Constrained Trajectory Optimization for Planetary Entry via Sequential Convex Programming

Article *in* Journal of Guidance Control and Dynamics · July 2017

DOI: 10.2514/1.G002150

2 authors, including:

Zhenbo Wang
University of Tennessee
13 PUBLICATIONS **60** CITATIONS

Constrained Trajectory Optimization for Planetary Entry via Sequential Convex Programming

Zhenbo Wang* and Michael J. Grant†
Purdue University, West Lafayette, Indiana 47907-2045

DOI: 10.2514/1.G002150

In this paper, the highly nonlinear planetary-entry optimal control problem is formulated as a sequence of convex problems to facilitate rapid solution. The nonconvex control constraint is avoided by introducing a new state variable to the original three-dimensional equations of motion. The nonconvex objective function and path constraints are convexified by first-order Taylor-series expansions, and the nonconvex terms in the dynamics are approximated by successive linearizations. A successive solution procedure is developed to find an approximated solution to the original problem, and its convergence is discussed. In each iteration, a convex optimization problem is solved by the state-of-the-art interior-point method with deterministic convergence properties. Finally, the proposed method is verified and compared to a general-purpose optimal control solver by numerical solutions of minimum terminal-velocity and minimum heat-load entry problems. The sequential method converges to accurate solutions with faster speed than the general-purpose solver using MATLAB on a desktop computer with a 64-bit operating system and an Intel Xeon E3-1225 V2 3.2 GHz processor, which demonstrates its potential real-time application for computational guidance.

I. Introduction

THIS research was mainly motivated by the requirement of solution methods with guaranteed convergence and fast computational speed to enable onboard entry guidance applications. Entry trajectory optimization problems have been generally solved as nonlinear optimal control problems, and most of these approaches find the optimal solution by solving general nonlinear programming (NLP) problems [1]. However, these methods have undesirable properties. First, there is no known bound on the solution time. Second, there exists no guarantee on the convergence to an optimal or even a feasible solution in a predetermined number of iterations. Finally, good initial guesses are often required for nonlinear optimization algorithms, and in many cases, these guesses must be supplied by a user. Consequently, such approaches may not be appropriate for onboard applications, and it is essential to exploit more effective structures of the problem to design algorithms with guaranteed convergence to the optimum with deterministic convergence criteria.

If the problem is formulated in a convex optimization framework, specifically formulated as a subclass of a convex optimization problem, such as linear programming, quadratic programming (QP), second-order cone programming (SOCP), or semidefinite programming (SDP), the problem will have low complexity and can be solved in polynomial time [2–4]. There exist algorithms, such as interior-point methods [5], which compute an optimal solution with deterministic stopping criteria and with a prescribed level of accuracy. As such, the optimum can be computed to any given accuracy with a deterministic upper bound on the number of iterations needed for convergence. In addition, primal–dual interior-point algorithms employ a self-dual embedding technique that allows the algorithm to start from a self-generated feasible point without the need for any user-supplied initial guesses [6]. All of these traits offer the kind of advantages not matched by other direct or indirect optimization methods. As such, convex optimization approaches are very promising for onboard applications.

One of the earliest applications of convex optimization for nonlinear optimal control problems in aerospace is the study of powered descent guidance for Mars pinpoint landing [7–10]. The powered-descent guidance problem was firstly formulated as a propellant-optimal trajectory optimization problem with state and control constraints. The primary nonconvex constraint in this problem is the thrust magnitude, which has a lower nonzero bound that defines a nonconvex feasible region in the control space. A slack variable was introduced to relax this nonconvex constraint and approximate the relaxed optimal control problem as a finite-dimensional convex optimization problem via discretization, which leads to an SOCP problem. Based on the convexification results, a real-time interior-point-method algorithm was developed in [11] to increase the computational speed. By using this customized solver, the fuel-optimal powered-descent guidance was demonstrated on eight flight tests using a vertical-takeoff/vertical-landing rocket [12]. Similar methods are also used in trajectory optimization for rendezvous and proximity operations [13–15]. First, the rendezvous problem was formulated as a nonlinear optimal control problem with various state and control constraints on interior points and terminal conditions. Then, the original problem was approximated by a successive solution process, in which the solutions of a sequence of constrained SOCP subproblems with linear, time-varying dynamics were sought.

Trajectory optimization for hypersonic cases is potentially more complicated than these problems because of the existence of highly nonlinear dynamics, aerodynamic forces, and path constraints. Recently, the sequential SOCP method for rendezvous was extended to address entry trajectory optimization [16–18]. First, the equations of motion were reformulated with respect to energy, which simplifies the problem formulation by replacing the differential equation of velocity with an approximated formula. Then, the original highly constrained entry trajectory optimization problem was formulated in a fashion suitable to be solved by SOCP with a combination of successive linearization and relaxation techniques. Numerical simulations and rigorous analysis were provided to support the effectiveness and soundness of this approach. In addition, similar methods have been applied to address the closed-loop optimization of guidance for constrained impact and fuel-optimal rocket-landing problems [19,20].

Motivated by the aforementioned research work, an alternative approach is proposed in this paper to solve entry trajectory optimization problems using the sequential convex programming (SCP) method. The focus of this investigation is on gliding entry trajectory optimization problems. Different from the energy

Presented as Paper 2016-3241 at the AIAA Atmospheric Flight Mechanics Conference, Washington, D.C., 13–17 June 2016.; received 6 April 2016; revision received 3 March 2017; accepted for publication 6 May 2017; published online 13 July 2017.

*Ph.D. Candidate, School of Aeronautics and Astronautics; wang2351@purdue.edu. Student Member AIAA.

†Assistant Professor, School of Aeronautics and Astronautics; mjgrant@purdue.edu. Senior Member AIAA.

approach in [16], the original system dynamics is used. By introducing new variables, the control is decoupled from the states in the new dynamic system. All common path constraints on heat rate, dynamic pressure, and load factor are included via successive linearization, along with a trust-region constraint. The terminal constraints and control constraint can be satisfied within the SCP framework, and good performance can be achieved as well without the usage of the relaxation technique.

II. Problem Formulation

A. Entry-Trajectory-Optimization Problem

The energy-based equations of motion for entry vehicles have been applied and improved over the years, especially for entry guidance design of the space shuttle based on drag acceleration [21]. In this paper, however, the original equations of motion are considered. The dimensionless equations of motion of a three-dimensional (3-D) unpowered flight for an entry vehicle over a spherical, rotating Earth can be expressed in the wind-relative frame as follows [22]:

$$\dot{r} = V\sin\gamma \tag{1}$$

$$\dot{\theta} = V\cos\gamma\sin\psi/(r\cos\phi) \tag{2}$$

$$\dot{\phi} = V\cos\gamma\cos\psi/r \tag{3}$$

$$\dot{V} = -D - \sin\gamma/r^2 + \Omega^2 r\cos\phi(\sin\gamma\cos\phi - \cos\gamma\sin\phi\cos\psi) \tag{4}$$

$$\dot{\gamma} = L\cos\sigma/V + (V^2 - 1/r)\cos\gamma/(Vr) + 2\Omega\cos\phi\sin\psi + \Omega^2 r\cos\phi(\cos\gamma\cos\phi + \sin\gamma\sin\phi\cos\psi)/V \tag{5}$$

$$\dot{\psi} = L\sin\sigma/(V\cos\gamma) + V\cos\gamma\sin\psi\tan\phi/r - 2\Omega(\tan\gamma\cos\psi\cos\phi - \sin\phi) + \Omega^2 r\sin\phi\cos\phi\sin\psi/(V\cos\gamma) \tag{6}$$

in which r is the radial distance from Earth's center to the vehicle, which is normalized by the radius of Earth $R_0 = 6378$ km. The variables θ and ϕ are longitude and latitude, respectively. The Earth-relative velocity V is normalized by $\sqrt{R_0 g_0}$, in which $g_0 = 9.81$ m/s^2. The flight-path angle is denoted as γ, and ψ is the heading angle of the velocity vector, measured clockwise in the local horizontal plane from the north. The dimensionless lift and drag accelerations, L and D, are normalized by g_0, and the simplified functions are shown next. The dimensionless constant Ω is the Earth self-rotation rate normalized by $\sqrt{g_0/R_0}$. The differentiation of the equations in Eqs. (1–6) is with respect to dimensionless time normalized by $\sqrt{R_0/g_0}$.

$$L = R_0\rho V^2 A_{\text{ref}} C_L/(2m)$$
$$D = R_0\rho V^2 A_{\text{ref}} C_D/(2m)$$

in which m is the dimensional mass; A_{ref} is the dimensional reference area of the vehicle; and $\rho = \rho_0 e^{-h/h_s}$ is the dimensional atmospheric density, which is a function of altitude h. The aerodynamic lift and drag coefficients are denoted as C_L and C_D, respectively, which are given in tabulated data as functions of Mach number and angle of attack α. In this paper, the profile of α is prespecified as a function of Mach number based on the considerations of thermal protection system and maneuverability requirements. Consequently, the bank angle σ is assumed to be the only control variable for entry trajectory optimization.

With the entry equations of motion in Eqs. (1–6), a typical entry trajectory optimization problem is considered and defined as a nonlinear optimal control problem as follows.

Problem 0:

$$\min_{\boldsymbol{x},\sigma} \; J = \varphi[\boldsymbol{x}(t_f)] + \int_{t_0}^{t_f} g(\boldsymbol{x},\sigma)\,\mathrm{d}t \tag{7}$$

$$\text{subject to} \quad \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x},\sigma) \tag{8}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f \tag{9}$$

$$\boldsymbol{x} \in [\boldsymbol{x}_{\min}, \boldsymbol{x}_{\max}] \tag{10}$$

$$\sigma_{\min} \le \sigma \le \sigma_{\max} \tag{11}$$

$$\dot{Q} = f_1(r,V) = k_Q\sqrt{\rho}V^{3.15} \le \dot{Q}_{\max} \tag{12}$$

$$q = f_2(r,V) = 0.5\rho V^2 \le q_{\max} \tag{13}$$

$$n = f_3(r,V) = \sqrt{L^2 + D^2} \le n_{\max} \tag{14}$$

in which $\boldsymbol{x} = [r;\theta;\phi;V;\gamma;\psi]$ is a six-dimensional state vector, $\boldsymbol{x}_0$ is the initial state, $\boldsymbol{x}_f$ is the desired terminal state, and $\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max}$ are the lower and upper bounds of the states, respectively. The minimum and maximum values of bank angle are denoted by $\sigma_{\min}$ and $\sigma_{\max}$, respectively. The variables $\dot{Q}_{\max}$, $q_{\max}$, and $n_{\max}$ are the dimensionless maximum values of heat rate $\dot{Q}$, dynamic pressure q, and normal load n.

Instead of propagating Eq. (4), an algebraic equation, $V = \sqrt{2(1/r - e)}$, was used in [16] to compute the velocity V. In this paper, however, the original equations of motion in Eqs. (1–6) are used for the entry trajectory optimization problem, and the velocity is calculated from the propagation of the original velocity differential equation (4).

B. Choice of New Control

Equations (1–6) are highly nonlinear in both state variables and control, σ. When applying successive linearization to the dynamics, high-frequency jitters will emerge. One reason for these jitters is the coupling of states and control in the dynamics. A similar phenomenon has also been observed when the problem is solved by NLP solvers, such as the sequential QP method [23]. To address this problem, a pair of new controls was introduced in [16] as follows:

$$u_1 = \cos\sigma, \qquad u_2 = \sin\sigma$$

and the energy-based system dynamics would be in the following form:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + B(\boldsymbol{x})\boldsymbol{u} \tag{15}$$

in which $\boldsymbol{x} = [r;\theta;\phi;\gamma;\psi]$ is the state vector, and $\boldsymbol{u} = [u_1;u_2]$ is the control vector, from which the following constraint must be satisfied:

$$u_1^2 + u_2^2 = 1 \tag{16}$$

Equation (16) is a quadratic nonconvex equality constraint and cannot be directly addressed by convex programming. It was relaxed to a second-order cone constraint $u_1^2 + u_2^2 \le 1$ in [16], and a large amount of work was done to prove that the relaxed inequality second-order cone constraint remains active at the optimal solution. Adding further complication, a regularization term is required for the proof.

