



Asteroid Precision Landing via Multiple Sliding Surfaces Guidance Techniques

Roberto Furfaro*

University of Arizona, Tucson, Arizona 85721

Dario Cersosimo†

University of Missouri–Columbia, Columbia, Missouri 65211

and

Daniel R. Wibben‡

University of Arizona, Tucson, Arizona 85721

DOI: 10.2514/1.58246

Autonomous close-proximity operations (hovering, landing) in the low-gravity environment exhibited by asteroids are particularly challenging. A novel nonlinear landing guidance scheme has been developed for spacecraft that are required to execute autonomous closed-loop guidance to a designated point on the asteroid surface. Based on high-order sliding-mode control theory, the proposed multiple sliding surface guidance algorithm has been designed to take advantage of the ability of the system to reach the sliding surface in a finite time. High control activity typical of sliding control design is avoided, resulting in a guidance law that is robust against unmodeled yet bounded perturbations. The proposed multiple sliding surface guidance does not require any off-line trajectory generation, and therefore it is flexible enough to target a large variety of points on the surface without the need of ground-based trajectory analysis. The global stability of the proposed guidance algorithm is proven using a Lyapunov-based approach. The behavior of the multiple sliding surface guidance-based feedback asteroid landing trajectories is investigated via a parametric analysis, and a full set of Monte Carlo simulations in realistic landing scenarios is implemented to evaluate the guidance performance. Based on such results, the multiple sliding surface guidance algorithm is demonstrated to be very accurate and flexible, and it has the potential to be implemented as real-time guidance during asteroid landing and possibly for close-proximity operations.

I. Introduction

THE space exploration community is currently experiencing a renewed interest in near-Earth asteroids (NEAs) missions. The surge in NEA mission activities, for which various space agencies around the world (e.g., NASA, ESA, and the Japan Aerospace Exploration Agency) are commissioning studies to determine the next generation of feasible robotic precursors and human missions, is justified by a variety of scientific and technological reasons, including 1) the scientific contribution provided by NEAs exploration to the understanding of the origin and evolution of the solar system [1]; 2) the characterization and quantification of the risk associated with possible Earth collision [2]; and 3) the characterization and quantification of NEAs as a source of extraterrestrial natural resources, especially metals [3]. Indeed, robotic science missions are currently planned to investigate carbonaceous asteroids [4]. It is expected that the elemental composition of such objects, which are generally considered as “pristine” (i.e., virtually unprocessed since their formation from the solar protodisk), may provide scientists with a window to unveil the early history of the solar system.

Asteroids and comets are very small when compared with other objects in the solar system (e.g., Mars, the moon, Titan, etc.). Their small size and irregular shape induce a weak gravitational pull, which

tends to rapidly vary in intensity and direction as a function of the spatial coordinates of the visiting spacecraft. In addition, the contribution from solar gravity and radiation pressure is an important factor. Orbital trajectories are generally complex and nonperiodic, and, for many small bodies, stability is guaranteed only for a limited set of latitudes [5]. Accurate information about size, density, rotational state, the number of orbital companions, surface characteristics, and heliocentric orbit is vital for both scientists and mission engineers. Acquisition of such critical information requires the execution of a set of close-proximity maneuvers to allow the spacecraft to collect data at specific locations, both remotely and in situ. It is important to note that, design and implementation of close-proximity maneuvers is highly coupled to the environmental factors around the target [6]. Two particular types of close-proximity maneuvers that are especially critical are hovering and landing.

Before landing, a global mapping and surface characterization effort may be carried out to identify a safe and scientifically valuable landing site. A potential strategy for this task is hovering at an arbitrary altitude from the surface of an asteroid. There are two types of hovering modes: inertial hovering and body-fixed hovering. Inertial hovering consists of fixing the spacecraft position relative to the body with respect to the asteroid–sun reference frame by creating an artificial equilibrium point by applying the necessary thrust for canceling the gravitational pull from the asteroid. Body-fixed hovering consists of fixing the position of the spacecraft relative to the body-fixed frame of the asteroid [7]. In this way, the spacecraft appears static for an observer on the surface asteroid. Analyses on the dynamics of these hovering modes were performed during the past decade, concluding that both hovering modes can be sustained by implementing multidimensional deadband controllers [7–10]. In fact, the Hayabusa spacecraft operated in inertial hovering while performing global observations of asteroid Itokawa, and, during its final descent maneuver, Hayabusa switched to body-fixed hovering at about 20 m from the surface of Itokawa [11–13].

Descent and landing on a small body demands the design of systems that provide high levels of navigation accuracy and

Presented as Paper AAS 11-647 at the AAS/AIAA Astrodynamics Specialist Conference, Girdwood, Alaska, USA, 31 July 2011–4 August 2011; received 12 March 2012; revision received 7 September 2012; accepted for publication 10 October 2012; published online 22 April 2013. Copyright © 2012 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the \$10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 1533-3884/13 and \$10.00 in correspondence with the CCC.

*Assistant Professor, Department of Systems and Industrial Engineering, 1127 E. James E. Rogers Way.

†Postdoctoral Fellow, Department of Mechanical and Aerospace Engineering, E1412 Laffer Hall.

‡Graduate Student, Department of Systems and Industrial Engineering, 1127 E. James E. Rogers Way.

autonomy. Yet, these systems must be flexible and robust enough to implement a closed-loop approach that guides the spacecraft to the desired targeted point on the asteroid surface with zero velocity (e.g., a soft landing is considered when the impact velocity is < 3 m/s [14]). Alternatives for precision landing guidance and control strategies were proposed in the literature, including autonomous closed-loop guidance based on optical navigation [15–18]. For example, Hayabusa used laser altimetry and landmark tracking to navigate during its descent on Itokawa [11,12].

This paper investigates a novel class of nonlinear guidance algorithms applicable to closed-loop guidance for asteroid descent and landing. Here, we propose a multiple sliding surface guidance (MSSG) approach to autonomous asteroid landing that is robust against perturbations and unmodeled dynamics. MSSG has its roots in the high-order sliding-mode (HOSM) control theory [19,20] in which multiple sliding surfaces are employed to generate online targeting trajectories that are guaranteed to be globally stable under bounded perturbations (with a known upper bound). Two sliding surface vectors are concatenated in such a way that an acceleration command program that drives the second surface to zero automatically drives the dynamic system on the first surface in a finite time, forcing a zero error in both position and velocity.

This paper is organized as follows. In Sec. II, the asteroid landing guidance problem is formulated and the equations of motion are described. In Sec. III, the principles behind the HOSM methodology are introduced and the MSSG law is derived. In Sec. IV, a parametric analysis of the guidance law is executed and discussed. The results of a set of Monte Carlo simulations are presented and discussed to evaluate the accuracy of the guidance algorithm in realistic scenarios. Conclusions and future work are presented in Sec. V.

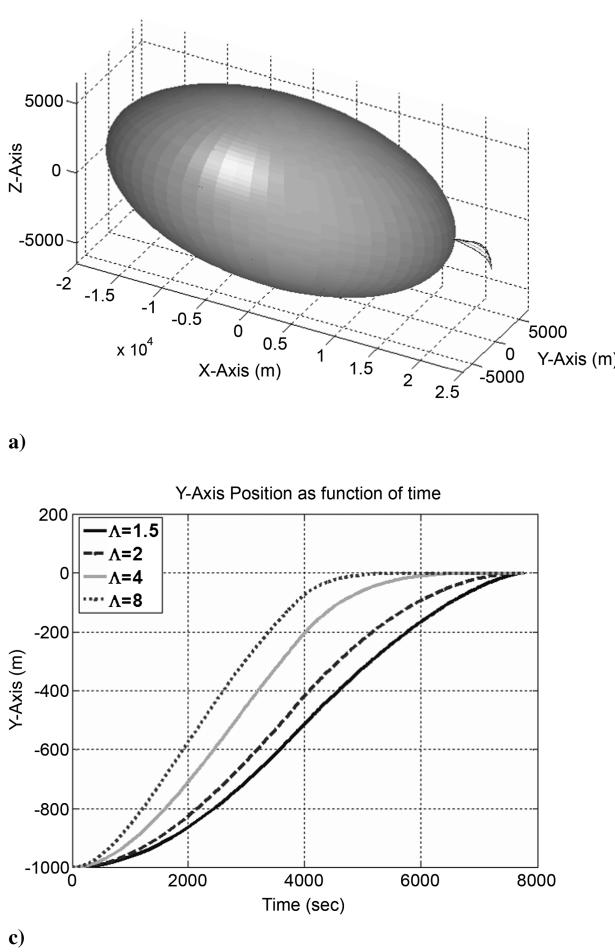


Fig. 1 Parametric analysis of the MSSG algorithms: effect of the guidance gain Λ on the trajectory history.

II. Guidance Problem Formulation

Here, the asteroid powered descent and landing guidance problem is considered. The guidance problem can be formulated as follows: Given the current state of the spacecraft (i.e., position and velocity), determine a real-time acceleration command program that reaches the target point on the asteroid surface with zero velocity. Next, the guidance model employed to derive the MSSG equations is presented.

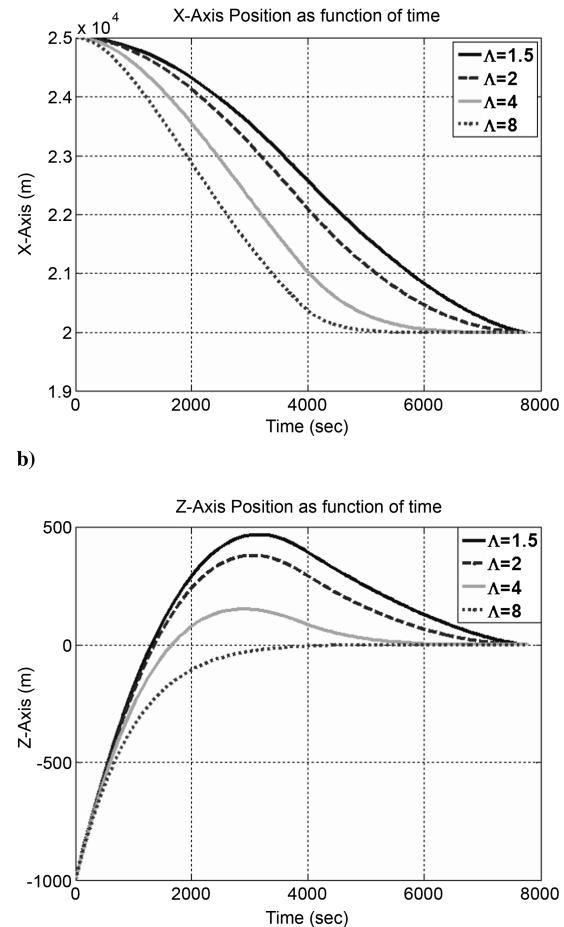
In formulating the asteroid landing guidance problem, we model the spacecraft dynamics near the asteroid using a two-body gravitational model (the spacecraft has negligible mass). The equations of motion for the spacecraft in a uniformly rotating, asteroid-fixed Cartesian coordinate frame, having the origin at the asteroid center of mass, are written as follows:

$$\dot{\mathbf{r}}_L = \mathbf{v}_L \quad (1)$$

$$\dot{\mathbf{v}}_L = 2\boldsymbol{\omega} \times \mathbf{v}_L + \boldsymbol{\omega} \times \boldsymbol{\omega} \times \mathbf{r}_L + \mathbf{g}(\mathbf{r}_L) + \mathbf{a}_c + \mathbf{a}_p \quad (2)$$

Here, $\mathbf{r}_L = [x, y, z]^T$ is the position vector in the body-fixed rotating frame, $\mathbf{v}_L = [v_x, v_y, v_z]^T$ is the velocity vector, $\mathbf{g}(\mathbf{r}_L) = [g_x, g_y, g_z]^T$ is the local gravitational field, $\mathbf{a}_c = [a_{cx}, a_{cy}, a_{cz}]^T$ is the acceleration command, and $\mathbf{a}_p = [a_{px}, a_{py}, a_{pz}]^T$ is the perturbing acceleration, accounting for unmodeled/unknown forces (e.g., gravity field inaccuracies, solar radiation pressure, and n th-body perturbations).

For this analysis, it is assumed that the shape of the asteroid can be modeled as a triaxial ellipsoid, allowing analytical determination of the asteroid gravitational field [21]. The gravitational field can



d)

be expressed as a partial derivative of the potential field, that is, $\mathbf{g}(\mathbf{r}_L) = \partial V / \partial \mathbf{r}_L^T$.

The equations of motion can be explicitly written in their scalar form:

$$\dot{x} = v_x \quad (3)$$

$$\dot{y} = v_y \quad (4)$$

$$\dot{z} = v_z \quad (5)$$

$$\dot{v}_x = 2\omega v_y + \omega^2 x + \frac{\partial V}{\partial x} + a_{Cx} + a_{Px} \quad (6)$$

$$\dot{v}_y = -2\omega v_x + \omega^2 y + \frac{\partial V}{\partial y} + a_{Cy} + a_{Py} \quad (7)$$

$$\dot{v}_z = \frac{\partial V}{\partial z} + a_{Cz} + a_{Pz} \quad (8)$$

The mathematical model described in Eqs. (1–8) is employed to derive the guidance equations. In the development of the guidance law, the mass of the spacecraft is assumed to be constant. However, in more realistic Monte Carlo simulations, required to test the performance of the proposed MSSG law, the model is upgraded to account for mass variation as given by the classical rocket equation:

$$\dot{m} = -\frac{\|\mathbf{T}\|}{I_{sp} g_c} \quad (9)$$

where $\|\mathbf{T}\| = \sqrt{T_x^2 + T_y^2 + T_z^2}$ is the magnitude of the thrust vector $\mathbf{T} = [T_x, T_y, T_z]^T$, and g_0 is the gravitational acceleration at sea level. The thrust vector is linked to the acceleration command, according to the conventional thrust-to-mass ratio $a_c = \mathbf{T}/m$ relationship.

III. Nonlinear Asteroid Landing Guidance Laws Development

A. Sliding Control Theory for Systems with Higher Relative Degree

The sliding control methodology can be best defined as an elementary approach to robust control [22]. The methodology relies on the fundamental principle that it is much easier to control nonlinear and uncertain first-order systems (i.e., systems described by first-order differential equations) than n th-order systems (i.e., systems described by n th-order differential equations). The key element behind the sliding theory is the ability to replace or transform an n th-order problem into a first-order problem. If such transformation exists, it can be shown that, for the derived problem, perfect performance can be achieved in spite of the system parameters uncertainty.

