Engineering Notes

Optimal Thrust Profile for Planetary Soft Landing Under Stochastic Disturbances

Ioannis Exarchos,* Evangelos A. Theodorou,† and
Panagiotis Tsiotras‡
Georgia Institute of Technology, Atlanta, Georgia 30332

DOI: 10.2514/1.G003598

I. Introduction

BY AND large, the literature on optimal control deals with the minimization of a performance index that penalizes control *energy*, because the input appears in quadratic form as part of the running cost. Such problems are typically referred to as *minimum energy* problems in optimal control theory—they involve the minimization of the $\mathcal{L}^2$-norm of an otherwise unconstrained control signal. While $\mathcal{L}^2$ minimization can be useful in addressing several optimal control problems in engineering (preventing engine overheating, avoiding high-frequency control input signals etc.), there are practical applications in which the control input is bounded (e.g., due to actuation constraints), and the $\mathcal{L}^1$-norm is a more suitable choice to penalize. These problems are also called *minimum-fuel* problems, due to the nature of the running cost, which involves an integral of the absolute value of the input signal. Minimum-fuel control appears as a necessity in several settings, especially in spacecraft guidance and control [1,2], in which fuel is a limited resource. Indeed, in such applications, using the $\mathcal{L}^2$-norm results in significantly more propellant consumption, as well as undesirable continuous thrusting. In some illustrative examples, this fuel penalty can be as high as 50% [3].

In this paper, we address a stochastic version of the so-called soft-landing problem (SLP). The objective of the SLP is to find the optimal thrust profile for a spacecraft attempting to make a soft landing on a planet, using the minimum amount of fuel. The problem was originally addressed by considering only one spatial dimension (namely, the altitude with respect to the planet), in which case its deterministic formulation offers a closed-form solution (initially obtained by Miele [4,5] during the 1960s; see also [6,7]). In more recent years, there has been renewed interest in the topic, appearing under the name *powered-descent guidance* (PDG), mainly due to the success of NASA's Mars Science Laboratory program. Several results appear in the literature, treating a more complex problem involving all three spatial dimensions, more accurate modeling of the dynamics to account for planetary rotation, and several state and control constraints [8–12]. An analysis for fuel optimality is also included in [13,14]. The challenges faced in the implementation of planetary PDG controllers are the twofold: 1) the environmental uncertainty and stochastic disturbances present, and 2) the limited capabilities for onboard computation.

The aim of this paper is to present an application of the framework of stochastic control using forward and backward sampling, developed in previous work by the authors [15–17], to the SLP. The problem considered within this paper is a combination of a stochastic $\mathcal{L}^1$-optimal control problem with a first-exit type of formulation. Such a combination has not been previously considered in our work, and no results have been published containing applications of our algorithm on first-exit problems in general. The motivation for this problem formulation stems from the fact that the proposed algorithm is intended to be deployed during the very last stage of the descent (i.e., seconds before landing), when the altitude is relatively low. In this setting, it is assumed that any constraints with respect to navigating toward the landing site have been already satisfied, and the final objective is to ensure that the spacecraft touches the ground smoothly, despite environmental disturbances. In fact, to the best of our knowledge, the majority of PDG controllers involve a terminal stage (i.e., just before landing), where the trajectory is vertical (see, e.g., [18]). Furthermore, this paper illustrates the advantages of a control law stemming from our framework, as opposed to a closed-loop deterministic control law that does not directly take stochasticity into consideration. Specifically, it is shown that simply employing a deterministic closed-loop controller in a model-predictive control fashion does not mitigate the risk of a crash during landing. In contrast, the proposed feedback controller, while subject to the same control structure restrictions, drastically reduces the chances of a crash, making them arbitrarily small. The proposed algorithm demonstrates superior performance, offering a much lower mean and variance for the touchdown speed. Depending on the given safety specifications, we can further reduce this mean and variance, thus gaining a more robust, safer controller, at the expense of slightly increased fuel expenditure. Finally, the nature of the algorithm allows for a complete solution of the problem a priori and off-line, thus minimizing the required onboard computing capabilities of the spacecraft.

II. Problem Description

In this section, we formally define the SLP. We first introduce the deterministic setting and its closed-form solution, which we will use later on for validation and comparison purposes. In the next section we present a stochastic version of the problem, on which we will apply the proposed algorithm. Finally, in the simulation section, we compare the numerical results obtained by the proposed algorithm to those of the closed-form solution (both in open-loop and closed-loop implementation).

A. Deterministic Setting

Consider the problem of a spacecraft attempting to make a soft landing on a planet, using the minimum amount of fuel. The dynamical equations are given by

$$\dot{h}(t) = v(t) \tag{1}$$

$$\dot{v}(t) = -g + \frac{u(t)}{m(t)}, \qquad u(t) \in [u_{\min}, u_{\max}] \tag{2}$$

$$\dot{m}(t) = -\alpha u(t) \tag{3}$$

with $t \in [0, t_f]$, $h(0) = h_0$, $v(0) = v_0$, and $m(0) = m_0$. Here, $h: [0, t_f] \to \mathbb{R}_+$, $v: [0, t_f] \to \mathbb{R}$, and $m: [0, t_f] \to \mathbb{R}_+$ denote the

altitude, vertical speed, and mass of the spacecraft at time t, respectively; g is the gravitational acceleration, assumed to be constant; α is a positive constant that describes the mass flow rate; and $u: [0, t_f] \to [u_{\min}, u_{\max}]$ is the control input (thrust), with $u_{\min}$, $u_{\max} \in \mathbb{R}_+$. As admissible controls, we consider all piecewise continuous control functions taking values in the aforementioned interval. The initial conditions are (h_0, v_0, m_0), whereas the terminal conditions are $h(t_f) = v(t_f) = 0$. Here, t_f denotes the time instant of landing, whose particular value is otherwise left unspecified. For the mass, we assume that a reasonable value has been assigned to m_0 so that landing with remaining mass at or above the dry mass (mass of the spaceship without fuel) is feasible. We wish to obtain the optimal control $u^*(\cdot)$ that satisfies the above conditions, while minimizing the amount of fuel spent:

$$J_{\det}(u(\cdot); h_0, v_0, m_0) = \int_0^{t_f} |u(t)| \, \mathrm{d}t \qquad (4)$$