The control matrix of energy-based dynamics (15) is as follows:

$$B(\boldsymbol{x}) = \begin{bmatrix} 0_{3\times1} & 0_{3\times1} \\ L/(DV^2) & 0 \\ 0 & L/(DV^2\cos\gamma) \end{bmatrix}$$

Note that the control $\boldsymbol{u}$ is coupled with the states, including flight-path angle through $\cos\gamma$ in the B matrix. To decouple the control from the states, a small flight-path angle was assumed, and the approximated velocity formula was used in [16].

There exists a simpler approach to address the coupling issue and nonconvex control constraint, and obtain more accurate solutions with the same rate of convergence. To this end, we define the bank-angle rate $\dot{\sigma}$ as the new control input, and add an additional state equation to the 3-D equations of motion:

$$\dot{\sigma} = u \tag{17}$$

in which u is the scalar control normalized by $\sqrt{g_0/R_0}$ and must be selected from some control domain $U \in \mathbb{R}$, which could be convex. By adding the rate dynamics (17) to the original equations of motion in Eqs. (1–6), the augmented equations of motion take the form:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + Bu + \boldsymbol{f}_\Omega(\boldsymbol{x}) \tag{18}$$

in which the state vector is given $\boldsymbol{x} = [r;\theta;\phi;V;\gamma;\psi;\sigma]$ and a new control variable, u. The column vectors $\boldsymbol{f}(\boldsymbol{x}) \in \mathbb{R}^7$ and $B \in \mathbb{R}^7$ are shown next. Because the effect of Earth self-rotation is very small, the Earth-rotation-dependent terms are separated out and included in the column vector, $\boldsymbol{f}_\Omega(\boldsymbol{x}) \in \mathbb{R}^7$, which can be obtained from Eqs. (1–6) and (17).

$$\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} V\sin\gamma \\ V\cos\gamma\sin\psi/(r\cos\phi) \\ V\cos\gamma\cos\psi/r \\ -D - \sin\gamma/r^2 \\ L\cos\sigma/V + (V^2 - 1/r)\cos\gamma/(Vr) \\ L\sin\sigma/(V\cos\gamma) + V\cos\gamma\sin\psi\tan\phi/r \\ 0 \end{bmatrix} \tag{19}$$

$$B = [\,0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1\,]^T \tag{20}$$

The control matrix B in Eq. (20) is constant and independent of the states, $\boldsymbol{x}$. The original control, σ, is still coupled with the states, but the new control, u, is naturally decoupled from the states in the new model, which is significant to eliminate the high-frequency jitters due to the successive linearization of dynamics with respect to σ and to the convergence of the sequential method developed in the following section. As such, the nonconvex control constraint encountered in [16] will be avoided. The smoothness of the bank angle σ can be enhanced by enforcing a constraint on the new control, u. Furthermore, numerical simulations will show that no regularization terms are needed to ensure the satisfaction of control constraints and the convergence of the sequential method. Thus, the new approach developed in this paper is potentially more accurate for entry trajectory optimization.

C. Reformulated Optimal Control Problem

With the original equations of motion in Eq. (8) replaced by the augmented dynamics in Eqs. (18–20), problem 0 can be reformulated into a new optimal control problem as follows. Because the bank angle σ is treated as a new state variable, the new control, u, is not included in the objective functional.

Problem 1:

$$\min_{\boldsymbol{x},u} \; J = \varphi[\boldsymbol{x}(t_f)] + \int_{t_0}^{t_f} g(\boldsymbol{x})\,\mathrm{d}t \tag{21}$$

$$\text{subject to} \quad \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + Bu + \boldsymbol{f}_\Omega(\boldsymbol{x}) \tag{22}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f \tag{23}$$

$$\boldsymbol{x} \in [\boldsymbol{x}_{\min}, \boldsymbol{x}_{\max}] \tag{24}$$

$$|u| \le u_{\max} \tag{25}$$

$$\dot{Q} = f_1(r,V) = k_Q\sqrt{\rho}V^{3.15} \le \dot{Q}_{\max} \tag{26}$$

$$q = f_2(r,V) = 0.5\rho V^2 \le q_{\max} \tag{27}$$

$$n = f_3(r,V) = \sqrt{L^2 + D^2} \le n_{\max} \tag{28}$$

in which Eq. (25) is the constraint on the new control, and $u_{\max}$ is the maximum bank-angle rate.

Lemma 1: If there exists an optimal solution for problem 1 given by $[\boldsymbol{x}^*,u^*] = [r^*,\theta^*,\phi^*,V^*,\gamma^*,\psi^*,\sigma^*,u^*]$, then $[r^*,\theta^*,\phi^*,V^*,\gamma^*,\psi^*,\sigma^*]$ is a feasible solution of problem 0.

Proof: Problem 1 is obtained by introducing a new state variable and a new control variable to the original system dynamics in Eqs. (1–6). The original control constraint on σ becomes a boundary condition on the new state σ, which is included in Eq. (24). An additional constraint in Eq. (25) is introduced as a new control constraint on the bank-angle rate. Thus, the set of feasible controls of problem 1 is strictly contained in the set of feasible controls of problem 0. Therefore, because problem 0 is merely a relaxation of problem 1, an optimal solution of problem 1 satisfies all the constraints of problem 0 and clearly defines a feasible solution of problem 0. However, a feasible solution of problem 0 does not necessarily define a feasible solution of problem 1. In addition, the equivalence of the dynamics with and without $\dot{\sigma} = u$ has also been demonstrated by numerical simulations.

Remark 1: Lemma 1 presents a relationship between the solutions of problem 0 and problem 1. It implies that the nonconvex constraint shown in Eq. (16) in the energy approach is avoided by introducing an additional state variable in Eq. (17) and adding a new control constraint in Eq. (25) on the bank-angle rate in problem 1. Consequently, a feasible solution to problem 0 can be obtained by solving for the optimal solution of problem 1 with convex control constraints, as well as decoupled control and states. However, the nonconvex constraint, $|\sigma| \ge \sigma_{\min}$, cannot be handled by the proposed method, even if it is beneficial to impose such a constraint and gain certain amount of margin in the resulting reference trajectory. This is a limitation of the method proposed in this paper.

III. Convexification

Problem 1 is a highly nonlinear optimal control problem with nonlinear dynamics and path constraints, which adds complexity when compared to the Mars pinpoint landing problem and rendezvous problem. In many cases, the objective function is nonlinear as well. Fortunately, the boundary conditions in Eq. (23) and state constraints in Eq. (24), as well as the control constraint in Eq. (25), are all linear, convex constraints. However, the nonlinear dynamics, nonlinear objective, and nonlinear path constraints must be converted to tractable formulations for convex programming. A successive small-disturbance-based linearization method is used to approximate problem 1 by a sequence of convex optimization problems, specifically to a sequence of SOCP problems. Numerical simulations in Sec. V demonstrate that the successive linearization approximation of problem 1 is very accurate.

A. Successive Linear Approximations

First, a general objective is considered, and many objectives, such as minimum heat load, fall into the integral form shown in Eq. (21). The integrand can be linearized by a first-order Taylor-series expansion about a given state history $\boldsymbol{x}_*(t)$, as shown in Eq. (29):

$$g(\boldsymbol{x}) \approx g(\boldsymbol{x}_*) + \left.\frac{\partial g(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{x=x_*} (\boldsymbol{x} - \boldsymbol{x}_*) \tag{29}$$

In addition, the terminal cost in Eq. (21) could also be nonconvex. It can be approximated by a linear approximation with respect to a reference terminal point $\boldsymbol{x}_*(t_f)$ shown in Eq. (30). As such, after discretization, the objective will be a linear combination of the states at all nodes.

$$\varphi[\boldsymbol{x}(t_f)] \approx \varphi[\boldsymbol{x}_*(t_f)] + \left.\frac{\partial \varphi[\boldsymbol{x}(t_f)]}{\varphi \boldsymbol{x}(t_f)}\right|_{x(t_f)=x_*(t_f)} [\boldsymbol{x}(t_f) - \boldsymbol{x}_*(t_f)] \tag{30}$$

As mentioned before, the control, u, has been decoupled from the states, $\boldsymbol{x}$, by introducing an additional state variable. Consequently, the nonlinear term in Eq. (22) is $\boldsymbol{f}(\boldsymbol{x})$, given by Eq. (19), and could be linearized about the fixed state history, $\boldsymbol{x}_*(t)$, as well. Because the magnitudes of the terms in $\boldsymbol{f}_\Omega(\boldsymbol{x})$ are very small, we will approximate $\boldsymbol{f}_\Omega(\boldsymbol{x}) \approx \boldsymbol{f}_\Omega(\boldsymbol{x}_*)$. The linearization of the dynamics is as follows:

$$\dot{\boldsymbol{x}} \approx \boldsymbol{f}(\boldsymbol{x}_*) + A(\boldsymbol{x}_*)(\boldsymbol{x} - \boldsymbol{x}_*) + B u + \boldsymbol{f}_\Omega(\boldsymbol{x}_*) \tag{31}$$

in which

$$A(\boldsymbol{x}_*) = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_*} = \begin{bmatrix} 0 & 0 & 0 & \sin\gamma & V\cos\gamma & 0 & 0 \\ a_{21} & 0 & a_{23} & a_{24} & a_{25} & a_{26} & 0 \\ a_{31} & 0 & 0 & a_{34} & a_{35} & a_{36} & 0 \\ a_{41} & 0 & 0 & a_{44} & a_{45} & 0 & 0 \\ a_{51} & 0 & 0 & a_{54} & a_{55} & 0 & a_{57} \\ a_{61} & 0 & a_{63} & a_{64} & a_{65} & a_{66} & a_{67} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{\boldsymbol{x}=\boldsymbol{x}_*}$$

With the approximation of the nonlinear dynamics in Eq. (22) by Eq. (31), the nonlinear objective functional in Eq. (21) by Eqs. (29) and (30), and the nonlinear path constraints in Eqs. (26–28) by Eq. (32), problem 1 can be converted into problem 2, which is a continuous-time optimal control problem with a convex objective, as well as convex state and control constraints.

