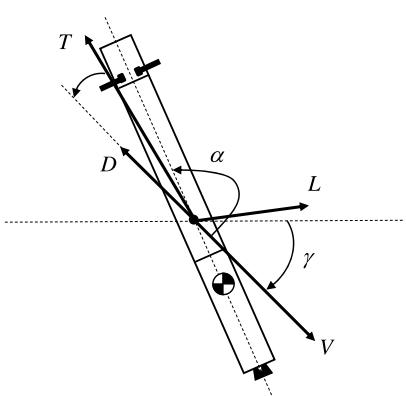


Fig. 1 Geometry of a rocket.

$$\begin{aligned} \dot{r} &= V \sin \gamma \\ \dot{s} &= V \cos \gamma \\ \dot{V} &= \frac{-T \cos \epsilon - D}{m} - \frac{\sin \gamma}{r^2} \\ \dot{\gamma} &= \frac{-T \sin \epsilon + L}{mV} - \frac{\cos \gamma}{r^2 V} \\ \dot{m} &= -T/I_{sp} \end{aligned} \quad (2)$$

Note that in Eq. (2) all variables except γ and ϵ are dimensionless and the differentiation is with respect to the dimensionless time. Equation (2) can be directly derived from Eq. (1) via normalizing the time and I_{sp} by $\sqrt{R_0/g_0}$, r and s by R_0 , V by $\sqrt{g_0 R_0}$, m by m_0 (the initial mass of the rocket), and T , L , and D all by $m_0 g_0$. The dimensionless aerodynamic accelerations L and D in Eq. (2) are computed by

$$L = 0.5\rho(V\sqrt{g_0 R_0})^2 S_{ref} C_L / (m_0 g_0) \quad (3)$$

$$D = 0.5\rho(V\sqrt{g_0 R_0})^2 S_{ref} C_D / (m_0 g_0) \quad (4)$$

where V is the dimensionless velocity, ρ is the dimensional air density, S_{ref} is the dimensional reference area of the rocket, C_L is the lift coefficient, and C_D is the drag coefficient. Note that C_L and C_D are functions of the Mach number M and angle of attack α (which is defined as the angle between the velocity vector and the nose of the rocket).

The controls in Eq. (2) include the angle of attack α , thrust magnitude T , and thrust direction ϵ . In rocket vertical landing, those control variables may be constrained by their lower and upper bounds, that is,

$$\alpha_{\min} \leq \alpha \leq \alpha_{\max} \quad (5)$$

$$T_{\min} \leq T \leq T_{\max} \quad (6)$$

$$\epsilon_{\min} \leq \epsilon \leq \epsilon_{\max} \quad (7)$$

where the bounds on the thrust direction are assumed to satisfy $-\pi/2 < \epsilon_{\min} < \epsilon_{\max} < \pi/2$. It is worth noting that the thrust gimbal angle, the angle between the thrust vector and the longitudinal body axis of the rocket, can be computed by $(\pi + \epsilon - \alpha)$. The bounds on the thrust direction ϵ may be adjusted to make the thrust gimbal angle meet its physical limits. In this paper the way of defining the direction of the thrust vector, as in Refs. [33,34], is adopted because it is more convenient to handle the nonconvexity caused by the controls in the system dynamics.

For a safe landing, the constraints on the final states and controls of the rocket can include

$$r(t_f) = 1, \quad s(t_f) = s_f^* \quad (8)$$

$$V(t_f) \leq V_f^*, \quad \gamma(t_f) = -\pi/2 \quad (9)$$

$$|\alpha(t_f) - \pi| \leq \alpha_f^*, \quad \epsilon(t_f) = 0 \quad (10)$$

where s_f^* is the required final downrange, $V_f^* \geq 0$ is a small safe threshold of the final velocity, and $\alpha_f^* \geq 0$ is a small safe threshold of the final angle-of-attack deviated from π (note that when $\alpha_f^* = 0$ the final longitudinal body axis is aligned with the velocity vector). The constraints in Eq. (8) ensure that the rocket reaches a specific landing pad, whereas the constraints in Eq. (9) make the rocket have a vertical velocity not greater than V_f^* but zero horizontal velocity at touchdown. Constraints on the longitudinal body axis of the rocket and the thrust direction at touchdown are imposed by Eq. (10).

B. Optimal Control Problem

In this paper, the optimization objective is selected to minimize the fuel cost or equivalently maximize the final mass. Hence, for a minimization problem, we have the following objective function:

$$J = -m(t_f) \quad (11)$$

The optimal control problem corresponding to the fuel-optimal rocket landing problem is as follows:

$$P0: \min -m(t_f) \quad (12)$$

$$\text{s.t. Eqs. (2), (5) – (10)} \quad (13)$$

For precise landing, we need to solve the proceeding problem onboard in real time. However, the problem is nonconvex because the (equality) dynamics in Eq. (2) are highly nonlinear due to the simultaneous presence of the thrust and aerodynamic forces, while solving nonconvex problems, by general NLP solvers, for instance, is time-consuming and has sensitivity problems (see the discussion in Sec. V). To take advantage of the high efficiency of existing convex optimization algorithms [14], we will make efforts to transform the problem P0 into a convex optimization problem. These upfront investments will be the focus of the next section.

Remark 1: Our discussion in this paper is restricted in the two-dimensional motion of a rocket in the vertical plane. Note that this is a reasonable simplification because the flight trajectory of a landing rocket is mostly in the vertical plane (or orbital plane) in order to save fuel consumption in the landing process. The small out-of-plane motion may be controlled separately. Nevertheless, applying convex optimization in the three-dimensional rocket landing problem is, though much more complex, still a topic of future research. In addition, this paper considers the fuel-optimal rocket landing problem with fixed time of flight, whereas a line search method may be applied to find the optimal time of flight (one can refer to Ref. [10] for such a method).

III. Second-Order Cone Programming Formulation

The goal of this section is to elaborate on the innovative techniques used to convexify P0 into a second-order cone programming (SOCP) problem. SOCP is a special class of convex optimization in which the objective function is linear and all other constraints are either linear or second-order cone constraints [35]. Obviously, we need to transform the rocket nonlinear dynamics into linear dynamics. This transformation process is likely to result in new nonconvex constraints, and it may also make some original convex constraints become nonconvex. How to deal with the nonconvexity will be discussed in this section as well.

A. Handling the Nonlinear Dynamics

Transforming the nonlinear dynamics in Eq. (2) into linear dynamics is critical and also a major work of this paper. In Refs. [36,37], it is proposed that general nonlinear dynamics can be first converted into a control-affine system and then further transformed into a partially linearized system. This two-step process, instead of the sole linearization method in one step, has its mission to try to preserve the nonlinearities in the original dynamics [32], which is considered advantageous for rapid convergence of a successive solution procedure (to be described in the next section). It should be pointed out that the two-step process just provides a general idea, whereas how to implement the idea to specific nonlinear dynamics is nontrivial. In the following, we will present how to handle the rocket nonlinear dynamics based on the idea.

In the first step, we will construct a control-affine system; that is, the differential equation becomes linear with respect to the controls related to both the aerodynamic forces and thrust. To that end, for the aerodynamic part a similar idea to that from Ref. [36] will be adopted by first defining a new variable η as follows:

$$\eta(\alpha, M) := C_L(\alpha, M)/\hat{C}_L(M) \quad (14)$$

where $\hat{C}_L(M)$ is the lift coefficient corresponding to the maximum lift-to-drag ratio at a certain M , which can be explicitly obtained given the tabulated aerodynamic data. Equation (14) means that C_L can be expressed as a function of η , that is,

$$C_L(\alpha, M) = \eta \hat{C}_L(M) \quad (15)$$

We can also express C_D as a function of η using the well-known quadratic drag polar [36]:

$$C_D(\alpha, M) = 0.5 \hat{C}_D(M)[1 + \eta(\alpha, M)^2] \quad (16)$$

where $\hat{C}_D(M)$ is the drag coefficient corresponding to the maximum lift-to-drag ratio at a certain M . Consequently, both L and D are functions of η , that is,

$$L = \hat{L}(M)\eta(\alpha, M), \quad D = 0.5\hat{D}(M)[1 + \eta(\alpha, M)^2] \quad (17)$$

where the aerodynamic forces \hat{L} and \hat{D} correspond to \hat{C}_L and \hat{C}_D , respectively.

Now, we can replace L and D in Eq. (2) by the expressions in Eq. (17). The resulting dynamics are functions of η , but are still nonlinear with respect to η due to the presence of η^2 . To overcome this difficulty, we can define two new control variables:

$$u_1 := \eta/m \quad (18)$$

$$u_2 := \eta^2/m^2 \quad (19)$$

and subsequently obtain the terms L/m and D/m^2 as follows:

$$L/m = \hat{L}(M)u_1, \quad D/m^2 = 0.5\hat{D}(M)(1/m^2 + u_2) \quad (20)$$

It is seen from Eq. (20) that the nonlinear terms related to L and D are now linear with respect to u_1 and u_2 . Note that the way of defining the new controls in Eqs. (18) and (19) is different from that in Ref. [36]. In this paper, the mass m is time-varying and including m in the definition of u_1 and u_2 can reduce the nonlinearity of the dynamics.

For the thrust part, we propose to define three new control variables as follows:

$$u_3 := T \cos \epsilon / m \quad (21)$$

$$u_4 := T \sin \epsilon / m \quad (22)$$

$$u_5 := T / m \quad (23)$$

which will make the nonlinear terms related to the thrust be linear with respect to u_3 , u_4 , and u_5 .

When u_5 is defined as in Eq. (23), we need to rewrite the original linear differential equation for the mass in Eq. (2), that is, $\dot{m} = -T/I_{sp}$. It then becomes

$$\dot{m} = -(m/I_{sp})u_5 \quad (24)$$

which is nonlinear. Nevertheless, if the following new state variable z is defined

$$z := \log(m) \quad (25)$$

Equation (24) can be rewritten as the following linear differential equation:

$$\dot{z} = -u_5/I_{sp} \quad (26)$$

In sum, with the new controls defined in Eqs. (18), (19) and (21–23) and the new state defined in Eq. (25), the dynamics in Eq. (2) are finally converted into a control-affine system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{B}(\mathbf{x})\mathbf{u} \quad (27)$$

where $\mathbf{x} := [r \ s \ V \ \gamma \ z]^T$, $\mathbf{u} := [u_1 \ u_2 \ u_3 \ u_4 \ u_5]^T$, and

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} V \sin \gamma \\ V \cos \gamma \\ -\sin \gamma / r^2 - 0.5 \hat{D} e^{-z} \\ -\cos \gamma / (r^2 V) \\ 0 \end{bmatrix} \quad (28)$$

$$\mathbf{B}(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -0.5 \hat{D} e^{-z} & -1 & 0 & 0 \\ \hat{L}/V & 0 & 0 & -1/V & 0 \\ 0 & 0 & 0 & 0 & -1/I_{sp} \end{bmatrix} \quad (29)$$

This makes the first step of constructing a control-affine system completed. Before going to the next step, we should realize that the \mathbf{u} in Eq. (27) must satisfy the following relationships:

$$u_1^2 = u_2 \quad (30)$$

$$u_3^2 + u_4^2 = u_5^2 \quad (31)$$

due to their definitions. Obviously, the preceding method has preserved some nonlinearities we desire as in Eqs. (30) and (31). Note that the nonconvexity from these constraints can be overcome using techniques of linearization or relaxation, which will be presented in the next subsection.