It can be shown [4,6,7] that the solution to this free final time $\mathcal{L}^1$-optimal control problem yields a unique optimal *bang-bang* controller. It is also known that the problem is *normal* (meaning that singular control does not appear within the optimal control sequence), and that there is at most one switch time. The optimal control sequence is

$$u^*(t) = \begin{cases} u_{\min}, & t \in [0, t_s), \\ u_{\max}, & t \in [t_s, t_f] \end{cases} \qquad (5)$$

where t_s denotes the switching time. The values of t_s and t_f satisfy the following system of equations:

$$h_0 + v_0 t_s + \frac{t_f}{\alpha} - \frac{1}{\alpha}\left(t_s - \frac{m_0}{\alpha u_{\min}}\right)\ell_n\left(1 - \frac{\alpha u_{\min}}{m_0} t_s\right) - \frac{1}{2}g t_s^2$$
$$+ \frac{m_0 - \alpha u_{\min} t_s}{\alpha^2 u_{\max}}\ell_n\left(1 - \frac{\alpha u_{\max}}{m_0 - \alpha u_{\min} t_s}(t_f - t_s)\right) + \frac{1}{2}g(t_f - t_s)^2 = 0 \quad (6)$$

$$\alpha(u_{\max} - u_{\min})t_s = \alpha u_{\max} t_f + m_0(\exp(\alpha(v_0 - g t_f) - 1)) \quad (7)$$

The above system can be solved numerically for given values of t_s and t_f.

B. Stochastic Setting

To construct a stochastic extension to the previous model, we assume that randomness appears in the spacecraft's acceleration because of unmodeled environmental disturbance forces. This uncertainty is then integrated into the speed and altitude variables. Furthermore, the exact thrust value exerted by the spacecraft is assumed to be uncertain, due to limitations in the precision of our control. Thus, we introduce the dynamics

$$\mathrm{d}h(t) = v(t)\mathrm{d}t \qquad (8)$$

$$\mathrm{d}v(t) = \left(-g + \frac{u(t)}{m(t)}\right)\mathrm{d}t + \sigma \frac{u_{\max}}{m(t)}\mathrm{d}W_t, \qquad u(t) \in [u_{\min}, u_{\max}] \qquad (9)$$

$$\mathrm{d}m(t) = -\alpha u(t)\mathrm{d}t - \sigma \alpha u_{\max}\mathrm{d}W_t \qquad (10)$$

with $t \in [0, t_f]$, $h(0) = h_0$, $v(0) = v_0$, and $m(0) = m_0$. Here, $\mathrm{d}W_t$ are increments of Brownian motion, and σ is a constant. For notational compactness, we define $f \triangleq [v, -g, 0]$, $G \triangleq [0, 1/m, -\alpha]^T$, and $\Sigma \triangleq [0, \sigma u_{\max}/m, -\sigma \alpha u_{\max}]^T$ and write Eqs. (8–10) as

$$\mathrm{d}x = f(x)\mathrm{d}t + G(x)u\mathrm{d}t + \Sigma \mathrm{d}W_t \qquad (11)$$

where $x = [h, v, m]^T$.

In contrast to the deterministic case, terminal state conditions are not meaningful in a stochastic setting, because whenever the system dynamics are modeled by controlled diffusions, the probability of hitting a particular point in state space *exactly* is zero. Furthermore, in a stochastic setting, free final time problems without cost discounting can be troublesome due to the absence of boundedness guarantees. Because the approach used in this paper is based on trajectory sampling (see Sec. III), allowing the process to continue without imposing an upper bound on its duration may yield trajectory samples that have a very large—or even possibly infinite—time duration, and thus cannot be simulated (see relevant discussion in [19]). Instead, we formulate a *first-exit* problem with time upper bound, in which the process terminates as soon as the task has been achieved (here, landing), *or* a specified maximum time duration has passed, whichever event occurs first. Thus, for the problem under investigation, we consider a first-exit formulation, in which the process terminates when the hyperplane $h = 0$ is crossed, or an upper bound T on the time duration has passed. We thus define the state space by $\mathcal{G} = \{h, v, m: h \in \mathbb{R}_+, v \in \mathbb{R}, m \in R_+\}$, with $\partial\mathcal{G} = \{h, v, m \in \mathcal{G}: h = 0\}$ being its boundary, the crossing of which signals early termination. To account for the lack of terminal constraints, we introduce them as *soft* constraints within the cost, which we define as follows:

$$J(u(\cdot); h_0, v_0, m_0, T) = \mathbb{E}\left[\Psi(\mathcal{T}, h(\mathcal{T}), v(\mathcal{T})) + \int_0^{\mathcal{T}} q|u(t)| \, \mathrm{d}t\right] \qquad (12)$$

with q being a positive constant, and $\mathcal{T}$ the minimum between the time of first exit, τ_{exit}, and the upper bound T, namely,

$$\mathcal{T} \triangleq \min\{\tau_{\text{exit}}, T\}, \quad \text{with} \quad \tau_{\text{exit}} \triangleq \inf\{s \in [0, T]: x(s) \in \partial\mathcal{G}\} \qquad (13)$$

Finally, we define the terminal cost Ψ as

$$\Psi(t, h, v) \triangleq \begin{cases} c_1 h^2(t) + c_2 v^2(t) \triangleq g(h, v), & (t, h, v, m) \in \{T\} \times \mathcal{G}, \\ c_3 v^2(t) \triangleq \psi(v), & (t, h, v, m) \in [0, T) \times \partial\mathcal{G} \end{cases} \qquad (14)$$

where c_1, c_2, and c_3 are positive constants. The motivation behind this choice of terminal cost $\Psi(\cdot)$ is that trajectories that terminate earlier than $t = T$ because of touchdown ($h = 0$) are penalized a high touchdown speed, whereas trajectories that terminate at $t = T$ (i.e., without a touchdown) are penalized for both residual altitude and speed.