Problem 2:

$$\min_{\boldsymbol{x},u} J = \left\{\varphi\left[\boldsymbol{x}_*(t_f)\right] + \partial\varphi\left[\boldsymbol{x}_*(t_f)\right]\left[\boldsymbol{x}(t_f) - \boldsymbol{x}_*(t_f)\right]\right\} + \int_{t_0}^{t_f} [\partial g(\boldsymbol{x}_*)\boldsymbol{x} + g(\boldsymbol{x}_*) - \partial g(\boldsymbol{x}_*)\boldsymbol{x}_*]\,\mathrm{d}t \tag{33}$$

$$\text{subject to } \dot{\boldsymbol{x}} = A(\boldsymbol{x}_*)\boldsymbol{x} + Bu + \boldsymbol{f}(\boldsymbol{x}_*) - A(\boldsymbol{x}_*)\boldsymbol{x}_* + \boldsymbol{f}_\Omega(\boldsymbol{x}_*) \tag{34}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f \tag{35}$$

$$\boldsymbol{x} \in [\boldsymbol{x}_{\min}, \boldsymbol{x}_{\max}] \tag{36}$$

$$|u| \le u_{\max} \tag{37}$$

$$f_i(r_*, V_*) + f_i'(r_*, V_*)[r - r_*; V - V_*] \le f_{i,\max}, \quad i = 1, 2, 3 \tag{38}$$

$$\|\boldsymbol{x} - \boldsymbol{x}_*\| \le \delta \tag{39}$$

$$a_{21} = -\frac{V\cos\gamma\sin\psi}{r^2\cos\phi}, \quad a_{23} = \frac{V\cos\gamma\sin\psi\sin\phi}{r\cos^2\phi}, \quad a_{24} = \frac{\cos\gamma\sin\psi}{r\cos\phi}, \quad a_{25} = -\frac{V\sin\gamma\sin\psi}{r\cos\phi}$$
$$a_{26} = \frac{V\cos\gamma\cos\psi}{r\cos\phi}, \quad a_{31} = -\frac{V\cos\gamma\cos\psi}{r^2}, \quad a_{34} = \frac{\cos\gamma\cos\psi}{r}, \quad a_{35} = -\frac{V\sin\gamma\cos\psi}{r}$$
$$a_{36} = -\frac{V\cos\gamma\sin\psi}{r}, \quad a_{41} = -D_r + \frac{2\sin\gamma}{r^3}, \quad a_{44} = -D_V, \quad a_{45} = -\frac{\cos\gamma}{r^2}$$
$$a_{51} = \frac{\cos\sigma}{V}L_r - \frac{V\cos\gamma}{r^2} + \frac{2\cos\gamma}{r^3V}, \quad a_{54} = \frac{\cos\sigma}{V}L_V - \frac{L\cos\sigma}{V^2} + \frac{\cos}{r} + \frac{\cos\gamma}{r^2V^2}$$
$$a_{55} = \left(\frac{1}{r^2V} - \frac{V}{r}\right)\sin\gamma, \quad a_{57} = -\frac{L\cos\sigma}{V}, \quad a_{61} = \frac{\sin\sigma}{V\cos\gamma}L_r - \frac{V\cos\gamma\sin\psi\tan\phi}{r^2}$$
$$a_{63} = \frac{V\cos\gamma\sin\psi}{r\cos^2\phi}, \quad a_{64} = \frac{\sin\sigma}{V\cos\gamma}L_V - \frac{L\sin\sigma}{V^2\cos\gamma} + \frac{\cos\gamma\sin\psi\tan\phi}{r}$$
$$a_{65} = \frac{L\sin\sigma\sin\gamma}{V\cos^2\gamma} - \frac{V\sin\gamma\sin\psi\tan\phi}{r}, \quad a_{66} = \frac{V\cos\gamma\cos\psi\tan\phi}{r}, \quad a_{67} = \frac{L\cos\sigma}{V\cos\gamma}$$
$$D_r = \frac{\partial D}{\partial r} = -\frac{R_0^2\rho V^2 A_{\text{ref}} C_D}{2\,mh_s} = -\frac{R_0}{h_s}D, \quad D_V = \frac{\partial D}{\partial V} = \frac{R_0\rho V A_{\text{ref}} C_D}{m}$$
$$L_r = -\frac{R_0}{h_s}L, \quad L_V = \frac{R_0\rho V A_{\text{ref}} C_L}{m}$$

In addition, the nonlinear path constraints in Eqs. (26–28) are functions of r and V, and can be linearized with respect to r and V about a fixed state (r_*, V_*). The linearized path constraints have the following form:

$$f_i(r, V) \approx f_i(r_*, V_*) + f_i'(r_*, V_*)[r - r_*; V - V_*], \qquad i = 1, 2, 3 \tag{32}$$

in which

$$f_1'(r_*, V_*) = \left[\frac{\partial f_1}{\partial r}, \frac{\partial f_1}{\partial V}\right] = \left[-0.5R_0k_Q(V\sqrt{R_0g_0})^{3.15}\sqrt{\rho}/h_s, 3.15k_Q(\sqrt{R_0g_0})^{3.15}V^{2.15}\sqrt{\rho}\right]_{(r_*,V_*)}$$
$$f_2'(r_*, V_*) = \left[\frac{\partial f_2}{\partial r}, \frac{\partial f_2}{\partial V}\right] = \left[-0.5R_0\rho(V\sqrt{R_0g_0})^2/h_s, (\sqrt{R_0g_0})^2\rho V\right]_{(r_*,V_*)}$$
$$f_3'(r_*, V_*) = \left[\frac{\partial f_3}{\partial r}, \frac{\partial f_3}{\partial V}\right] = \left[-0.5R_0^2A_{\text{ref}}\sqrt{C_L^2 + C_D^2}\rho V^2/(mh_s), R_0A_{\text{ref}}\sqrt{C_L^2 + C_D^2}\rho V/m\right]_{(r_*,V_*)}$$

Note that a trust-region constraint is enforced in Eq. (39), which is a second-order cone constraint. The variable δ defines the radius of the trust region, which is very important to ensure the convergence of successive linear approximations in Eqs. (33), (34), and (38).

B. Existence of an Optimal Solution

After discretization, problem 2 can be converted into an SOCP problem for a given $\boldsymbol{x}_*$, and problem 1 will be solved as a sequence of such SOCP problems in Sec. IV. Therefore, an important theoretical issue is the existence of an optimal solution to problem 2.

Assumption 1: Let Γ represent the set of all feasible state trajectories and control signals for problem 2 [i.e., $\boldsymbol{x}(\cdot), u(\cdot) \in \Gamma$], which implies that, for $t \in [t_0, t_f]$, $\boldsymbol{x}(t)$ and $u(t)$ define a feasible state trajectory and control signal for problem 2. For this problem, we assume that Γ is nonempty.

Theorem 1: Consider the following general optimal control problem [24]:

$$\min_{\boldsymbol{x},u} J[\boldsymbol{x}(\cdot), u(\cdot)] = \varphi[\boldsymbol{x}(t_f)] + \int_{t_0}^{t_f} g(\boldsymbol{x}, u)\,\mathrm{d}t$$

$$\text{subject to } \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}, u)$$

$$\boldsymbol{x}(t_0) \in \boldsymbol{X}_0, \quad \boldsymbol{x}(t_f) \in \boldsymbol{X}_f$$

$$\boldsymbol{x}(t) \in \boldsymbol{X}, \quad u(t) \in U, \quad \forall\, t \in [t_0, t_f]$$

Suppose that

1) There exists a compact set $\boldsymbol{K}$, such that, for all feasible trajectories $\boldsymbol{x}(\cdot)$, we have $[t, \boldsymbol{x}(t)] \in \boldsymbol{K}$ for all $t \in [t_0, t_f]$.

2) The set of all feasible $[\boldsymbol{x}(t_0), t_f, \boldsymbol{x}(t_f)] \in \boldsymbol{M}$ is closed, such that $\boldsymbol{x}(t_0) \in \boldsymbol{X}_0$ and $\boldsymbol{x}(t_f) \in \boldsymbol{X}_f$.

3) The control set U is compact.

4) Both g and $\boldsymbol{f}$ are continuous functions on $\boldsymbol{X} \times U$.

5) For each $[t, \boldsymbol{x}(t)] \in \boldsymbol{K}$, the set $\boldsymbol{Q}^+(t, \boldsymbol{x})$ defined as follows is convex:

$$\boldsymbol{Q}^+(t, \boldsymbol{x}) = \{(z_1, z_2)\colon z_1 \geq g(\boldsymbol{x}, u), z_2 = \boldsymbol{f}(\boldsymbol{x}, u), u \in U\}$$

Then, there exists a solution pair $[\boldsymbol{x}^*, u^*]$ in Γ, such that

$$J(\boldsymbol{x}^*, u^*) \leq J[\boldsymbol{x}(\cdot), u(\cdot)], \quad \forall [\boldsymbol{x}(\cdot), u(\cdot)] \in \Gamma$$

Additional descriptions of the preceding existence theorem and its proof can be found in [24]. This theorem was also used in [7] to prove the existence of an optimal solution for the relaxed optimal control problem associated with Mars powered descent guidance. For entry problems, however, all the conditions need to be verified based on the new problem formulation shown in problem 2 with decoupled control and linear approximations.

Proposition 1: Under assumption 1 (i.e., if the set of all feasible states and controls for problem 2 is nonempty), there exists an optimal solution to problem 2.

Proof: We will prove the existence of the optimal solution by showing that all conditions in Theorem 1 can be satisfied by problem 2.

1) For problem 2, the time of entry flight t_f considered in this paper is fixed, and both the position vector $[r; \theta; \phi]$ and velocity vector $[V; \gamma; \psi]$ are constrained in compact sets, as shown in Eq. (36). There are also lower and upper bounds on the new state σ. Thus, we have $[t, \boldsymbol{x}(t)] \in \boldsymbol{K}$ with $t \in [t_0, t_f]$ for all feasible state trajectories, and condition 1 is satisfied.

2) The satisfaction of condition 2 can be guaranteed as well, because both the initial condition, $\boldsymbol{x}(t_0) \in \boldsymbol{X}_0$, and terminal condition, $\boldsymbol{x}(t_f) \in \boldsymbol{X}_f$, are specified. Therefore, $[\boldsymbol{x}(t_0), t_f, \boldsymbol{x}(t_f)] \in \boldsymbol{M}$ is closed and bounded.

3) The control set U defined by Eq. (37) is also compact, which leads to the third condition.

4) Condition 4 is obviously satisfied, because both g and $\boldsymbol{f}$ are continuous functions on $\boldsymbol{X} \times U$ regardless if they are linearized or not.

5) The set $\boldsymbol{Q}^+(t, \boldsymbol{x})$ in condition 5 for problem 2 is as follows:

$$\boldsymbol{Q}^+(t, \boldsymbol{x}) = \Big\{(z_1, z_2)\colon z_1 \geq \partial g(\boldsymbol{x}_*)\boldsymbol{x} + g(\boldsymbol{x}_*) - \partial g(\boldsymbol{x}_*)\boldsymbol{x}_*$$
$$z_2 = A(\boldsymbol{x}_*)\boldsymbol{x} + Bu + \boldsymbol{f}(\boldsymbol{x}_*) - A(\boldsymbol{x}_*)\boldsymbol{x}_* + \boldsymbol{f}_\Omega(\boldsymbol{x}_*)$$
$$z_1 \in \mathbb{R}, z_2 \in \mathbb{R}^7, |u| \leq u_{\max}\Big\}$$

Assume that $A(\boldsymbol{x}_*)$ is invertible almost everywhere for $t \in [t_0, t_f]$; thus, $(z_1, z_2) \in \boldsymbol{Q}^+(t, \boldsymbol{x})$ if and only if the following conditions are satisfied:

$$z_1 \geq \partial g(\boldsymbol{x}_*)A^{-1}(\boldsymbol{x}_*)[z_2 - Bu - \boldsymbol{f}(\boldsymbol{x}_*) + A(\boldsymbol{x}_*)\boldsymbol{x}_* - \boldsymbol{f}_\Omega(\boldsymbol{x}_*)]$$
$$+ g(\boldsymbol{x}_*) - \partial g(\boldsymbol{x}_*)\boldsymbol{x}_*$$
$$z_2 = A(\boldsymbol{x}_*)\boldsymbol{x} + Bu + \boldsymbol{f}(\boldsymbol{x}_*) - A(\boldsymbol{x}_*)\boldsymbol{x}_* + \boldsymbol{f}_\Omega(\boldsymbol{x}_*)$$
$$\|B^T[z_2 - A(\boldsymbol{x}_*)\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{x}_*) + A(\boldsymbol{x}_*)\boldsymbol{x}_* - \boldsymbol{f}_\Omega(\boldsymbol{x}_*)]\| \leq u_{\max}$$

Because the preceding equality is linear and the inequalities are either linear or second-order cone equations, the set $\boldsymbol{Q}^+(t, \boldsymbol{x})$ for problem 2 is convex, and condition 5 is satisfied.

According to Theorem 1, we can draw the conclusion that problem 2 has an optimal solution when its feasible set is nonempty.

