Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 55-22 (2022) 91–96

Mars Atmospheric Entry Guidance using MPSP with State and Control Constraints

Prayag Sharma * **Radhakant Padhi** *

* *Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India (e-mail: padhi@iisc.ac.in)*

Abstract: An optimal Mars entry guidance scheme is presented in this paper using the Model Predictive Static Programming (MPSP) technique accounting for the applicable state and control constraints. The guidance scheme is designed to maximize the terminal parachute deployment altitude while applying minimum control effort and satisfying hard constraints on desired terminal conditions such as final velocity and downrange. The proposed guidance computes the optimal bank angle profile to shape the trajectory of the spacecraft. Path constraints on heat rate, dynamic pressure, aerodynamic load, and bounds on bank angle are considered to guide the vehicle safely through the martian atmosphere. Moreover, in order to generate practically realizable bank angle profiles, an additional constraint on the bank angle rate is also applied. Next, using the MPSP technique, the nonlinear constrained optimal control problem is converted into a static quadratic optimization problem with linear equality and inequality constraints to solve it in a computationally efficient manner. The concept of flexible final time MPSP is incorporated to update the final time in an optimal fashion. Numerical simulations illustrate the ability of the proposed method to solve the guidance problem efficiently while satisfying the path and terminal constraints within the desired accuracy.

Keywords: Mars entry guidance, Optimal entry guidance, Model Predictive Static Programming, MPSP guidance

1. INTRODUCTION

Landing exploration-class vehicles safely on the surface of Mars has always been an exciting yet challenging space exploration mission. Starting from Viking in the 1970s to the more recent Perseverance Rover in 2021, so far, only nine spacecraft have successfully landed on the surface of Mars. More than two-thirds of the Mars landing exploration missions ended in failure, which amply demonstrates the mission's difficulty (Li and Jiang (2014)). The execution of the Entry, Descent, and Landing (EDL) sequence is one of the most challenging problems for safely landing spacecraft on the surface of Mars, and the entry phase of this EDL sequence begins when the spacecraft first enters the Martian atmosphere and ends with the parachute deployment is a critical part of the mission. The MSL (Mars Science laboratory) or Curiosity 2012 executed a guided entry phase which significantly helped in reducing the landing error ellipse down to 20 km, which was four times less as compared to the previous missions (Steinfeldt et al. (2010)). However, future Mars missions demand pinpoint landing performance with sub-100-meters accuracy (Steinfeldt et al. (2010)), which requires significant improvement in the current Guidance, Navigation, and Control (GNC) technologies to ensure success in such missions.

Guiding the vehicle safely through the Martian atmosphere is a challenging task. The vehicle enters the Martian atmosphere traveling at nearly 6 km/sec at approximately 125 km above the surface of Mars, and by just modulating the bank angle of the spacecraft, it needs to slow down to safe parachute deployment velocities while traversing the required downrange and crossrange to reach the target location. Moreover, peak heating and loading also occur during this phase and must be maintained within their respective bounds. Other than this, the objective of the entry guidance is to maximize the parachute deployment altitude to enable landing at higher elevation sites (Braun and Manning (2006)). In order to fulfill these objectives, the MSL 2012 adopted the Apollo guidance algorithm with a few modifications (Mendeck and Craig McGrew (2014)), which controlled the spacecraft based on the perturbations experienced during the flight about a pre-designed reference trajectory. However, the ability to compute the reference trajectory in real-time onboard has evident advantages such as robustness against perturbations and generation of safe trajectories in case of off-nominal conditions.