Consider the following single-input/single-output (SISO) n th-order dynamic system:

$$\frac{d^n}{dt^n} x = f(x) + b(x)u \quad (10)$$

Here, x is the scalar output, u is the control variable, and $\mathbf{x} = [x, \dot{x}, \dots, x^{(n-1)}]^T$ is the state vector. Both the nonlinear plant

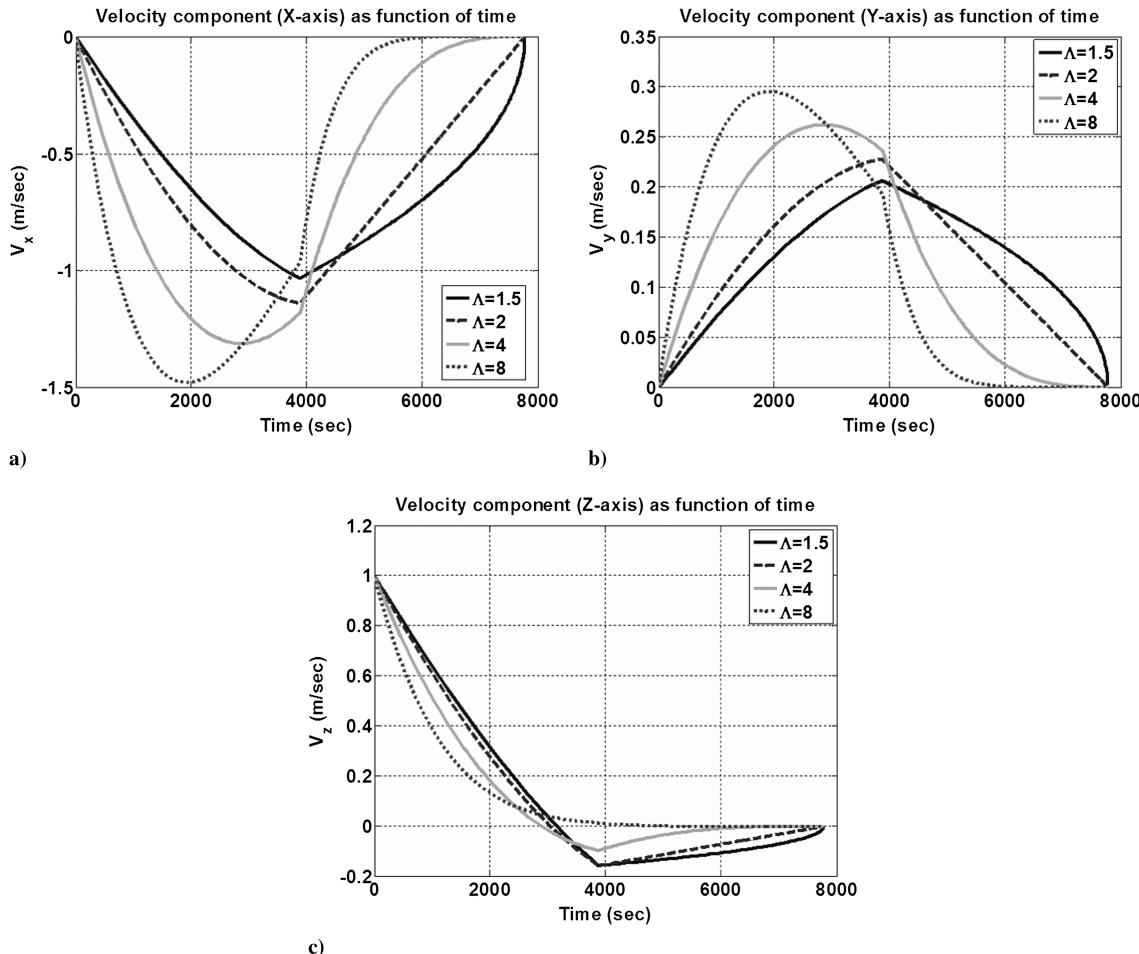


Fig. 2 Parametric analysis of the MSSG algorithm for various guidance gains Λ .

dynamics $f(\mathbf{x})$ and the control gain $b(\mathbf{x})$ are assumed to be not exactly known. Under the condition that both $f(\mathbf{x})$ and $b(\mathbf{x})$ have a known upper bound, the sliding control goal is to get the state \mathbf{x} to track the desired state $\mathbf{x}_d = [x_d, \dot{x}_d, \dots, \dot{x}_d^{(n-1)}]^T$ in the presence of model uncertainties. The time-varying sliding surface is introduced as a function of the tracking error $\tilde{\mathbf{x}} = \mathbf{x} - \mathbf{x}_d = [\tilde{x}, \dot{\tilde{x}}, \dots, \dot{\tilde{x}}^{(n-1)}]^T$ by the following scalar equation:

$$s(\mathbf{x}, t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} \tilde{x} = 0 \quad (11)$$

Here, λ is a strictly positive constant. For example, if $n = 2$, we obtain

$$s(\mathbf{x}, t) = \dot{\tilde{x}} + \lambda \tilde{x} = 0 \quad (12)$$

Under the preceding conditions, the general tracking problem is reduced by forcing the dynamic system in Eq. (10) to remain on the time-varying sliding surface described by Eq. (11). Using this approach, the specified SISO control problem is greatly simplified because the tracking of an n -dimensional vector \mathbf{x}_d has been reduced to the problem of keeping the scalar sliding surface at zero [i.e., the problem has been reduced to a first-order stabilization problem in the sliding surface $s(\mathbf{x}, t)$]. The simplified first-order stabilization problem can be achieved by selecting a control law such that, outside the sliding surface $s(\mathbf{x}, t)$, the following condition is satisfied:

$$\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s| \quad (13)$$

Here, η is a strictly positive constant. Equation (13), also called the "sliding condition," explicitly states that the distance from the sliding surface decreases exponentially along all system trajectories. Gen-

erally, constructing a control law that satisfies the sliding condition is fairly straightforward. For example, using the Lyapunov direct method [22,23], one can select a candidate Lyapunov function as follows:

$$V(s) = \frac{1}{2} s^T s \quad (14)$$

Equation (14) satisfies the following two conditions: $V(0) = 0$ and $V(s) > 0$ for $s > 0$. By taking the derivative of Eq. (14), and applying the sliding condition [Eq. (13)], it is easily verified that the derivative of the selected Lyapunov function is negative everywhere (i.e., the dynamic system is globally stable). The control law is generally obtained by substituting the sliding control definition, Eq. (11), and the system dynamic equations, Eq. (10), into Eq. (13). It is worth mentioning that, if the system is multi-input multi-output (MIMO), i.e. both controller and output states are vectors, the sliding surface becomes a vector. Generally, the sliding vector contains as many components as the controller (e.g., see Slotine and Li [22]). Here, we anticipate that, for our guidance development, any defined sliding surface will be a three-dimensional vector [e.g., Eq. (15) in the next section].

Generally, constraining the system to "slide" on the surface defined by Eq. (11) can be achieved only at the price of higher control activity. This represents one major drawback of the sliding methodology because high-frequency control switching may cause undesirable chattering. The conventional sliding methodology can be applied only if the system is of relative degree one [i.e., the controller explicitly appears on the first derivative of the sliding surface (Eq. (11))]. If the dynamic system has a higher relative degree, the application of a high-order sliding control (HOSC) mode can be an effective way to eliminate chattering, yet maintain closed-loop robustness in a highly uncertain environment. Here, the following definition is introduced [19,24,25]:

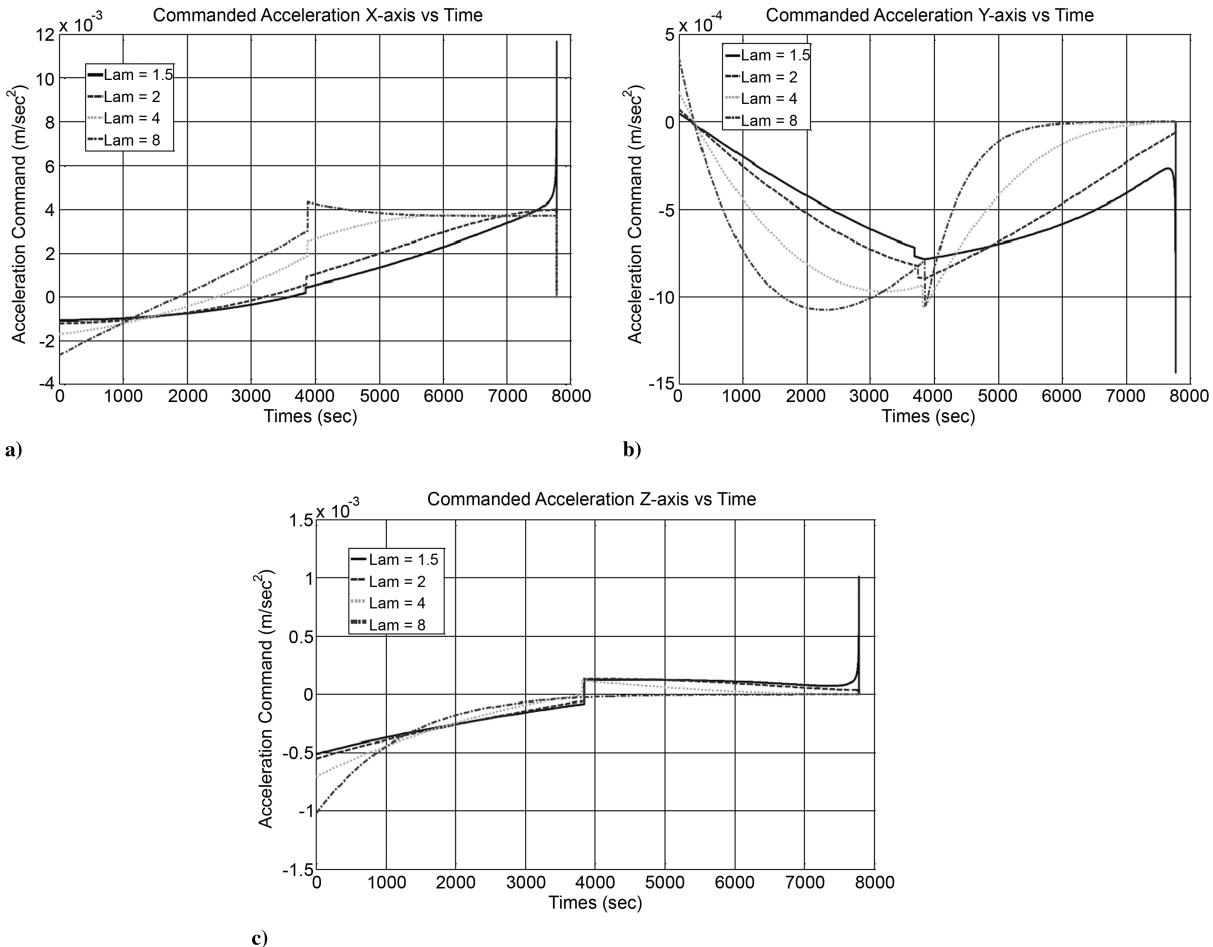


Fig. 3 Parametric analysis of the MSSG algorithm: effect of the guidance gain Λ on the thrust history during the guided descent.

Consider a smooth dynamic system with a smooth output $s(\mathbf{x}, t)$ (sliding function). Then, provided that $s, \dot{s}, \ddot{s}, \dots, s^{r-1}$ are continuous and that $s = \dot{s} = \ddot{s} = \dots = s^{r-1} = 0$, the motion on the set $\{s, \dot{s}, \ddot{s}, \dots, s^{r-1}\} = \{0, 0, 0, \dots, 0\}$ is said to exist on an r -sliding mode.

In the dynamic problem of descending and landing on an asteroid, as well as in many other close-proximity operations around small bodies, the acceleration command shows up at the second derivative of a properly defined sliding vector. Consequently, two-sliding control principles can be applied to take advantage of such properties and eliminate or attenuate chattering. Recently, HOSC has been one of the central topics of modern nonlinear control theory. Indeed, asymptotically stable higher-order sliding modes appear naturally in systems that are traditionally treated with conventional sliding-mode control [26]. Whereas theoretical studies on the finite convergence properties of arbitrary-order sliding-mode control are currently underway [25], two-sliding controllers have already been applied in practical problems of interest in space and aerospace applications, including missile guidance [27,28], reentry terminal guidance [29], as well as lunar landing guidance [30].

Harl and Balakrishnan [29] set the stage on applying HOSC principles for terminal landing guidance problems. The key point is to guarantee that both the sliding surface and its derivative will reach zero in a finite time, while ensuring that the sliding surface will not cross zero until the final time, in contrast with the approach described by Levant [19], where two-sliding homogeneous control can “twist” around the sliding surface, zeroing it out in a finite time. However, Levant’s approach is not suitable for guidance applications because the problem is considered over when the sliding surface is crossed.

Notably, the idea of devising robust guidance algorithms such that the sliding surface is reached in a finite time for the first time at the landing location can be effectively employed using standard sliding-mode control as demonstrated for lunar landing [30]. The application of such methodology for the problem of asteroid landing guidance is investigated in the following sections.

B. Multiple Sliding Surface Guidance Design for Asteroid Powered Descent

The goal of this section is to develop a novel nonlinear guidance approach for asteroid powered descent and landing that employs the recent advancements in HOSC theory. The proposed guidance law, named the MSSG algorithm, is built on the basic principles of two-sliding control mode [19,29]. The overall objective is to derive a guidance law (acceleration command) that is 1) robust against unmodeled dynamics and 2) guarantees higher performance (i.e., accuracy in targeting with zero velocity) as required by stringent precision requirements. The guidance model employed to develop the guidance algorithm is the three-degree-of-freedom (3-DOF) model described by Eqs. (1–8) assuming $a_p = 0$. For the class of sliding surfaces that are of interest to the asteroid descent and landing guidance problem, the dynamics of the sliding system has relative degree two (i.e., the acceleration command appears at the second derivative of the sliding surface vector). Let us define the first sliding vector surface in the following way:

$$\mathbf{s}_1 = \mathbf{r}_L - \mathbf{r}_{Ld} \quad (15)$$

Here, \mathbf{r}_{Ld} is the position of the desired (target) landing point on the asteroid surface. Taking the derivative of \mathbf{s}_1 with respect to time, one obtains:

$$\dot{\mathbf{s}}_1 = \dot{\mathbf{r}}_L - \dot{\mathbf{r}}_{Ld} = \mathbf{v}_L - \mathbf{v}_{Ld} \quad (16)$$

Here, \mathbf{v}_{Ld} is the desired landing velocity (set to zero for soft landing). The guidance problem can be formulated as a standard control problem: Find the acceleration command law such that, in a finite time t_F , $\mathbf{s}_1 \rightarrow 0$ and $\dot{\mathbf{s}}_1 \rightarrow 0$. It is easily verified that, for landing problems, the sliding surface is of relative degree two:

$$\ddot{\mathbf{s}}_1 = \dot{\mathbf{v}}_L = 2\boldsymbol{\omega} \times \mathbf{v}_L + \boldsymbol{\omega} \times \boldsymbol{\omega} \times \mathbf{r}_L + \frac{\partial V^T}{\partial r} + \mathbf{a}_C \quad (17)$$

The goal is achieved by setting $\dot{\mathbf{s}}_1$ as a virtual controller and using a backstepping approach. More specifically, $\dot{\mathbf{s}}_1$ is found such that the first sliding surface is driven to zero in a finite time. The virtual controller can be conveniently selected as follows:

$$\dot{\mathbf{s}}_1 = -\frac{\boldsymbol{\Lambda}}{(t_F - t)} \mathbf{s}_1 \quad (18)$$