IV. SCP Method

The only difference between problem 1 and problem 2 are the approximations of the objective functional (21) by Eq. (33), dynamics (22) by Eq. (34), and path constraints (26–28) by Eq. (38), and so the solution to problem 2 is potentially a good approximation to the corresponding solution to problem 1 if the reference trajectory, $\boldsymbol{x}_*(t)$, for linearization is close enough to the real trajectory, $\boldsymbol{x}(t)$. In this section, we attempt to find the solution to problem 1 by solving a sequence of convex optimal control problems defined by problem 2.

A. Solution Procedure

Similar to the successive SOCP method developed for the rendezvous problem [13–16], an SCP method is devised in this section to approximate and solve problem 1. In this approach, a sequence of convex optimal control subproblems is formed using the state from the previous iteration. The process is described as follows:

1) Set $k = 0$. Initialize the states $r(t_0) = r_0$, $\theta(t_0) = \theta_0$, $\phi(t_0) = \phi_0$, $V(t_0) = V_0$, $\gamma(t_0) = \gamma_0$, $\psi(t_0) = \psi_0$, and $\sigma(t_0) = \sigma_0$. Propagate the equations of motion (18–20) with these initial conditions, and a specific constant control to provide an initial trajectory $\boldsymbol{x}^{(0)}$ for the solution procedure.

2) For $k \geq 1$, solve the following optimal control problem to find the solution pair $\{\boldsymbol{x}^{(k)}, u^{(k)}\}$. Given the state equations

$$\dot{\boldsymbol{x}} = A\left(\boldsymbol{x}^{(k-1)}\right)\boldsymbol{x} + Bu + \boldsymbol{f}\left(\boldsymbol{x}^{(k-1)}\right) - A\left(\boldsymbol{x}^{(k-1)}\right)\boldsymbol{x}^{(k-1)} + \boldsymbol{f}_\Omega\left(\boldsymbol{x}^{(k-1)}\right) \tag{40}$$

Minimize the objective functional:

$$\min_{\boldsymbol{x},u} J = \left\{\varphi\left[\boldsymbol{x}^{(k-1)}(t_f)\right] + \partial\varphi\left[\boldsymbol{x}^{(k-1)}(t_f)\right]\left[\boldsymbol{x}(t_f) - \boldsymbol{x}^{(k-1)}(t_f)\right]\right\}$$
$$+ \int_{t_0}^{t_f}\left[\partial g\left(\boldsymbol{x}^{(k-1)}\right)\boldsymbol{x} + g\left(\boldsymbol{x}^{(k-1)}\right) - \partial g\left(\boldsymbol{x}^{(k-1)}\right)\boldsymbol{x}^{(k-1)}\right]\mathrm{d}t \tag{41}$$

subject to

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f \tag{42}$$

$$\boldsymbol{x} \in [\boldsymbol{x}_{\min}, \boldsymbol{x}_{\max}] \tag{43}$$

$$|u| \leq u_{\max} \tag{44}$$

$$f_i\left(r^{(k-1)}, V^{(k-1)}\right) + f_i'\left(r^{(k-1)}, V^{(k-1)}\right)\left[r - r^{(k-1)}; V - V^{(k-1)}\right] \le f_{i,\max}, \quad i = 1,2,3 \tag{45}$$

$$\left\|\boldsymbol{x} - \boldsymbol{x}^{(k-1)}\right\| \le \delta \tag{46}$$

3) Check the convergence condition:

$$\sup_{t_0 \le t \le t_f} \left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right\| \le \varepsilon, \qquad k > 1 \tag{47}$$

in which ε is a prescribed tolerance value for convergence. If the preceding condition is satisfied, go to step 4; otherwise, set $k = k + 1$ and go to step 2.

4) The solution of the problem is found to be $\boldsymbol{x}^* = \boldsymbol{x}^{(k)}$ and $u^* = u^{(k)}$.

Remark 2: For each $k \ge 1$, a nonlinear optimal control problem is defined by Eqs. (40–46). Because only linear, time-varying dynamics, affine equality constraints, and second-order cone inequality constraints are included in these subproblems, they can be discretized into SOCP problems and solved efficiently by convex solvers, as described in the next subsection. The conventional linearization techniques are used to approximate problem 1 based on small perturbations. A trust-region constraint is introduced in Eq. (46) to ensure the convergence of the successive linearization method and improve the convergence performance of the solution procedure.

B. Discretization

The subproblems defined by Eqs. (40–46) are still infinite-dimensional optimal control problems. To find their numerical solutions, a trapezoidal discretization is applied to convert the infinite-dimensional optimization problems to finite-dimensional ones [13]. To do this, the time domain is discretized into N equal time intervals, and the constraints are imposed at $N + 1$ nodes. The step size is $\Delta t = (t_f - t_0)/N$, and the discretized nodes are denoted by $\{t_0, t_1, t_2, \ldots, t_{N-1}, t_N\}$ with $t_i = t_{i-1} + \Delta t$, $i = 1, 2, \ldots, N$. The corresponding state and control are discretized into the sequences $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N-1}, \boldsymbol{x}_N\}$ and $\{u_0, u_1, u_2, \ldots, u_{N-1}, u_N\}$. Then, the dynamics can be integrated numerically as follows:

$$\boldsymbol{x}_i = \boldsymbol{x}_{i-1} + \frac{\Delta t}{2}\left[\left(A_{i-1}^{k-1}\boldsymbol{x}_{i-1} + Bu_{i-1} + \boldsymbol{f}_{i-1}^{k-1} - A_{i-1}^{k-1}\boldsymbol{x}_{i-1}^{k-1} + \boldsymbol{f}_{\Omega,i-1}^{k-1}\right) + \left(A_i^{k-1}\boldsymbol{x}_i + Bu_i + \boldsymbol{f}_i^{k-1} - A_i^{k-1}\boldsymbol{x}_i^{k-1} + \boldsymbol{f}_{\Omega,i}^{k-1}\right)\right] \tag{48}$$

in which $\boldsymbol{x}_i^k = \boldsymbol{x}^k(t_i)$, $A_i^k = A(\boldsymbol{x}_i^k)$, B is constant, $\boldsymbol{f}_i^k = \boldsymbol{f}(\boldsymbol{x}_i^k)$, and $\boldsymbol{f}_{\Omega,i}^k = \boldsymbol{f}_\Omega(\boldsymbol{x}_i^k)$. After rearrangement, Eq. (48) can be written as

$$\left(\boldsymbol{I} - \frac{\Delta t}{2}A_i^{k-1}\right)\boldsymbol{x}_i - \left(\boldsymbol{I} + \frac{\Delta t}{2}A_{i-1}^{k-1}\right)\boldsymbol{x}_{i-1} - \frac{\Delta t}{2}Bu_i - \frac{\Delta t}{2}Bu_{i-1} = \frac{\Delta t}{2}\left(\boldsymbol{f}_{i-1}^{k-1} - A_{i-1}^{k-1}\boldsymbol{x}_{i-1}^{k-1} + \boldsymbol{f}_{\Omega,i-1}^{k-1} + \boldsymbol{f}_i^{k-1} - A_i^{k-1}\boldsymbol{x}_i^{k-1} + \boldsymbol{f}_{\Omega,i}^{k-1}\right) \tag{49}$$

If we choose $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N-1}, \boldsymbol{x}_N\}$ and $\{u_0, u_1, u_2, \ldots, u_{N-1}, u_N\}$ as the design variables, then the augmented optimization vector will be $\boldsymbol{y} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N, u_0, u_1, \ldots, u_N)$, and the state equation (40) is converted into linear equality constraints on $\boldsymbol{y}$ defined by Eq. (49). Note that the linearized objective functional in Eq. (41) can be discretized into a linear function of $\boldsymbol{y}$ as well. All the constraints defined in Eqs. (42–46) will be enforced at each node and become linear equality constraints and second-order cone inequality constraints. After discretization, each optimal control subproblem defined in Eqs. (40–46) is converted into a large-scale NLP problem as follows.

Problem 3:

$$\min_{\boldsymbol{y}} \boldsymbol{c}^T\boldsymbol{y} \tag{50}$$

$$\text{subject to } \|M_i\boldsymbol{y} + \boldsymbol{p}_i\| \le \boldsymbol{q}_i^T\boldsymbol{y} + r_i, \; i = 1, 2, \ldots, N_d \tag{51}$$

in which Eq. (51) is a general description of the discretized constraints. The parameters $\boldsymbol{c} \in \mathbb{R}^{n_y}$, $M_i \in \mathbb{R}^{n_i \times n_y}$, $\boldsymbol{p}_i \in \mathbb{R}^{n_i}$, $\boldsymbol{q}_i \in \mathbb{R}^{n_y}$, and $r_i \in \mathbb{R}$ are calculated based on the solution $\boldsymbol{x}^{(k-1)}$ from the previous iteration before solving $\boldsymbol{y}^{(k)} = \{\boldsymbol{x}^{(k)}, u^{(k)}\}$ in the current iteration. The variable n_y denotes the dimension of the optimization vector $\boldsymbol{y}$ after discretization, and N_d is the total number of the constraints in the discretized problem 3. Note that the linear inequality constraints are included in Eq. (51) because each linear constraint is a special case of the second-order cone constraint. Additionally, linear equality constraints can be expressed by two linear inequality constraints, which are also included in Eq. (51). Thus, the discretized problem is a nonlinear, but an SOCP, problem, and the discretization is accurate for a sufficiently large N.

Assumption 2: We assume that the existence of the optimal solution to problem 2 and each subproblem (which has been proved in the previous section) is guaranteed after the discretization process. Finally, we can solve a sequence of problem 3 to obtain an approximate numerical solution to problem 1.

C. Convergence Analysis

Because the existence of the optimal solution to each of the subproblems has been demonstrated in the proceeding section by proposition 1, then based on assumption 2, our attention turns to the convergence of the sequential solution procedure of problem 3 and the equivalence of the converged solution to the solution to problem 1. A complete proof of the convergence of the sequential method for entry trajectory optimization problems is still unreachable. However, some characteristics of the problem can be used to provide confidence of convergence in the context of preceding research in sequential methods [25–27].

1. Sequential Linear-Quadratic Approximations

The work in [26] is a close resemblance to the method in this paper, in which the convergence of successive approximations has been proved for nonlinear optimal control problems with pseudolinear systems and a standard quadratic cost functional. The nonlinear dynamics considered take the following form:

$$\dot{\boldsymbol{x}} = A(\boldsymbol{x})\boldsymbol{x} + B(\boldsymbol{x})\boldsymbol{u}, \qquad \boldsymbol{x}(t_0) = \boldsymbol{x}_0$$

together with a standard quadratic cost functional

$$J = \frac{1}{2}\boldsymbol{x}^T(t_f)F\boldsymbol{x}(t_f) + \frac{1}{2}\int_{t_0}^{t_f}\left[\boldsymbol{x}^TQ(\boldsymbol{x})\boldsymbol{x} + \boldsymbol{u}^TR(\boldsymbol{x})\boldsymbol{u}\right]\mathrm{d}t$$

in which $F \ge 0$, $Q(\boldsymbol{x}) \ge 0$, and $R(\boldsymbol{x}) > 0$ for all $\boldsymbol{x}$. The solution to this problem can be found by solving a sequence of time-varying linear-quadratic approximations. Because each subproblem has a quadratic performance index and linear, time-varying dynamics, the optimal control can be solved explicitly. Assume that $A(\boldsymbol{x})$ and $B(\boldsymbol{x})$ satisfy

1) $\mu[A(\boldsymbol{x})] = \lim_{h\to 0^+}\frac{\|\boldsymbol{I} - hA(\boldsymbol{x})\| - 1}{h} \le \mu_0 \quad \forall\, \boldsymbol{x} \in \mathbb{R}^n$
2) $\|A(\boldsymbol{x}_1) - A(\boldsymbol{x}_2)\| \le c_1\|\boldsymbol{x}_1 - \boldsymbol{x}_2\| \quad \forall\, \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^n$
3) $\|B(\boldsymbol{x}_1) - B(\boldsymbol{x}_2)\| \le c_2\|\boldsymbol{x}_1 - \boldsymbol{x}_2\| \quad \forall\, \boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^n$, and
4) $\|B(\boldsymbol{x})\| \le b_0 \quad \forall\, \boldsymbol{x} \in \mathbb{R}^n$

for some constants μ_0, c_1, c_2, and b_0. The logarithmic norm of the matrix is denoted by μ. Under the preceding boundary conditions and local Lipschitz continuity conditions, the convergence of the sequential approximation method for the previous linear-quadratic problem can be proved [26].