III. Solution of the Stochastic Problem

In this section, we provide a brief overview of the methodology developed by the authors in [15–17] to address stochastic $\mathcal{L}^2$- and $\mathcal{L}^1$-optimal control problems, which is suitably modified in this paper to address the peculiarities of the SLP. The interested reader is referred to the above publications for a detailed, self-contained exposition of the framework used herein.

Through Bellman's principle of optimality, it is shown [20,21] that minimizing Eq. (12) subject to the dynamics given by Eqs. (8–10) is associated with the solution of the Hamilton-Jacobi-Bellman (HJB) partial differential equation (PDE) for the Value function V, which—omitting function arguments for brevity—assumes for the case at hand the following form:

$$
\begin{cases}
V_t + \inf_{u \in U}\left\{\dfrac{1}{2}\mathrm{tr}(V_{xx}\Sigma\Sigma^T) + V_x^T f + (V_x^T G + q)u\right\} = 0, & (t,x) \in [0,T) \times \mathbb{R}^n, \\
V(T,x) = g(x), & x \in \mathcal{G}, \\
V(t,x) = \psi(x), & (t,x) \in [0,T) \times \partial\mathcal{G}
\end{cases}
\tag{15}
$$

where $x \triangleq [h, v, m]$, $g(\cdot)$ and $\psi(\cdot)$ represent the boundary conditions given for this problem by Eq. (14), and V_x and V_{xx} denote the gradient and the Hessian of V, respectively. The optimal control is given by

$$
u^* = \begin{cases}
u_{\max}, & V_x^T G + q \le 0, \\
u_{\min}, & V_x^T G + q > 0
\end{cases}
\tag{16}
$$

and substitution to the original HJB PDE (15) yields the following equivalent form:

$$
\begin{cases}
V_t + \dfrac{1}{2}\mathrm{tr}(V_{xx}\Sigma(t,x)\Sigma^T(t,x)) + V_x^T f + \min\{(V_x^T G + q)u_{\max}, (V_x^T G + q)u_{\min}\} = 0, & (t,x) \in [0,T) \times \mathcal{G}, \\
V(T,x) = c_1 h^2(t) + c_2 v^2(t), & x \in \mathcal{G}, \\
V(t,x) = c_3 v^2(t), & (t,x) \in [0,T) \times \partial\mathcal{G}
\end{cases}
\tag{17}
$$

The cornerstone of our approach is the nonlinear Feynman-Kac lemma, which states that instead of solving the PDE problem (17), one may solve an equivalent system of *forward and backward stochastic differential equations* (FBSDEs), to obtain the solution $V(t,x)$, and through that, the optimal control $u(t,x)$. The corresponding system of FBSDEs [15] consists of the forward process (forward stochastic differential equation, FSDE)

$$
\begin{cases}
\mathrm{d}X_t = (f(X_t) + G(X_t)\bar{u}(t))\mathrm{d}t + \Sigma(X_t)\mathrm{d}W_t, & t \in [0,T], \\
X_0 = x_0
\end{cases}
\tag{18}
$$

along with the backward process (backward stochastic differential equation, BSDE)

$$
\begin{cases}
\mathrm{d}Y_t = -F(t,Z)\mathrm{d}t + Z_t \mathrm{d}W_t, & t \in [0,T], \\
Y_T = \Psi(X_T)
\end{cases}
\tag{19}
$$

wherein, for the problem at hand,

$$
F(t,Z) \triangleq \min\left\{\left(\frac{Z_t}{bu_{\max}\sigma} + q\right)u_{\max}, \left(\frac{Z_t}{bu_{\max}\sigma} + q\right)u_{\min}\right\} - \frac{Z_t}{bu_{\max}\sigma}\bar{u}(t)
$$

and $\bar{u}$ is *any* control input (it can be taken as zero, or an initial guess of the optimal control). Note that the forward process X_t is simply a copy of the dynamics of the system, and is subject to a given *initial* condition. In contrast, the backward process Y_t needs to satisfy a *terminal* condition. The solution $V(t,x)$ is obtained by solving the above system of FBSDEs, in which the process Y_t is an evaluation of the value function along trajectories of X_t, and the process Z_t is an evaluation of $\Sigma^T V_x$, again along trajectories of X_t, namely,

$$
Y_t = V(t, X_t), \qquad Z_t = \Sigma^T(X_t)V_x(t, X_t)
\tag{20}
$$

The main difficulty in dealing with backward SDEs is that, in contrast to forward SDEs (which can be easily simulated by sampling a noise profile and performing Euler integration), due to the presence of a terminal condition, integration must be performed in the same manner that is backward in time. This presents a complication in practice, because the BSDE involves the noise term $Z_t \mathrm{d}W_t$, and thus a mere backward integration would yield a solution that at each time instant depends on future values of the noise. A way to avoid this issue is to backpropagate the *conditional expectation* of Y_t. Thus, if one discretizes time, Eq. (19) takes the form

$$
Y_{t_i} = \mathbb{E}[Y_{t_{i+1}} + \Delta t F(t_{i+1}, Z_{t_{i+1}}) | X_{t_i}] = \mathbb{E}[\tilde{Y}_{t_i} | X_{t_i}]
\tag{21}
$$

in which the noise term vanishes due to the conditional expectation (it has an expectation of zero), and in which we have denoted $\tilde{Y}_{t_i} \triangleq Y_{t_{i+1}} + \Delta t F(t_{i+1}, Z_{t_{i+1}})$. The conditional expectation is then numerically approximated using the least squares Monte Carlo method (LSMC). Specifically, for any time t_i and given sample pairs of X_{t_i} and $\tilde{Y}_{t_i}$, LMSC consists of performing least squares regression to obtain the function v_{t_i} that minimizes $|v_{t_i}(X_{t_i}) - \tilde{Y}_{t_i}|^2$. Then, the expectation operator is approximated by taking the projection $Y_{t_i} = v_{t_i}(X_{t_i})$. The regressed function $v_{t_i}(x)$ is in fact an approximation of the value function for that particular time t_i, namely, $v_{t_i}(x) \approx V(t_i, x)$. The entire numerical process is as follows:

1) Simulate M forward processes X_t according to the system dynamics (18) along a time grid $\{t_i\}_{i=0}^N$, using a suitable time discretization (e.g., the Euler-Maruyama scheme [22]), until each trajectory reaches the boundary condition states ($h = 0$, or $t = T$).