In the second step, the control-affine system (27) can be transformed into the following partially linearized system:

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}^{(k)})\mathbf{x} + \mathbf{B}(\mathbf{x}^{(k)})\mathbf{u} + \mathbf{c}(\mathbf{x}^{(k)}) \quad (32)$$

where $\mathbf{x}^{(k)}$ is the states at the k th iteration, $\mathbf{A}(\mathbf{x}^{(k)}) = \partial \mathbf{f}(\mathbf{x}^{(k)}) / \partial \mathbf{x}$, and $\mathbf{c}(\mathbf{x}^{(k)}) = \mathbf{f}(\mathbf{x}^{(k)}) - \mathbf{A}(\mathbf{x}^{(k)})\mathbf{x}^{(k)}$. Specifically, $\mathbf{A}(\mathbf{x}^{(k)})$ has the following form:

$$\mathbf{A}(\mathbf{x}^{(k)}) = \begin{bmatrix} 0 & 0 & \sin(\gamma^{(k)}) & V^{(k)} \cos(\gamma^{(k)}) & 0 \\ 0 & 0 & \cos(\gamma^{(k)}) & -V^{(k)} \sin(\gamma^{(k)}) & 0 \\ 2 \sin(\gamma^{(k)}) / (r^{(k)})^3 & 0 & 0 & -\cos(\gamma^{(k)}) / (r^{(k)})^2 & \frac{1}{2} \hat{D}^{(k)} e^{-z^{(k)}} \\ 2 \cos(\gamma^{(k)}) / [V^{(k)} (r^{(k)})^3] & 0 & \cos(\gamma^{(k)}) / [(V^{(k)})^2 (r^{(k)})^2] & \sin(\gamma^{(k)}) / [V^{(k)} (r^{(k)})^2] & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (33)$$

Note that Eq. (32) is called partially linearized because we do not linearize the term $\mathbf{B}(\mathbf{x})\mathbf{u}$ in Eq. (27), but instead just approximate the coefficient $\mathbf{B}(\mathbf{x})$ with $\mathbf{B}(\mathbf{x}^{(k)})$ [36]. Such a strategy has two advantages. First, since $\mathbf{u}^{(k)}$ does not appear in the resulting linear system in Eq. (32), there is no need to provide $\mathbf{u}^{(0)}$ to initialize the successive solution procedure to be described in the next section. Second, the $\mathbf{u}^{(k)}$ from the k th iteration does not affect the coefficients of the linear system in the $(k+1)$ th iteration. Otherwise, in the successive solution procedure small oscillations in $\mathbf{u}^{(k)}$ will make the coefficients of the linear system in the next iteration oscillate, and consequently the obtained $\mathbf{u}^{(k+1)}$ in the $(k+1)$ th iteration is likely to have oscillations as well [37], which will be adverse to the convergence of the iterative process.

Until now, in problem P0 we have transformed the nonlinear dynamics Eq. (2) into the linear dynamics Eq. (32). Because the

controls are redefined, any constraints relating to the original controls α , T , and e in Eqs. (5–10) need to be changed correspondingly. Specifically, the angle-of-attack constraint in Eq. (5) is to put constraints on u_1 and u_2 as follows:

$$\eta_{\min} e^{-z} \leq u_1 \leq \eta_{\max} e^{-z} \quad (34)$$

$$0 \leq u_2 \leq \bar{\eta}^2 e^{-2z} \quad (35)$$

where $e^{-z} = 1/m$ [cf. Eq. (25)], $e^{-2z} = 1/m^2$, and η_{\min} (η_{\max}) are the minimum (maximum) values of η when $\alpha_{\min} \leq \alpha \leq \alpha_{\max}$ and the Mach number M is approximated by $M^{(k)}$, and $\bar{\eta} = \max(|\eta_{\min}|, |\eta_{\max}|)$. Note that η_{\min} , η_{\max} , and $\bar{\eta}$ are functions of time and have known values. The thrust magnitude constraint in Eq. (6) becomes

$$T_{\min} e^{-z} \leq u_5 \leq T_{\max} e^{-z} \quad (36)$$

and the thrust direction constraint in Eq. (7) yields

$$u_4 \geq u_3 \tan \epsilon_{\min} \quad (37)$$

$$u_4 \leq u_3 \tan \epsilon_{\max} \quad (38)$$

Finally, the constraints on the final angle of attack and thrust direction in Eq. (10) mean

$$|u_1(t_f)| \leq \eta_f^* e^{-z(t_f)}, \quad u_4(t_f) = 0 \quad (39)$$

where η_f^* is the value of η corresponding to the angle of attack α_f^* [cf. Eq. (14)] and η is assumed to be zero when $\alpha = \pi$.

With z defined as a new state variable in Eq. (25), it is straightforward that minimizing the original objective function (11) is equivalent to minimizing

$$J = -z(t_f) \quad (40)$$

Based on the above discussion, the original optimal control problem P0 is now approximated by

$$\text{P1: } \min -z(t_f) \quad (41)$$

$$\text{s.t. } \dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}^{(k)})\mathbf{x} + \mathbf{B}(\mathbf{x}^{(k)})\mathbf{u} + \mathbf{c}(\mathbf{x}^{(k)}) \quad (42)$$

$$|\mathbf{x}(t) - \mathbf{x}^{(k)}(t)| \leq \delta \quad (43)$$

$$[u_1(t)]^2 = u_2(t) \quad (44)$$

$$\eta_{\min} e^{-z(t)} \leq u_1(t) \leq \eta_{\max} e^{-z(t)} \quad (45)$$

$$0 \leq u_2(t) \leq \bar{\eta}^2 e^{-2z(t)} \quad (46)$$

$$[u_3(t)]^2 + [u_4(t)]^2 = [u_5(t)]^2 \quad (47)$$

$$T_{\min}e^{-z(t)} \leq u_5(t) \leq T_{\max}e^{-z(t)} \quad (48)$$

$$u_4(t) \geq u_3(t) \tan \epsilon_{\min} \quad (49)$$

$$u_4(t) \leq u_3(t) \tan \epsilon_{\max} \quad (50)$$

$$\phi(\mathbf{x}(t_f)) = 0 \quad (51)$$

$$|u_1(t_f)| \leq \eta_f^* e^{-z(t_f)}, \quad u_4(t_f) = 0 \quad (52)$$

where Eq. (43) is a trust-region constraint to ensure validity of the linearization used in handling the original nonlinear dynamics, Eqs. (44–46) are control constraints related to the aerodynamic forces, Eqs. (47–50) are control constraints on the thrust, Eq. (51) represents the linear terminal constraints in Eqs. (8) and (9), and Eq. (52) includes the terminal constraint on u_1 and u_4 .

From P0 to P1, we have successfully converted the original nonlinear dynamics into linear dynamics that can then be discretized as (convex) linear equality constraints. At the same time, this process can generate nonconvex constraints, such as the nonconvex admissible sets formed by the aerodynamic forces [cf. Eqs. (44–46)] and the thrust [cf. Eqs. (47–50)]. Convexification of the nonconvexity will be the subject of the next subsection.

B. Convexification of the Control Constraints

This subsection aims to convexify all the nonconvexity in P1. The nonconvex constraints caused by the existence of the terms e^{-z} and e^{-2z} include Eqs. (45), (46), (48), and (52). By linearizing e^{-z} and e^{-2z} about $z^{(k)}$, where $z^{(k)} = \log(m^{(k)})$, that is,

$$e^{-z} = e^{-z^{(k)}}[1 - (z - z^{(k)})] \quad (53)$$

$$e^{-2z} = e^{-2z^{(k)}}[1 - 2(z - z^{(k)})] \quad (54)$$

they can be convexified as the following linear constraints, respectively,

$$k_1 z(t) + \underline{d}_1 \leq u_1(t) \leq \bar{k}_1 z(t) + \bar{d}_1 \quad (55)$$

$$0 \leq u_2(t) \leq \bar{k}_2 z(t) + \bar{d}_2 \quad (56)$$

$$k_5 z(t) + \underline{d}_5 \leq u_5(t) \leq \bar{k}_5 z(t) + \bar{d}_5 \quad (57)$$

$$|u_1(t_f)| \leq \bar{k}_{1f} z(t_f) + \bar{d}_{1f} \quad (58)$$

where it is straightforward to obtain $\underline{k}_1 = -\eta_{\min} e^{-z^{(k)}(t)}$, $\underline{d}_1 = -\bar{k}_1[1 + z^{(k)}(t)]$, $\bar{k}_1 = -\eta_{\max} e^{-z^{(k)}(t)}$, $\bar{d}_1 = -\bar{k}_1[1 + z^{(k)}(t)]$, $\bar{k}_2 = -2\bar{k}^2 e^{-2z^{(k)}(t)}$, $\bar{d}_2 = -\bar{k}_2[1 + 2z^{(k)}(t)]/2$, $k_5 = -T_{\min} e^{-z^{(k)}(t)}$, $d_5 = -k_5[1 + z^{(k)}(t)]$, $\bar{k}_5 = -T_{\max} e^{-z^{(k)}(t)}$, $\bar{d}_5 = -\bar{k}_5[1 + z^{(k)}(t)]$, $\bar{k}_{1f} = -\eta_f^* e^{-z^{(k)}(t_f)}$, and $\bar{d}_{1f} = -\bar{k}_{1f}[1 + z^{(k)}(t_f)]$.

Now, the remaining nonconvex control constraints in P1 are $u_1^2 = u_2$ [Eq. (44)] and $u_3^2 + u_4^2 = u_5^2$ [Eq. (47)]. In the following, we will first discuss convexification of Eq. (47), and then followed by that of Eq. (44).

Equation (47) is a nonconvex constraint related to the thrust. Its combination with the constraints in Eqs. (48–50) generates an admissible control set of the thrust visually shown in Fig. 2a. That is a part of the curved surface of a cone, and obviously nonconvex. We propose to relax Eq. (47) into the following convex second-order cone constraint:

$$[u_3(t)]^2 + [u_4(t)]^2 \leq [u_5(t)]^2 \quad (59)$$

Consequently, the admissible control set defined by the thrust becomes convex, as can be seen in Fig. 2b. This is very advantageous in lowering computational cost if a numerical algorithm can search a solution over a convex admissible set.

Simply changing the equality sign “=” to the inequality sign “≤” is a simple way to relax a nonconvex constraint, as also done in other engineering problems such as terminal guidance [36] and entry trajectory optimization [37–39]. Nevertheless, ensuring validity of the relaxation technique is a critical part that deserves rigorous analysis, which is to answer the question of whether the optimal solution of the corresponding relaxed problem is still feasible to the original constraints. It may happen that the solution lies in the interior or on the planar surfaces of the solid (see Fig. 2b), which is then infeasible to the admissible set in Fig. 2a. To make the relaxation technique (or the sign change) valid, we need to ensure that the solution can only lie on the curved surface of the solid in Fig. 2b; that is, to prove $[u_3^*(t)]^2 + [u_4^*(t)]^2 = [u_5^*(t)]^2$ is satisfied for the optimal solution. Fortunately, this is the case and the statement (with proof) will be provided later in Sec. III.C (see Proposition 1).

Next, for the nonconvex constraint (44) related to the aerodynamic forces, we may also simply relax it into (by a sign change)

$$[u_1(t)]^2 \leq u_2(t) \quad (60)$$

which is a convex constraint. Then, the next step is to prove that the optimal solution will make $[u_1^*(t)]^2 = u_2^*(t)$ hold. However, it turns out that the proof is unachievable. Our numerical demonstrations also indeed show that $[u_1^*(t)]^2 < u_2^*(t)$ is satisfied. Hence, the proposed relaxation technique is invalid.