In our problem, matrix B in Eq. (20) is constant; thus, B is Lipschitz continuous, and conditions 3 and 4 are met. However, the objective functional in Eq. (21) is not generally in standard quadratic form, and

the entry dynamics in Eq. (22) do not naturally fall into the pseudolinear form. Additionally, there are various constraints in Eqs. (23–28) for our problem. As such, a complete theoretical proof of the successive convex method in this paper is much more difficult than for the linear-quadratic problem. Even if the convexified system in Eq. (34) is considered, the properties of matrix $A(\boldsymbol{x}_*)$ and its satisfaction of conditions 1 and 2 cannot be determined explicitly. However, some similar characteristics can be identified.

Consider the system in Eq. (22) subject to the constraint in Eq. (25). For a given control u, there exist appropriate constants $r_{\min}$, $\theta_{\min}$, $\phi_{\min}$, $V_{\min}$, $\gamma_{\min}$, $\psi_{\min}$, $\sigma_{\min}$, $r_{\max}$, $\theta_{\max}$, $\phi_{\max}$, $V_{\max}$, $\gamma_{\max}$, $\psi_{\max}$, $\sigma_{\max}$, such that, for any fixed t_f, the following conditions are satisfied for any admissible control:

$$r_{\min} \le \|r(t)\| \le r_{\max}, \theta_{\min} \le \|\theta(t)\| \le \theta_{\max}, \phi_{\min} \le \|\phi(t)\| \le \phi_{\max}, V_{\min} \le \|V(t)\| \le V_{\max}$$

$$\gamma_{\min} \le \|\gamma(t)\| \le \gamma_{\max}, \psi_{\min} \le \|\psi(t)\| \le \psi_{\max}, \sigma_{\min} \le \|\sigma(t)\| \le \sigma_{\max}, \quad \forall\ t \in [t_0, t_f]$$

The preceding conditions are equivalent to the state constraints defined in Eq. (24), based on which all the differential equations in Eq. (22) have maximum and minimum values on a compact set. Thus, the linearized system matrix $A(\boldsymbol{x}_*)$ is expected to be bounded and satisfy condition 1. In addition, $A(\boldsymbol{x}_*)$ is continuously differential on a closed interval, and so the Lipschitz continuous condition 2 is expected to be satisfied as well.

Remark 3: Because of the complex structure of $A(\boldsymbol{x}_*)$ shown in Eq. (31), explicit expressions for the logarithmic norm of $A(\boldsymbol{x}_*)$ and the left-hand side of the Lipschitz continuity condition 2 are hard to derive; however, the satisfaction of conditions 1 and 2 by the system matrices $A(\boldsymbol{x}^{(k-1)})$ of the sequential SOCP subproblems could be verified numerically for the entry trajectory optimization problems considered in this paper.

2. *Sequential SOCP Approximations*

Another sequential convergence study can be found in [15]. The nonlinear optimal control problems considered there have concave state inequality constraints and nonlinear terminal equality constraints, as shown next.

Problem 4:

$$\min_{\boldsymbol{x},u} J = \varphi[\boldsymbol{x}(t_f)] + \int_{t_0}^{t_f} g(\boldsymbol{x}, u, t)\,\mathrm{d}t \tag{52}$$

$$\text{subject to } \dot{\boldsymbol{x}} = A(t)\boldsymbol{x} + B(t)u \tag{53}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f \tag{54}$$

$$\|M_i\boldsymbol{y} + \boldsymbol{p}_i\| \le \boldsymbol{q}_i^T\boldsymbol{y} + r_i, \qquad i = 1, 2, \ldots, l \tag{55}$$

$$s_j(\boldsymbol{x}, t) \le 0, \qquad j = 1, 2, \ldots, m \tag{56}$$

in which a linear, time-varying system dynamics is defined. Matrices $A(t)$ and $B(t)$ are continuous, and $\boldsymbol{y} = (\boldsymbol{x}; u)$ is an augmented vector used to define all linear and second-order cone constraints shown in Eq. (55). The integrand $g(\boldsymbol{x}, u, t)$ in the performance index is assumed to be second-order cone representable, which can always be discretized into a linear cost function with second-order cone constraints. The only nonconvex part of problem 4 is the inequality constraints in Eq. (56), which is assumed to be concave. When the concave constraints are linearized about a reference point, problem 4 can be discretized into the form of problem 3. The solution of problem 4 can be obtained by solving a sequence of problem 3 [15].

Let $\boldsymbol{y}^{(k)}$ be a sequence of solutions obtained by solving problem 3 successively. Assume that the quadratic-cone constraints in each problem 3 are strictly feasible, and there exists a unique solution for each problem 3. Under these assumptions, the convergence of the sequential SOCP method is proved in [15]. The converged solution $\boldsymbol{y}^*$ satisfies the Karush–Kuhn–Tucker condition for problem 4, and $\boldsymbol{y}^*$ is at least a local minimum of problem 4. More information about this sequential method and proofs can be found in [15].

The entry trajectory optimization problem considered in this paper is much more complicated than problem 4, and addressing the convergence and optimality of the solutions are quite challenging, but we have the following assurance about the feasibility of the solutions.

Proposition 2: If a feasible state trajectory $\boldsymbol{x}$ of problem 2 satisfies the constraints in Eq. (38), then it also satisfies the original path constraints in Eqs. (26–28) based on the linearization defined by Eq. (32).

Proof: Let $\boldsymbol{x}$ be any feasible state of problem 2, that is

$$f_i(r_*, V_*) + f_i'(r_*, V_*)[r - r_*; V - V_*] \le f_{i,\max}, \quad i = 1, 2, 3 \tag{57}$$

in which $(r, V) \in \boldsymbol{x}$ and $f_i(r, V)$ from Eqs. (26–28) are all linearized with respect to the reference point $(r_*, V_*) \in \boldsymbol{x}_*$. The Jacobian f_i' can be found in Eq. (32).

Now, we can further represent $f_i(r, V)$ using the second-order Taylor-series expansion.

$$f_i(r, V) = f_i(r_*, V_*) + f_i'(r_*, V_*)[r - r_*; V - V_*] + \frac{1}{2}[r - r_*; V - V_*]^T\nabla^2 f_i(r_*, V_*)[r - r_*; V - V_*] \tag{58}$$

in which $\nabla^2 f_i$ is the Hessian of f_i.

By substituting Eq. (57) into Eq. (58), we have

$$f_i(r, V) \le f_{i,\max} + \frac{1}{2}[r - r_*; V - V_*]^T\nabla^2 f_i(r_*, V_*)[r - r_*; V - V_*], \quad i = 1, 2, 3 \tag{59}$$

The Hessian $\nabla^2 f_i$ can be obtained by taking the second derivatives of Eqs. (26–28). Let us take the heat-rate constraint as an example with its Hessian as follows:

$$\nabla^2 f_1(r, V) = \begin{bmatrix} \frac{\partial^2 f_1}{\partial r^2} & \frac{\partial^2 f_1}{\partial r \partial V} \\ \frac{\partial^2 f_1}{\partial V \partial r} & \frac{\partial^2 f_1}{\partial V^2} \end{bmatrix} = k_Q(\sqrt{R_0 g_0})^{3.15}\sqrt{\rho}\begin{bmatrix} \frac{0.25R_0^2V^{3.15}}{h_s^2} & \frac{-1.575R_0V^{2.15}}{h_s} \\ \frac{-1.575R_0V^{2.15}}{h_s} & 6.7725V^{1.15} \end{bmatrix}$$

Note that $\nabla^2 f_1(r, V)$ is symmetric and negative definite for all $(r, V) \in \boldsymbol{x}$. Similarly, the Hessian matrices $\nabla^2 f_2$ and $\nabla^2 f_3$ for dynamic pressure and normal load constraints are all negative definite. Thus, the following inequalities hold from Eq. (59), which are exactly the constraints in Eqs. (26–28):

$$f_i(r, V) \le f_{i,\max}, \qquad i = 1, 2, 3$$

Therefore, the original path constraints are all satisfied by the feasible state trajectory $\boldsymbol{x}$ of problem 2.

Based on the preceding analysis, the following remarks can be made:

1) Without the concavity of the second-order Taylor-series expansion of $f_i(r, V)$, there is no guarantee that any feasible state trajectory $\boldsymbol{x}$ satisfying the linearized path constraints in Eq. (38) will also satisfy the original path constraints in Eqs. (26–28). Thus, the feasibility to the original path constraints in problem 1 can be guaranteed by the feasible solutions of the successive subproblems.

2) Although the existence of the optimal solution for each approximated problem can be proved for our specific entry

applications and the feasibility of the path constraints can be preserved by the sequential solutions, a complete theoretical proof of the convergence of the SCP method for generic entry flight is much more difficult than the problems considered in the literature because of the following reasons. First, the objective functional in Eq. (21) is not in quadratic or general convex form for most cases. As such, it cannot be directly discretized into a linear function. Second, strong nonlinear terms, such as aerodynamic forces and an exponential atmospheric model, are naturally included in the entry dynamics in Eq. (22). All of these reasons make a general proof of the existence of the solution to problem 0 or problem 1 rather elusive, let alone the convergence and equivalence of the SCP method.

3) A complete theoretical proof of the convergence of the sequential method is still an open challenge; however, our numerical simulations show a strong evidence of convergence in solving hypersonic entry trajectory optimization problems, and the accuracy of the successive approximation method can be guaranteed as well. A faster rate of convergence is expected when a more accurate initial trajectory could be provided. In addition, the accuracy of the converged solution can be demonstrated numerically by simply substituting the converged optimal control into the original system, and verifying the similarity between the propagated trajectory and the optimal trajectory obtained by the sequential method. Satisfaction of the original constraints can be verified as well.

V. Numerical Demonstration

The methodology described in this paper has been implemented in SDPT3, a MATLAB-based convex optimization solver that uses a primal–dual interior-point algorithm for linear, quadratic, and SDP problems [28]. For each fixed k, the CPU time taken by SDPT3 to solve each of the subproblems in Sec. IV on a typical desktop computer is 0.3–0.5 s. The execution time is expected to be considerably shorter if the simulation environment is changed that can take advantage of compiled or parallel programs.

A reusable-launch-vehicle model used for numerical demonstrations is much like the configuration examined in [29], which is a vertical-takeoff, horizontal-landing, winged-body vehicle. For entry flight, the mass of the vehicle is set to be $m = 104{,}305$ kg and the reference area $A_{\text{ref}} = 391.22\ \text{m}^2$. The 1976 U.S. Standard Atmosphere is used in the simulations. For the conceptual entry trajectory optimization in this paper, the aerodynamic coefficients are approximated by fitting the data in hypersonic regime with [30]:

$$C_L = -0.041065 + 0.016292\alpha + 0.0002602\alpha^2 \tag{60}$$

$$C_D = 0.080505 - 0.03026C_L + 0.86495C_L^2 \tag{61}$$

in which α is in degrees and is scheduled with respect to velocity as shown next.

$$\alpha = \begin{cases} 40, & \text{if } V > 4570\ \text{m/s} \\ 40\text{–}0.20705(V-4570)^2/340^2, & \text{otherwise} \end{cases} \tag{62}$$

A nominal profile of angle of attack and profiles of aerodynamic coefficients are shown in Fig. 1. For simplicity, we only consider fixed-flight-time entry trajectory optimization problems; however, the approach developed in this paper is also applicable for free-flight-time entry problems by normalizing the time to τ, such that $\tau \in [0, 1]$. When the time is normalized in this manner, we must multiply the dynamics by the t_f parameter. Then, it can be solved as a fixed-flight-time problem, in which the parameter t_f is also optimized. In the simulations, we choose $t_f = 1600$ s. The time domain $[t_0, t_f]$ is discretized into 500 intervals, and the maximum number of iterations is set to be 100. The simulation parameters are shown in Table 1. Based on the data, this entry mission has an initial downrange of 7792 km and right cross range of 1336 km from an altitude of 100 km and velocity of 7450 m/s at entry interface. The desired terminal altitude is 25 km, the terminal flight-path angle must be −10 deg, and the terminal heading angle must be 90 deg, pointing to the east. The limit on the magnitude of the bank-angle rate is 10 deg/s. The upper and lower bounds of the bank angle and the values of limits on the path constraints are also given in Table 1.