Here, $\boldsymbol{\Lambda} = \text{diag}\{\Lambda_1, \Lambda_2, \Lambda_3\}$ is a diagonal matrix of guidance gains. To drive the first sliding surface to zero, the virtual controller $\dot{\mathbf{s}}_1$ must be globally stable. Global stability of $\dot{\mathbf{s}}_1$ can be shown by choosing the following candidate Lyapunov function:

$$V_1 = \frac{1}{2} \mathbf{s}_1^T \mathbf{s}_1 \quad (19)$$

V_1 has the following properties:

$$\begin{aligned} V_1(0) &= 0 \quad \text{if } \mathbf{s}_1 = 0 \\ V_1(\mathbf{s}_1) &> 0 \quad \forall \mathbf{s}_1 \neq 0 \\ V_1(\mathbf{s}_1) &\rightarrow \infty \quad \text{if } \mathbf{s}_1 \rightarrow \infty \end{aligned} \quad (20)$$

In addition, for stability, the time derivative of V_1 must be negative definite everywhere. Imposing positive guidance gains $\{\Lambda_1, \Lambda_2, \Lambda_3\} > 0$ and setting $\mathbf{s}_1 = \{s_{1i}\}_{i=1}^3$, one obtains

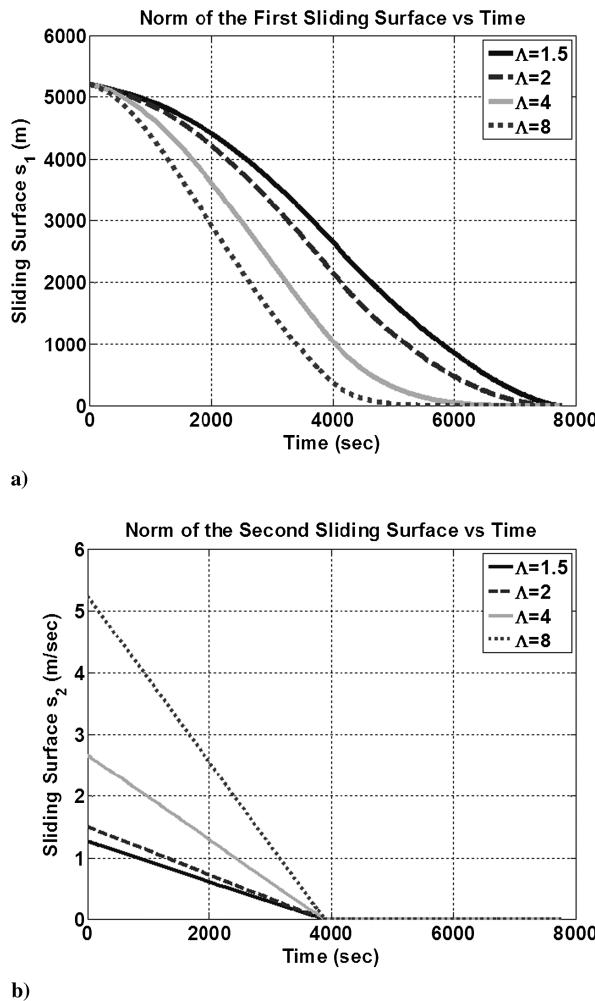


Fig. 4 Norm of the first and second surfaces as a function of time with the guidance gain Λ as a parameter.

$$\begin{aligned}\dot{V}_1 &= s_1^T \dot{s}_1 = -\frac{1}{(t_F - t)} s_1^T \Lambda s_1 \\ &= -\frac{1}{(t_F - t)} (\Lambda_1 s_{11}^2 + \Lambda_2 s_{12}^2 + \Lambda_3 s_{13}^2) < 0\end{aligned}\quad (21)$$

However, to ensure that both the sliding surface and its derivative approach zero in a finite time, it is generally desirable for the matrix gains to be all greater than one. Indeed, the time variation of the sliding surface vector s_1 can be explicitly derived as a function of the guidance gains. Applying separation of variables to Eq. (18), one obtains

$$\frac{ds_{1i}}{s_{1i}} = -\frac{\Lambda_i dt}{t_F - t} \quad (22)$$

where $i = 1, 2, 3$ are the components of the sliding surface vector. Equation (18) can be formally integrated:

$$\ln(s_{1i}) = \Lambda_i \ln(t_F - t) + C_i \quad (23)$$

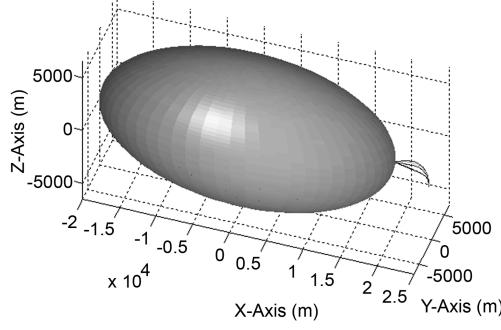
By imposing the initial conditions $s_1(0) = s_{10}$ and taking the exponential of both sides, the solution becomes

$$s_{1i}(t) = s_{1i}(t_F - t)^{\Lambda_i} \quad (24)$$

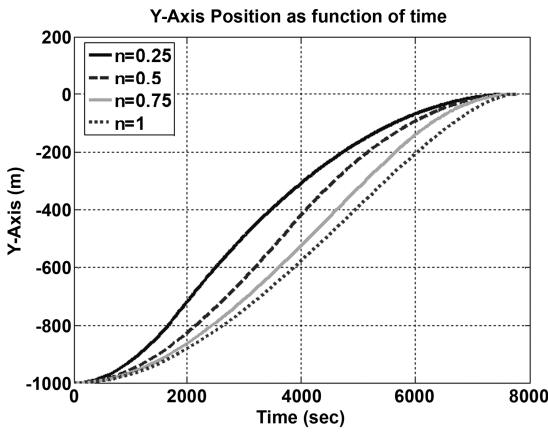
or in vector form

$$s_1(t) = s_{10}(t_F - t)^{\Lambda} \quad (25)$$

The derivative of the sliding surface vector can also be computed explicitly:



a)



c)

$$\dot{s}_{1i}(t) = \Lambda_i s_{1i}(t_F - t)^{\Lambda_i - 1} \quad (26)$$

or in vector form

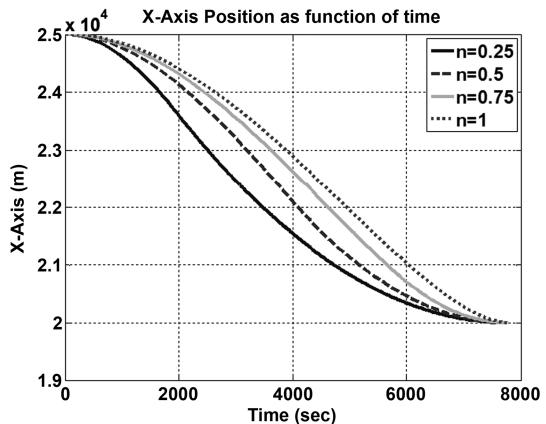
$$\dot{s}_1(t) = \Lambda s_{10}(t_F - t)^{\Lambda - I} \quad (27)$$

As can be seen from Eq. (24), as long as $\Lambda_i > 0$ ($i = 1, 2, 3$), the sliding surface vector will reach zero in finite time. However, if $\Lambda_i < 1$ ($i = 1, 2, 3$), the derivative of the sliding surface vector blows up for $t = t_F$. Therefore, if the matrix gains are selected such that $\Lambda_i > 1$ ($i = 1, 2, 3$), both the sliding surface vector and its derivative go to zero as $t \rightarrow t_F$.

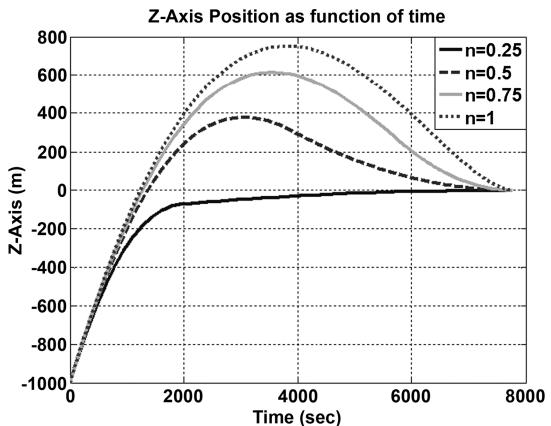
At the time when the descent maneuver is initiated, the spacecraft is generally characterized by position and velocity such that Eq. (18) is not satisfied. A meaningful guidance law requires \dot{s}_1 to be explicitly linked to the acceleration command a_C , which must be executed to drive both s_1 and \dot{s}_1 to zero. The correct acceleration command that drives \dot{s}_1 from its original state to a trajectory defined by Eq. (18) can be obtained by employing a second sliding surface vector s_2 . In addition, the commanded acceleration must be able to keep the system on the second surface until $s_1, \dot{s}_1 \rightarrow 0$ is achieved. Consequently, the second sliding surface vector is defined in the following way:

$$s_2 = \dot{s}_1 + \frac{\Lambda}{(t_F - t)} s_1 = 0 \quad (28)$$

The new sliding surface vector s_2 is of relative degree one with respect to the acceleration command. Indeed, it can be easily verified that the acceleration command appears explicitly within the expression of the first derivative of s_2 :



b)



d)

Fig. 5 Parametric analysis of the MSSG algorithms: effect of the parameter n on the trajectory history.

$$\ddot{s}_2 = \ddot{s}_1 + \frac{\Lambda}{(t_F - t)} \dot{s}_1 + \frac{\Lambda}{(t_F - t)^2} s_1 \quad (29)$$

Using Eq. (17), it is explicitly found that

$$\begin{aligned} \dot{s}_2 &= 2\omega \times v_L + \omega \times \omega \times r_L + \frac{\partial V^T}{\partial r} \\ &\quad + a_c(t) + \frac{\Lambda}{(t_F - t)} \dot{s}_1 + \frac{\Lambda}{(t_F - t)^2} s_1 \end{aligned} \quad (30)$$

The desired acceleration command $a_c(t)$ (guidance law) is determined using the Lyapunov direct method. Here, second Lyapunov candidate function V_2 is defined as follows:

$$V_2 = \frac{1}{2} s_2^T s_2 \quad (31)$$

V_2 satisfies conditions similar to the one defined for V_1 [see Eq. (20)]. Moreover, its time derivative can be explicitly computed as follows:

$$\begin{aligned} \dot{V}_2 &= s_2^T \dot{s}_2 = s_2^T \left\{ 2\omega \times v_L + \omega \times \omega \times r_L \right. \\ &\quad \left. + \frac{\partial V^T}{\partial r} + a_c(t) + \Lambda \frac{(t_F - t)\dot{s}_1 + s_1}{(t_F - t)^2} + \Phi sgn(s_2) \right\} \end{aligned} \quad (32)$$

The acceleration command can be selected accordingly:

$$\begin{aligned} a_c(t) &= - \left\{ 2\omega \times v_L + \omega \times \omega \times r_L \right. \\ &\quad \left. + \frac{\partial V^T}{\partial r} + \Lambda \frac{(t_F - t)\dot{s}_1 + s_1}{(t_F - t)^2} + \Phi sgn(s_2) \right\} \end{aligned} \quad (33)$$

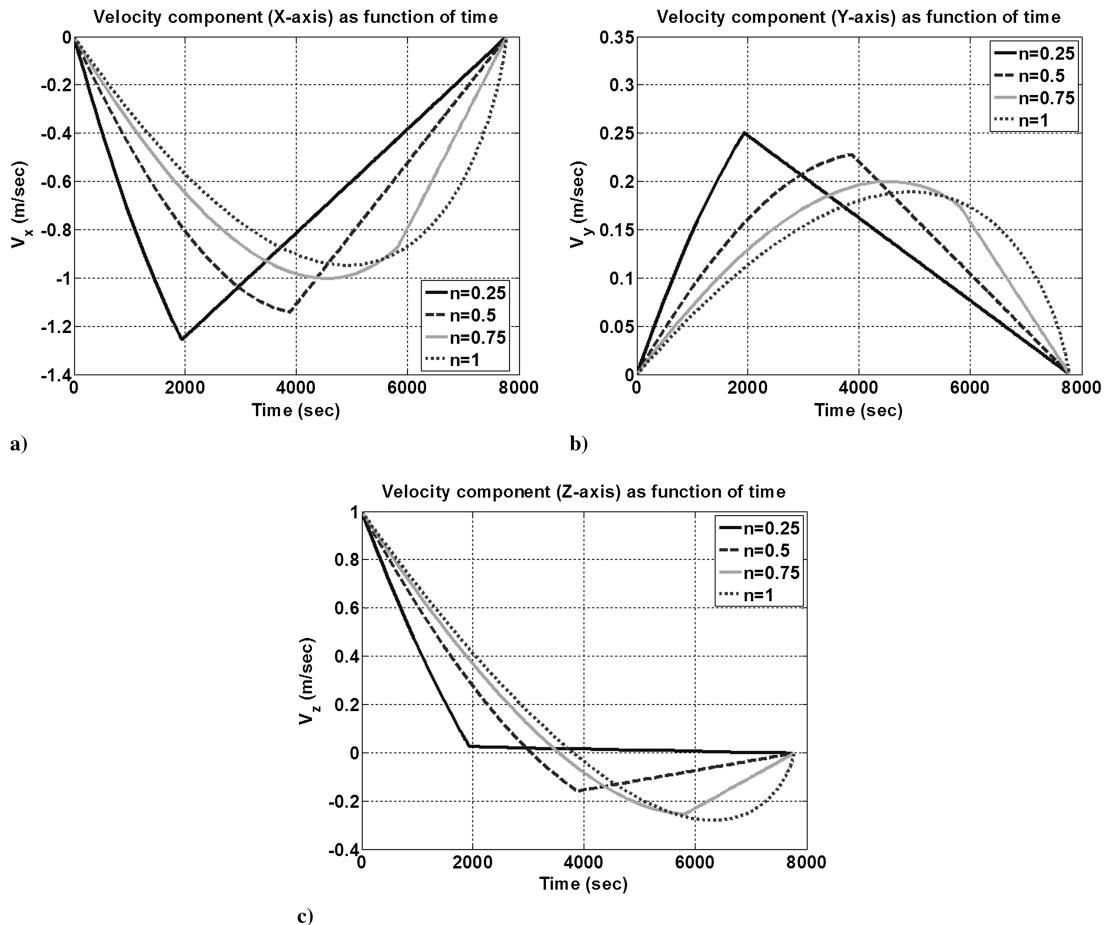


Fig. 6 Parametric analysis of the MSSG algorithm for various values of n .