Generating an efficient guidance trajectory that can satisfy these objectives requires solving a constrained non-linear optimal control problem. Numerical methods such as indirect and direct methods have been proposed to solve such optimal control problems. The indirect methods convert the original OCP into a two-point boundary value problem (TPBVP) using Pontryagin's maximum principle (PMP), and the resulting TPBVP can be solved using classical numerical techniques such as transcription and shooting methods. Even though the solution has high accuracy, the resulting TPBVPs are sensitive to the initial guess. To get past this difficulty, Jacob et al. (2014) used particle swarm optimization to generate the initial values of the costate variables, and Zheng et al. (2017) explored the simplicial

10.1016/j.ifacol.2023.03.016

homotopy method (SHM) to circumvent these issues with the initial guess. However, indirect methods are computationally intensive and difficult to implement and therefore are ill-suited for online applications. In comparison, direct methods convert the OCP into a constrained static optimization problem which can then be solved using various non-linear programming (NLP) techniques. To reduce the effects of uncertainties on the resultant trajectory, Li and Peng (2011) implemented desensitized optimal control (DOC) to formulate the trajectory optimization problem and solved using the direct collocation-based NLP method. Similarly, Zhao and Li (2019) formulated the optimization problem using the local collocation methods and a mesh refinement technique to solve the resulting NLP efficiently. However, in all these proposed methods, the time taken by the algorithms to compute the optimal solution still lies in the order of several seconds and thus requires further development to bring the computational time down to ensure real-time capability.

To overcome these limitations, the co-author of this paper proposed a computationally efficient technique called Model Predictive Static Programming (MPSP) in (Oza and Padhi (2012)) to solve non-linear OCPs with terminal constraints. Later, Mondal and Padhi (2020) extended the MPSP technique to solve the path and control constrained OCPs with minimum control effort as the performance index. The core idea behind MPSP lies in deriving a sensitivity relation between the states and the control variables, using which the optimization problem can be written in terms of control variables only, substantially reducing the problem's size. Moreover, these sensitivity matrices can be computed recursively, further decreasing the computational load. Tacking into account these promising benefits, a Mars entry guidance scheme is developed in this paper using the philosophy of constrained MPSP. A cost function maximizing the terminal altitude while applying minimum control effort is selected. Path constraints on heat rate, dynamic pressure, aerodynamic load, and control constraints on bank angle are considered. The final velocity and downrange also need to be satisfied as terminal constraints. An additional constraint on bank angle rate is also placed to generate less aggressive bank angle profiles that can be tracked suitably. The necessary algebra to include the terminal state-dependent term in the cost function is derived, and by integrating the idea of flexible final time MPSP (Maity et al. (2012)), the final time is updated in an optimal fashion. The non-linear constrained OCP is converted into an equivalent quadratic optimization problem with linear equality and inequality constraints using the MPSP technique and solved using the interior-point method.

2. PROBLEM FORMULATION

2.1 State Dynamics

Both the bank angle and the angle of attack can be controlled during the flight to guide the spacecraft safely through the Martian atmosphere, but only the bank angle modulation is generally opted to simplify the implementation of the guidance law (Li and Jiang (2014)), and the angle of attack is kept at a trim (or constant) value throughout the entry phase. In this formulation, only the longitudinal dynamics of the spacecraft are considered using which the energy and downrange of the vehicle can be managed. Since the longitudinal and lateral dynamics are decoupled, cross-range could be controlled by performing bank reversals whenever necessary to maintain the cross-range error within the reversal deadbands (Mendeck and Craig McGrew (2014)). The following equations denote the longitudinal dynamics of the vehicle in the planetocentric frame assuming a spherical non-rotating planet:

$$\begin{bmatrix} \dot{h} \\ \dot{V} \\ \dot{\gamma} \\ \dot{x} \end{bmatrix} = \begin{bmatrix} V \sin\gamma \\ -\dfrac{D}{m} - g_M \sin\gamma \\ \left(\dfrac{V}{r_M + h} - \dfrac{g_M}{V}\right)\cos\gamma + \dfrac{L}{mV}u \\ V\cos\gamma \end{bmatrix} \tag{1}$$

where, x denotes the downrange travelled by the vehicle and $u = \cos\sigma$ is the control input. h is the altitude above the Martian surface, V is the velocity, γ is the flight path angle, σ is the bank angle, r_M is the radius of Mars, $g_M = \mu/(r_M + h)^2$ is the gravitational acceleration at Mars, m specifies the weight of the spacecraft and the aerodynamic lift L and Drag D forces are given by:

$$L = \frac{1}{2}\rho V^2 C_L S_{ref},\ D = \frac{1}{2}\rho V^2 C_D S_{ref} \tag{2}$$

where, C_L and C_D are the aerodynamic lift and drag coefficients which are a function of angle of attack, S_{ref} is the reference surface area of the vehicle, $\rho = \rho_0 e^{-\left(\frac{h}{h_s}\right)}$, ρ_0 is the density at the surface of Mars, and h_s is the height scale density.

2.2 Objective function and Constraints:

The objective of mars exploration missions has always been to land at scientifically interesting landmarks, but almost half of the Martian surface, which lies on a higher elevation, had been inaccessible to the missions before the MSL (Mars Science Laboratory) in 2012 (Braun and Manning (2006)). This was primarily attributed to the difficulty in slowing down the vehicle to parachute deployment velocities at a higher altitude due to the relatively thin Martian atmosphere as compared to the Earth (only $0.01 \times \rho_{earth}$).Therefore, the vehicle needs to slow down to parachute deployment velocities while maximizing its terminal deploy altitude to enable landing at higher elevation sites. Path constraints on heat rate, dynamic pressure, and aerodynamic acceleration must be maintained within their maximum allowable limits $\dot{Q}_{MAX}$, q_{MAX}, and A_{MAX} to guide the vehicle safely throughout the entry phase. Finally, the terminal constraints on final velocity and downrange required to be traveled need to be met at the final time t_f. An equivalent optimal control problem fulfilling the objectives mentioned above can be written in the following form:

$$\min_{u(t),t_f} J = \varsigma \cdot (-h(t_f)) + (1-\varsigma) \cdot \int_{t_0}^{t_f} \left[u^T(t) R(t) u(t)\right] dt \tag{3}$$

Subject to the following constraints:

$$\begin{aligned} &k_q\left(\frac{\rho}{r_n}\right)^N V^M \leq \dot{Q}_{MAX},\ \frac{1}{2}\rho V^2 \leq q_{MAX} \\ &\frac{\sqrt{L^2+D^2}}{m} \leq A_{MAX},\ \sigma_{MIN} \leq \sigma \leq \sigma_{MAX} \\ &V\left(t_f\right) = V_f^*,\ x\left(t_f\right) = x_f^* \end{aligned} \tag{4}$$

k_q, r_n, N, M are constants and $V_f^*,\ x_f^*$ denote the desired terminal velocity and downrange. The second term in the cost function expression (3) is introduced to achieve maximum terminal deploy altitude while applying minimum control effort, and the variable ς can be used to set the priority between the two cost components. Moreover, to generate guidance demands that are suitable for tracking $\dot{\sigma}$ or the rate of change of bank angle is also constrained within its allowable limits as below

$$\begin{aligned} \dot{\sigma}_{MIN} \leq \dot{\sigma} \leq \dot{\sigma}_{MAX} \\ \sigma_{MIN}\left(\sqrt{1-u^2}\right) \leq -\dot{u} \leq \sigma_{MAX}\left(\sqrt{1-u^2}\right) \end{aligned} \tag{5}$$

3. CONSTRAINED MPSP

The discrete time optimal control problem for the above formulation can be expressed in a general form as below

$$\begin{aligned} &\min_{[U,t_f]^{i+1}} J = G\left(X_N^{i+1}\right) + \frac{1}{2}\sum_{k=1}^{N-1}\left(U_k^{i+1}\right)^T R_k^i\left(U_k^{i+1}\right) \\ &\text{s.t. } X_{k+1} = F(X_k, U_k), Y = h_k\left(X_k\right), Y_N^{i+1} = Y_N^* \\ &Z_{\min_k} \leq Z\left(X_k^{i+1}\right) \leq Z_{\max_k}, U_{\min_k} \leq U_k^{i+1} \leq Z_{\max_k} \end{aligned} \tag{6}$$