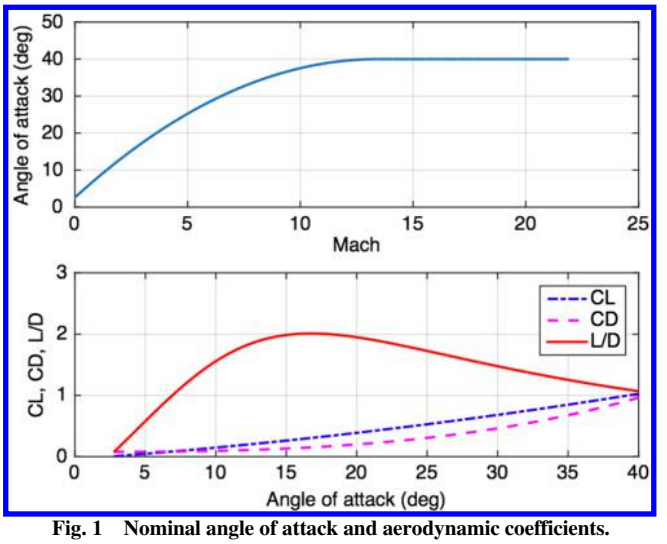

Fig. 1 Nominal angle of attack and aerodynamic coefficients.

For the two examples considered in this section, the trust-region size in Eq. (46) and the stopping criteria in Eq. (47) are selected as

$$\delta = \left[\frac{10{,}000}{R_0}, \frac{20\pi}{180}, \frac{20\pi}{180}, \frac{500}{V_0}, \frac{20\pi}{180}, \frac{20\pi}{180}, \frac{20\pi}{180}\right]^T$$

$$\varepsilon = \left[\frac{100}{R_0}, \frac{0.05\pi}{180}, \frac{0.05\pi}{180}, \frac{1}{V_0}, \frac{0.05\pi}{180}, \frac{0.05\pi}{180}, \frac{1\pi}{180}\right]^T$$

To verify the performance of the SCP method developed in this paper, numerical results are compared to the solutions obtained by GPOPS-II, which is a general-purpose MATLAB software program for solving multiple-phase optimal control problems [31]. In the simulations, GPOPS solves exactly the same entry trajectory optimization problems with Eq. (18) as the system dynamics and the bank-angle rate as the control. GPOPS employs a Legendre–Gauss–Radau quadrature orthogonal collocation method to convert the continuous-time optimal control problem to a large sparse NLP problem. An adaptive mesh-refinement method is implemented to determine the number of nodes required and the degree of the approximating polynomial to achieve a specified accuracy. The default NLP solver used by GPOPS is IPOPT [32].

A. Example 1: Minimum Terminal-Velocity Entry

With the given initial and terminal conditions, we firstly considered the minimum terminal-velocity entry with specific terminal constraints and path constraints. The objective function is shown in Eq. (63), which is a linear function of V. Thus, no linearization is needed to convexify the objective function.

$$J = V(t_f) \tag{63}$$

Table 1 Parameters for entry flight

Parameter	Value	Parameter	Value
h_0	100 km	ϕ_f	70 deg
θ_0	0 deg	γ_f	−10 deg
ϕ_0	0 deg	ψ_f	90 deg
V_0	7450 m/s	$u_{\max}$	10 deg/s
γ_0	−0.5 deg	$\sigma_{\min}$	−80 deg
ψ_0	0 deg	$\sigma_{\max}$	80 deg
σ_0	0 deg	$\dot{Q}_{\max}$	1,500 kW/m^2
h_f	25 km	$q_{\max}$	18,000 N/m^2
θ_f	12 deg	$n_{\max}$	2.5g

Figure 2 shows that the value of the objective function decreases to the converged solution, and the sequential method converges in 17 steps without any initial guess for each subproblem. If looser tolerance is used, fewer steps will be needed for convergence. In each step, SDPT3 solves a convex problem in about 0.35 s CPU time. To make the progression of the convergence clearer, the trajectories for all iterations are shown by curves from "cool" (thin dark blue) to "warm" (thick red) in Fig. 3. As can be seen from the

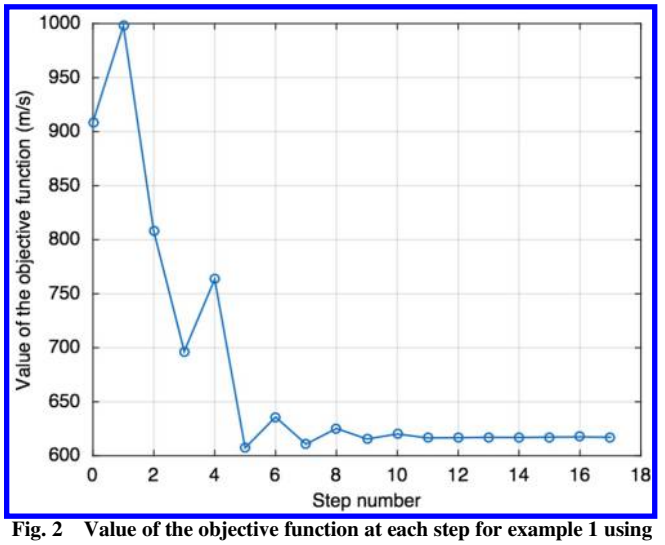

Fig. 2 Value of the objective function at each step for example 1 using SCP.

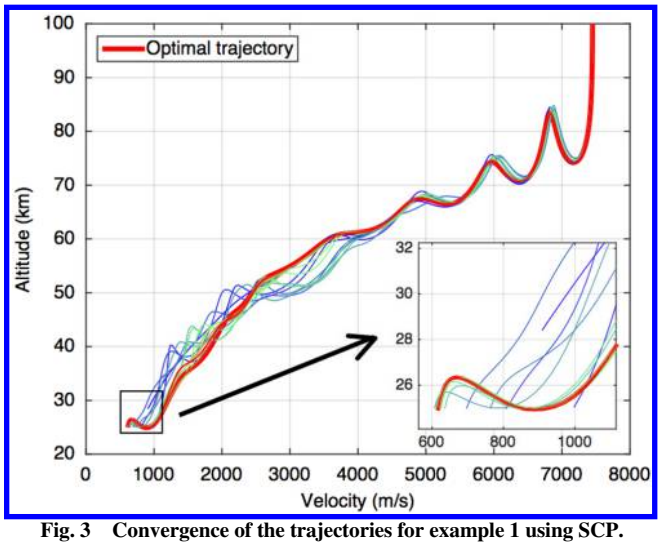

Fig. 3 Convergence of the trajectories for example 1 using SCP.

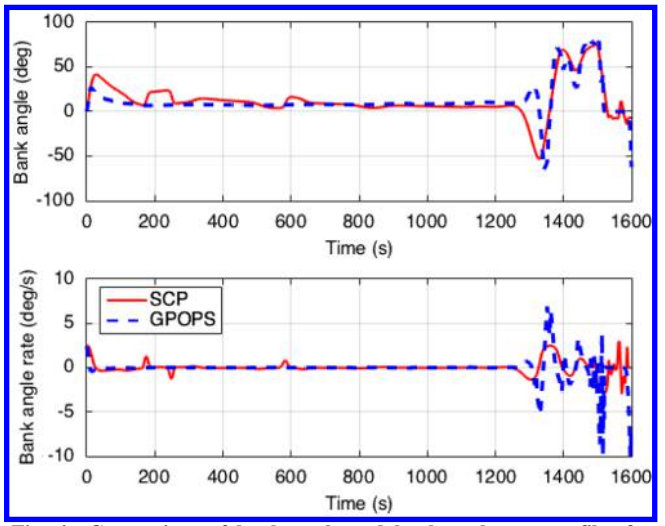

Fig. 4 Comparison of bank angle and bank-angle-rate profiles for example 1.

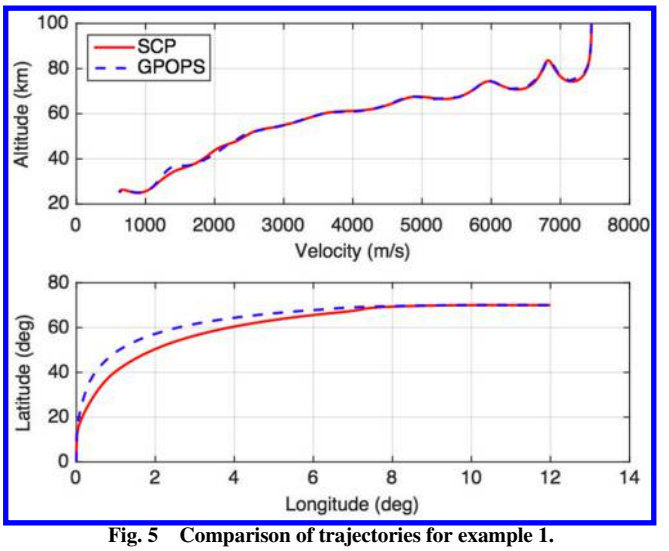

Fig. 5 Comparison of trajectories for example 1.

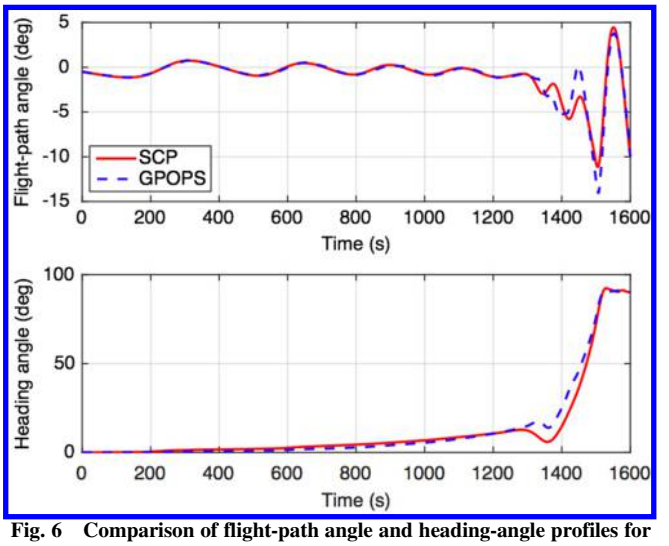

Fig. 6 Comparison of flight-path angle and heading-angle profiles for example 1.