An alternative way to convexify Eq. (44) is simply by linearization to get

$$k_u u_1(t) + d_u = u_2(t) \quad (61)$$

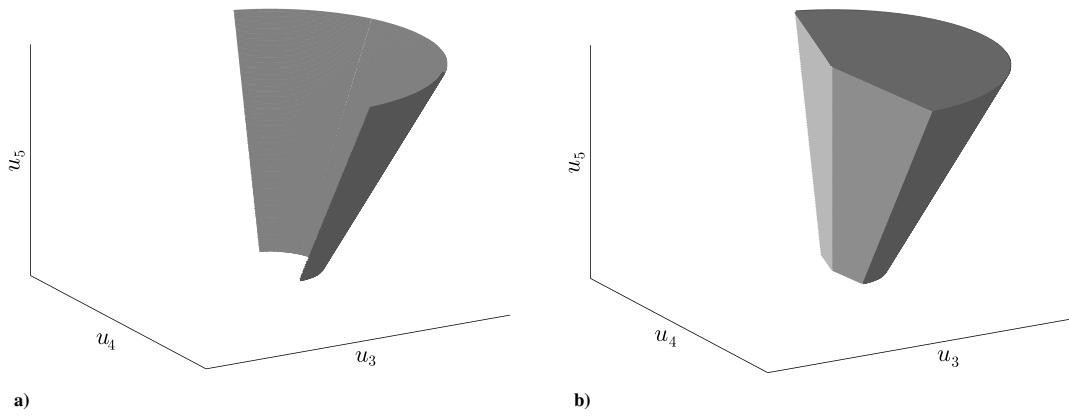


Fig. 2 Relaxing the control constraints defined by the thrust: a) original (nonconvex) admissible control set [cf. Eq. (47)]; b) relaxed (convex) admissible control set [cf. Eq. (59)].

where $k_u = 2u_1^{(k)}(t)$ and $d_u = [u_1^{(k)}(t)]^2 - k_u u_1^{(k)}(t)$ with $u_1^{(k)}(t)$ being the control u_1 at the k th iteration. This is the method we will use in this paper to convexify Eq. (44). Because linearization is used, a trust-region constraint on u_1 is imposed as follows:

$$|u_1(t) - u_1^{(k)}(t)| \leq \Delta_{u_1} \quad (62)$$

where Δ_{u_1} is a user-defined constant.

Remark 2: The preceding analysis indicates that the relaxation technique is not adopted to handle Eq. (44). This implies that validity of such a technique is problem dependent. We apply the technique to convexify $u_3^2 + u_4^2 = u_5^2$ and its validity can be guaranteed (see Proposition 1 in the next subsection). Note that this method, when compared with the linearization technique, will preserve all the nonlinearities (no approximation is involved by the sign change), which is generally beneficial to improve the rate of convergence of an iterative solution process [32].

C. SOCP Problem

With all the steps proposed in the previous two subsections, P1 is convexified as

$$\text{P2: } \min -z(t_f) \quad (63)$$

$$\text{s.t. } \dot{\mathbf{x}} = A(\mathbf{x}^{(k)})\mathbf{x} + B(\mathbf{x}^{(k)})\mathbf{u} + \mathbf{c}(\mathbf{x}^{(k)}) \quad (64)$$

$$|\mathbf{x}(t) - \mathbf{x}^{(k)}(t)| \leq \delta \quad (65)$$

$$k_u u_1(t) + d_u = u_2(t) \quad (66)$$

$$|u_1(t) - u_1^{(k)}(t)| \leq \Delta_{u_1} \quad (67)$$

$$\underline{k}_1 z(t) + \underline{d}_1 \leq u_1(t) \leq \bar{k}_1 z(t) + \bar{d}_1 \quad (68)$$

$$0 \leq u_2(t) \leq \bar{k}_2 z(t) + \bar{d}_2 \quad (69)$$

$$[u_3(t)]^2 + [u_4(t)]^2 \leq [u_5(t)]^2 \quad (70)$$

$$\underline{k}_5 z(t) + \underline{d}_5 \leq u_5(t) \leq \bar{k}_5 z(t) + \bar{d}_5 \quad (71)$$

$$u_4(t) \geq u_3(t) \tan \epsilon_{\min} \quad (72)$$

$$u_4(t) \leq u_3(t) \tan \epsilon_{\max} \quad (73)$$

$$\phi(\mathbf{x}(t_f)) = 0 \quad (74)$$

$$|u_1(t_f)| \leq \bar{k}_{1f} z(t_f) + \bar{d}_{1f}, u_4(t_f) = 0 \quad (75)$$

One crucial step from P1 to P2 is relaxing the constraint in Eq. (47) into the constraint in Eq. (70), which is also shown in Fig. 2. As mentioned in the preceding subsection, we need to ensure that the relaxed constraint is active for the optimal solution. This is given in Proposition 1 with the following two assumptions:

Assumption 1: The trust-region constraint in Eq. (65) is always inactive, that is, $|\mathbf{x}(t) - \mathbf{x}^{(k)}(t)| < \delta, \forall t \in [t_0, t_f]$.

Assumption 2: For the thrust magnitude constraints in Eq. (71), the lower bound constraint $\underline{k}_5 z(t) + \underline{d}_5 \leq u_5(t)$ cannot be active for all $[t_0, t_f]$.

Proposition 1: Let $\{\mathbf{x}^*(t); \mathbf{u}^*(t)\}$ be the solution of P2 over the fixed interval $[t_0, t_f]$. Then, under Assumptions 1 and 2, $[u_3^*(t)]^2 + [u_4^*(t)]^2 = [u_5^*(t)]^2$ is satisfied almost everywhere on $[t_0, t_f]$.

Proof: See Appendix A.

Remark 3: It can be seen from Appendix B that the proof is completed without any information or assumptions on the status of the thrust direction constraints in Eqs. (72) and (73). It is also independent of the status of the mixed state and control constraints from the aerodynamic forces, that is, Eqs. (68) and (69). This implies that the conclusion in Proposition 1 always holds no matter what the status of those inequality constraints is. Note that the two assumptions are all mild. Assumption 1 is easy to be met as long as we choose a large-enough value for the tolerance δ in Eq. (65) [see Eq. (81) in Sec. V for an example]. Though the size of the trust region is allowed to be large, our numerical observation shows that the difference between the new state profile and the previous one where linearization is applied turns out to be small (the reader can refer to Fig. 3 later in Sec. V for the difference between consecutive state profiles). That is, the value of $|\mathbf{x} - \mathbf{x}^{(k)}|$ is found to be small, and it even becomes smaller and smaller as the successive solution

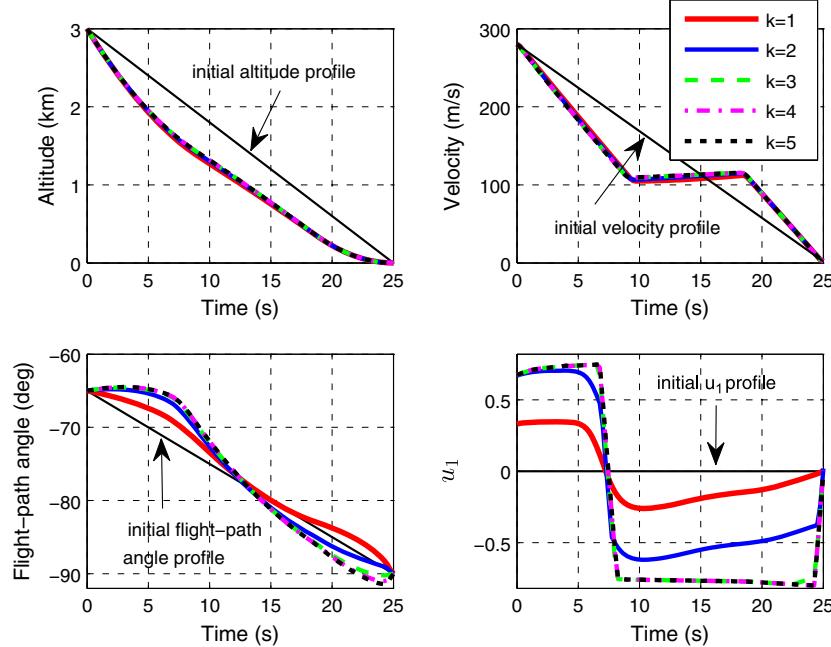


Fig. 3 Successive solutions in the first 5 iterations for mission 1 ($s_f^* = 950$ m).

procedure is converging. Hence, the linearization on the system dynamics is considered to be valid.

Assumption 2 means that minimum thrust magnitude occurs on at most a finite interval that is smaller than the whole interval $[t_0, t_f]$. In other words, there must exist a finite interval over which either maximum thrust magnitude is used or singular thrust control (an intermediate value between the lower and upper bound) is used. This assumption is easy to satisfy and always holds in all numerical examples we have tested.

Next, problem P2 can then be discretized over $[t_0, t_f]$ with $(N + 1)$ equally spaced discretized points (i.e., $\{t_0, \dots, t_N\}$), leading to an SOCP problem in the following form:

$$\text{P3: } \min \quad I^T \mathbf{y} \quad (76)$$

$$\text{s.t. } F(\mathbf{y}^{(k)})\mathbf{y} = \mathbf{g}(\mathbf{y}^{(k)}) \quad (77)$$

$$\|C_i(\mathbf{y}^{(k)})\mathbf{y} + \mathbf{d}_i(\mathbf{y}^{(k)})\|_2 \leq \mathbf{p}_i^T(\mathbf{y}^{(k)})\mathbf{y} + q_i(\mathbf{y}^{(k)}), \quad i = 1, \dots, \nu \quad (78)$$

where $\mathbf{y} \in \mathbb{R}^n$ is the optimization variable that collects all the states $\{\mathbf{x}(t_i)\}_{i=0, \dots, N}$ and controls $\{\mathbf{u}(t_i)\}_{i=0, \dots, N}$ from all discretized points, and coefficients of the constraints $F \in \mathbb{R}^{m \times n}$, $\mathbf{g} \in \mathbb{R}^m$, $C_i \in \mathbb{R}^{n_i \times n}$, $\mathbf{d}_i \in \mathbb{R}^{n_i}$, $\mathbf{p}_i \in \mathbb{R}^n$, and $q_i \in \mathbb{R}$ are dependent on $\mathbf{y}^{(k)}$, specifically on $\mathbf{x}^{(k)}$ and $u_1^{(k)}$ at the k th iteration, as can be seen in P2. Note that Eq. (77) in P3 is from the equality constraints in P2, and Eq. (78) in P3 is from all the inequality constraints in P2. To save space, the straightforward discretization process is not given here and the readers may refer to [37] for reference.

IV. Successive Solution Procedure

The goal of this section is to find the solution to P0 by proposing a successive solution procedure of iteratively solving the SOCP problem P3 until convergence is achieved. Specifically, at each iteration we first compute the $\mathbf{y}^{(k)}$ -dependent parameters in P3, and then solve the problem to get a new solution that will be used to update those parameters in the next iteration. This process is repeated until the current solution is within a small tolerance from the previous solution. The successive solution procedure is stated in detail as follows:

1) Set $k = 0$. Select an initial state profile $\mathbf{x}^{(0)} = [r^{(0)} \ s^{(0)} \ V^{(0)} \ \gamma^{(0)} \ z^{(0)}]^T$ and an initial control profile $u_1^{(0)}$, which can be used to construct $\mathbf{y}^{(0)}$.

2) At the $(k + 1)$ th iteration ($k \geq 0$), compute the $\mathbf{y}^{(k)}$ -dependent parameters in P3 [specifically dependent on $\mathbf{x}^{(k)}$ and $u_1^{(k)}$]. Then, solve P3 for a solution denoted as $\mathbf{y}^{(k+1)} = \{\mathbf{x}^{(k+1)}; \mathbf{u}^{(k+1)}\}$.