2) For each terminated process, assign a final condition for Y_T according to Eq. (19), and $Z_T = \Sigma(X_T)\Psi_x(X_T)$, where Ψ_x denotes the gradient of Ψ. Then, back-propagate each data point to the previous time step on the grid by assigning $\tilde{Y}_{t_i} = Y_{t_{i+1}} + F(t_{i+1}, Z_{t_{i+1}})\Delta t$. Perform regression (e.g., linear regression using polynomial basis functions) on the $\tilde{Y}_{t_i}$ dataset to obtain the value function $V(t_i, x)$. Take $Y_{t_i} = V(t_i, X_{t_i})$, and calculate $Z_i = \Sigma(X_{t_i})V_x(t_i, X_{t_i})$.

3) Repeat step (2) going backward along the time grid until t_0.

This procedure allows us to obtain regressions for the value function for each time step t_i, namely, $V(t_i, x)$, and thus obtain a feedback control law according to Eq. (16). Initializing with $\bar{u} = 0$, or another initial guess, the procedure is repeated by using $\bar{u} = u^*$, in which u^* is the optimal control law resulting from a previous run of the procedure. In this way, results are refined and improved in an iterative manner. The trajectory blending technique [16,17] is also helpful in order to improve the overall convergence properties of the algorithm. For more details on both the theory as well as the algorithmic aspects of this section, we refer the reader to the above publications. More on the theory of FBSDEs can also be found in [23–25].

IV. Results

For the purposes of simulation, we assumed the following constants: $g = 3.71$ [m/s^2], $\alpha = 4.83 \times 10^{-4}$ [s/m], $u_{min} = 4.97 \times 10^3$ [N], $u_{max} = 1.33 \times 10^4$ [N], $\sigma = 0.06$, and initial conditions $(h_0, v_0, m_0) = (80$ [m], -10 [m/s], 1905 [kg]). The upper bound on the duration of the descent is $T = 8.5$ s. Comparison of performance is done via two metrics, namely, the touchdown speed, and the fuel mass used; in both cases, both mean and variance are calculated. Another indicator is the percentage of trajectories that lead to a touchdown.

A. Deterministic Control: Open-Loop Implementation

In this case, we calculate the switching time and apply the deterministic control law (5) in an open-loop fashion. The results are depicted in Fig. 1. Out of the 1000 trajectories simulated, only 50.3% lead to touchdown. The remaining trajectories lead to a hovering above the ground, which also explains the spike in fuel expenditure (shown in Fig. 1b). Of the 50.3% of the trajectories for which a touchdown occurs, most of them are considered a crash, due to the high-speed impact. Indeed, the mean touchdown speed is -5.24 m/s, with a variance of 5.40 m/s.

B. Deterministic Control: Closed-Loop Implementation

Next, we simulated the control law (5) in a closed-loop fashion, that is, at each time instant we recalculated the switching time. Switching back and forth between controls (due to the influence of the noise) is allowed. This is a deterministic feedback in the form of a model predictive control (MPC) scheme. The results are depicted in Fig. 2. All of the 1000 trajectories simulated now lead to a touchdown. However, most of them are still considered a crash, due to the high-speed impact. Indeed, the mean touchdown speed this time is -3.19 m/s, with a variance of 1.96 m/s.
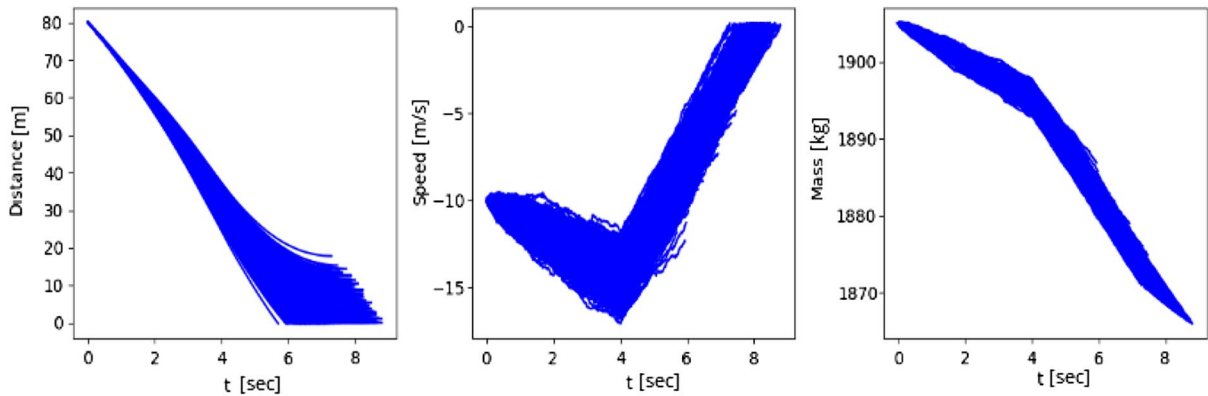
C. Proposed Algorithm