where, $X_k \in \Re^n, U_k \in \Re^r, Y_k \in \Re^m$ denote the state, control, and output at the kth grid point $k = 1, 2, \ldots, N$. The superscript i denotes the iteration number. The functions $F(\cdot)$, $Y(\cdot)$ correspond to any general nonlinear state dynamics and output equations, and $G(\cdot)$, $Z(\cdot)$ denote the state-dependent terminal cost and state constraint functions respectively. Y_N denotes the output at the final time and Y_N^* being the desired output. $(\cdot)_{\max_k}, (\cdot)_{\min_k}$ are upper and lower bounds on state and control constraints at the kth grid point. The constrained MPSP solution presented here is an iterative algorithm, therefore requiring a guess control history to start, since the control is updated in an iterative manner the corresponding update in state and output at a given grid point k between any two successive iterations can be given by the following relation:

$$\begin{aligned} U_k^{i+1} &\triangleq U_k^i + dU_k^i \\ X_k^{i+1} &\triangleq X_k^i + dX_k^i \\ Y_k^{i+1} &\triangleq Y_k^i + dY_k^i \end{aligned} \tag{7}$$

Where i denotes the previous iteration, and dU_k^i is the control deviation at the kth grid point. U_k^{i+1}, X_k^{i+1}, and Y_k^{i+1} denote the corresponding updated control, state and output equations. It can be observed that the objective function J and constraints in (6) are a function of the updated control as well as the updated state. Using the philosophy of MPSP, the state and output deviations at a given grid point can be written in terms of the control deviations, and this sensitivity relation between the state and control can be used to re-write the problem (6) as a function of updated control only. The objective of constrained MPSP is to compute the desired control deviations at the grid points $k = 1, 2, \ldots, N-1$, such that the updated state X_k^{i+1} and updated control U_k^{i+1} minimizes the cost function J while adhering to the state and control bounds and satisfying the terminal constraints. First, the relation between the state error and control deviation is derived, then using this relation the state and output constraints, and cost function are modified accordingly.

3.1 State error and sensitivity matrix computation

The state dynamics equation in (6) at iteration $(i+1)$ can be expanded using Taylor series expansion to obtain the relation between the deviation in state and the deviation in control as follows:

$$\begin{aligned} X_k^{i+1} &= F\left(X_{k-1}^{i+1}, U_{k-1}^{i+1}\right) \\ X_k^{i+1} &= X_k^i + \left[\frac{\partial F}{\partial X}\right]_{X_{k-1}^i, U_{k-1}^i} dX_{k-1}^i \\ &+ \left[\frac{\partial F}{\partial U}\right]_{X_{k-1}^i, U_{k-1}^i} dU_{k-1}^i + \cdots \\ dX_k^i &= \left[\frac{\partial F}{\partial X}\right]_{[X,U]_{k-1}^i} dX_{k-1}^i + \left[\frac{\partial F}{\partial U}\right]_{[X,U]_{k-1}^i} dU_{k-1}^i \end{aligned} \tag{8}$$

Deviation in the state at the grid $k-1$ (dX_{k-1}^i) can be further expanded in term of dX_{k-2}^i and dU_{k-2}^i, and dX_{k-2}^i can further be expanded in terms of dX_{k-3}^i and dU_{k-3}^i and so on uptill dX_1^i i.e., deviation in state at the initial time as below

$$dX_k^i = \left[A^k\right]^i dX_1^i + \left[B_1^k\right]^i dU_1^i + \ldots + \left[B_{k-1}^k\right]^i dU_{k-1}^i \tag{9}$$

$$\begin{aligned} \left[A^k\right]^i &\triangleq \left[\frac{\partial F}{\partial X}\right]_{X_{k-1}^i, U_{k-1}^i} \cdots \left[\frac{\partial F}{\partial X}\right]_{X_1^i, U_1^i} \\ \left[B_j^k\right]^i &\triangleq \left[\frac{\partial F}{\partial X}\right]_{X_{k-1}^i, U_{k-1}^i} \cdots \left[\frac{\partial F}{\partial X}\right]_{X_{j+1}^i, U_{j+1}^i} \left[\frac{\partial F}{\partial U}\right]_{X_j^i, U_j^i} \end{aligned} \tag{10}$$