Equation (33) is what we named MSSG. The matrix coefficients $\Phi = \text{diag}\{\Phi_1, \Phi_2, \Phi_3\}$ are given as follows:

$$\Phi_i = \frac{s_{2i}(0)}{t_F^*} \quad (34)$$

Using Eq. (34), one can show that the second sliding surface vector is driven to zero in a finite time $t_F^* < t_F$. In fact, by replacing the guidance law explicitly derived in Eq. (33) into Eq. (30), the dynamics of the second sliding surface vector becomes

$$\dot{s}_2 = -\Phi sgn(s_2) \quad (35)$$

Noting that s_2 does not change sign before reaching zero, Eq. (35) can be integrated between zero and t to yield

$$s_{2i}(t) = s_{2i}(0) - \frac{|s_{2i}(0)|}{t_F^*} t \quad (36)$$

Clearly, the second sliding surface vector goes to zero as $t \rightarrow t_F^*$. The derived MSSG law is globally stable. Inserting Eq. (33) into Eq. (32) and augmenting the equations of motion to account for the perturbing acceleration, the derivative of V_2 has the following expression:

$$\dot{V}_2 = s_2^T \dot{s}_2 = s_2^T \{a_p(t) - \Phi sgn(s_2)\} < 0 \quad (37)$$

The time derivative of the second Lyapunov function is always less than zero if an upper bound for the perturbing acceleration a_p^{MAX} is available. In such a case, the matrix coefficients Φ can be selected such that $\Phi_i > |a_p^{\text{MAX}}|$. The second Lyapunov function is therefore decrescent and, by virtue of the Lyapunov stability theorem for

nonautonomous systems, $s_2 \rightarrow 0$ as $t \rightarrow t_F^*$. Consequently, $s_1, \dot{s}_1 \rightarrow 0$ as $t \rightarrow t_F$.

As pointed out by Harl and Balakrishnan [29], the adaptive nature of the guidance law is such that the system cannot be maintained on the first sliding surface for $t > t_F$ [see Eqs. (24–27)]. However, for the soft landing guidance problem, the latter is a nonissue because the problem is over as soon as the system reaches the final time (landing conditions).

IV. Multiple Sliding Surface Guidance Implementation and Performance Analysis

The goal of this section is to implement the MSSG algorithm in a simulation environment to 1) evaluate the behavior of the class of closed-loop landing trajectories guided by the proposed MSSG algorithm as a function of guidance parameters; 2) explore the fuel efficiency of the proposed guidance law; and 3) integrate the continuous MSSG-based acceleration command in a pulse-width pulse-frequency (PWPF) modulation scheme and run a set of Monte Carlo simulations under realistic conditions to evaluate the landing accuracy of the proposed guidance law.

A. Parametric Analysis of the Guidance Algorithm

The first step toward validation and verification of the proposed algorithm is to analyze the behavior of the landing trajectories guided by the derived MSSG law as a function of the selected guidance parameters. The 3-DOF model described by Eqs. (1–8) has been implemented in a MATLAB simulation environment to generate MSSG-guided landing trajectories.

The analysis of the guidance algorithm is conducted on a flight dynamics scenario simulating the powered landing descent on an Eros-like asteroid. The gravitational field is modeled assuming that the asteroid is a perfect triaxial ellipsoid with semi-axis ($a > b > c$)

such that $a = 20,000$, $b = 7000$, and $c = 6500$ m. The density ρ of the asteroid is assumed to be constant and equal to 3200 kg/m^3 . Tri-axial ellipsoids are often reasonable approximations to small-bodies and provide with a closed form solution for the gravitational potential as reported by MacMillan [21] and Scheeres [31]. Once the expression for the analytical potential is known, the gravity acceleration vector can be determined as a function of the position by directly deriving its gradient in closed form [31]. Finally, the asteroid rotation rate has been set to be $\|\omega\| = 3.3118 \times 10^{-4}$ rad/s, which is consistent with an Eros rotation period of 5 h and 16 min [32].

For this set of simulations, it is assumed that the spacecraft is in a near-hoovering state above the asteroid surface, in a location identified by the vector $\mathbf{r}_L(t_0) = [25,000, -1000, -1000]^T$ m. The latter identifies the coordinates of the spacecraft with respect to a body-fixed reference frame, having its origin at the center of the ellipsoid and the axis oriented along its principal axis. In the same coordinate frame, the initial velocity of the spacecraft is assumed to be $\mathbf{v}_L(t_0) = [0, 0, -1]^T$ m/s. The guidance algorithm is required to drive the spacecraft to a surface point located at an equatorial region, with coordinates specified by $\mathbf{r}_{Ld}(t_F) = [20,000, 0, 0]^T$ m and zero final velocity (soft landing). The spacecraft is assumed to have an initial (wet) mass of 500 kg. It is also assumed that the spacecraft is configured such that the attitude is kept fixed and the thrusters are oriented to generate an independent thrust-based acceleration in each of the three directions of the asteroid-centered, body-fixed reference frame. For this initial analysis, it is also assumed that each of the thrusters generates a continuous thrust. The guidance law formulation is implemented in the simulation platform according to Eq. (33). The algorithm depends on the available gravitational model as well as on five guidance parameters (i.e., t_F , t_F^* and $\Lambda = \text{diag}\{\Lambda_1, \Lambda_2, \Lambda_3\}$), where, t_F^* has been parameterized as a function of the time of flight according to $t_F^* = nt_F$, for $0 < n < 1$.

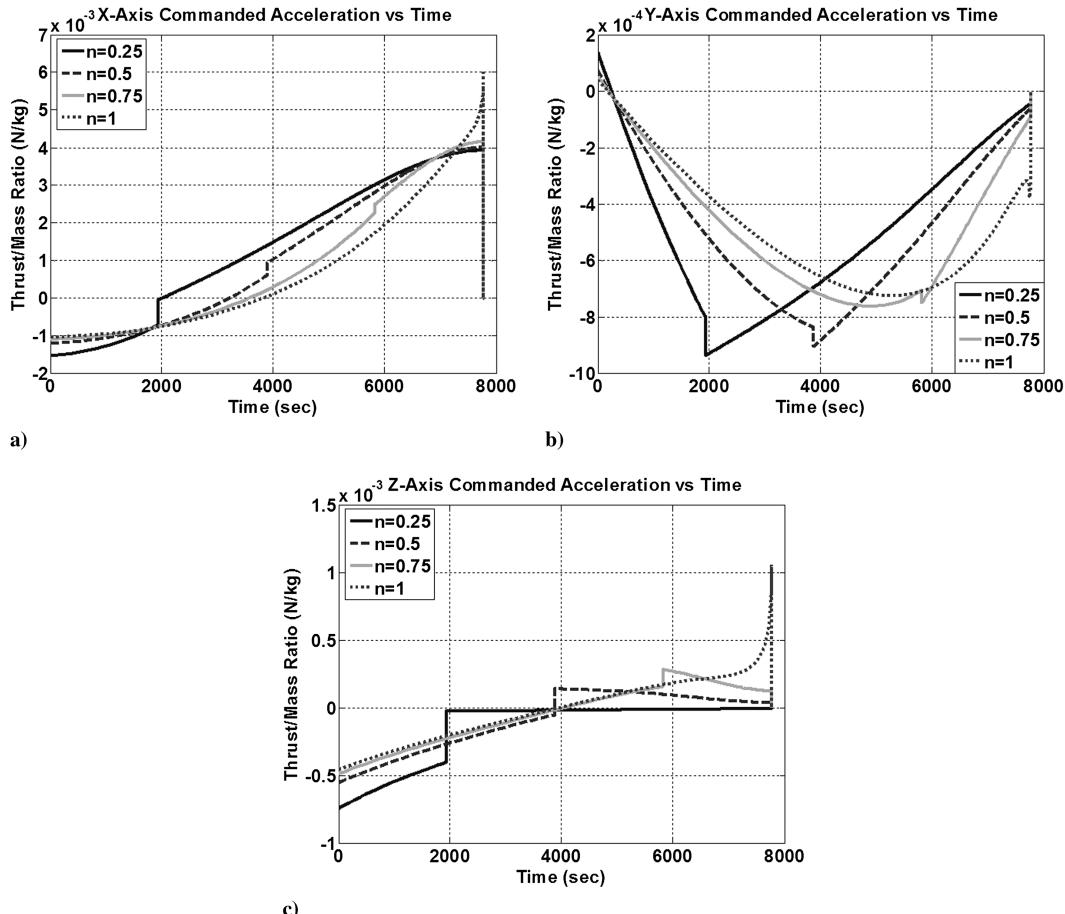


Fig. 7 Parametric analysis of the MSSG algorithm: effect of the parameter n on the thrust history during the guided descent.

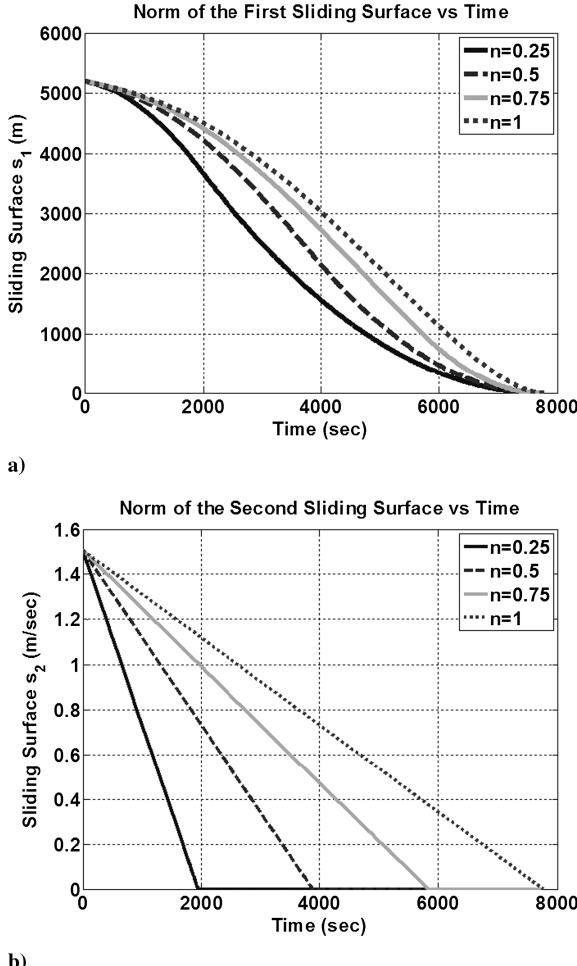


Fig. 8 Norm of the first and second surfaces as a function of time with n as a parameter.

1. Effects on Performance due to Λ and t_F^*

The first set of simulations is directed at analyzing the effect of the guidance matrix Λ on the guided trajectories. For this specific case, t_F and t_F^* are fixed. The simulations are initiated at time $t = 0$ and conducted until a fixed final time selected to be $t_F = 7778.3$ s. Assuming that the guidance matrix is spherical (i.e., $\Lambda = \text{A} \text{diag}\{1, 1, 1\}$), the trajectory and the acceleration command histories depend on only one single guidance gain Λ as well as n . Assuming a fixed value of $n = 0.5$ (i.e., the second sliding surface is reached in half of the total flight time), the guidance gain Λ is varied parametrically. Four MSSG-guided trajectories have been simulated with Λ equal to 1.5, 2, 4, and 8, respectively. Figure 1 shows the three-dimensional (3-D) guided descent and landing trajectories, as well as the history of the position components along the asteroid-centered body-fixed coordinates. Figure 2 shows the history of the three velocity components. Figure 3 shows the history of the three components of the acceleration command. The MSSG algorithm generates trajectories that can be subdivided in two phases. In the first phase of the flight ($0 < t < t_F^*$), the acceleration command drives the second sliding surface to zero. Once the second surface is reached ($t = t_F^*$), the second phase is initiated. During this phase, the first

surface is driven to zero according to the nonlinear first-order dynamics dictated by Eq. (18). For each of the three components, the magnitude of the acceleration command tends to increase with the parameter Λ , which regulates the rate at which the first surface is reached. Although the time of flight is fixed (i.e., the first surface is reached exactly at $t = t_F$), the rate of convergence depends on Λ . Figure 4 shows the time history of the first and second sliding surface (Figs. 4a and 4b, respectively). The norm of the second sliding surface is driven to zero at $t = t_F^*$ and maintained within a prescribed tolerance for the rest of the flight. The effect of the rate of convergence can be clearly seen in Fig. 4a, in which the guidance parameter influences the shape of the sliding surface norm. All cases precisely achieved the desired location with a very low terminal landing velocity, but the performances degraded for lower values of Λ . Indeed, it is observed that, whereas the three components of the terminal velocity are always less than 0.0015 m/s for $\Lambda = 2, 4$, and 8, the values of the terminal velocity components are close to 0.02 m/s if $\Lambda = 1.5$. In this specific case, the rate of convergence of the sliding surface is slower and the MSSG algorithm tends to generate, on average, acceleration commands of lower magnitude during the flight. For a fixed time of flight, the errors tend to be larger toward the end. Consequently, the MSSG algorithm tries to compensate by increasing the terminal acceleration command. Despite such guidance command reaction (see Fig. 3), the accuracy is slightly degraded.

The second set of simulations is executed to analyze the behavior of MSSG-guided landing trajectories as function of the parameter n (i.e., the time at which the second sliding surface is achieved). In this case, both final time and guidance gain are kept fixed at values $t_F = 7778.3$ s and $\Lambda = 2$ s, respectively. The parameter n assumes four possible values (i.e., 0.25, 0.5, 0.75, and 1.0). Figure 5 shows the 3-D guided descent and landing trajectories, as well as the history of the position components along the asteroid-centered body-fixed coordinates. Figure 6 shows the history of the three velocity components. Figure 7 shows the history of the three components of the acceleration command, and Fig. 8 shows the histories of the norm of the two sliding surfaces. It is important to note that, t_F^* has a strong influence in both shaping the trajectory and on the overall acceleration magnitude. As seen in the previous case, the components of the acceleration command are generally discontinuous at $t = t_F^*$. If $t_F^* = t_F$ the acceleration exhibits a smooth behavior during the powered descent phase and drastically increases close to the final time. In this case, the two sliding surfaces are reached at the same time (i.e., at the desired landing point).

B. Guidance Parameters Selection Analysis: Multiple Sliding Surface Guidance Fuel Performance

To shed light on how to properly select the guidance parameters, an MSSG fuel-efficiency study has been conducted. Here, the goal is to understand how the propellant mass required for landing varies as a function of the guidance parameters. To this end, MSSG-based closed-loop trajectories are compared with an open-loop, fuel-optimal landing guidance solution. The latter is determined by finding an acceleration program that, starting from a selected initial condition, lands the spacecraft at a desired location using the minimum amount of propellant. Optimal trajectories for asteroid soft landing have been studied and are shown to be computationally intensive [33]. For the sake of comparison, in our analysis, we have selected an open-loop powered descent profile, in which the spacecraft descent is initiated exactly above the selected (Eros-like) asteroid north pole with the following initial conditions: $\mathbf{r}_L(t_0) = [0, 0, 7300]^T$ m, $\mathbf{v}_L(t_0) = [0, 0, -1]^T$ m/s, and terminates at the final desired condition defined by $\mathbf{r}_{Ld}(t_F) = [0, 0, 6500]^T$ m (north-pole landing) with zero velocity. The initial and final conditions have been selected to simplify the problem and obtain an optimal guided descent that occurs only in the vertical dimension. The problem has been solved numerically using pseudospectral methods instantiated via the general pseudospectral optimal control software (GPOPS [34]). Details of the problem formulation and implementation are reported in the Appendix.