For $T = 8.5$ [s], $q = 1$, we used 3000 trajectory samples on a time grid of $\Delta t = 0.005$ [s], and a trajectory blending ratio of 0.98. The algorithm was run for 35 iterations. The results are depicted in Fig. 3. After the final iteration of the proposed algorithm, we evaluate the performance of the control law by simulating 1000 trajectories for time intervals long enough to achieve touchdown (see Fig. 3b). For $t > T$, we use the same control law as for $t = T$. In contrast to the deterministic setting, the cost given by Eq. (12) can be used to shape trajectories based on whether we place more importance on minimizing the touchdown speed even for worst-case disturbances (at the expense of increased fuel usage), or whether fuel expenditure is critical and should be thus done in a parsimonious manner. Two such cases are depicted in Fig. 4. In case I, fuel is relatively expensive, and thus for some noise profiles the spacecraft has a high touchdown speed (mean -0.62 m/s, variance 0.061 m/s). In contrast, case II corresponds to relatively cheap fuel, and thus the algorithm increases the effort to contain the spread of trajectories, thus avoiding a crashing impact even for bad noise profiles (mean touchdown speed -0.55 m/s, variance 0.006 m/s). This increases the fuel expenditure (used fuel mass of case II: mean 43.2 kg variance 1.5 kg, as opposed to 39.7/1.1 kg for case I).
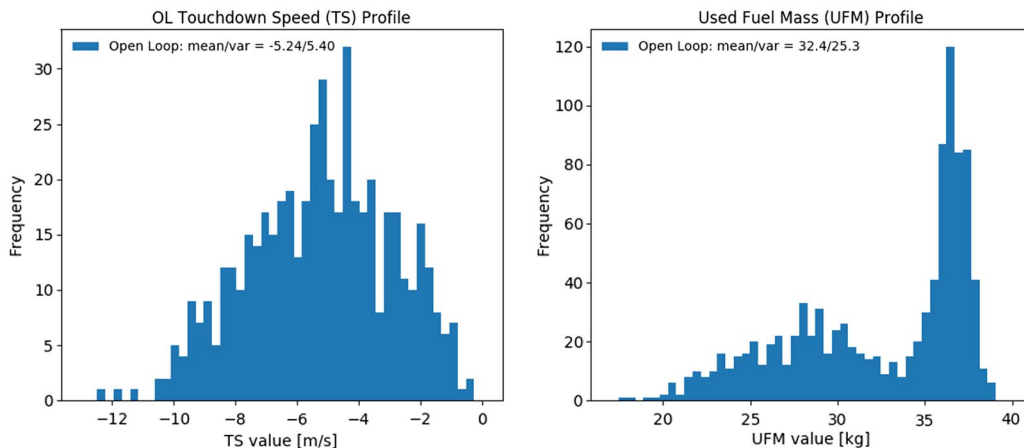
D. Comparison to the Apollo Powered Descent Guidance Law

The venerable Apollo powered descent guidance (APDG) is a simple feedback control law that receives initial and terminal conditions for a given guidance phase and outputs an acceleration profile that meets these conditions. Expressed in feedback form for the thrust, the APDG law reads [14]

$$u(t) = -\frac{6\,m(t)}{t_{go}}(v^* - v(t)) + \frac{12\,m(t)}{t_{go}^2}(r^* - r(t) - v(t)t_{go}) + u_f^* \quad (22)$$
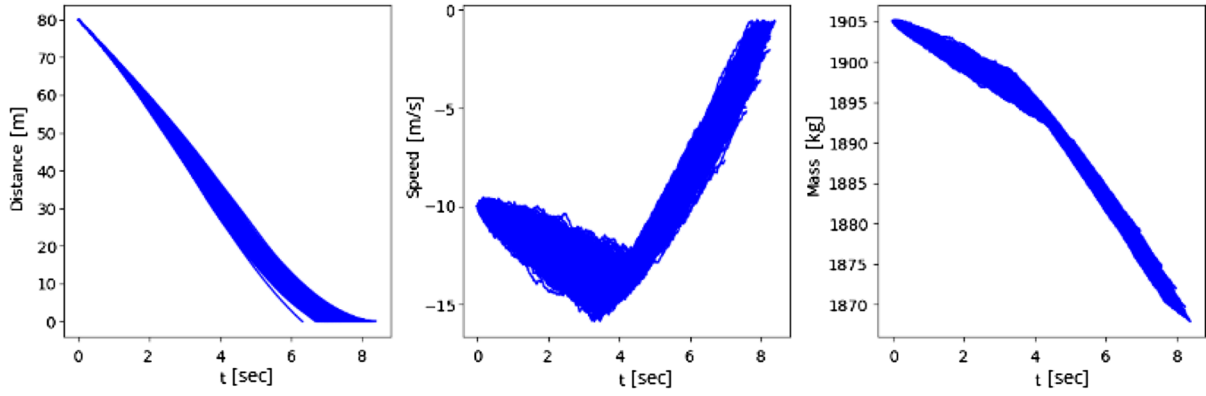

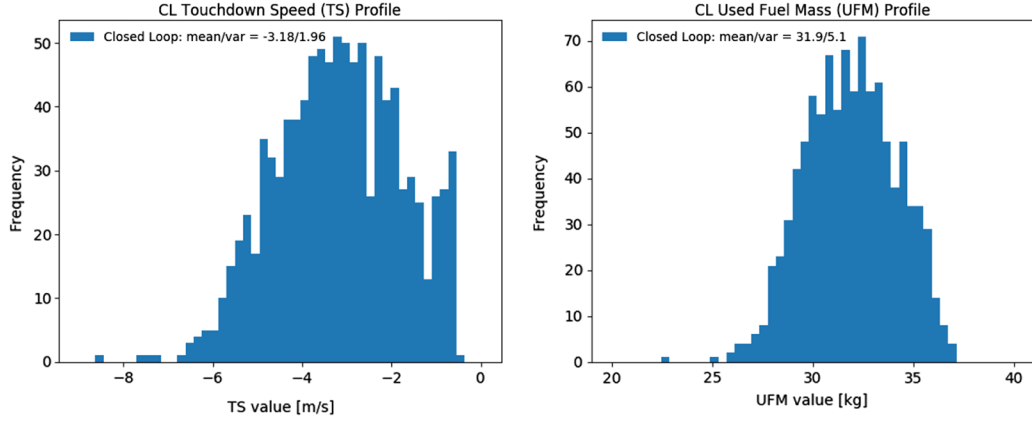a) Trajectories resulting from the open-loop implementation of control law (5)

b) Touchdown speed profile (left) and fuel consumption profile (right)