Since the initial condition is assumed to be known, therefore $dX_1^i = 0$

$$dX_k^i = \sum_{j=1}^{k-1}\left[B_j^k\right]^i dU_j^i \tag{11}$$

where, $\left[B_j^k\right]^i$ matrix is called the sensitivity matrix of the state at the kth time step due to the deviation in control input at the jth time step. Equation (11) provides the relation between the deviation in state at any given time step due to the deviation in control inputs at all the prior time steps. Moreover, the computation of the sensitivity matrix can be significantly reduced, since it can be computed recursively. An Interested reader can refer to Maity et al. (2012) for further details on the recursion formula.

3.2 Inequality and Equality constraints

Inequality constraints: The state constraint equation is also a function of the updated state or X_k^{i+1}. Using Taylor series expansion and the expression (11), the state constraint function can also be simplified as follows:

$$Z_{\min_k} - Z\left(X_k^i\right) \leq \sum_{j=1}^{k-1} \left[P_j^k\right]^i dU_j^i \leq Z_{\max_k} - Z\left(X_k^i\right) \quad (12)$$

where, $\left[P_j^k\right]^i \triangleq \left[\frac{\partial Z}{\partial X}\right]_{X_k^i} \left[B_j^k\right]^i$, writing the above equation for all nodes i.e., $k = 1, 2, \ldots, N$ in a compact form:

$$Z_{low} = \left[Z_{low_1}^T \ \ldots \ Z_{low_{N-1}}^T\right]^T, \ Z_{up} = \left[Z_{up_1}^T \ \ldots \ Z_{up_{N-1}}^T\right]^T$$
$$Z_{low} \leq P^i dU^i \leq Z_{up}, \ P^i = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ \left[P_1^2\right]^i & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \left[P_1^N\right]^i & \left[P_2^N\right]^i & \cdots & \left[P_{N-1}^N\right]^i \end{bmatrix} \quad (13)$$

where, $Z_{up_k} = Z_{\max_k} - Z\left(X_k^i\right)$, and $dU^i = \left[\left(dU_1^i\right)^T \ \ldots \ \left(dU_{N-1}^i\right)^T\right]^T$. By invoking the relation $U_k^{i+1} \triangleq U_k^i + dU_k^i$ in the equation (13), and combining both the state and control constraints the following matrix inequality can be obtained

$$\begin{aligned} &WU^{i+1} \leq V, \ W \triangleq \left[P^i \ -P^i \ I \ -I\right]^T \\ &V \triangleq \left[Z_{up} - P^i U^i \ \ Z_{low} + P^i U^i \ \ U_{\max} \ \ -U_{\min}\right]^T \end{aligned} \quad (14)$$

Terminal Equality Constraint with Flexible final time: As the time $t \to t_f$ we want the output at the final time t_f to approach its desired value i.e., $Y_N^{i+1} = Y_N^*$. Using the relation in (7) as below

$$dY_N^i = Y_N^* - Y_N^i = \Delta Y_N^* \quad (15)$$

ΔY_N^* can be computed based on the final output at the previous iteration and the desired final output. The expression of dY_N^i can be computed by expanding the output dynamics equation (6) using Taylor series and small error approximation at the final grid point N, (Maity et al. (2012))

$$dY_N^i = \left[\frac{\partial h}{\partial X}\right]_{X_N} \sum_{j=1}^{N-1} \left[B_j^N\right]^i dU_j^i + \left[\frac{\partial h}{\partial X}\right]_{X_N} \dot{X}_N^i dt_f^i \quad (16)$$