Table 1 Comparison of MSSG and optimal solution fuel usage as a function of Λ

	Optimal, kg	$\Lambda = 2$ s, kg	$\Lambda = 4$ s, kg	$\Lambda = 8$ s, kg	$\Lambda = 10$ s, kg
$t_F^* = 0.5t_F$	1.01	1.39	1.87	2.30	2.42
$t_F^* = 0.75t_F$	1.01	1.20	1.57	1.91	2.00
$t_F^* = t_F$	1.01	1.11	1.43	1.71	1.79

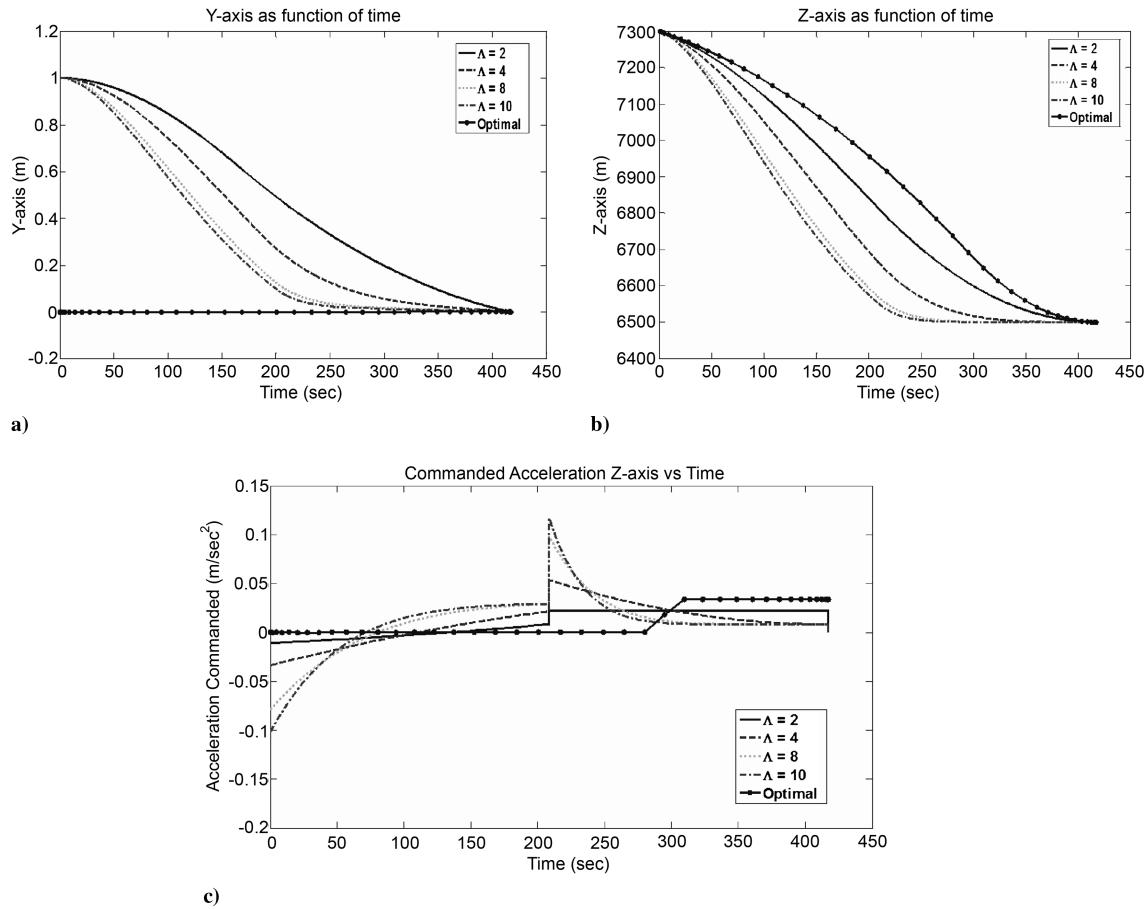


Fig. 9 Comparison of MSSG trajectories and acceleration command history to optimal solution for $n = 0.5$.

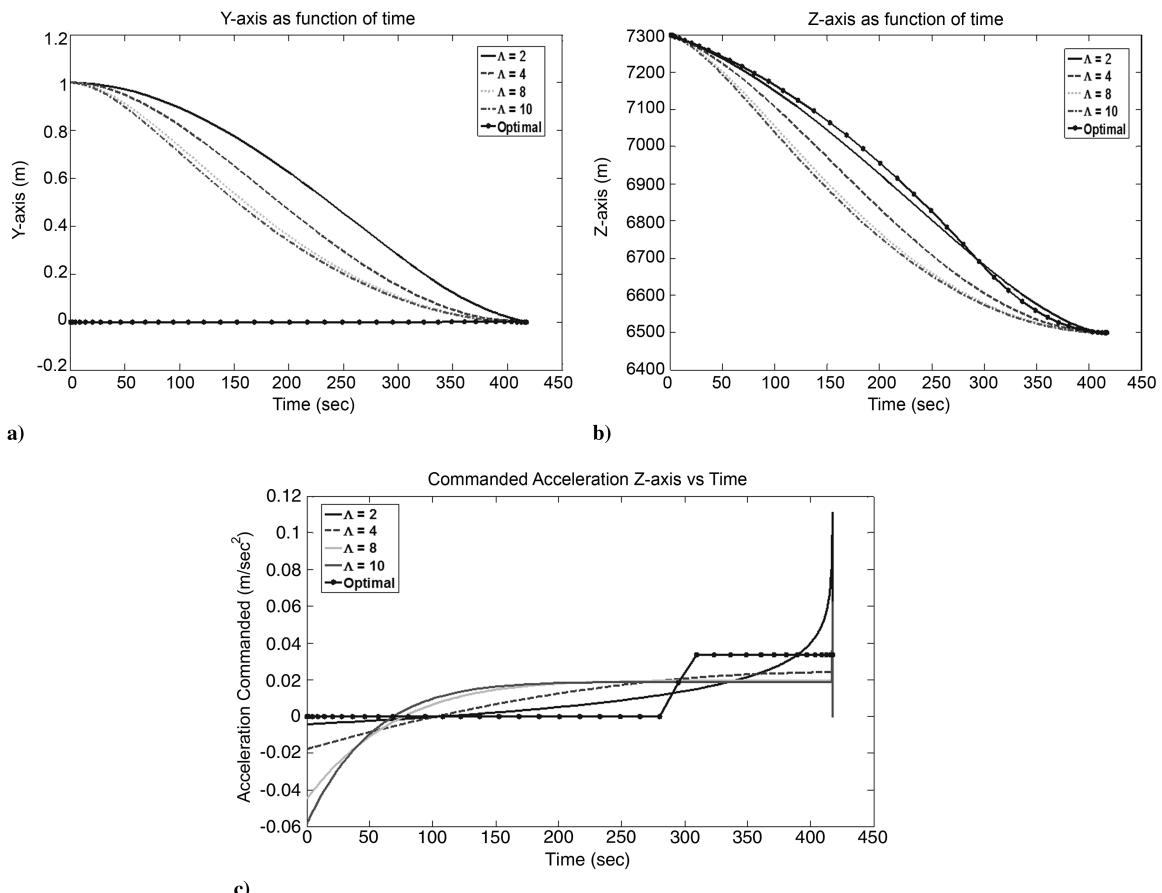
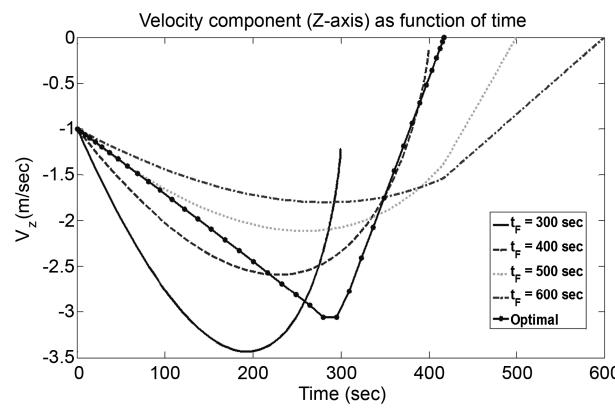


Fig. 10 Comparison of MSSG trajectories and acceleration command history to optimal solution for $n = 1$.

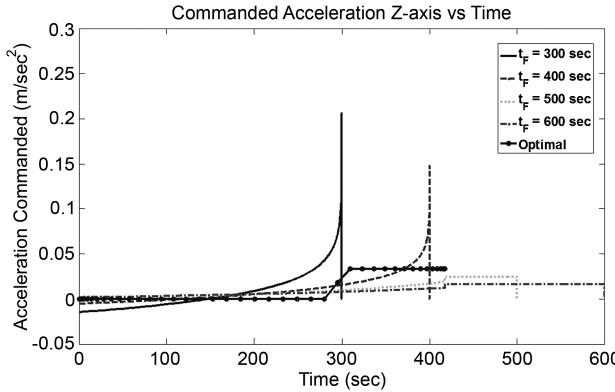
Table 2 Comparison of MSSG and optimal solution fuel usage as a function of t_F

	Optimal, kg	$t_F = 300$ s, kg	$t_F = 400$ s, kg	$t_F = 500$ s	$t_F = 600$ s
$\Lambda = 2$ s	1.01	1.07	1.09	1.16	1.34
$\Lambda = 10$ s	1.01	1.80	1.78	1.76	1.79

A set of MSSG-guided trajectories starting very close to the optimal descent problem have been generated and the mass of propellant consumed during the closed-loop descent recorded. All MSSG simulations are implemented with the following initial conditions: $\mathbf{r}_L(t_0) = [0, 0, 7300]^T$ m and $\mathbf{v}_L(t_0) = [0, 0, -1]^T$ m/s. The guidance algorithm is required to drive the spacecraft to the asteroid north pole with zero velocity for a variety of guidance parameters. For the initial set of simulations, the time of flight is set to be equal to the optimal value obtained via GPOPS. Table 1 shows the mass of propellant employed by the MSSG trajectories with fixed $t_F = 417.3$ s for a set of three t_F^* ($n = 0.5, 0.75, 1$) and four guidance gains ($\Lambda = 2, 4, 8, 10$ s). For this specific case, the vertical minimum-fuel, open-loop powered descent requires a mass of propellant equal to 1.01 kg. As seen in Table 1, the best MSSG fuel performances are obtained for lower guidance gains Λ and $t_F^* = t_F n = 1$. The best case is achieved for $\Lambda = 2$, requiring 10% more propellant with respect to the optimal case. As seen in the preceding section, guidance gains lower than 2 s are not recommended because it may result in performance degradation (e.g., higher impact velocity). Figure 9 shows the trajectory histories in the y and z directions, as well as the vertical acceleration commands as compared with the optimal solution for $t_F^* = 0.5t_F$ ($n = 0.5$). Consistent with the analysis reported in the preceding section, the $\Lambda = 2$ s case exhibits a discontinuous behavior exactly when the second sliding surface is achieved. Nevertheless, the acceleration command spike is less than the one observed for the higher guidance gains, resulting in less fuel consumption. Figure 10



a)

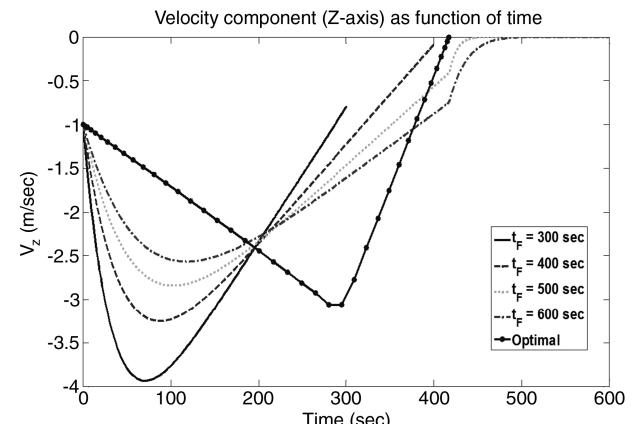


b)

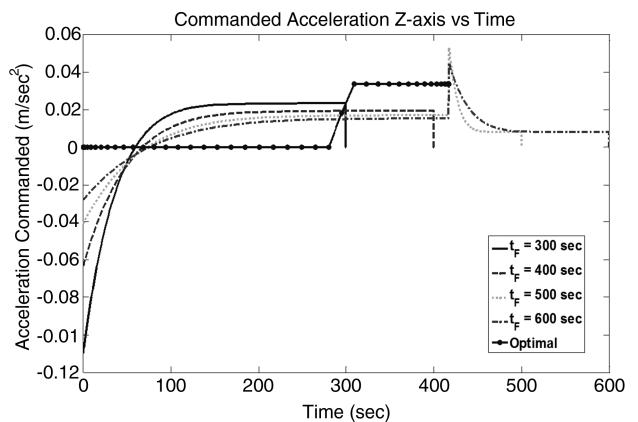
Fig. 11 Comparison of velocity and acceleration command histories of MSSG and optimal solution for $\Lambda = 2$.

shows the trajectory histories in the y and z directions, as well as the vertical acceleration commands as compared with the optimal solution for $t_F^* = t_F$ ($n = 1$). Here, the vertical acceleration has values close to zero for most of the descent and dramatically increases as the final location is approached. Despite the final spike, such guidance parameters achieve the best MSSG fuel consumption.

Next, a second set of simulations has been implemented to analyze the impact of t_F on fuel efficiency. Here, the parameter n has been kept constant and equal to one, whereas four values for t_F (300, 400, 500, 600) s and two values for Λ (2, 10) s have been considered. Table 2 shows the mass of propellant employed to drive the spacecraft to the landing location via MSSG. Figure 11 shows the vertical velocity and acceleration command histories as compared with the optimal solution for $\Lambda = 2$ s. It is observed that, if t_F is selected to be less than the optimal value, MSSG fuel efficiency increases (e.g., 1.07 kg for $t_F = 300$ s) but exhibits a significant degradation of performance. Indeed, for such cases, zero terminal velocity is never achieved. As shown in Fig. 12, increasing the guidance gain value may improve terminal impact velocity but results in higher propellant consumption. Similarly, it was found that, for final times lower than the optimal, reducing the value of t_F^* increases the ability of MSSG to meet terminal conditions, but results in higher propellant consumption. In conclusion, underestimating t_F (with respect to the optimal t_F) may result in a dramatic increase in fuel consumption if



a)



b)

Fig. 12 Comparison of velocity and acceleration command histories of MSSG and optimal solution for $\Lambda = 10$.