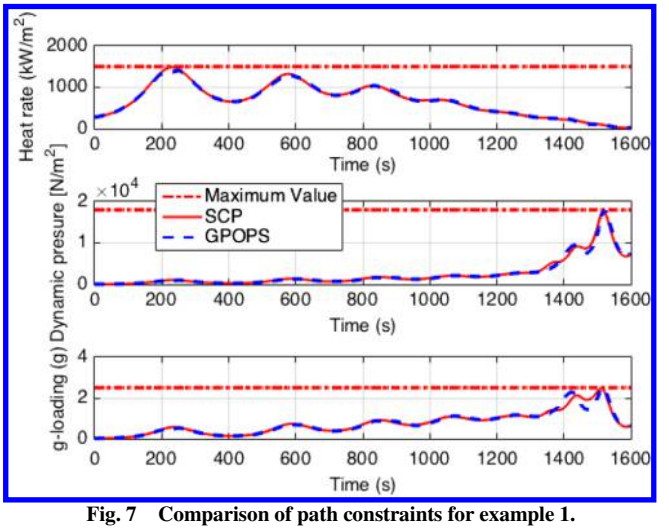

Fig. 7 Comparison of path constraints for example 1.

Table 2 Difference of the states between consecutive iterations for example 1

Iteration number	$\lvert\Delta r\rvert$, km	$\lvert\Delta\theta\rvert$, deg	$\lvert\Delta\phi\rvert$, deg	$\lvert\Delta V\rvert$, m/s	$\lvert\Delta\gamma\rvert$, deg	$\lvert\Delta\psi\rvert$, deg	$\lvert\Delta\sigma\rvert$, deg
1	4.2219	2.7591	0.3651	89.1868	8.6809	12.0018	19.9999
2	4.0227	1.9495	0.3696	190.3813	5.1802	11.7014	19.9992
3	5.3001	1.6708	0.3035	110.9115	4.6261	9.1813	19.9977
4	4.0136	0.9492	0.2606	66.6928	4.0521	7.3689	19.9937
5	3.8226	0.7047	0.1821	156.2186	3.2078	5.0572	19.8149
6	2.9610	0.6749	0.1202	28.5263	2.4287	2.4244	17.9734
7	2.6880	0.4954	0.0783	25.0989	1.7382	1.6570	15.0628
8	1.3337	0.4631	0.0698	14.5745	1.3135	1.3620	13.1139
9	0.6669	0.3729	0.0495	9.7738	0.9986	1.2908	11.7351
10	0.3370	0.2981	0.0420	4.5417	0.7009	1.2006	9.2602
11	0.3033	0.1037	0.0231	2.9160	0.3975	1.1652	8.1199
12	0.2463	0.0695	0.0076	2.2816	0.1959	0.8639	6.6593
13	0.1746	0.0277	0.0052	1.9628	0.0691	0.4125	5.2126
14	0.1218	0.0040	0.0039	1.5663	0.0305	0.1846	4.1054
15	0.0896	0.0028	0.0016	0.9092	0.0083	0.0589	2.9869
16	0.0611	0.0020	0.0010	0.3940	0.0053	0.0124	1.6177
17	0.0332	0.0011	0.0006	0.3143	0.0044	0.0037	0.7794

Table 3 Comparison of solutions for example 1

Method	Terminal velocity, m/s	CPU time, s
SCP	617.22	6.74
GPOPS	628.54	68.35

zoom-in views, the trajectories are quite close after 10 steps. To further demonstrate the convergence performance of the solution procedure, more quantitative results are reported in Table 2, which presents the difference of each state variable between consecutive iterations. The difference of the states shown in the table is defined as $|\Delta z| \coloneqq \max |z^{(k)}(t_i) - z^{(k-1)}(t_i)|, i = 1, 2, \ldots, N$, in which z could be each state variable.

The trajectory profiles from the SCP method and GPOPS are shown in Figs. 4–7, and the optimal solutions and CPU time consumption are summarized in Table 3. As can be seen from Fig. 4, both the profiles of the bank angle and bank-angle rate have the same trend. The typical high-frequency jitters are not present in the bank-angle profiles under both SCP and GPOPS due to the decoupling of the control from the states in the new problem formulation developed in this paper. The profiles are very smooth, because a constraint is enforced on the new control, which sets the limit on the bank-angle rate. Another noteworthy thing about the bank-angle profiles is a change of the sign of bank angle between 1330 and 1400 s, which is called bank reversal for entry guidance. The bank reversals presented in Fig. 4 are much smoother than the instant reversals observed in traditional entry trajectory optimization. To make these changes, obvious oscillations are observed in the bank-angle-rate profiles in the later part of flight. Figures 5 and 6 present the histories of the other state variables, and the solutions of SCP and GPOPS are very similar. All the terminal constraints and path constraints are satisfied under these two approaches. In addition, Fig. 7 indicates that the heat rate reaches its highest value at about 230 s, the dynamic pressure constraint is active at about 1520 s, and the maximum value of g loading happens at about 1515 s. The small differences in the optimal solutions could be caused by several potential reasons, such as the linearization and approximation of dynamics, different discretization strategies used by the two approaches, etc. Generally speaking, the NLP solver GPOPS is much slower than the SCP method for entry problems, good initial guesses are required for GPOPS to converge, and the time cost of GPOPS is unpredictable.

B. Example 2: Minimum Heat-Load Entry

To further investigate the efficiency of the proposed method, the minimum heat-load entry is considered as the second example. It is more complicated than the minimum terminal-velocity entry, because the objective becomes a highly nonlinear integral function shown in Eq. (64). All the constraints and parameters are the same as example 1.

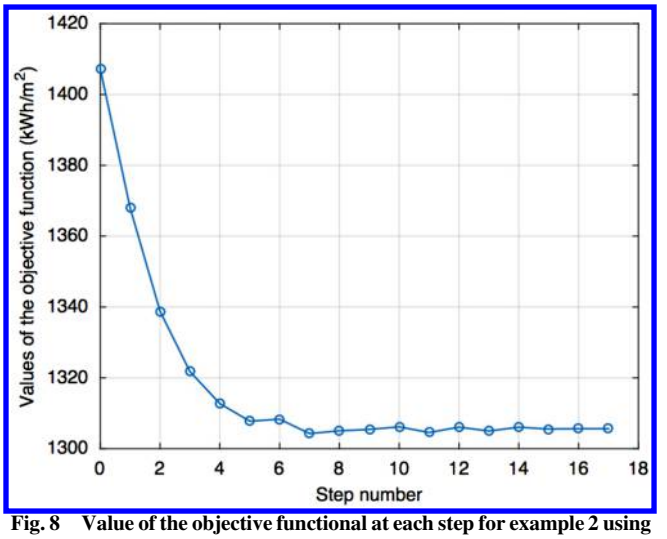

Fig. 8 Value of the objective functional at each step for example 2 using SCP.

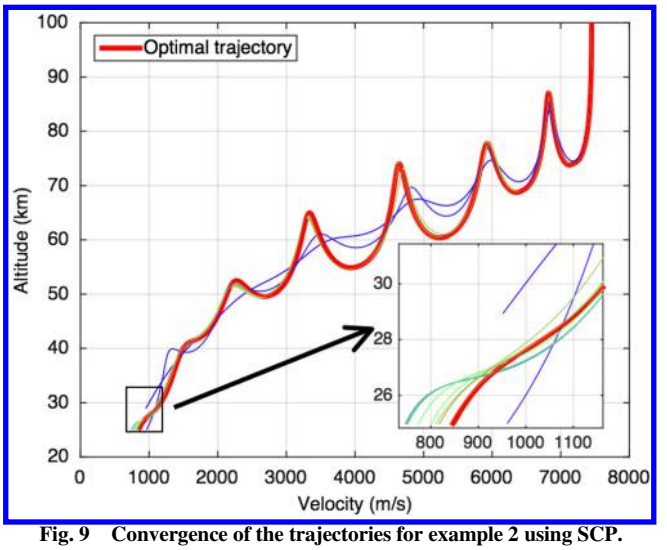

Fig. 9 Convergence of the trajectories for example 2 using SCP.

Table 4 Difference of the states between consecutive iterations for example 2

Iteration number	$\lvert\Delta r\rvert$, km	$\lvert\Delta\theta\rvert$, deg	$\lvert\Delta\phi\rvert$, deg	$\lvert\Delta V\rvert$, m/s	$\lvert\Delta\gamma\rvert$, deg	$\lvert\Delta\psi\rvert$, deg	$\lvert\Delta\sigma\rvert$, deg
1	3.9376	1.7135	0.4975	165.3609	6.7366	13.1749	20.0000
2	2.8673	1.9567	0.4671	245.2744	3.6572	17.6450	19.9999
3	3.1045	1.1547	0.1029	106.1395	3.2549	7.8184	19.9993
4	1.5523	1.3186	0.1069	110.5910	2.4116	3.1988	19.9983
5	1.4254	1.1349	0.1147	99.9065	2.0859	3.8067	19.7880
6	1.2064	0.8702	0.0892	74.7424	1.4889	3.0627	19.9996
7	1.1582	0.7109	0.0646	53.2254	0.7312	2.7035	19.9982
8	0.8601	0.5369	0.0507	32.2859	0.3464	2.0926	17.1535
9	0.6882	0.3186	0.0471	19.7247	0.2692	1.4540	15.3530
10	0.4539	0.1348	0.0347	8.3208	0.1314	0.6236	13.3392
11	0.2483	0.0871	0.0221	3.0767	0.0729	0.2442	11.9793
12	0.1483	0.0348	0.0091	1.9917	0.0422	0.1365	9.0197
13	0.1137	0.0165	0.0069	1.7110	0.0255	0.0833	7.6837
14	0.0925	0.0067	0.0045	1.4331	0.0148	0.0548	5.1405
15	0.0825	0.0051	0.0024	1.2069	0.0126	0.0171	2.4342
16	0.0481	0.0028	0.0019	0.6569	0.0113	0.0075	1.1968
17	0.0386	0.0017	0.0012	0.5263	0.0092	0.0014	0.3113

$$J = \int_{t_0}^{t_f} \dot{Q}\,\mathrm{d}t = \int_{t_0}^{t_f} f_1(r, V)\,\mathrm{d}t \approx \int_{t_0}^{t_f} \{f_1(r_*, V_*) + f_1'(r_*, V_*)[(r - r_*; V - V_*)]\}\,\mathrm{d}t \quad (64)$$

The convergence of the value of the objective functional is shown in Fig. 8. For the minimum heat-load entry flight, the SCP method converges in 17 steps. In each step, it takes SDPT3 about 0.5 s CPU time to solve a subproblem. The trajectories for all iterations and the zoom-in views are depicted in Fig. 9, in which the curves go from thin dark blue to thick red as well, so that the progression of solutions can be visualized. We can see that the trajectories are very close after nine steps. Table 4 shows the difference of state variables between consecutive iterations. One may notice from Fig. 8 (and Fig. 2 for example 1) that the converged solution is not the minimum value on this plot. For example, iteration 7 has a lower heat load than the converged point. The reason is that the solution to each subproblem in the intermediate iterations is not necessarily an accurate solution of the original problem. In this paper, instead of solving a single convex problem, we try to find an approximate solution to the original problem by solving a sequence of convex subproblems, until the convergence criterion in Eq. (47) is satisfied, that is, the difference of the solutions between two consecutive steps becomes very small.

Figures 10–13 compare the results obtained by the SCP and GPOPS. To reduce the severe oscillations observed in the bank-angle-rate profile in example 2, an additional constraint is imposed on the bank-angle acceleration. The limit on the magnitude of the bank-angle acceleration is 10 deg/s^2. The bank angle and bank-angle-rate profiles of the SCP in Fig. 10 have the similar trend as those of GPOPS. In this example, both the profiles of bank angle and bank-angle rate are very smooth. No obvious bank reversals are observed in these bank-angle profiles for the minimum heat-load entry case. The profiles for each state are quite similar under these two approaches.