where, the first term in the above expression (16) corresponds to the error gathered in the final output due to errors in the control input and the second term refers to the error in the final output due to the sub-optimal selection of the final time in the previous iteration. $\dot{X}_N^i = f\left(X_N, U_N\right)$ and also, t_f follows a similar update rule i.e., $t_f^{i+1} = t_f^i + dt_f^i$. Similar to equation (13), dY_N^i can also be written in compact form as below

$$S^i \triangleq \left[\frac{\partial h}{\partial X}\right]_{X_N} \left[B_1^N \ \ldots \ B_{N-1}^N\right], Q^i \triangleq \left[\frac{\partial h}{\partial X}\right]_{X_N} \dot{X}_N^i \quad (17)$$

$$\underbrace{\left[S^i \ Q^i\right]}_{A_{eq}} \underbrace{\begin{bmatrix} U^{i+1} \\ t_f^{i+1} \end{bmatrix}}_{U_{time}^{i+1}} = \underbrace{\Delta Y_N^* + S^i U^i + Q^i t_f^i}_{B_{eq}} \quad (18)$$

$$A_{eq} U_{time}^{i+1} = B_{eq} \quad (19)$$

The above equation represents the equality constraint that needs to be satisfied. Since there is no inequality constraint on the final time t_f, therefore, the equation (14) can be modified to represent in terms of the new optimization vector U_{time}^{i+1} as below

$$A_{ineq} U_{time}^{i+1} \leq B_{ineq} \quad (20)$$

where, A_{ineq} and B_{ineq} are similar to the matrices W and V the only difference being the addition of the constraint $-\infty \leq t_f^{i+1} \leq \infty$, which implies that the final time is unbounded, the detailed expressions are omitted for brevity.

3.3 Cost Function

The updated final time t_f^{i+1} can also be minimized along with the previous objectives introduced in the cost function (6).

$$J = \underbrace{G\left(X_N^{i+1}\right)}_{J_1} + \underbrace{\frac{1}{2} \sum_{k=1}^{N-1} \left(U_k^{i+1}\right)^T R_k^i \left(U_k^{i+1}\right)}_{J_2} + \underbrace{c_f \left(t_f^{i+1}\right)^2}_{J_3} \quad (21)$$

The term J_1 is dependent on the updated state and can be written in terms of control update using the expression (11), and J_2, J_3 can be re-written in a compact form as below:

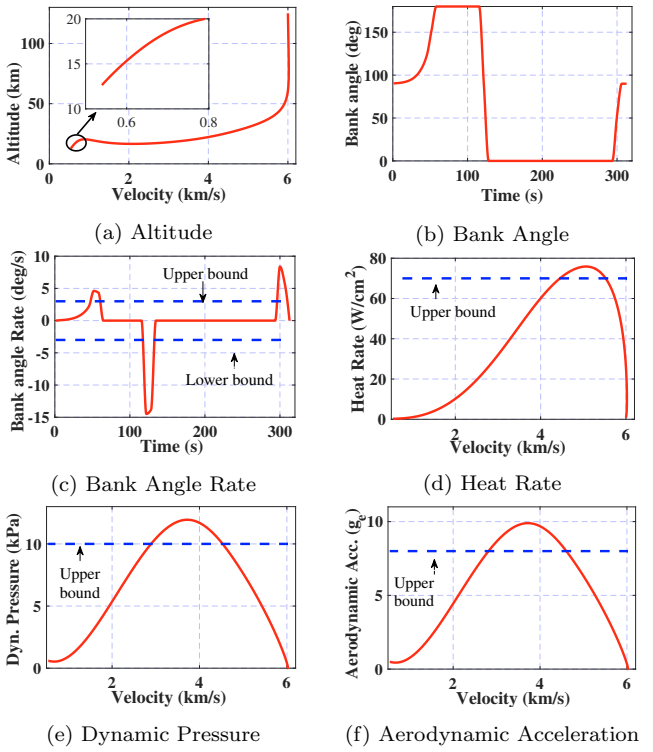
$$\begin{aligned} J &= \frac{1}{2} \left(U_{time}^{i+1}\right)^T R_{time}^i U_{time}^{i+1} + E_{time}^i U_{time}^{i+1} \\ R^i &= \begin{bmatrix} R_1^i & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & R_{N-1}^i & 0 \\ 0 & 0 & 0 & c_f \end{bmatrix}, E_{time}^i = \left[E^i \ 0\right] \\ E^i &\triangleq \left[\frac{\partial G}{\partial X}\right]_{X_N^i} \left[B_1^N \ \ldots \ B_{N-1}^N\right] \end{aligned} \quad (22)$$