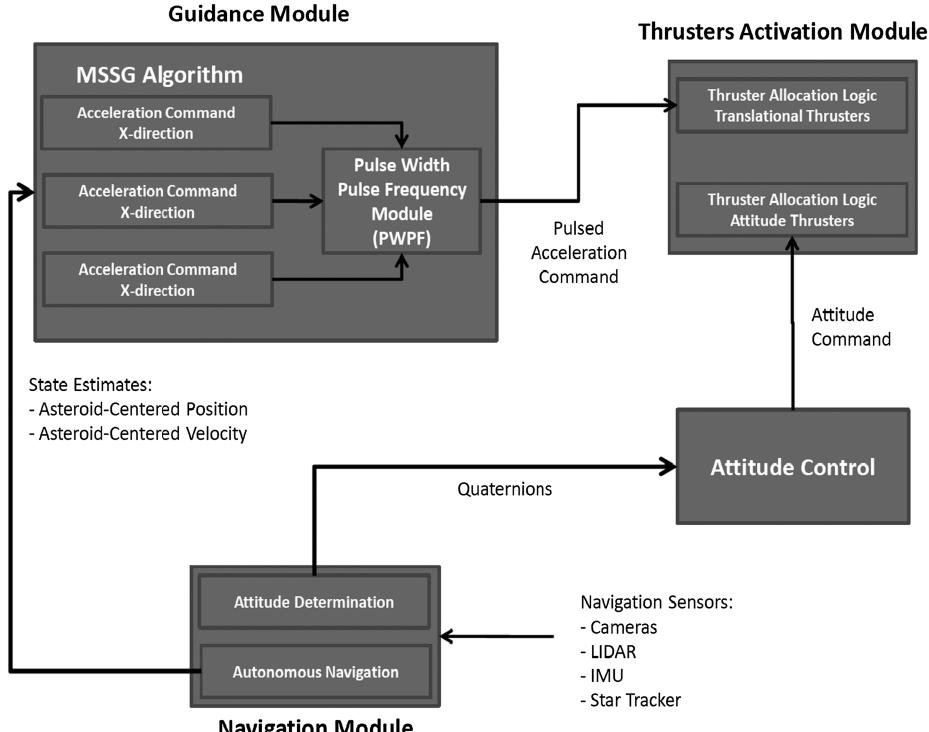


Fig. 13 Schematic of the GNC architecture for MSSG implementation.

stringent accuracy landing requirements are desired. Conversely, overestimating t_F has a mild effect on fuel efficiency.

C. Multiple Sliding Surface Guidance Pulse-Width Pulse-Frequency Implementation and Monte Carlo Simulations

Figure 13 shows the schematic of a possible guidance navigation and control (GNC) architecture hosting the proposed guidance algorithm. The next generation of robotic spacecraft for autonomous close-proximity operations around asteroids will have the ability to autonomously perform navigation and guidance functions, including landing site selection and obstacle detection and avoidance [16,35].

Position and velocity determination comes from filters capable of processing optical navigation data (e.g. camera and LIDAR) to correctly estimate the relative position and velocity of the spacecraft around the asteroid. Attitude is determined using a combination of inertial measurement units (IMUs) and star trackers. Position and velocity are fed to the guidance module, which implements the MSSG logic to determine the three components of the acceleration command with respect to the body-fixed, asteroid-centered reference frame. In the configuration shown in Fig. 13, it is assumed that the guidance and attitude functions are independent. More specifically, it is assumed that the only function of the attitude module is to maintain the body-fixed spacecraft reference frame aligned with the asteroid-centered frame. In this case, the guidance algorithm can generate

three independent acceleration commands along the asteroid-fixed directions.

The acceleration command generated by the MSSG is usually continuous. Conventional hydrazine thrusters have an on-off operation mode and cannot directly process continuous signals. The three acceleration commands are therefore fed to a set of PWPF algorithms to generate on-off acceleration commands, which approximates the three continuous commands generated by the guidance algorithms. Indeed, the PWPF modulator produces a pulsed command sequence of the thrusters' valves by adjusting both pulse width and frequency. The PWPF is composed of a Schmidt trigger, a linear prefilter, and a feedback loop [36]. The continuous acceleration command is compared with the modulator output and the error processed by a first-order filter. The filter output is then fed to the Schmidt module, which triggers a constant positive, zero, or negative command. The outputs of the three modulators are passed to the thruster activation logic module, which selects the thrusters to be fired accordingly. The parameters of interest are the prefilter parameters K_m , τ_m , as well as the Schmidt-trigger's parameters U_{on} , U_{off} .

In the preceding guidance parameters analysis, the MSSG algorithm is shown to have good performance in driving the system toward the desired point on the asteroid surface. Indeed, a properly designed guidance algorithm is expected to perform well under idealized conditions (e.g., perfect knowledge of the dynamic environment, no guidance delays, etc.). To verify the robustness of the MSSG algorithm and its ability to drive the spacecraft to the desired point under an uncertain environment, a set of Monte Carlo simulations has been implemented. In this set of simulations, the

Table 3 Dispersion parameters used in the Monte Carlo simulations

	Mean value	Standard deviation
Cross range (x axis)	-1,000 m	50 m
Downrange (y axis)	1,000 m	50 m
Altitude (z axis)	11,500 m	50 m
Velocity (x axis)	0 m/s	0.5 m/s
Velocity (y axis)	0 m/s	0.5 m/s
Velocity (z axis)	0 m/s	0.5 m/s
Mass	500 kg	—
Mass flow rate	0%	Uniform ($\pm 10\%$ max)
Solar pressure (x, y axes)	10^{-5} m/s^2	10^{-6} m/s^2
Gravity (x, y, z axes)	Nominal	10%

Table 4 Guidance gains and PWPF parameters

Λ Matrix	diag{4, 10, 2} s
t_f	1200 s
t_f^*	600 s
K_m	1 (adimensional)
τ_m	1 s
U_{On}	0.0042 m/s ²
U_{Off}	0.0021 m/s ²

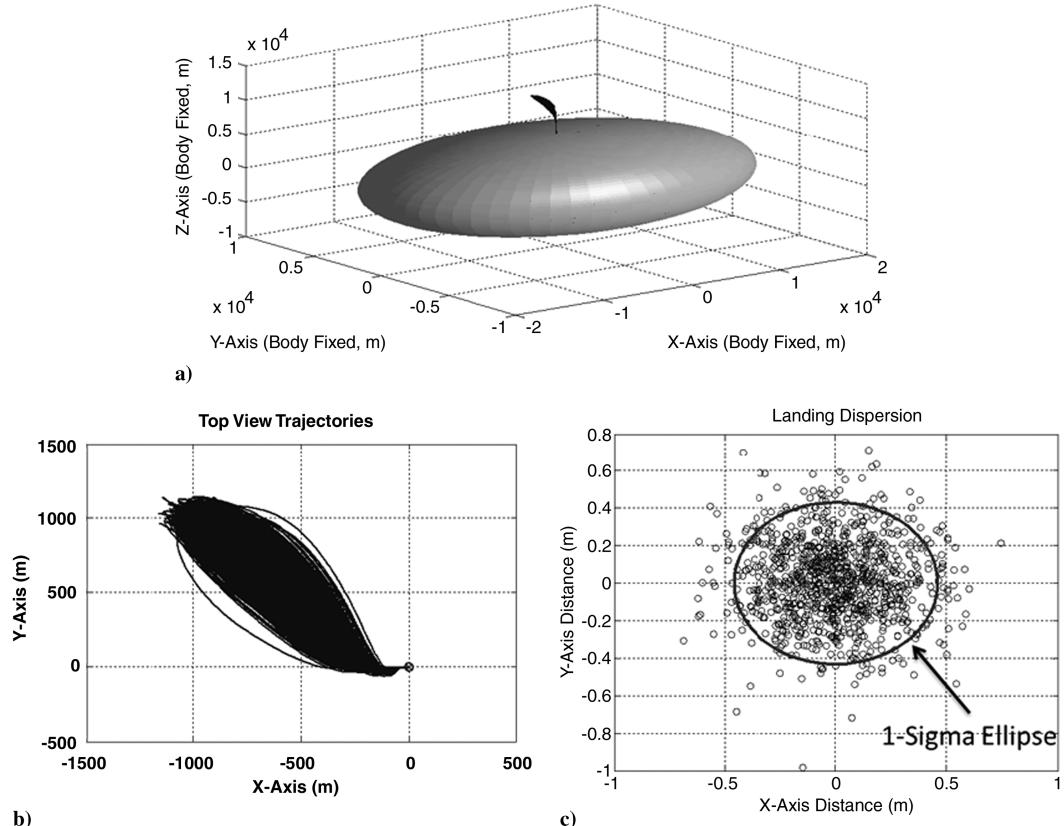


Fig. 14 Monte Carlo histories for the MSSG algorithm: a) three-dimensional landing trajectories, b) top-view trajectories (from the asteroid north pole), and c) landing ellipse (1σ value).

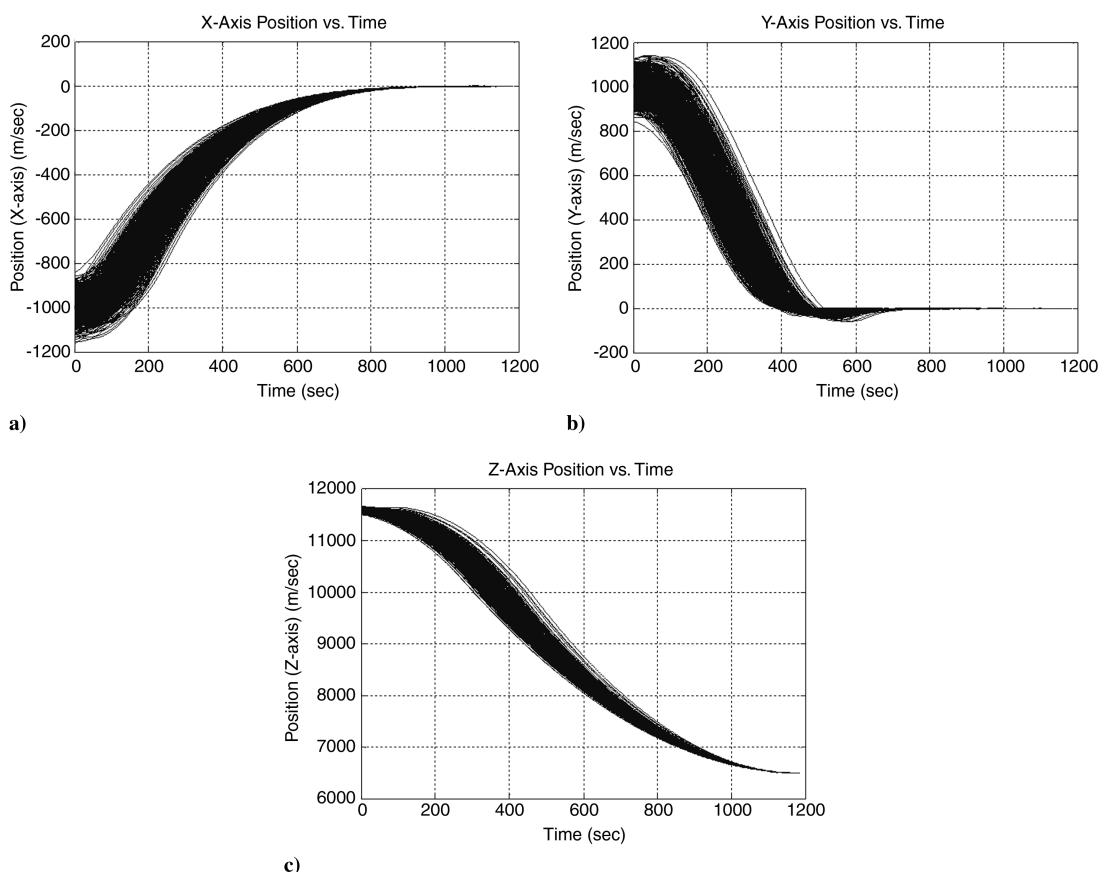


Fig. 15 Monte Carlo position histories for the MSSG algorithm: a) crossrange vs. time, b) downrange vs. time, and c) altitude vs. time.

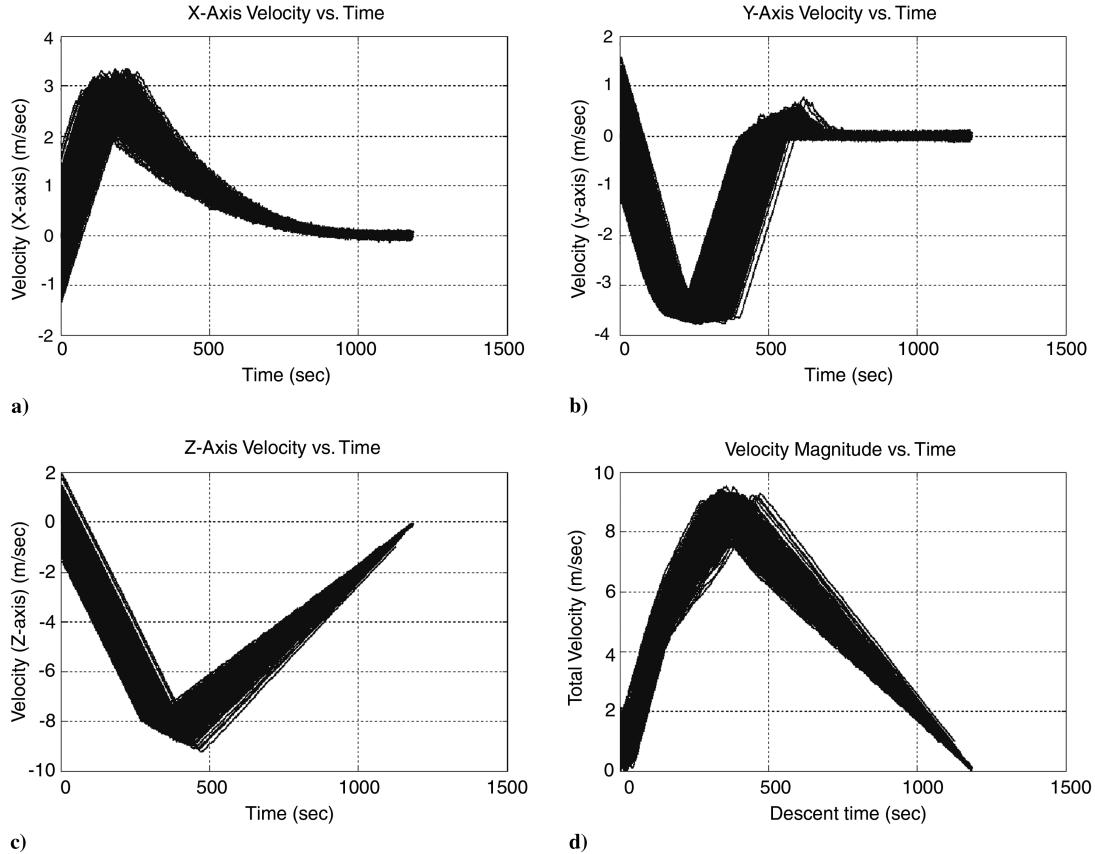


Fig. 16 Monte Carlo velocity histories for the MSSG algorithm: a) cross-range velocity vs time, b) downrange velocity vs time, and c) vertical velocity vs time.

guidance algorithm is asked to determine and execute commands under more realistic conditions. The 3-DOF model [Eqs. (1–9)], describing the mass-varying translational dynamics of the spacecraft in close proximity of the triaxial asteroid described in the preceding section, has been implemented in a MATLAB environment. The model includes 1) a linearly, time-varying mass model with a nominal mass flow-rate subjected to perturbation; 2) perturbing acceleration that accounts for unmodeled dynamics, including accelerations due to mass-flow rate perturbations, perturbing accelerations simulating the influence of solar radiation pressure, and perturbing acceleration coming from uncertain knowledge of the gravitational acceleration.