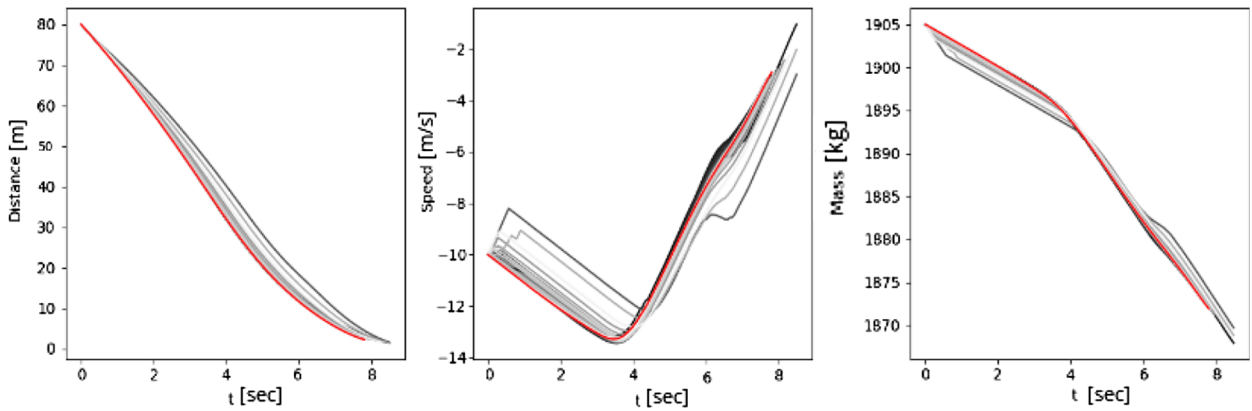**Fig. 1   Solution of the open-loop implementation of control law (5).**

a) Trajectories resulting from the closed-loop implementation of control law (5)
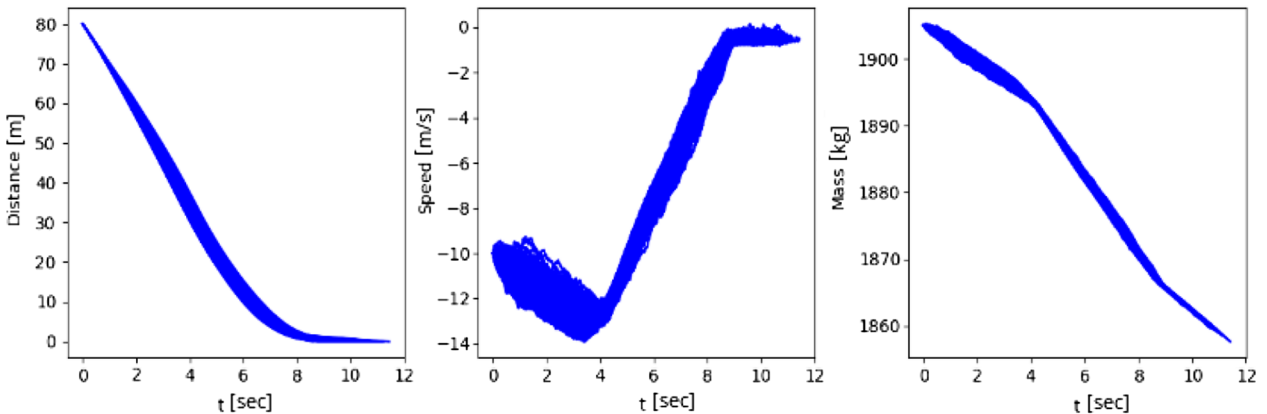

b) Touchdown speed profile (left) and fuel consumption profile (right)

Fig. 2 Solution of the closed-loop implementation of control law (5).

a) The mean of the controlled system trajectories of each iteration (grayscale) and after the final iteration (red)

b) Optimally controlled trajectories, simulated until touchdown

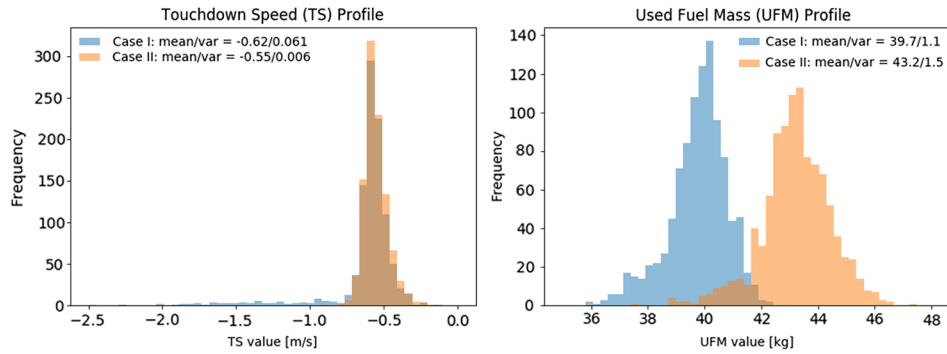Fig. 3 Solution of the proposed algorithm.

Fig. 4   Comparison between the touchdown speed and fuel consumption profiles for cases I and II.

Table 1   Comparison of all methods

Method	Touchdown speed [m/s] (mean/variance)	Fuel usage [kg] (mean/variance)	Touchdown percentage	Crash percentage
Deterministic, open loop	−5.24/5.40	32.4/25.3	50.3	95.0
Deterministic, closed loop	−3.18/1.96	31.9/5.1	100	86.8
Stochastic, case I	−0.62/0.061	39.7/1.1	100	2.3
Stochastic, case II	−0.55/0.006	43.2/1.5	100	0[a]
APDG, unconstrained thrust	−0.08/0.003	37.9/0.002	100	0[a]
APDG, constrained thrust	−1.96/1.302	34.8/3.32	100	61.9

[a]No crashes occur during simulation.