3) Check whether the following stopping criteria for convergence are satisfied:

$$\max_i |\mathbf{x}^{(k+1)}(t_i) - \mathbf{x}^{(k)}(t_i)| \leq \epsilon_x \quad (79)$$

$$\max_i |u_1^{(k+1)}(t_i) - u_1^{(k)}(t_i)| \leq \epsilon_{u_1} \quad (80)$$

where $\epsilon_x \in \mathbb{R}^5$ and $\epsilon_{u_1} \in \mathbb{R}$ are user-defined small tolerance for convergence. If the conditions in Eqs. (79) and (80) hold, go to step 4; otherwise, replace $\mathbf{y}^{(k)}$ with $\mathbf{y}^{(k+1)}$, set $k = k + 1$, and go back to step 2.

4) The successive solution procedure converges and $\{\mathbf{x}^{(k+1)}; \mathbf{u}^{(k+1)}\}$ is found to be the solution to P0. Stop.

The implementation and effectiveness of this successive solution procedure will be seen by numerical demonstrations in the next section.

Remark 4: To initialize the parameters in P3, $\mathbf{x}^{(0)}$ and $u_1^{(0)}$ need to be selected (note that there is no need to provide $u_i^{(0)}$, $i = 2, \dots, 5$). In the numerical examples presented in this paper, each element in $\mathbf{x}^{(0)}$ is chosen as a linear function of time from its initial value to the

desired (or estimated) final value, and $u_1^{(0)}$ is simply set as zero. Despite the rough selection, we will see in the next section that the successive solution procedure still converges quickly. It should be noted that it may speed up convergence if other methods can be used to find better $\mathbf{x}^{(0)}$ and $u_1^{(0)}$, such as those consistent with the physical motion of a landing rocket. Though a convergence proof is not available due to the complexity of the original highly nonlinear and constrained rocket landing problem, rapid convergence is observed in extensive numerical tests. The good convergence property benefits from the upfront investment in appropriately handling the nonlinear rocket dynamics and applying the relaxation technique, especially the attempt in keeping some nonlinearities in the process of convexifying the nonlinear dynamics (cf. Sec. III). In another successive convex optimization approach aiming to solve nonconvex optimal control problems with nonlinear system dynamics, the size of trust regions is regulated to ensure convergence of the approach [40].

V. Numerical Examples

In this section, numerical examples are provided to demonstrate effectiveness of the proposed method and show optimal characteristics of the controls. The rocket model used has parameters $m_0 = 55,000$ kg, $S_{\text{ref}} = 12.54$ m², $I_{\text{sp}} = 443$ s, and $T_{\text{max}} = 1375.6$ kN (which corresponds to an initial acceleration of 25 m/s²). In P2, the parameters in the trust-region constraints are set as

$$\delta = \left[\frac{10e3}{R_0} \ \frac{10e3}{R_0} \ \frac{200}{\sqrt{g_0 R_0}} \ \frac{50\pi}{180} \ \frac{3000}{m_0} \right]^T \quad (81)$$

$$\Delta_{u_1} = 0.5 \max(\bar{\eta}) \quad (82)$$

where $\bar{\eta}$ is given in Eq. (35). In the stopping criteria for convergence in Eqs. (79) and (80), we set

$$\epsilon_x = \left[\frac{1}{R_0} \ \frac{1}{R_0} \ \frac{0.1}{\sqrt{g_0 R_0}} \ \frac{0.1\pi}{180} \ \frac{1}{m_0} \right]^T, \quad \epsilon_{u_1} = 0.01 \quad (83)$$

The software MOSEK [41] is run on a desktop with Intel Core i7-3370 3.40 GHz to iteratively solve the SOCP problems in which the number of discretized points is selected as 101 ($N = 100$) and the constraints on the controls are listed in Table 1. Note that the angle-of-attack rate and thrust-direction rate constraints are also incorporated to avoid fast changes in α and ϵ , and in discretized form they can be approximated by linear constraints on consecutive discretized values of \mathbf{u} (derivation is omitted to save space).

In the following, numerical examples will be provided to illustrate efficiency and effectiveness of the proposed method, and also compare the performance between the proposed method and a general optimal control software called PSOPT (data for ‘PSOPT Optimal Control Solver User Manual, Release 3’ available online at <http://code.google.com/p/psopt/downloads/list> [retrieved 14 February 2015]).

A. Landing with Different Downrange Requirements

The initial and terminal conditions of the rocket landing problem are given in Table 2, and the time of flight is set as $t_f = 25$ s. We consider three different missions with $s_f^* = 950, 1050$, and 1150 m, denoted by mission 1, 2, and 3, respectively. The number of iterations

Table 1 Constraints on the controls

| Variable | Range |
|------------------------|--|
| Thrust magnitude | $T \in [30\%, 100\%]T_{\text{max}}$ |
| Angle of attack | $\alpha \in [165^\circ, 195^\circ]$ |
| Thrust direction | $\epsilon \in [-10^\circ, 10^\circ]$ |
| Angle-of-attack rate | $\dot{\alpha} \in [-20^\circ/\text{s}, 20^\circ/\text{s}]$ |
| Thrust-direction rate | $\dot{\epsilon} \in [-5^\circ/\text{s}, 5^\circ/\text{s}]$ |
| Final angle of attack | $\alpha(t_f) \in [-178^\circ, 182^\circ]$ |
| Final thrust direction | $\epsilon(t_f) = 0$ |

Table 2 Initial and terminal conditions for rocket landing with different downrange requirements

| States | Initial values | Final values |
|--------------|----------------|----------------|
| h , km | 3 | 0 |
| s , km | 0 | 0.95/1.05/1.15 |
| V , m/s | 280 | ≤ 2 |
| γ , ° | -65 | -90 |
| m , kg | 55,000 | Maximized |

(or the number of P3 that needs to be solved in the successive solution procedure) for the missions is 7, 6, and 6, respectively, and in each iteration it takes MOSEK about 0.1–0.2 s to solve P3. To visually show convergence performance of the successive solution procedure, we plot the first five successive solutions for mission 1 in Fig. 3. It is seen that the successive state and u_1 profiles are almost indiscernible to the scale of the plots after the third iteration. Note that this (pleasant) rapid convergence of the successive solution procedure is largely attributed to the appropriate convexification methods used to handle the rocket nonlinear dynamics and the nonconvex control constraints (see Sec. III). These methods try to preserve the nonlinearities inherent in the original problem.

The converged solutions plotted in Figs. 4–7 show that the control constraints and terminal constraints are all met for the missions. The

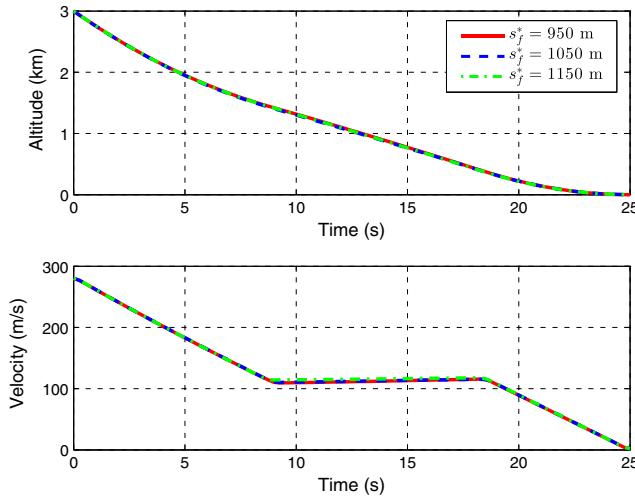


Fig. 4 Altitude and velocity histories for different downrange requirements.

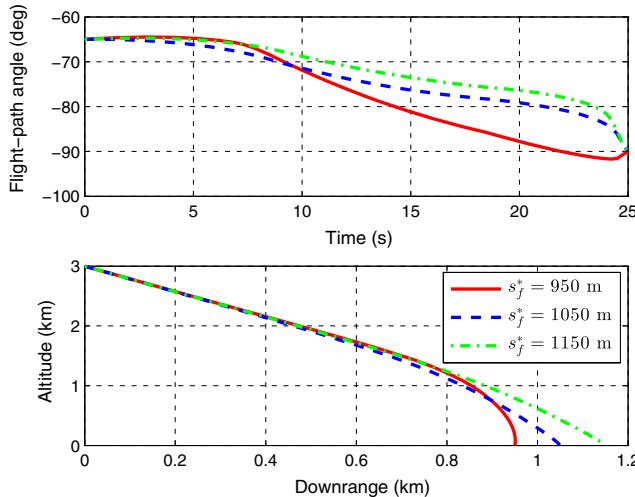


Fig. 5 Flight-path angle histories and altitude vs downrange profiles for different downrange requirements.

fuel consumption is found to be 5757.4, 5736.8, and 5703.9 kg, respectively. For convenience of discussion on the angle of attack α , we define

$$\alpha_{\text{tail}} := \alpha - \pi \quad (84)$$

which is the angle between the velocity vector and the tail of the rocket. Then, the constraint on α in Table 1 becomes $|\alpha_{\text{tail}}| \in [-15^\circ, 15^\circ]$. It is seen from Figs. 6 and 7 that the thrust magnitude profiles are almost the same, whereas the thrust direction and α_{tail} profiles show obvious differences for different downrange requirements. Specifically, compared with mission 1, the α_{tail} for mission 2 keeps being positive for a longer period of time, which will yield smaller $|\gamma|$, as shown in the top plot of Fig. 5. This in turn makes the rocket have a larger horizontal velocity and thus cover more downrange. For mission 3, the duration of positive α_{tail} is further increased. In addition, the thrust direction becomes negative almost for the whole flight (see Fig. 7), which is, together with the positive lift force, to generate smaller $|\gamma|$. Then, as before, more downrange can be covered. These analyses indicate that the angle of attack and the thrust direction are closely cooperated to meet the downrange requirements.

It should be pointed out that the thrust magnitude profile has a bang-bang structure, as shown in Fig. 6. Interestingly, the optimal angle of attack also shows the same pattern in that either the

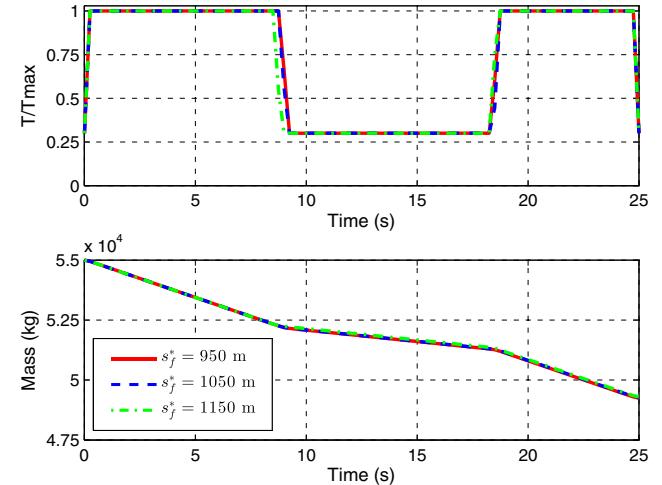


Fig. 6 Thrust magnitude and mass histories for different downrange requirements.

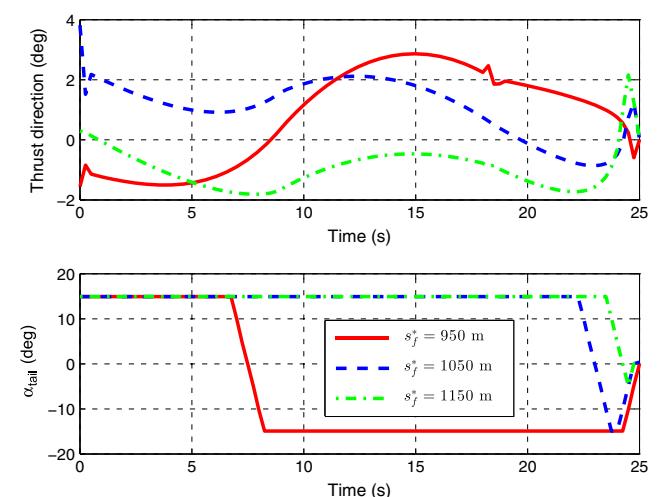


Fig. 7 Thrust direction and angle-of-attack histories for different downrange requirements.