The Monte Carlo analysis has been conducted by running 1000 simulations of the MSSG algorithm in the described 3-DOF framework. Table 3 shows the parameters employed in the simulations, as well as their dispersion. Table 4 shows the MSSG guidance and PWPF parameters. The nominal unperturbed case assumes that a 500 kg spacecraft is hovering with zero velocity on a

location above the asteroid polar region determined in the body-fixed, asteroid-centered reference frame by the vector $\mathbf{r}_L(t_0) = [-1000, 1000, 11, 500]^T$ m. The MSSG algorithm is required to drive the system to a location on the north pole of the asteroid defined by the vector $\mathbf{r}_L(t_0) = [0, 0, 6500]^T$ m. The spacecraft is assumed to be equipped with a set of two thrusters per principal axis. The thrusters can generate a 10 N thrust in each direction and have a specific impulse of 170 s (pulsed mode). The selected asteroid is assumed to be perfectly triaxial, mimicking an Eros-like body (see preceding section). For each of the simulations, both initial position and velocity are sampled from a normal (Gaussian) distribution with standard deviation reported in Table 1. For each firing, mass flow rate is perturbed using a uniform distribution ($\pm 10\%$ upper value). Additional perturbing acceleration acting on the x - y plane is added to simulate the influence of the solar radiation pressure. During the Monte Carlo simulations, for each of the two components, the perturbing acceleration is sampled from a normal distribution with mean and standard deviation as specified in Table 1. The nominal gravity field computed using the closed-form solution for a homogeneous, Eros-like triaxial ellipsoid is perturbed using a Gaussian noise with zero mean and 10% standard deviation with respect to the nominal value. The MSSG is implemented assuming that the guidance module generates a command every 0.5 s (2 Hz).

The MSSG law requires knowledge of the current spacecraft state, i.e., position and velocity. Accurate navigation around small bodies may require onboard autonomous navigation schemes (e.g., AutoNav [37]), which rely on real-time optical images analysis and onboard state and covariance propagation algorithms [35]. Here, we model the navigation error using a simple statistical model, in which the true state is perturbed using a Gaussian noise with zero mean and 0.5 m and 0.1 m/s standard deviation (1σ) for each component of the position and velocity, respectively.

Figures 14–16 show the history of the MSSG-guided trajectories for the 1000 Monte Carlo simulations. Note that in Fig. 15, altitude is with respect to the x - y plane (equatorial plane of the asteroid). The

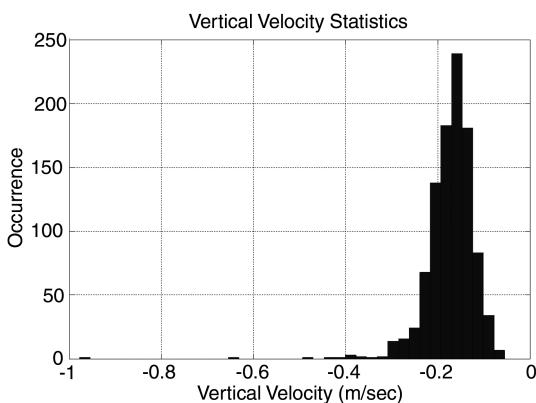


Fig. 17 Terminal vertical velocity statistics.

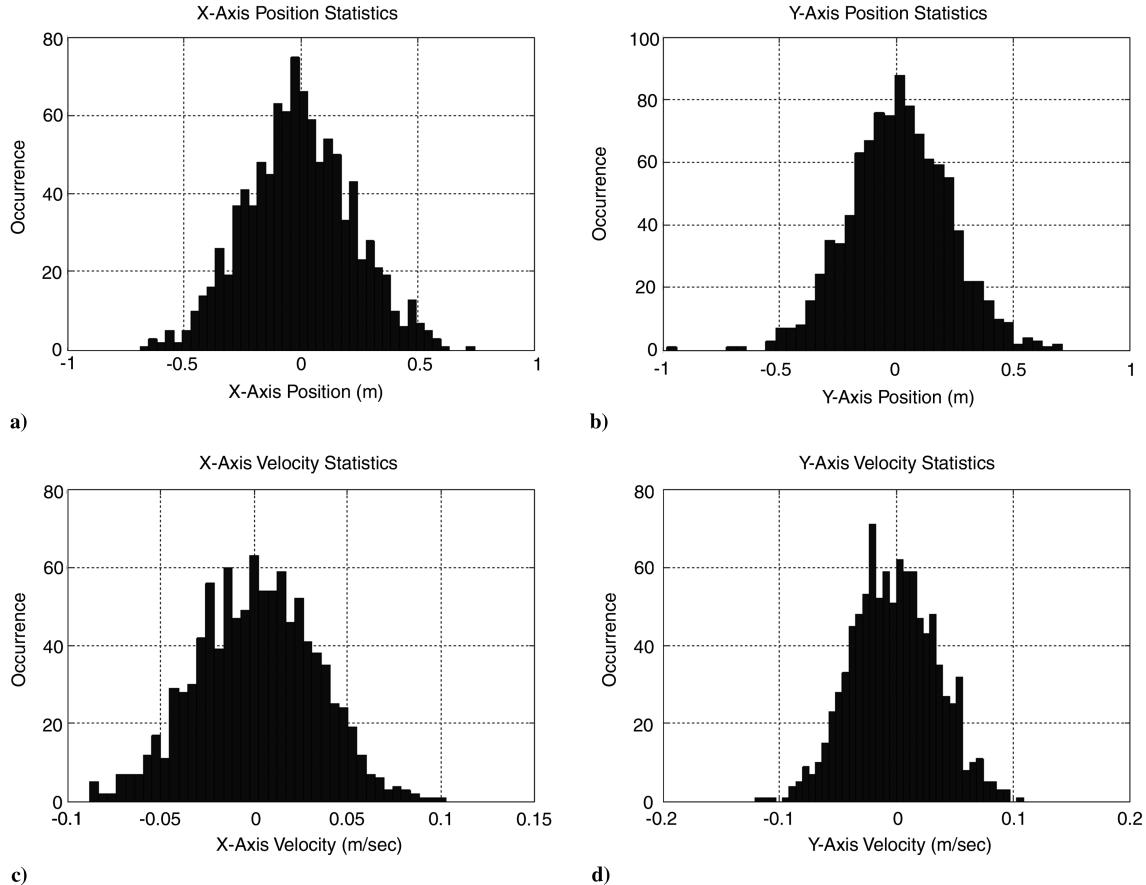


Fig. 18 Landing statistics for the Monte Carlo simulations: a) cross range, b) downrange, c) cross-range velocity, and d) downrange velocity.

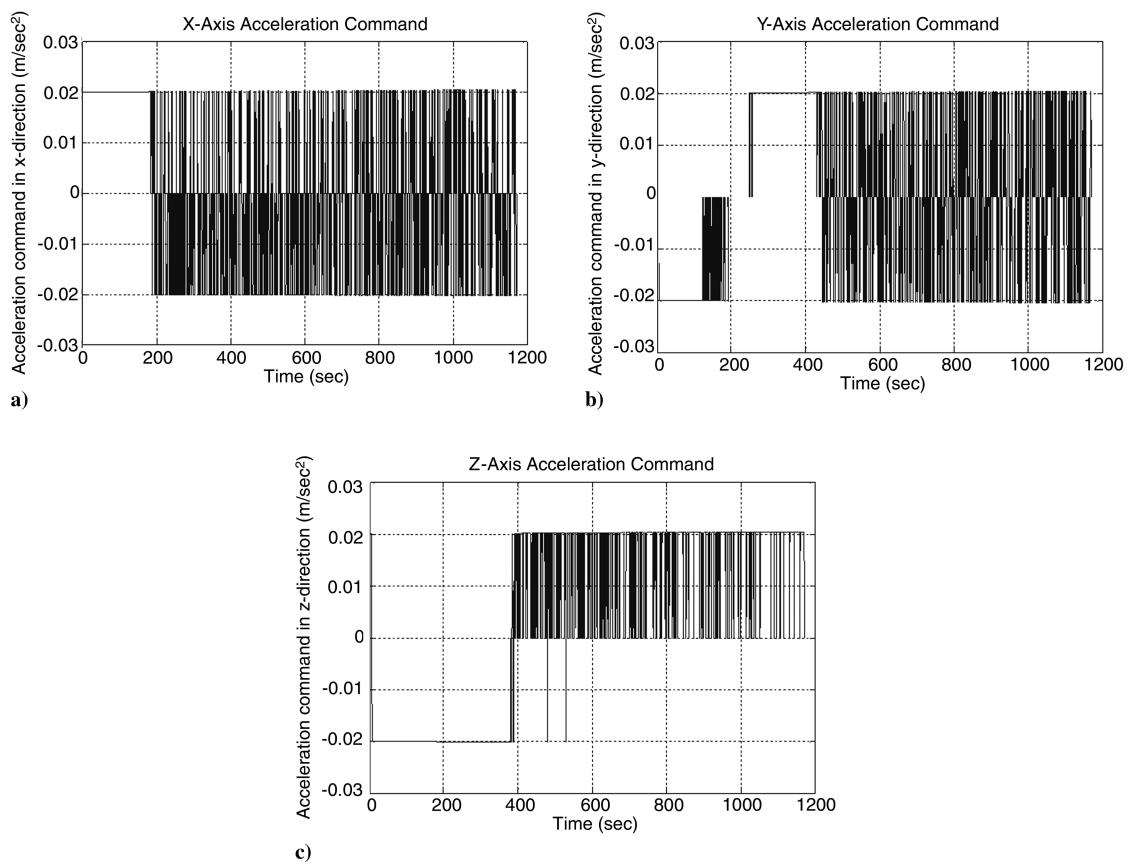


Fig. 19 One example of the acceleration command histories generated by the MSSG algorithm.

terminal state statistics are reported in Figs. 17 and 18. As evident, the MSSG algorithm performs very well. For example, Fig. 14c shows the landing dispersion ellipse for the 1000 simulated cases along with a 1σ ellipse. MSSG is shown to be very accurate in spite of perturbing accelerations and navigation errors. Maximum errors in the x and y directions are 0.75 m with a 1σ value of 0.23 m, and 1.0 m with a 1σ value of 0.215 m, respectively, with their statistical distribution reported in Figs. 18a and 18b. The x - and y -axis terminal velocities are reported to be in the range $[-0.09, 0.10]$ and $[-0.12, 0.11]$ m/s, respectively (Figs. 18c and 18d). The maximum terminal vertical velocity is -0.98 m/s with a mean of -0.17 m/s (Fig. 17). Figure 19 shows the three components of the acceleration command both continuous and modulated.

V. Conclusions

In this paper, a novel nonlinear guidance algorithm for asteroid powered descent and landing is presented. The algorithm is based on higher-order sliding-mode control theory and it has been named multiple sliding surface guidance (MSSG) algorithm. The guidance algorithm is robust against perturbations and unmodeled dynamics. It is theoretically demonstrated to be globally stable if an upper bound for the perturbing acceleration is known. The class of closed-loop landing trajectories generated by this algorithm has been investigated via an in-depth parametric analysis. A comparison with a nominal minimum-fuel, open-loop guided solution has been presented to understand the performance of the proposed guidance in terms of accuracy and propellant consumption. Moreover, a set of Monte Carlo simulations has been implemented to evaluate the performance of the algorithm in realistic asteroid descent scenarios and its feasibility to be operated in a modulated fashion. Such scenarios included perturbing accelerations (thrust, solar radiation pressure, and uncertain gravitational field), as well as navigation errors in both position and velocity. It is shown that the MSSG algorithm is very

accurate and yields very low residual guidance errors in both position and velocity. The algorithm is extremely fast and suitable for real-time onboard implementation. Indeed, no precomputed reference trajectory is required. The only input is the state to be targeted, that is, position on the asteroid surface and desired landing velocity (typically zero for a soft landing). The ability of the MSSG algorithm to target a desired state makes it extremely flexible for close-proximity operations. For example, landing retargeting can be implemented by simply shifting the desired landing point while en route toward the asteroid surface. Future work will investigate the implementation of the algorithm for other close-proximity operations including 1) transition between two hovering states and 2) transition between two points on the surface.

Appendix: Powered Descent Optimization Using General Pseudospectral Optimal Control Software

For the asteroid powered descent case, the minimum-fuel optimal guidance problem can be formulated as follows: Find the thrust program that minimizes the following cost function (negative of the lander final mass; equivalent to minimizing the amount of propellant during descent):

$$\max_{t_F, T_c(\cdot)} m_L(t_F) = \min_{t_F, T_c(\cdot)} \int_0^{t_F} \|T_c\| dt \quad (A1)$$

subject to the following constraints (equations of motion)

$$\dot{r}_L = v_L \quad (A2)$$

$$\dot{v}_L = 2\omega \times v_L + \omega \times \omega \times r_L + g(r_L) + \frac{T_c}{m_L} \quad (A3)$$

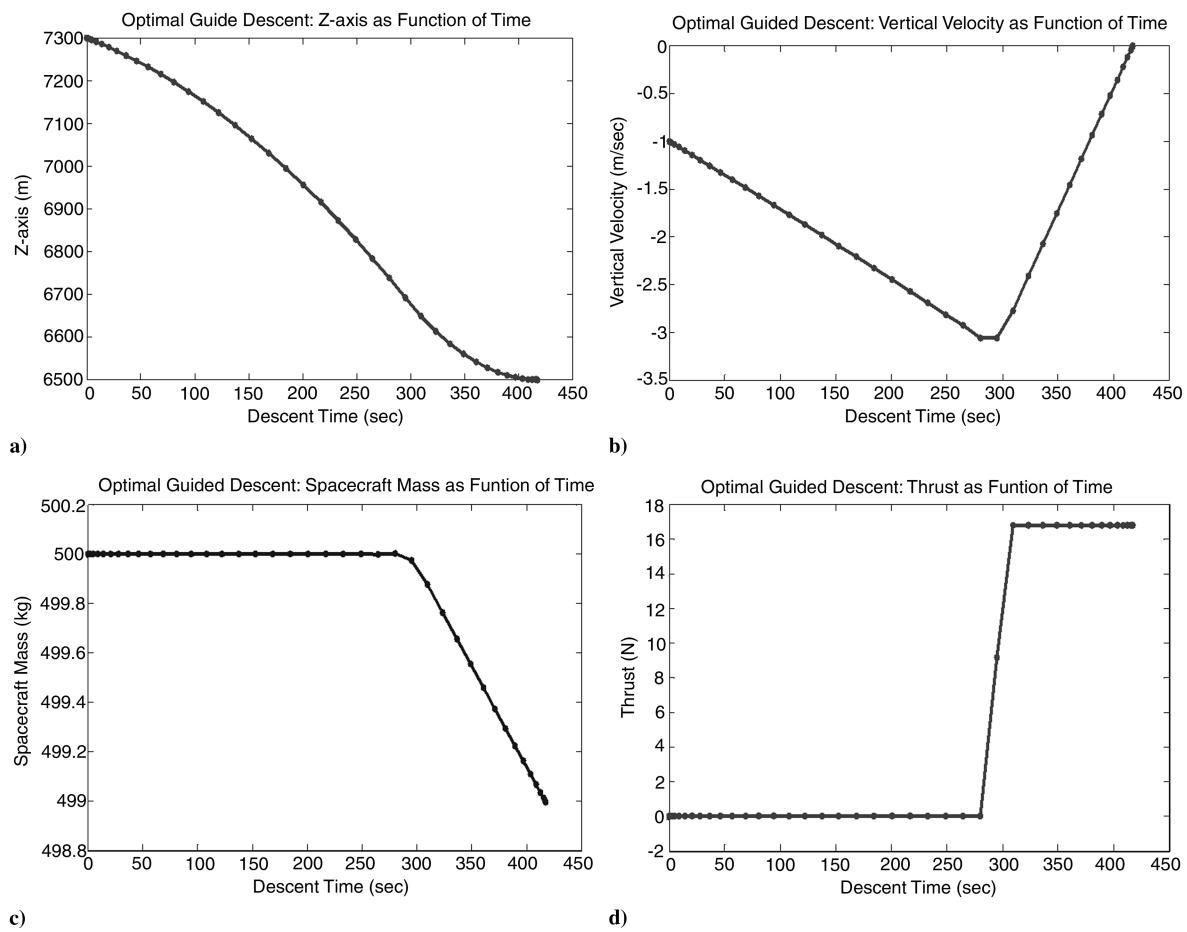


Fig. A1 Optimal trajectory generation results: a) z -axis position, b) vertical velocity, c) total spacecraft mass, and d) thrust command.