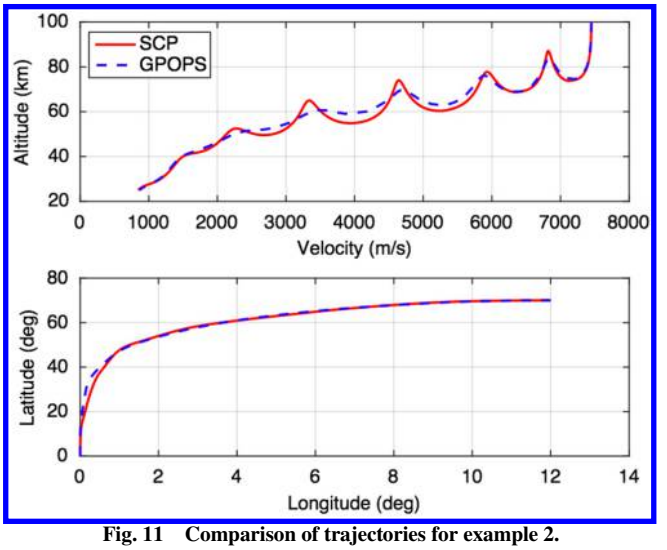

Fig. 11 Comparison of trajectories for example 2.

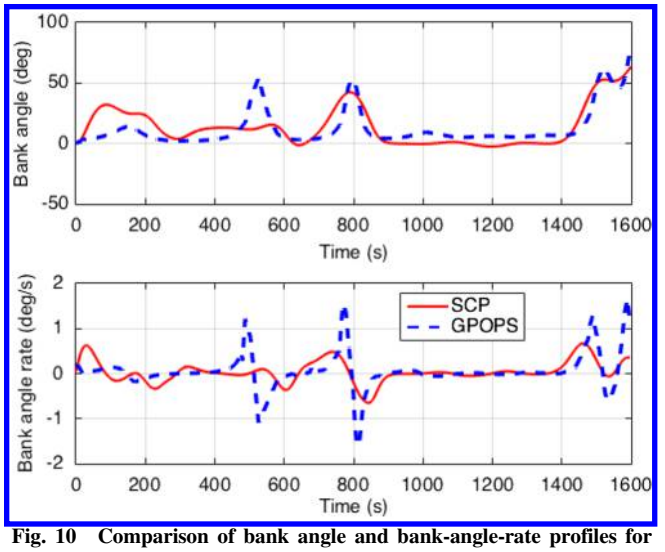

Fig. 10 Comparison of bank angle and bank-angle-rate profiles for example 2.

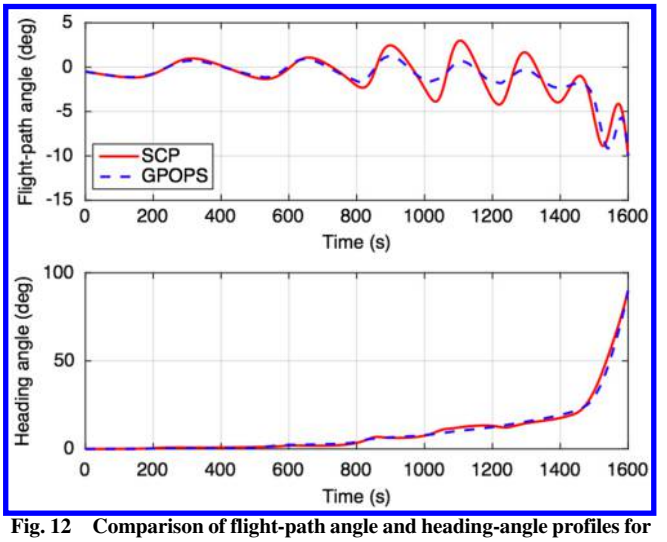

Fig. 12 Comparison of flight-path angle and heading-angle profiles for example 2.

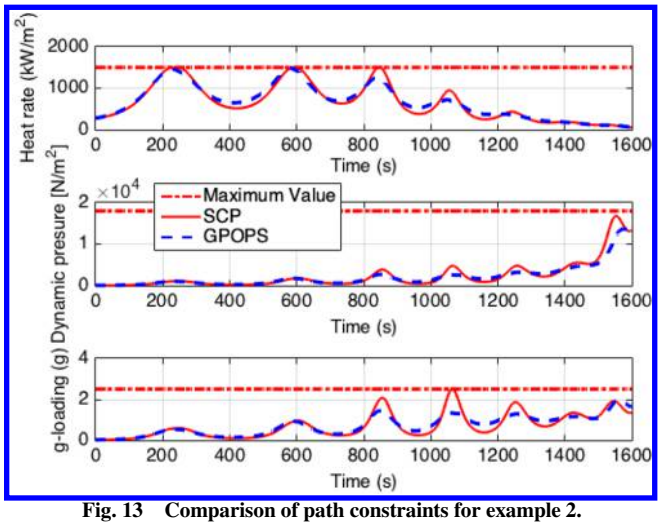

Fig. 13 Comparison of path constraints for example 2.

Table 5 Comparison of solutions for example 2

Method	Heat load, kWh/m^2	CPU time, s
SCP	1306.25	8.16
GPOPS	1311.17	178.43

The terminal constraints, including the hard constraints on the terminal longitude and latitude, are all satisfied under the SCP and GPOPS. Figure 13 confirms that the heat-rate constraint becomes active twice between finite intervals [225, 250 s] and [586, 600 s], respectively, along the minimum heat-load trajectories. As such, the vehicle is commanded to decrease the heat load under the downrange requirement and path constraints. In addition, small differences in heat load are reported in Table 5, which further demonstrates the accuracy of the SCP method. For the SCP method, the heat load of the converged trajectory is about 1306 kWh/m^2 with a total reduction of about 101 kWh/m^2. It is indicated from Table 5 that the optimal value of the heat load by GPOPS is about 1311.17 kWh/m^2 with a difference of about 4.92 kWh/m^2 from the converged solution of the SCP. The difference is less than 5% of the total reduction by the SCP. If a customized convex-optimization algorithm were applied, the computational speed of each subproblem would be further improved. However, GPOPS takes much longer time to converge, and its time cost is unpredictable.

VI. Conclusions

In this paper, an sequential convex programming (SCP) method is developed for 3-D planetary-entry trajectory optimization problems. This method is designed to solve the highly nonlinear optimal control problems associated with hypersonic entry via convex programming methods. The main contribution of this study is the decoupling of states and control by defining new state and control variables, and the formulation of the trajectory optimization problem with nonlinear dynamics and nonlinear path constraints as a sequence of finite-dimensional convex optimization problems, specifically as a sequence of second-order cone programming problems. The original equations of motion are used, and no assumptions or approximations are made to calculate the velocity. There exist convex programming algorithms, such as interior-point methods, that can solve each convex subproblem very efficiently. The existence and convergence of the sequential solutions are partially proved where currently possible, and the accuracy of the proposed method has been demonstrated by numerical solutions to minimum terminal-velocity and minimum heat-load entry trajectory optimization problems through comparisons with GPOPS. This rapid trajectory-generation method is more attractive for planetary-entry problems than genetic nonlinear programming algorithms, which are generally very slow to converge. Consequently, the SCP method has a potential for onboard implementation.

References

[1] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207. doi:10.2514/2.4231

[2] Boyd, S., and Vandenberghe, L., *Convex Optimization*, Cambridge Univ. Press, Cambridge, England, U.K., 2004, pp. 127–187, Chap. 4.

[3] Vandenberghe, L., and Boyd, S., "Semidefinite Programming," *SIAM Review*, Vol. 38, No. 1, 1996, pp. 49–95. doi:10.1137/1038003

[4] Nesterov, Y., *Introductory Lectures on Convex Optimization*, Kluwer, Boston, MA, 2004, pp. 51–110, Chap. 2.

[5] Nesterov, Y., and Nemirovsky, A., *Interior-Point Polynomial Methods in Convex Programming*, Soc. for Industrial and Applied Mathematics, Philadelphia, PA, 1994, pp. 57–99, Chap. 3, pp. 217–271, Chap. 6.

[6] Wright, S. J., *Primal–Dual Interior-Point Methods*, Soc. for Industrial and Applied Mathematics, Philadelphia, PA, 1997, pp. 1–47, Chaps. 1–2.

[7] 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

[8] 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

[9] 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

[10] Acikmese, B., Carson, J. M., 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

[11] 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

[12] 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

[13] Lu, P., and Liu, X., "Autonomous Trajectory Planning for Rendezvous and Proximity Operations by Conic Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 2, 2013, pp. 375–389. doi:10.2514/1.58436

[14] Liu, X., and Lu, P., "Robust Trajectory Optimization for Highly Constrained Rendezvous and Proximity Operations," AIAA Paper 2013-4720, Aug. 2013. doi:10.2514/6.2013-4720

[15] Liu, X., and Lu, P., "Solving Nonconvex Optimal Control Problems by Convex Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 3, 2014, pp. 750–765. doi:10.2514/1.62110

[16] 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

[17] 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

[18] 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

[19] Liu, X., Shen, Z., and Lu, P., "Closed-Loop Optimization of Guidance Gain for Constrained Impact," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 453–460. doi:10.2514/1.G000323

[20] Liu, X., "Fuel-Optimal Rocket Landing with Aerodynamic Controls," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper

2017-1732, Jan. 2017.
doi:10.2514/6.2017-1732

[21] Roenneke, A. J., and Markl, A., "Reentry Control of a Drag vs. Energy Profile," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 5, 1994, pp. 916–920.
doi:10.2514/3.21290

[22] Lu, P., "Entry Guidance: A Unified Method," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 3, 2014, pp. 713–728.
doi:10.2514/1.62605

[23] Shaffer, P. J., Ross, I. M., Oppenheimer, M. W., Doman, D. B., and Bollino, K. B., "Fault-Tolerant Optimal Trajectory Generation for Reusable Launch Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 6, 2007, pp. 1794–1802.
doi:10.2514/1.27699

[24] Berkovitz, L. D., *Optimal Control Theory*, Springer–Verlag, Berlin, 1975, pp. 39–117, Chap. 3.

[25] Palacios-Gomez, F., Lasdon, L., and Engquist, M., "Nonlinear Optimization by Successive Linear Programming," *Management Science*, Vol. 28, No. 10, 1982, pp. 1106–1120.
doi:10.1287/mnsc.28.10.1106

[26] Banks, S. P., and Dinesh, K., "Approximate Optimal Control and Stability of Nonlinear Finite-and Infinite-Dimensional Systems," *Annals of Operations Research*, Vol. 98, No. 1, 2000, pp. 19–44.
doi:10.1023/A:1019279617898

[27] Tomas-Rodriguez, M., and Banks, S. P., *Linear, Time-Varying Approximations to Nonlinear Dynamical Systems with Applications in Control and Optimization*, Springer–Verlag, London, 2010, pp. 101–121, Chap. 6.

[28] Toh, K. C., Todd, M. J., and Tutuncu, R. H., "On the Implementation and Usage of SDPT3—A MATLAB Software Package for Semidefinite–Quadratic–Linear Programming, Version 4.0," *Handbook on Semidefinite, Conic and Polynomial Optimization*, edited by M. F. Anjos, and J. B. Lasserre, Vol. 166, International Series in Operations Research & Management Science, Springer US, New York, 2011, pp. 715–754.

[29] Stanley, D. O., Engelund, W. C., Lepsch, R. A., McMillin, M., Wurster, K. E., Powell, R. W., Guinta, T., and Unal, R., "Rocket-Powered Single-Stage Vehicle Configuration Selection and Design," *Journal of Spacecraft and Rockets*, Vol. 31, No. 5, 1994, pp. 792–798.
doi:10.2514/3.26514

[30] Lu, P., "Entry Guidance and Trajectory Control for Reusable Launch Vehicle," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 143–149.
doi:10.2514/2.4008

[31] Patterson, M. A., and Rao, A. V., "GPOPS-II: A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using hp-Adaptive Gaussian Quadrature Collocation Methods and Sparse Nonlinear Programming," *ACM Transactions on Mathematical Software (TOMS)*, Vol. 41, No. 1, 2014, pp. 1–37.
doi:10.1145/2684421

[32] Wachter, A., and Biegler, L. T., "On the Implementation of a Primal–Dual 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