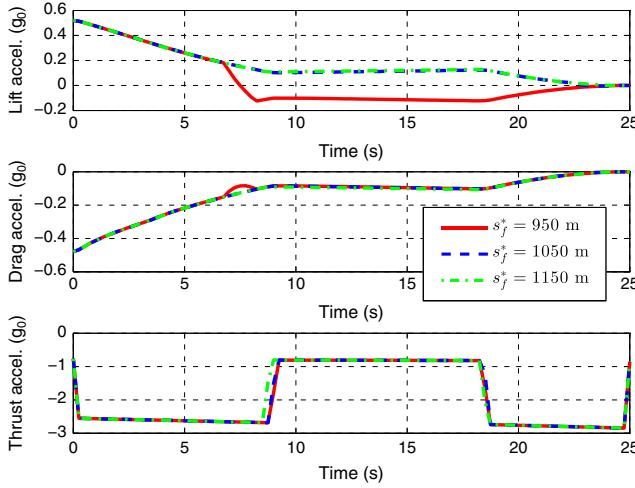


Fig. 8 Lift, drag, and thrust acceleration histories for different downrange requirements.

maximum α_{tail} or minimum α_{tail} is used (see Fig. 7). This is consistent with the objective of minimizing fuel cost, since flying at maximum $|\alpha_{tail}|$, or equivalently at maximum drag, can greatly reduce the energy of the rocket so as to save the consumption of fuel. The lift, drag, and thrust acceleration histories are illustrated in Fig. 8, which can tell us the relative acceleration of the thrust and aerodynamic forces compared with the gravitational acceleration g_0 . For missions in which aerodynamic accelerations are larger than g_0 , the readers can refer to the conference version of this paper in Ref. [42].

B. Landing with Singular Control

In this subsection, another example is provided with its initial and terminal conditions given in Table 3 (cf. Table 1 for constraints on the controls), and the time of flight is set as $t_f = 15$ s. The solution obtained by the proposed method is illustrated by the red solid lines in Figs. 9–11. In Fig. 11, it shows that the final pitch angle is close to 90° and the final thrust gimbal angle is almost zero, which is useful to avoid tipping the rocket over. Figure 12 plots the optimal values of $u_3^2 + u_4^2$ and u_5^2 . The identical values mean that the relaxed constraint $u_3^2 + u_4^2 \leq u_5^2$ is always active and thus Proposition 1 in Sec. III is verified.

Again, as in the preceding subsection, both the thrust magnitude profile and the angle of attack profile have bang-bang structures (cf. Fig. 10). Note that this is the case in most examples we have tested. The bang-bang characteristic will benefit reusable rockets because there is no need to frequently manipulating the thrust magnitude and the angle of attack to achieve fuel-optimal landing.

To show the existence of intermediate thrust magnitude (also called singular control) in rocket landing, we set $T_{min} = 0$, that is, $T \in [0\%, 100\%]T_{max}$, and the optimal thrust is found to have this feature. The obtained solution is given in the blue dashed lines in Figs. 9–11. Clearly, the top plot in Fig. 10 shows that in a finite interval the thrust magnitude has intermediate values between 0 and T_{max} . It should be pointed out that when the time of flight is free the fuel-optimal thrust magnitude has been proved to be bang-bang for planetary landing in the absence of aerodynamic forces, such as in Mars powered descent landing [10]. Hence, it implies that the inclusion of the aerodynamic forces can affect the optimal

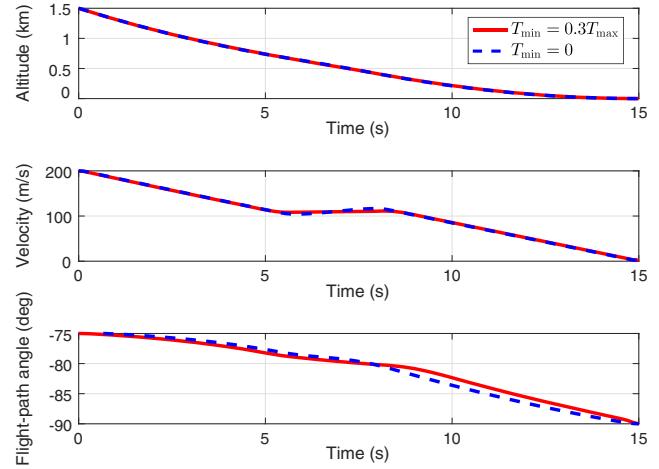


Fig. 9 Altitude and velocity histories for $T_{min} = 0.3T_{max}$ and $T_{min} = 0$.

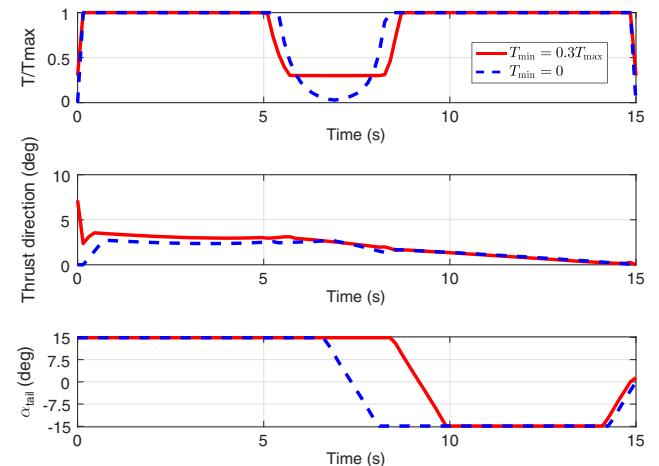


Fig. 10 Thrust magnitude, thrust direction, and angle-of-attack histories for $T_{min} = 0.3T_{max}$ and $T_{min} = 0$.

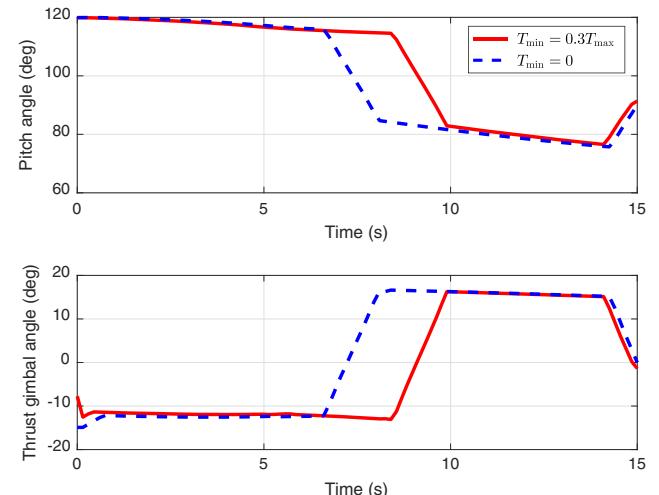


Fig. 11 Pitch angle and thrust gimbal angle histories for $T_{min} = 0.3T_{max}$ and $T_{min} = 0$. The pitch angle is defined as the angle between the nose of the rocket and the horizon.

Table 3 Initial and terminal conditions for rocket landing with singular control

| States | Initial values | Final values |
|--------------|----------------|--------------|
| h , km | 1.5 | 0 |
| s , km | 0 | 0.3 |
| V , m/s | 200 | ≤ 2 |
| γ , ° | -75 | -90 |
| m , kg | 55,000 | Maximized |

characteristics of the thrust. Indeed, singular control of the thrust was also previously found in optimal atmospheric rocket trajectories in the launch phase [43]. Nevertheless, unlike the indirect method that has a time-consuming trial-and-error process of guessing and adjusting the entry and departure times of singular subarcs [34,43], this convex optimization-based direct method can automatically and

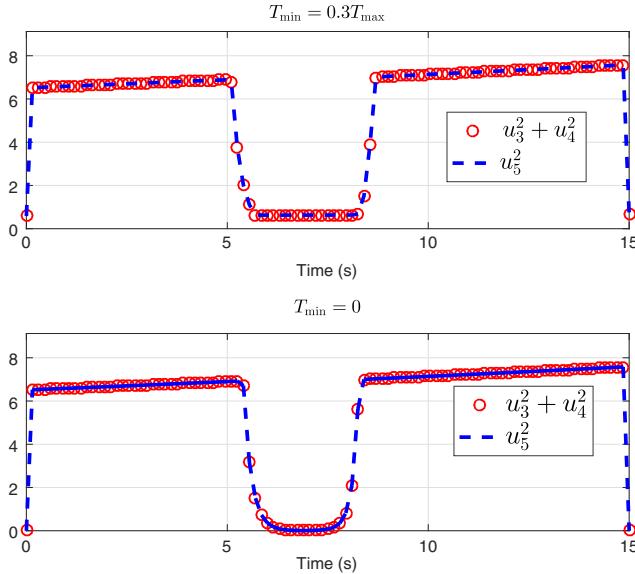


Fig. 12 Check the optimal values of $u_3^2 + u_4^2$ and u_5^2 for $T_{\min} = 0.3T_{\max}$ and $T_{\min} = 0$.

efficiently determine the location of and the number of the singular subarc(s) in the rocket landing trajectory (the successive solution procedure converges in just seven iterations).

C. Comparison with an Optimal Control Software

An open-source optimal control software PSOPT written in C++ is used to solve the original rocket landing problem P0 [cf. Eqs. (12) and (13)] and its solution will be compared with that of the proposed method (note that MOSEK is written in C). Note that directly solving a discretized version of P0 by general NLP solvers, such as IPOPT, is found to have difficulty to converge. In PSOPT, the P0 is first discretized by the Chebyshev pseudospectral method (different pseudospectral discretization method can be set by users), and then the resulting problem is solved by the default NLP solver IPOPT [44]. The mission 1 in Sec. V.A, with the angle-of-attack rate and thrust-direction rate constraints removed and the time of flight changed to 19 s, is selected as a test case. The solutions obtained by PSOPT and the proposed method are plotted in Figs. 13–15. It is seen that the state and thrust magnitude histories are almost identical, whereas the thrust direction and angle-of-attack histories show similar patterns. Hence, the comparison verifies the validity of the proposed method in this paper.

It should be noted that it is difficult to make PSOPT converge in solving P0, because PSOPT is sensitive to user-specified settings, including the initial guess of the optimization variables, the initial

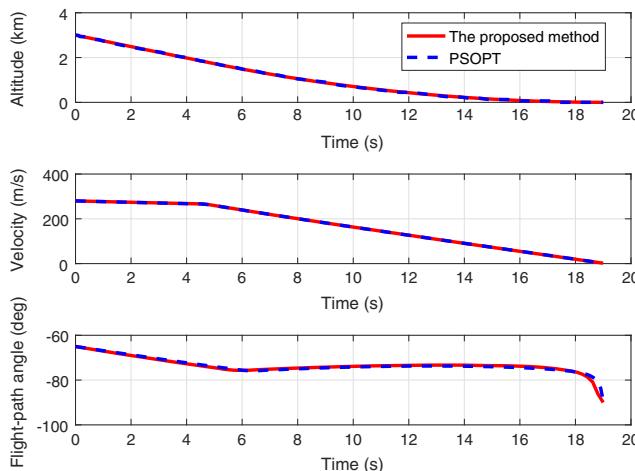


Fig. 13 Altitude, velocity, and flight-path angle histories for different solution methods.