wherein v^* and r^* are the terminal conditions to be met, t_{go} is a specified time-to-go to the target condition, and u_f^* is a specified final thrust acceleration vector. The main weakness of the APDG law is that it does not take thrust constraints (i.e., lower and upper bounds) into account, and exhibits sensitivity with respect to the parameter t_{go}. In fact, depending on the initial conditions and t_{go}, saturation in the thrust magnitude may occur, leading to a miss in the prescribed terminal conditions [14]. In this paper, we simulated a one-dimensional

expression of Eq. (22), implemented in closed-loop fashion (i.e., by recalculating the thrust profile at each instant of time). We used the same initial conditions as previously, and set $v^* = r^* = 0$, while t_{go} and u_f^* were optimized to yield the best results. We do note, however, that we enforced the same upper bound on the allowable time for the duration of the descent; that is, the maximum t_{go} was equal to $T = 8.5$ s. We tested the APDG law for both cases of unconstrained and constrained thrust.
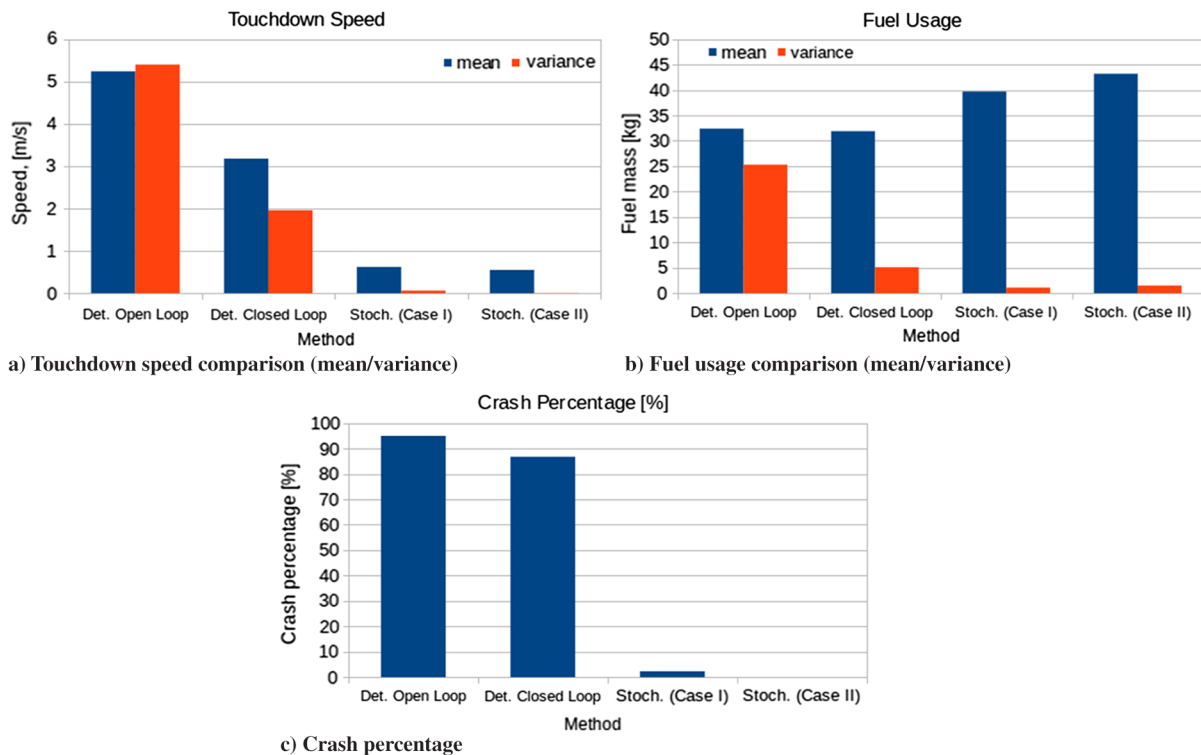

a) Touchdown speed comparison (mean/variance)

b) Fuel usage comparison (mean/variance)

c) Crash percentage

Fig. 5   Performance comparison of the three methods.

Assuming that any touchdown speed higher than 5 ft/s (1.52 m/s) is considered a crash,[§] we may summarize the comparison results in Table 1. The results are also shown in Fig. 5. For case II of the proposed method, no crashes occur during simulation; the Chebyshev-Cantelli Inequality gives an upper bound of 0.6% on the probability of a crash occurring in this case. The APDG law, when allowed to use unconstrained thrust, offers exceptional performance, but yields a thrust magnitude that can be as high as five times the upper bound in the final moments before touchdown. Saturating the values between $u_{\min}$ and $u_{\max}$ deteriorates the performance significantly, even if optimization over the parameters t_{go} and u_f^* is performed. The optimal values were 8.5 s (the maximum allowable value, which was similarly imposed to the proposed algorithm) and $u_{\max}$, respectively. We conjecture that the APDG performance could be improved by increasing the upper bound on the time, but used the same bound in order to be as fair as possible for all methods.

The superiority of the proposed algorithm in providing a control solution leading to a smooth landing, which is furthermore robust to stochastic disturbances, is evident, as it offers a much lower mean and variance on the touchdown speed. Depending on given safety specifications, we can further reduce this mean and variance, thus gaining a more robust, safer controller, at the expense of slightly increased fuel expenditure. In addition, all computations can be performed off-line, leading to a simple implementation, which does not require high on-board computational capability for the spacecraft. Furthermore, this result illustrates an important finding, namely, that deterministic model predictive control, in contrast to the proposed approach, might fail to perform adequately in the presence of stochastic disturbances.

E.   Note on Convergence

The main sources of errors in the proposed numerical algorithm consist of 1) the time discretization scheme, and 2) the LSMC method of approximating conditional expectations. The time discretization error in most FBSDE numerical schemes decreases at a rate $\sqrt{N}$, where N is the number of (equidistant) time steps [26]. The error due to the LSMC scheme can be reduced as the number of basis functions tends to infinity and is inversely proportional to the square root of the number of realizations, $\sqrt{M}$ [27]. Note that the PDE-FBSDE problem equivalence, illustrated by the nonlinear Feynman-Kac lemma that links Eqs. (17) and (18), being exact, does not introduce any errors. Proving the overall convergence of the proposed scheme, however, is more involved, and is part of our on-going research.

V.   Conclusions

This paper has studied a stochastic version of the SLP, in which the objective is to find the fuel-optimal thrust profile needed for a spacecraft to land on a planet. The deterministic version of this problem offers a closed-form solution. To address the case of a stochastic environment, the framework of optimal control via forward and backward sampling, developed recently by the authors, was modified to fit the particular requirements of this problem. The modified algorithm yields a feedback control law that is shown to outperform both the open loop, as well as the closed-loop implementation of the deterministic control law. More important, this illustrates that deterministic MPC, in contrast to the proposed approach, may fail to perform adequately in the presence of stochastic disturbances.