$$\frac{d}{dt}m_L = -\frac{\|T_c\|}{I_{sp}g_0} \quad (\text{A4})$$

and the following boundary conditions and additional constraints

$$0 < T_{\min} < T_{cx,y,z} < T_{\max} \quad (\text{A5})$$

$$\mathbf{r}_L(0) = \mathbf{r}_{L0}, \quad \mathbf{v}_L(0) = \dot{\mathbf{r}}_L(0) = \mathbf{v}_{L0} \quad (\text{A6})$$

$$\mathbf{r}_L(t_F) = \mathbf{r}_{LF}, \quad \mathbf{v}_L(t_F) = \dot{\mathbf{r}}_L(t_F) = \mathbf{v}_{LF} \quad (\text{A7})$$

Here, each of the thrust components in the asteroid-fixed reference frame is limited to operate between a minimum value T_{\min} and a maximum value T_{\max} . Generally, the problem formulated in Eqs. (A1–A7) does not have an analytical solution and must be solved numerically. To obtain the open-loop, fuel-optimal thrust program GPOPS [34] has been employed. GPOPS is an open-source optimal control software that implements Gauss and Radau HP-adaptive pseudospectral methods. After formulating the landing problem as described earlier, the software allows the direct transcription of the continuous-time, fuel-optimal control problem to a finite-dimensional nonlinear programming problem (NLP). In GPOPS, the resulting NLP is solved using the SNOPT solver [38]. The pseudospectral approach is very powerful because one can approximate both state and control using a basis of Lagrange polynomials. Moreover, the dynamics is collocated at the Legendre–Gauss–Radau points. The use of global polynomials coupled with Gauss quadrature collocation points is known to provide accurate approximations that converge exponentially to continuous problems with smooth solutions.

For our analysis, the open-loop, fuel-optimal asteroid landing problem is solved assuming that the motion of the spacecraft is constrained to occur in a vertical direction (altitude). The latter is imposed by assuming that the spacecraft is initially located above the asteroid north pole as defined by $\mathbf{r}_L(0) = [0, 0, 7300]^T$ m, that is, 800 m above the desired landing location defined by $\mathbf{r}_L(t_F) = [0, 0, 6500]^T$ m in the asteroid-fixed coordinate frame. The initial spacecraft velocity is downward vertical as defined by $\mathbf{v}_L(0) = [0, 0, -1]^T$, which constrains the motion to occur in the vertical direction only. The spacecraft is assumed to have an initial wet mass of 500 kg and is capable of a maximum (allowable) thrust of 17 N with $T_{\min} = 0$ N. A GPOPS scripts has been generated to determine a numerical solution of the optimal asteroid landing problem. The descent interval has been discretized in 40 intervals with two nodes per interval. The HP-adaptive method with a maximum of 10 mesh refinements (iterations) has been selected out of the options available via the GPOPS platform. Figure A1 shows the open-loop, fuel-optimal guided vertical descent. As expected, the optimal solution is extremal, that is, the control variable (thrust) switches between the minimum and maximum value. Indeed, starting from the initial condition, the spacecraft follows a free-fall trajectory, until it switches to the maximum allowable value and soft landing is achieved.

References

- [1] Ciesla, F. J., and Charnley, S. B., “The Physics and Chemistry of Nebular Evolution,” *Meteorites and the Early Solar System II*, Univ. of Arizona Press, Tucson, AZ, 2006, pp. 209–230.
- [2] “National Research Council, Defending Planet Earth: Near-Earth Object Surveys and Hazard Mitigation Strategies,” Final Rept., Com. to Review Near-Earth Object Surveys and Hazard Mitigation Strategies, National Academy Press, Washington, DC, Jan. 2010.
- [3] Kargel, J. S., “Metalliferous Asteroids as Potential Sources of Precious Metals,” *Journal of Geophysical Research*, Vol. 99, No. E10, 1994, pp. 21–129.
doi:10.1029/94JE02141
- [4] “Proposed Mission Would Return Sample from Asteroid,” *NASA News and Media Resources*, <http://www.nasa.gov/centers/goddard/news/features/2010/osiris-rex-science.html> [retrieved 1 Feb. 2012].
- [5] Lara, M., and Scheeres, D. J., “Stability Bounds for Three-Dimensional Motion Close to Asteroids,” *Journal of the Astronautical Sciences*, Vol. 50, No. 4, 2002, pp. 389–409.
- [6] Scheeres, D. J., “Close Proximity Operations at Small Bodies: Orbiting, Hovering, and Hopping,” *Mitigations of Hazardous Comets and Asteroids*, Cambridge Univ. Press, Cambridge, England, U.K., 2004, pp. 313–336.
- [7] Scheeres, D. J., “Stability of Hovering Orbits Around Small Bodies,” *AAS/AIAA Space Flight Mechanics Meeting*, American Astronautical Society Paper 99-159, Feb. 1999.
- [8] Sawai, S., Scheeres, D. J., and Broschart, S. B., “Control of Hovering Spacecraft Using Altimetry,” *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 4, 2002, pp. 786–795.
doi:10.2514/2.4947
- [9] Broschart, S. B., and Scheeres, D. J., “Control of Hovering Spacecraft near Small Bodies: Application to Asteroid 25143 Itokawa,” *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 2, 2005, pp. 343–354.
doi:10.2514/1.3890
- [10] Broschart, S. B., and Scheeres, D. J., “Boundedness of Spacecraft Hovering Under Dead-Band Control in Time-Invariant Systems,” *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 2, 2007, pp. 601–610.
doi:10.2514/1.20179
- [11] Kubota, T., Hashimoto, T., Sawai, S., Kawaguchi, J., Ninomiya, K., Uo, M., and Baba, K., “An Autonomous Navigation and Guidance System for MUSES-C Asteroid Landing,” *Acta Astronautica*, Vol. 52, Nos. 2–6, 2003, pp. 125–131.
- [12] Yoshimitsu, T., Kawaguchi, J., Hashimoto, T., Kubota, T., Uo, M., Morita, H., and Shirakawa, K., “Hayabusa-Final Autonomous Descent and Landing Based on Target Marker Tracking,” *Acta Astronautica*, Vol. 65, No. 5, 2009, pp. 657–665.
doi:10.1016/j.actaastro.2009.01.074
- [13] Scheeres, D. J., “Controlled Hovering Motion on an Asteroid,” *Orbital Motion in Strongly Perturbed Environments: Applications to Asteroid, Comet and Planetary Satellite Orbiters*, Springer Praxis, Chichester, England, U.K., 2012, pp. 243–254.
- [14] Dunham, D. W., Farquhar, R. W., McAdams, J. V., Holdridge, M., Nelson, R., Whittenburg, K., Antreasian, P., Chesley, S., Helfrich, C., Owen, W. M., Antreasian, P., Chesley, S., Helfrich, C., Owen, W. M., and Williams, B., “Implementation of the First Asteroid Landing,” *Icarus*, Vol. 159, No. 2, 2002, pp. 433–438.
doi:10.1006/icar.2002.6911
- [15] Xiangyu, H., Huta, C., and Pingyan, C., “An Autonomous Optical Navigation and Guidance for Soft Landing on Asteroids,” *Acta Astronautica*, Vol. 54, No. 10, 2003, pp. 763–771.
doi:10.1016/j.actaastro.2003.09.001
- [16] Shuang, L., and Pingyan, C., “Landmark Tracking Based Autonomous Navigation Schemes for Landing Spacecraft on Asteroids,” *Acta Astronautica*, Vol. 62, No. 6, 2008, pp. 391–403.
doi:10.1016/j.actaastro.2007.11.009
- [17] Gaskell, R. W., Barnouin-Jha, O. S., Scheeres, D. J., Konopliv, A. S., Mukai, T., Abe, S., Saito, J., Ishiguro, M., Kubota, T., Hashimoto, T., Kawaguchi, J., Yoshikawa, M., Shirakawa, K., Kominato, T., Hirata, N., and Demura, H., “Characterizing and Navigating Small Bodies With Imaging Data,” *Meteoritics & Planetary Science*, Vol. 43, No. 6, 2008, pp. 1049–1061.
doi:10.1111/j.1945-5100.2008.tb00692.x
- [18] Bhaskaran, S., Nandi, S., Broschart, S., Wallace, M., Olson, C., and Cangahuala, L. A., “Small Body Landing Accuracy Using In-Situ Navigation,” *AAS Guidance, Navigation and Control Meeting*, American Astronautical Society Paper 11-056, Feb. 2011.
- [19] Levant, A., “Construction Principles of 2-Sliding Mode Design,” *Automatica*, Vol. 43, No. 4, 2007, pp. 576–586.
doi:10.1016/j.automatica.2006.10.008
- [20] Won, M., and Hedrick, J., “Multiple Surface Sliding Control of a Class of Uncertain Nonlinear Systems,” *International Journal of Control*, Vol. 64, No. 4, 2003, pp. 693–706, 1996.
doi:10.1080/00207179608921650
- [21] MacMillan, W. D., *The Theory of Potential*, Dover, New York, NY, 1930, pp. 52–60.
- [22] Slotine, J., and Li, W., *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, 1991, pp. 276–310.
- [23] Vincent, T. L., and Grantham, W. J., *Nonlinear and Optimal Control Systems*, Wiley, New York, NY, 1997, pp. 169–260.
- [24] Levant, A., “Sliding Order and Sliding Accuracy in Sliding Mode Control,” *International Journal of Control*, Vol. 58, No. 6, 1993,

- pp. 1247–1263.
doi:10.1080/00207179308923053
- [25] Levant, A., "Higher-Order Sliding Modes, Differentiation and Output Feedback Control," *International Journal of Control*, Vol. 76, Nos. 9–10, 2003, pp. 924–941.
doi:10.1080/0020717031000099029
- [26] Fridman, L., "Chattering Analysis in Sliding Mode Systems with Inertial Sensors," *International Journal of Control*, Vol. 76, Nos. 9–10, 2003, pp. 906–912, 2003.
doi:10.1080/0020717031000099074
- [27] Shtessel, Y. B., and Shkolnikov, I. A., "Aeronautical and Space Vehicle Control in Dynamic Sliding Manifolds," *International Journal of Control*, Vol. 76, Nos. 9–10, 2003, pp. 1000–1017.
doi:10.1080/0020717031000099065
- [28] Shtessel, Y., Shkolnikov, I., and Levant, A., "Guidance and Control of Missile Interceptor Using Second Order Sliding Modes," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 45, No. 1, 2009, pp. 110–124.
doi:10.1109/TAES.2009.4805267
- [29] Harl, N., and Balakrishnan, S. N., "Reentry Terminal Guidance Through Sliding Control Mode," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 1, 2010, pp. 186–199.
doi:10.2514/1.42654
- [30] Furfaro, R., Selnick, S., Cupples, M. L., and Cribb, M. W., "Nonlinear Sliding Guidance Algorithms for Precision Lunar Landing," *21st AIAA/AAS Space Flight Mechanics Meeting*, American Astronautical Society Paper 11-167, Feb. 2011.
- [31] Scheeres, D. J., "Dynamics About Uniformly Rotating Triaxial Ellipsoids: Applications to Asteroids," *Icarus*, Vol. 110, No. 2, 1994, pp. 225–238.
doi:10.1006/icar.1994.1118
- [32] Scheeres, D. J., Williams, B. G., and Miller, J. K., "Evaluation of the Dynamic Environment of an Asteroid: Applications to 433 Eros," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 3, 2000, pp. 466–475.
doi:10.2514/2.4552
- [33] Lantoine, G., and Braun, R. D., "Optimal Trajectories for Soft Landing on Asteroids," *Advances in the Astronautical Sciences*, Vol. 129, Pt. 1, 2008, pp. 447–468. ; also American Astronautical Society Paper 07-280.
- [34] Rao, A. V., Benson, D. A., Darby, C., Patterson, M. A., Francolin, C., and Sanders, I., "Algorithm 902: GPOPS, a MATLAB Software for Solving Multiple Phase Optimal Control Problems Using the Gauss Pseudospectral Method," *ACM Transactions on Mathematical Software*, Vol. 37, No. 2, 2010, pp. 22:1–22:39.
doi:10.1145/1731022
- [35] Misu, T., Hashimoto, T., and Ninomiya, K., "Optical Guidance for Autonomous Landing of Spacecraft," *IEEE Transaction of Aerospace Electronic Systems*, Vol. 35, No. 2, 1999, pp. 459–472.
doi:10.1109/7.766929
- [36] Wie, B., *Space Vehicle Dynamics and Control*, 2nd ed., AIAA Educational Series, AIAA, Reston, VA, 1998, pp. 478–482.
- [37] Riedel, J. E., Bhaskaran, S., Desai, S. D., Han, D., Kennedy, B., McElrath, T., Null, G. W., Ryne, M., Synott, S. P., Wang, T. C., and Werner, R. A., "Using Autonomous Navigation for Interplanetary Missions: The Validation of Deep Space 1 AutoNav," *Fourth IAA International Conference on Low-Cost Planetary Missions*, International Academy of Astronautics Paper L-0807, May 2000.
- [38] Gill, P. E., Saunders, M. A., and Murray, W., "SNOPT: An SQP Algorithm for Large Scale Constrained Optimization," Univ. of California, TR NA-96-2, San Diego, CA, 1996.