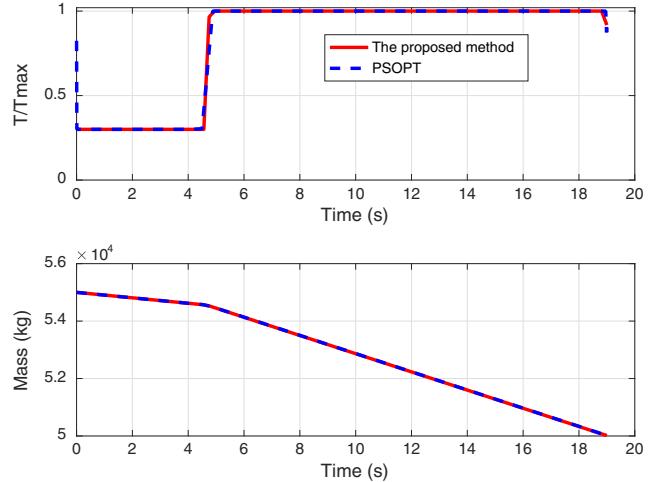


Fig. 14 Thrust magnitude and mass histories for different solution methods.

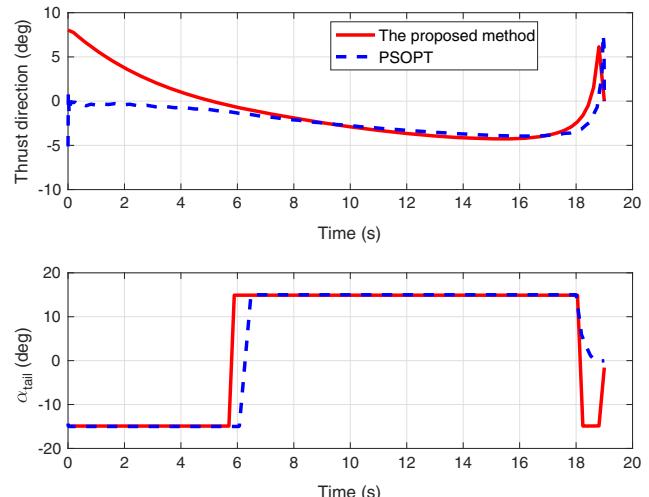


Fig. 15 Thrust direction and angle-of-attack histories for different solution methods.

number of discretized nodes, and the ordinary differential equation error (ODE) tolerance. For the rocket landing problem, inappropriate values on those parameters can easily make PSOPT fail to find a solution. For instance, the solution in Figs. 13–15 corresponds to an initial number of nodes as 40. However, if it is set as 30, 50, or 60, PSOPT does not converge. Furthermore, according to our numerical experiences changing the initial guess of the optimization variables or the ODE tolerance can also make PSOPT fail. It is a tedious trial-and-error process to specify proper settings for PSOPT to obtain the solution in Figs. 13–15. In contrast, these deficiencies in PSOPT do not exist in the proposed method for which it is not sensitive to the number of discretized nodes and no initial guess is required for the optimization variables. Additionally, the computation time by PSOPT and the proposed method is 125.4 and 1.6 s, respectively. Hence, for the highly constrained rocket landing problem the proposed method is more robust and efficient than the software PSOPT.

The work of reformulation and theoretical analysis in this paper contributes that only convex optimization problems need to be solved. Similar efforts were done when optimization-based guidance is sought in Mars powered landing [10,11,17], and customized interior point methods can be used to further improve the efficiency for engineering practice [17]. Nevertheless, it is certainly not trivial to make a nonconvex optimal control problem become convex and also ensure effectiveness and efficacy of the convex optimization approach. For many (complex) aerospace problems, pseudospectral methods and general NLP solvers are still good candidates to solve the problems.

VI. Conclusions

This paper represents an attempt to investigate the fuel-optimal rocket landing problem in the vertical plane with the rocket's angle of attack, thrust magnitude, and thrust direction as control inputs. Simultaneous presence of the aerodynamic forces and propulsion makes the corresponding optimal control problem challenging to solve. This paper presents a modern convex optimization approach to solve the original problem as a sequence of SOCP problems. The proposed method provides an efficient way, with potential for onboard real-time application, to determine fuel-optimal landing trajectories with full consideration on the aerodynamic forces and propulsion. Numerical demonstrations have shown that in general the optimal thrust magnitude profile has a bang-bang structure and maximum drag is almost always used to save fuel consumption.

In addition to providing an effective and efficient method to solve the rocket landing problem, this work embraces the trend known as computational guidance and control in which significant upfront investment in problem formulation and analysis is required to increase the computational efficiency. This work has also shown the power of applying convex optimization to highly nonlinear systems with both aerodynamic forces and propulsion as control inputs.

Appendix: Proof of Proposition 1

We first apply the direct adjoining approach in the maximum principle [45] to derive the necessary conditions of optimality for problem P2. Define the Hamiltonian and Lagrangian for problem P2 as follows:

$$\begin{aligned} H = & p_r(a_{13}V + a_{14}\gamma + c_1) + p_s(a_{23}V + a_{24}\gamma + c_2) \\ & + p_v(a_{31}r + a_{34}\gamma + a_{35}z + c_3 + b_{32}u_2 + b_{33}u_3) \\ & + p_\gamma(a_{41}r + a_{43}V + a_{44}\gamma + c_4 + b_{41}u_1 + b_{44}u_4) + p_z b_{55}u_5 \end{aligned} \quad (\text{A1})$$

$$\begin{aligned} L = & H + \mu_{u_1}^-(u_1 - u_1^{(k)} - \Delta_{u_1}) + \mu_{u_1}^+(u_1^{(k)} + \Delta_{u_1} - u_1) \\ & + \mu_1^-(u_1 - \underline{k}_1 z(t) - \underline{d}_1) + \mu_1^+(\bar{k}_1 z + \bar{d}_1 - u_1) \\ & + \mu_2^- u_2 + \mu_2^+(\bar{k}_2 z + \bar{d}_2 - u_2) + \mu_{345}(u_5^2 - u_3^2 - u_4^2) \\ & + \mu_5^-(u_5 - \underline{k}_5 z - \underline{d}_5) + \mu_5^+(\bar{k}_5 z + \bar{d}_5 - u_5) \\ & + \mu_{34}^-(u_4 - u_3 \tan \epsilon_{\min}) + \mu_{34}^+(u_3 \tan \epsilon_{\max} - u_4) \end{aligned} \quad (\text{A2})$$

where a_{ij} , b_{ij} , and c_i are the nonzero elements in $A(\mathbf{x}^{(k)})$, $B(\mathbf{x}^{(k)})$, and $\mathbf{c}(\mathbf{x}^{(k)})$, respectively; $\mathbf{p}(t) = [p_r(t) p_s(t) p_v(t) p_\gamma(t) p_z(t)]^T \in \mathbb{R}^5$ is the costate vector; and μ , with different superscripts and subscripts, are Lagrangian variables associated with each inequality state/control constraint in P2. Note that the trust-region constraint is not considered in the definition of L under Assumption 1. Then, the necessary conditions of optimality include the following:

The complementary slackness conditions are that the Lagrangian variables in L are all nonnegative, and any Lagrangian variable must be zero whenever its corresponding inequality constraint is inactive.

The costate vector satisfies

$$\dot{p}_r = -\partial_r L = -a_{31}p_v - a_{41}p_\gamma \quad (\text{A3})$$

$$\dot{p}_s = -\partial_s L = 0 \quad (\text{A4})$$

$$\dot{p}_v = -\partial_v L = -a_{13}p_r - a_{23}p_s - a_{43}p_\gamma \quad (\text{A5})$$

$$\dot{p}_\gamma = \partial_\gamma L = a_{14}p_r - a_{24}p_s - a_{34}p_v - a_{44}p_\gamma \quad (\text{A6})$$

$$\dot{p}_z = -\partial_z L = -a_{35}p_v + \underline{k}_1\mu_1^- - \bar{k}_1\mu_1^+ - \bar{k}_2\mu_2^+ + \underline{k}_5\mu_5^- - \bar{k}_5\mu_5^+ \quad (\text{A7})$$

where the exact form of a_{ij} can be found in Eq. (33).

There exists a constant $p_0 \leq 0$ such that a transversality condition is

$$p_z(t_f) = -p_0 \quad (\text{A8})$$

The stationary conditions are (here we only give $\partial_{u_3}L = 0$, $\partial_{u_4}L = 0$, and $\partial_{u_5}L = 0$, that will later be used)

$$b_{33}p_v - 2\mu_{345}u_3 - \mu_{34}^- \tan \epsilon_{\min} + \mu_{34}^+ \tan \epsilon_{\max} = 0 \quad (\text{A9})$$

$$b_{44}p_\gamma - 2\mu_{345}u_4 + \mu_{34}^- - \mu_{34}^+ = 0 \quad (\text{A10})$$

$$b_{55}p_z + 2\mu_{345}u_5 + \mu_5^- - \mu_5^+ = 0 \quad (\text{A11})$$

A nontriviality condition is

$$[p_0 \quad \mathbf{p}(t)^T]^T \neq 0, \quad \forall t \in [t_0, t_f] \quad (\text{A12})$$

Now, we start the proof by assuming that there exists a finite interval $[t_1, t_2] \in [t_0, t_f]$ over which $[u_3^*(t)]^2 + [u_4^*(t)]^2 < [u_5^*(t)]^2$. This assumption implies that $\mu_{345} = 0$. In the following, we will derive a condition that contradicts Eq. (A12).

First, when the thrust direction constraints in both Eqs. (72) and (73) are inactive, we have $\mu_{34}^- = \mu_{34}^+ = 0$. This condition, together with $\mu_{345} = 0$, can be substituted into Eqs. (A9) and (A10) to get $p_v = p_\gamma = 0$. Using Eqs. (A3–A6) and $p_v = p_\gamma = 0$ can give $p_r = p_s = p_z = 0$.

On the other hand, when one of those thrust constraints is active and without loss of generality the one in Eq. (72) is assumed to be active, we have $\mu_{34}^+ = 0$. Then, Eqs. (A9) and (A10) result in

$$p_v = k_\gamma p_\gamma \quad (\text{A13})$$

or

$$p_\gamma = k_v p_v \quad (\text{A14})$$

where $k_\gamma = \tan \epsilon_{\min}/V$ and $k_v = V \cot \epsilon_{\min}$. Note that the superscript (k) in the state variables is suppressed for notational simplicity.