Acknowledgments

[§]See NASA specifications, e.g., https://www.nasa.gov/mission_pages/station/structure/elements/soyuz/landing.html.

References

[1] Dixon, M., Edelbaum, T., Potter, J., and Vandervelde, W., "Fuel Optimal Reorientation of Axisymmetric Spacecraft," *Journal of Spacecraft and Rockets*, Vol. 7, No. 11, 1970, pp. 1345–1351.
doi:10.2514/3.30168

[2] Seywald, H., Kumar, R. R., Deshpande, S. S., and Heck, M. L., "Minimum Fuel Spacecraft Reorientation," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 1, 1994, pp. 21–29.
doi:10.2514/3.21154

[3] Ross, M. I., "How to Find Minimum-Fuel Controllers," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2004-5346, Aug. 2004.
doi:10.2514/6.2004-5346

[4] Miele, A., "The Calculus of Variations in Applied Aerodynamics and Flight Mechanics," *Optimization Techniques: With Applications to Aerospace Systems*, Vol. 5, 1962, pp. 99–170.
doi:10.1016/S0076-5392(08)62092-5

[5] Miele, A., "Extremization of Linear Integrals by Green's Theorem," *Optimization Techniques: With Applications to Aerospace Systems*, Vol. 5, 1962, pp. 69–98.
doi:10.1016/S0076-5392(08)62091-3

[6] Fleming, W., and Rishel, R., *Deterministic and Stochastic Optimal Control*, Springer–Verlag, New York, 1975, Chap. 2.
doi:10.1007/978-1-4612-6380-7

[7] Meditch, J., "On the Problem of Optimal Thrust Programming for a Lunar Soft Landing," *IEEE Transactions on Automatic Control*, Vol. 9, No. 4, 1964, pp. 477–484.
doi:10.1109/TAC.1964.1105758

[8] Açikmeşe, 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

[9] Dueri, D., Açikmeşe, B., Scharf, D., and Harris, M., "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

[10] Scharf, D., Açikmeşe, B., Dueri, D., Benito, J., and Cosoliva, 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

[11] Wong, E., Singh, G., and Masciarelli, J., "Guidance and Control Design for Hazard Avoidance and Safe Landing on Mars," *Journal of Spacecraft and Rockets*, Vol. 43, No. 2, 2006, pp. 378–384.
doi:10.2514/1.19220

[12] Pascucci, C. A., Bennani, S., and Bemporad, A., "Model Predictive Control for Powered Descent Guidance and Control," *European Control Conference*, IEEE, July 2015, pp. 1388–1393.
doi:10.1109/ECC.2015.7330732

[13] Lu, P., "Propellant-Optimal Powered Descent Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 4, 2018, pp. 813–826.
doi:10.2514/1.G003243

[14] Lu, P., "Fuel-Optimal and Apollo Powered Descent Guidance Compared for High-Mass Mars Mission," *AAS Guidance and Control Conference*, Breckenridge, CO, Feb. 2018.

[15] Exarchos, I., and Theodorou, E., "Stochastic Optimal Control via Forward and Backward Stochastic Differential Equations and Importance Sampling," *Automatica*, Vol. 87, Jan. 2018, pp. 159–165.
doi:10.1016/j.automatica.2017.09.004

[16] Exarchos, I., "Stochastic Optimal Control—A Forward and Backward Sampling Approach," Ph.D. Thesis, The Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, Dec. 2017.

[17] Exarchos, I., Theodorou, E., and Tsiotras, P., "Stochastic L^1-Optimal Control Using Forward and Backward Sampling," *Systems and Control Letters*, Vol. 118, Aug. 2018, pp. 101–108.
doi:10.1016/j.sysconle.2018.06.005

[18] Sostaric, R., "Powered Descent Trajectory Guidance and Some Considerations for Human Lunar Landing," *30th Annual AAS Guidance and Control Conference*, Breckenridge, CO, Feb. 2007.

[19] Fleming, W., "Exit Probabilities and Optimal Stochastic Control," *Applied Mathematics and Optimization*, Vol. 9, No. 1, 1971, pp. 329–346.
doi:10.1007/BF01442148

[20] Yong, J., and Zhou, X. Y., *Stochastic Controls: Hamiltonian Systems and HJB Equations*, Springer–Verlag, New York, 1999, Chap. 4.
doi:10.1007/978-1-4612-1466-3

[21] Fleming, W., and Soner, H., *Controlled Markov Processes and Viscosity Solutions*, 2nd ed., Stochastic Modelling and Applied Probability, Springer, New York, 2006, Chap. IV.
doi:10.1007/0-387-31071-1

[22] Kloeden, P., and Platen, E., *Numerical Solution of Stochastic Differential Equations*, 3rd ed., Vol. 23, Applications in Mathematics, Stochastic Modelling and Applied Probability, Springer–Verlag, Berlin, 1999, Chap. 9.
doi:10.1007/978-3-662-12616-5

[23] El Karoui, N., Peng, S., and Quenez, M. C., "Backward Stochastic Differential Equations in Finance," *Mathematical Finance*, Vol. 7, No. 1, 1997.
doi:10.1111/mafi.1997.7.issue-1

[24] Ma, J., and Yong, J., *Forward-Backward Stochastic Differential Equations and their Applications*, Springer–Verlag, Berlin, 1999, Chaps. 1–3.
doi:10.1007/978-3-540-48831-6

[25] Zhang, J., *Backward Stochastic Differential Equations*, Probability Theory and Stochastic Modelling, Springer Science+Business Media LLC, New York, 2017, Chaps. 4, 5, 8.
doi:10.1007/978-1-4939-7256-2

[26] Lemor, J. P., Gobet, E., and Warin, X., "Rate of Convergence of an Empirical Regression Method for Solving Generalized Backward Stochastic Differential Equations," *Bernoulli*, Vol. 12, No. 5, 2006, pp. 889–916.
doi:10.3150/bj/1161614951

[27] Xiu, D., *Numerical Methods for Stochastic Computations—A Spectral Method Approach*, Princeton Univ. Press, Princeton, NJ, 2010, p. 3.