We differentiate p_v in Eq. (A13), with the help of Eq. (A6), to get a formula to compute \dot{p}_v . This value is used to replace the left side of Eq. (A5) to yield a linear equation on p_r , p_s , and p_γ , that is,

$$k_{11}p_r + k_{12}p_s + k_{13}p_\gamma = 0 \quad (\text{A15})$$

where

$$\begin{aligned} k_{11} &= \sin \gamma + \cos \gamma \tan \epsilon_{\min} \\ k_{12} &= \cos \gamma + \sin \gamma \tan \epsilon_{\min} \\ k_{13} &= \frac{\cos \gamma \sec^2 \epsilon_{\min} - \sin \gamma \tan \epsilon_{\min} - r^2 \tan \epsilon_{\min} \dot{V}}{r^2 V^2} \end{aligned} \quad (\text{A16})$$

We also differentiate p_γ in Eq. (A14), with the help of Eq. (A5), to get a formula to compute \dot{p}_γ . This value can be used to replace the left side of Eq. (A6) to get another linear equation on p_r , p_s , and p_γ , that is,

$$k_{21}p_r + k_{22}p_s + k_{23}p_\gamma = 0 \quad (\text{A17})$$

where

$$\begin{aligned} k_{21} &= -V \cos \gamma - V \cot \epsilon_{\min} \sin \gamma \\ k_{22} &= -V \cos \gamma \cot \epsilon_{\min} - V \sin \gamma \\ k_{23} &= \frac{-\cos \gamma \csc \epsilon_{\min} \sec \epsilon_{\min} + \sin \gamma + r^2 \dot{V}}{r^2 V} \end{aligned} \quad (\text{A18})$$

It is seen that the linear equations (A15) and (A17) contain three variables, p_r , p_s , and p_γ . The software "Wolfram Mathematica" is

used to solve the equations for p_r and p_γ as functions of p_s :

$$p_r = \frac{\cos \gamma + \sin \gamma \tan \epsilon_{\min}}{\sin \gamma + \cos \gamma \tan \epsilon_{\min}} p_s \quad (\text{A19})$$

$$p_\gamma = 0 \quad (\text{A20})$$

When $p_\gamma = 0$ [cf. Eq. (A20)], Eq. (A13) gives $p_v = 0$. Substituting $p_\gamma = p_v = 0$ into Eq. (A3) can generate

$$\dot{p}_r = 0 \quad (\text{A21})$$

which means that p_r is a constant. Also, Eq. (A4) indicates that p_s is a constant. The fact that both p_r and p_s are constants will contradict Eq. (A19) unless $p_r = p_s = 0$, because the coefficient of p_s in Eq. (A19) is time-varying.

Until now, the preceding analysis has proven $p_r = p_s = p_v = p_\gamma = 0$ for $t \in [t_1, t_2]$. Since p_r , p_s , p_v , and p_γ are continuous and Eqs. (A3–A6) are affine with respect to the costate variables, we can actually get $p_r = p_s = p_v = p_\gamma = 0$ for $t \in [t_0, t_f]$. In the following, to contradict the condition in Eq. (A12), it is still desired to prove $p_z = 0$ and $p_0 = 0$. This will be completed based on the status of the thrust magnitude constraints in Eq. (71). According to Assumption 2, two cases are considered for Eq. (71): there exists a finite interval over which neither the lower nor the upper bound constraint is active (that is singular control), or there exists a finite interval over which the upper bound constraint $u_5(t) \leq \bar{k}_5 z(t) + \bar{d}_5$ is active.

Case 1: There exists a finite interval $t \in [t_i, t_j]$ over which neither the lower nor the upper bound constraint in Eq. (71) is active.

In this case, we have $\mu_5^- = \mu_5^+ = 0$ for $t \in [t_i, t_j]$. Then, Eq. (A11) results in $p_z = 0$ for $t \in [t_i, t_j]$. Note that H is a constant because it is not explicitly dependent on the time. This condition, together with $p = 0$ for $t \in [t_i, t_j]$, implies that

$$H = 0, \quad \forall t \in [t_0, t_f] \quad (\text{A22})$$

Specifically, when $t = t_f$ we have

$$H(t_f) = -(1/I_{\text{sp}}) p_z(t_f) u_5(t_f) = 0 \quad (\text{A23})$$

It is intuitive that the thrust magnitude is nonzero at $t = t_f$ in order to decelerate the rocket for a soft landing, that is, $u_5(t_f) \neq 0$. Hence, Eq. (A23) implies that $p_z(t_f) = 0$. This condition can be substituted into Eq. (A8) to get $p_0 = 0$.

Case 2: There exists a finite interval $t \in [t_m, t_n]$ over which the upper bound constraint $u_5(t) \leq \bar{k}_5 z(t) + \bar{d}_5$ is active.

The condition that the upper bound constraint is active gives $\mu_5^- = 0$ for $t \in [t_m, t_n]$. Substituting $\mu_5^- = 0$ into Eq. (A11) yields

$$-p_z/I_{\text{sp}} - \mu_5^+ = 0 \quad (\text{A24})$$

Since the Lagrangian multiplier $\mu_5^+ \geq 0$, Eq. (A24) implies that

$$p_z(t) \leq 0, \quad \forall t \in [t_m, t_n] \quad (\text{A25})$$

Note that the value of the Hamiltonian, $H = -(1/I_{\text{sp}}) p_z u_5$, is a constant and it is denoted as c . With Eq. (A25) and the condition $u_5 \geq 0$ (note that $u_5 = T/m$), the sign of the constant c can be determined as follows:

$$c = -(1/I_{\text{sp}}) p_z u_5 \geq 0 \quad (\text{A26})$$

This condition $c \geq 0$, together with $u_5 \geq 0$, results in

$$p_z(t) = -c I_{\text{sp}}/u_5 \leq 0, \quad \forall t \in [t_0, t_f] \quad (\text{A27})$$

When $t = t_f$, the preceding equation generates

$$p_z(t_f) \leq 0 \quad (\text{A28})$$

Recall Eq. (A8) and we can get

$$p_z(t_f) = -p_0 \geq 0 \quad (\text{A29})$$

since $p_0 \leq 0$. Combining Eqs. (A28) and (A29) immediately yields

$$p_z(t_f) = p_0 = 0 \quad (\text{A30})$$

The analysis in both Case 1 and Case 2 has shown that $p_z = 0$ and $p_0 = 0$. Hence, we have run into a contradiction to the condition in Eq. (A12). This is to say that the original assumption does not hold and the proof is completed.

Acknowledgments

This research was supported by the Beijing Institute of Technology Research Fund Program for Young Scholars. The author would like to thank Ping Lu from San Diego State University for his very valuable suggestions to improve this paper.

References

- [1] Cook, S., "The Reusable Launch Vehicle Technology Program," *International Space Planes and Hypersonic Systems and Technologies Conferences*, AIAA Paper 1995-6153, 1995.
doi:10.2514/6.1995-6153
- [2] Gallaher, M., Coughlin, D., and Krupp, D., "A Guidance and Control Assessment of Three Vertical Landing Options for RLV," *Guidance, Navigation, and Control Conference*, AIAA Paper 1996-3702, 1996.
doi:10.2514/6.1996-3702
- [3] Lu, P., Neal, D., Su, S., Horneman, K., and Schierman, J., "Pitch-Over Maneuver and Guidance for Rocket-Back Boosters," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2011-6501, Aug. 2011.
doi:10.2514/6.2011-6501
- [4] Su, S., Neal, D., Horneman, K., Schierman, J., and Lu, P., "Integrated Rocket-Back Guidance and a Comparative Rocket-Back Study," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2011-6504, Aug. 2011.
doi:10.2514/6.2011-6504
- [5] Ishijima, Y., Matsumoto, S., and Hayashi, K., "Re-Entry and Terminal Guidance for Vertical-Landing TSTO (Two-Stage to Orbit)," *Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 1998-4120, 1998.
doi:10.2514/6.1998-4120
- [6] Szmulik, M., Acikmese, B., and Berning, A. W., "Successive Convexification for Fuel-Optimal Powered Landing with Aerodynamic Drag and Non-Convex Constraints," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2016-0378, Jan. 2016.
doi:10.2514/6.2016-0378
- [7] Blackmore, L., "Autonomous Precision Landing of Space Rockets," *The Bridge*, Vol. 4, No. 46, 2016, pp. 15–20.
- [8] Lu, P., "Introducing Computational Guidance and Control," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 193–193.
doi:10.2514/1.G002745
- [9] Tsiotras, P., and Mesbah, M., "Toward an Algorithmic Control Theory," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 194–196.
doi:10.2514/1.G002754
- [10] Acikmese, B., and Ploen, S. R., "Convex Programming Approach to Powered Descent Guidance for Mars Landing," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 5, 2007, pp. 1353–1366.
doi:10.2514/1.27553
- [11] Blackmore, L., Acikmese, B., and Scharf, D. P., "Minimum Landing Error Powered Descent Guidance for Mars Landing Using Convex Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 4, 2010, pp. 1161–1171.
doi:10.2514/1.47202
- [12] Acikmese, B., Carson, J., and Blackmore, L., "Lossless Convexification of Nonconvex Control Bound and Pointing Constraints of the Soft Landing Optimal Control Problem," *IEEE Transactions on Control*

- Systems Technology*, Vol. 21, No. 6, 2013, pp. 2104–2113.
doi:10.1109/TCST.2012.2237346
- [13] Harris, M. W., and Acikmese, B., “Maximum Divert for Planetary Landing Using Convex Optimization,” *Journal of Optimization Theory and Applications*, Vol. 162, No. 3, 2014, pp. 975–995.
doi:10.1007/s10957-013-0501-7
- [14] Boyd, S., and Vandenberghe, L., *Convex Optimization*, Cambridge Univ. Press, New York, 2004, Chaps. 1, 11.
- [15] Domahidi, A., Chu, E., and Boyd, S., “ECOS: An SOCP Solver for Embedded System,” *European Control Conference*, IEEE Publ., Piscataway, NJ, July 2013, pp. 3071–3076.
doi:10.23919/ECC.2013.6669541
- [16] Dueri, D., Acikmese, B., Scharf, D. P., and Harris, M. W., “Customized Real-Time Interior-Point Methods for Onboard Powered-Descent Guidance,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 197–212.
doi:10.2514/1.G001480
- [17] Scharf, D. P., Acikmese, B., Dueri, D., Benito, J., and Casoliva, J., “Implementation and Experimental Demonstration of Onboard Powered-Descent Guidance,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 213–229.
doi:10.2514/1.G000399
- [18] Acikmese, B., and Blackmore, L., “Lossless Convexification for a Class of Optimal Problems with Nonconvex Control Constraints,” *Automatica*, Vol. 47, No. 2, 2011, pp. 341–347.
doi:10.1016/j.automatica.2010.10.037
- [19] Harris, M. W., and Acikmese, B., “Lossless Convexification of Non-Convex Optimal Control Problems for State Constrained Linear Systems,” *Automatica*, Vol. 50, No. 9, 2014, pp. 2304–2311.
doi:10.1016/j.automatica.2014.06.008
- [20] Harris, M. W., and Acikmese, B., “Minimum Time Rendezvous of Multiple Spacecraft Using Differential Drag,” *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 2, 2014, pp. 365–373.
doi:10.2514/1.61505
- [21] Morgan, D., Chung, S.-J., and Hadaegh, F. Y., “Model Predictive Control of Swarms of Spacecraft Using Sequential Convex Optimization,” *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1725–1740.
doi:10.2514/1.G000218
- [22] Dai, R., and Sun, C., “Path Planning of Spatial Rigid Motion with Constrained Attitude,” *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 8, 2015, pp. 1356–1365.
doi:10.2514/1.G000705
- [23] Misra, G., and Bai, X., “Optimal Path Planning for Free-Flying Space Manipulators via Sequential Convex Programming,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 11, 2017, pp. 3019–3026.
doi:10.2514/1.G002487
- [24] Misra, G., and Bai, X., “Task-Constrained Trajectory Planning of Free-Floating Space-Robotic Systems Using Convex Optimization,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 11, 2017, pp. 2857–2870.
doi:10.2514/1.G002405
- [25] Yang, H., Bai, X., and Baoyin, H., “Rapid Generation of Time-Optimal Trajectories for Asteroid Landing via Convex Optimization,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 3, 2017, pp. 628–641.
doi:10.2514/1.G002170
- [26] Sun, C., Liu, Y., Dai, R., and Grymin, D., “Two Approaches for Path Planning of Unmanned Aerial Vehicles with Avoidance Zones,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 8, 2017, pp. 2076–2083.
doi:10.2514/1.G002314
- [27] Wang, Z., and Grant, M. J., “Constrained Trajectory Optimization for Planetary Entry via Sequential Convex Programming,” *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 10, 2017, pp. 2603–2615.
doi:10.2514/1.G002150
- [28] Pinson, R. M., and Lu, P., “Trajectory Design Employing Convex Optimization for Landing on Irregularly Shaped Asteroids,” *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 6, 2018, pp. 1243–1256.
doi:10.2514/1.G003045
- [29] Szmuk, M., and Acikmese, B., “Successive Convexification for 6-DoF Mars Rocket Powered Landing with Free-Final-Time,” *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2018-0617, Jan. 2018.
doi:10.2514/6.2018-0617
- [30] Sagliano, M., and Mooij, E., “Optimal Drag-Energy Entry Guidance via Pseudospectral Convex Optimization,” *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2018-1315, Jan. 2018.
doi:10.2514/6.2018-1315
- [31] Sagliano, M., “Pseudospectral Convex Optimization for Powered Descent and Landing,” *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 2, 2018, pp. 320–334.
doi:10.2514/1.G002818
- [32] Liu, X., Lu, P., and Pan, B., “Survey of Convex Optimization for Aerospace Applications,” *Astrodynamic*s, Vol. 1, No. 1, 2017, pp. 23–40.
doi:10.1007/s42064-017-0003-8
- [33] Vinh, N. X., *Optimal Trajectories in Atmospheric Flight*, Elsevier, New York, 1981, Chap. 3.
- [34] Oberle, H. J., “Numerical Computation of Singular Control Functions in Trajectory Optimization Problems,” *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 1, 1990, pp. 153–159.
doi:10.2514/3.20529
- [35] Alizadeh, F., and Goldfarb, D., “Second-Order Cone Programming,” *Mathematical Programming*, Vol. 95, No. 1, 2003, pp. 3–51.
doi:10.1007/s10107-002-0339-5
- [36] Liu, X., Shen, Z., and Lu, P., “Exact Convex Relaxation for Optimal Flight of Aerodynamically Controlled Missiles,” *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 52, No. 4, 2016, pp. 1881–1892.
doi:10.1109/TAES.2016.150741
- [37] Liu, X., Shen, Z., and Lu, P., “Entry Trajectory Optimization by Second-Order Cone Programming,” *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 2, 2016, pp. 227–241.
doi:10.2514/1.G001210
- [38] Liu, X., Shen, Z., and Lu, P., “Solving the Maximum-Crossrange Problem via Successive Second-Order Cone Programming with a Line Search,” *Aerospace Science and Technology*, Vol. 47, Dec. 2015, pp. 10–20.
doi:10.1016/j.ast.2015.09.008
- [39] Liu, X., and Shen, Z., “Rapid Smooth Entry Trajectory Planning for High Lift/Drag Hypersonic Glide Vehicles,” *Journal of Optimization Theory and Applications*, Vol. 168, No. 3, 2016, pp. 917–943.
doi:10.1007/s10957-015-0831-8
- [40] Mao, Y., Szmuk, M., and Acikmese, B., “Successive Convexification of Non-Convex Optimal Control Problems and its Convergence Properties,” *IEEE 55th Conference on Decision and Control (CDC)*, IEEE Publ., Piscataway, NJ, Dec. 2016, pp. 3636–3641.
doi:10.1109/CDC.2016.7798816
- [41] Andersen, E. D., Roos, C., and Terlaky, T., “On Implementing a Primal-Dual Interior-Point Method for Conic Quadratic Optimization,” *Mathematical Programming*, Vol. 95, No. 2, 2003, pp. 249–277.
doi:10.1007/s10107-002-0349-3
- [42] Liu, X., “Fuel-Optimal Rocket Landing with Aerodynamic Controls,” *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2017-1732, 2017.
doi:10.2514/6.2017-1732
- [43] Kumar, R., and Kelley, H. J., “Singular Optimal Atmospheric Rocket Trajectories,” *Journal of Guidance, Control, and Dynamics*, Vol. 11, No. 4, 1988, pp. 305–312.
doi:10.2514/3.20312
- [44] Wachter, A., and Biegler, L. T., “On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming,” *Mathematical Programming*, Vol. 106, No. 1, 2006, pp. 25–57.
doi:10.1007/s10107-004-0559-y
- [45] Hartl, R., Sethi, S., and Vickson, R., “A Survey of the Maximum Principles for Optimal Control Problems with State Constraints,” *SIAM Review*, Vol. 37, No. 2, 1995, pp. 181–218.
doi:10.1137/1037043

This article has been cited by:

1. Boris Benedikter, Alessandro Zavoli, Guido Colasurdo, Simone Pizzurro, Enrico Cavallini. Convex Optimization of Launch Vehicle Ascent Trajectory with Heat-Flux and Splash-Down Constraints. *Journal of Spacecraft and Rockets*, ahead of print1-16. [[Abstract](#)] [[Full Text](#)] [[PDF](#)] [[PDF Plus](#)]
2. Boris Benedikter, Alessandro Zavoli, Zhenbo Wang, Simone Pizzurro, Enrico Cavallini. Covariance Control for Stochastic Low-Thrust Trajectory Optimization . [[Abstract](#)] [[PDF](#)] [[PDF Plus](#)]
3. Mohamed El Fatini, Driss Bouggar, Idriss Sekkak, Aziz Laaribi. 2022. Stochastic near-optimal control for drug therapy in a random viral model with cellular immune response. *Stochastic Analysis and Applications* **40**:1, 20-44. [[Crossref](#)]
4. Cong Wang, Zhengyu Song, Guoxing Shi, Qinghai Gong. Trajectory Planning for Landing Phase of Reusable Rocket with High Thrust-to-Weight Ratio 5311-5323. [[Crossref](#)]
5. Runqiu Yang, Xinfu Liu. Gravity-Turn-Based Precise Landing Guidance for Reusable Rockets 3423-3434. [[Crossref](#)]
6. Zhengyu Song, Cong Wang. 2021. Powered soft landing guidance method for launchers with non-cluster configured engines. *Acta Astronautica* **189**, 379-390. [[Crossref](#)]
7. Zichen Zhao, Haibin Shang, Yue Dong, Haoyu Wang. 2021. Multi-phase trajectory optimization of aerospace vehicle using sequential penalized convex relaxation. *Aerospace Science and Technology* **119**, 107175. [[Crossref](#)]
8. Jiateng Long, Shengying Zhu, Zixuan Liang, Chenglong Pan. 2021. Rapid Generation of the Fuel-Optimal Trajectory for Landing on Irregularly Shaped Asteroids. *IEEE Transactions on Aerospace and Electronic Systems* **57**:6, 4390-4400. [[Crossref](#)]
9. Ki-Wook Jung, Chang-Hun Lee, Junseong Lee, Sunghyuck Im, Keejoo Lee, Marco Sagliano, David Seelbinder. An Instantaneous Impact Point Guidance for Rocket with Aerodynamics Control 1489-1495. [[Crossref](#)]
10. Guanghui Cheng, Wuxing Jing, Changsheng Gao. 2021. Recovery trajectory planning for the reusable launch vehicle. *Aerospace Science and Technology* **117**, 106965. [[Crossref](#)]
11. Rolando Cortés-Martínez, Krishna D. Kumar, H. Rodríguez-Cortés. 2021. Precise power descent control of a lunar lander using a single thruster. *Acta Astronautica* **186**, 473-485. [[Crossref](#)]
12. Marco Sagliano, Erwin Mooij. 2021. Optimal drag-energy entry guidance via pseudospectral convex optimization. *Aerospace Science and Technology* **106**946. [[Crossref](#)]
13. Jian Zhao, Xiangyue He, Haiyang Li, Lin Lu. 2021. An adaptive optimization algorithm based on clustering analysis for return multi-flight-phase of VTVL reusable launch vehicle. *Acta Astronautica* **183**, 112-125. [[Crossref](#)]
14. Boris Benedikter, Alessandro Zavoli, Guido Colasurdo, Simone Pizzurro, Enrico Cavallini. 2021. Convex Approach to Three-Dimensional Launch Vehicle Ascent Trajectory Optimization. *Journal of Guidance, Control, and Dynamics* **44**:6, 1116-1131. [[Abstract](#)] [[Full Text](#)] [[PDF](#)] [[PDF Plus](#)]
15. Sixiong You, Changhuang Wan, Ran Dai, Jeremy R. Rea. 2021. Learning-Based Onboard Guidance for Fuel-Optimal Powered Descent. *Journal of Guidance, Control, and Dynamics* **44**:3, 601-613. [[Citation](#)] [[Full Text](#)] [[PDF](#)] [[PDF Plus](#)]
16. Ping Lu. 2021. Convex-Concave Decomposition of Nonlinear Equality Constraints in Optimal Control. *Journal of Guidance, Control, and Dynamics* **44**:1, 4-14. [[Abstract](#)] [[Full Text](#)] [[PDF](#)] [[PDF Plus](#)]
17. Danylo Malyuta, Yue Yu, Purnanand Elango, Behçet Açıkmeşe. 2021. Advances in trajectory optimization for space vehicle control. *Annual Reviews in Control* **52**, 282-315. [[Crossref](#)]
18. Jinbo Wang, Huixu Li, Hongbo Chen. 2020. An Iterative Convex Programming Method for Rocket Landing Trajectory Optimization. *The Journal of the Astronautical Sciences* **67**:4, 1553-1574. [[Crossref](#)]
19. Boris Benedikter, Alessandro Zavoli, Guido Colasurdo, Simone Pizzurro, Enrico Cavallini. Autonomous Upper Stage Guidance Using Convex Optimization and Model Predictive Control . [[Abstract](#)] [[PDF](#)] [[PDF Plus](#)]
20. Jian Zhao, Haiyang Li, Xiangyue He, Yuechen Huang, Jianghui Liu. 2020. Uncertainty Analysis for Return Trajectory of Vertical Takeoff and Vertical Landing Reusable Launch Vehicle. *Mathematical Problems in Engineering* **2020**, 1-18. [[Crossref](#)]
21. Runqiu Yang, Xinfu Liu. 2020. Fuel-optimal powered descent guidance with free final-time and path constraints. *Acta Astronautica* **172**, 70-81. [[Crossref](#)]
22. Ze An, FenFen Xiong, Chao Li. A Trajectory Tracking Method Using Convex Optimization 3281-3287. [[Crossref](#)]

23. Zheng-yu Song, Cong Wang, Stephan Theil, David Seelbinder, Marco Sagliano, Xin-fu Liu, Zhi-jiang Shao. 2020. Survey of autonomous guidance methods for powered planetary landing. *Frontiers of Information Technology & Electronic Engineering* 21:5, 652-674. [[Crossref](#)]
24. Rebecca Foust, Soon-Jo Chung, Fred Y. Hadaegh. 2020. Optimal Guidance and Control with Nonlinear Dynamics Using Sequential Convex Programming. *Journal of Guidance, Control, and Dynamics* 43:4, 633-644. [[Abstract](#)] [[Full Text](#)] [[PDF](#)] [[PDF Plus](#)]
25. Beldon Lin, Michele Carpenter, Olivier de Weck. Simultaneous Vehicle and Trajectory Design using Convex Optimization . [[Abstract](#)] [[PDF](#)] [[PDF Plus](#)]
26. Gaurav Misra, Xiaoli Bai. Iteratively feasible optimal spacecraft guidance with non-convex path constraints using convex optimization . [[Abstract](#)] [[PDF](#)] [[PDF Plus](#)]
27. Lei Xie, Hongbo Zhang, Xiang Zhou, Guojian Tang. 2020. A Convex Programming Method for Rocket Powered Landing With Angle of Attack Constraint. *IEEE Access* 8, 100485-100496. [[Crossref](#)]
28. Guangtong Xu, Teng Long, Zhu Wang, Yan Cao. 2019. Matrix Structure Driven Interior Point Method for Quadrotor Real-Time Trajectory Planning. *IEEE Access* 7, 90941-90953. [[Crossref](#)]