Hence, the optimization problem can be written as

$$\min_{U_{time}^{i+1}} \ J = \frac{1}{2} \left(U_{time}^{i+1}\right)^T R_{time}^i U_{time}^{i+1} + E_{time}^i U_{time}^{i+1} \quad (23)$$

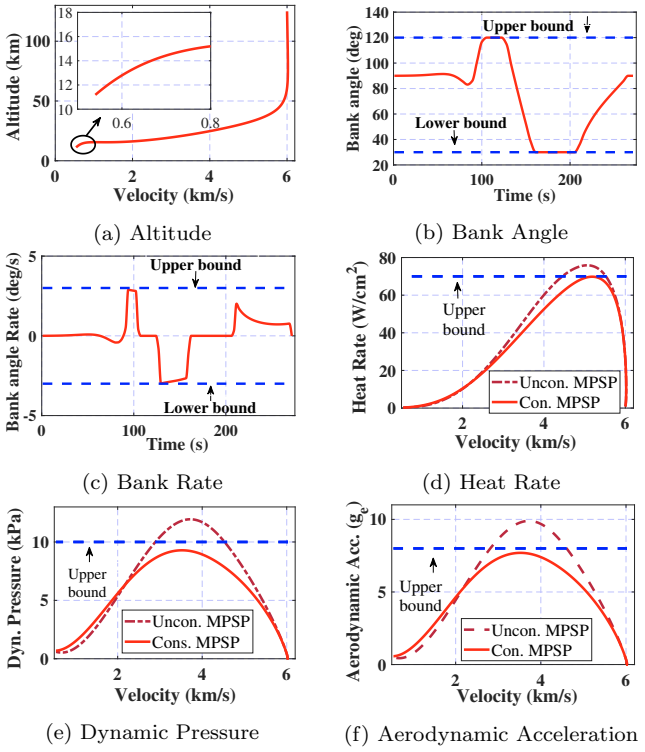
Subject to

$$\begin{aligned} A_{eq} U_{time}^{i+1} &= B_{eq} \\ A_{ineq} U_{time}^{i+1} &\leq B_{ineq} \end{aligned} \quad (24)$$



(a) Altitude (b) Bank Angle (c) Bank Angle Rate (d) Heat Rate (e) Dynamic Pressure (f) Aerodynamic Acceleration

Fig. 1. Solution to the path unconstrained Mars entry guidance using MPSP.



(a) Altitude (b) Bank Angle (c) Bank Rate (d) Heat Rate (e) Dynamic Pressure (f) Aerodynamic Acceleration

Fig. 2. Solution to the path and control constrained Mars entry guidance using MPSP.

The equations (23)-(24) form a quadratic optimization problem with quadratic cost, linear equality and inequality constraints. But since the matrix R^i_{time} can be selected to be a positive semidefinite matrix resulting in a convex quadratic optimization problem which can be solved using efficient solving techniques such as the Interior-Point (IP) method.

4. SIMULATION AND RESULTS

The simulations are performed for a MSL (2012) type vehicle with $S_{ref} = 15.9m^2$, $m = 2920kg$, $C_L = 0.248$, $C_D = 1.45$, $r_M = 3386km$, $\mu = 4.284 \times 10^{13}\ m^3/s^2$, $h_s = 9354m$, $\rho_0 = 0.0158kg/m^3$, $r_n = 0.6m$, $N = 0.5$, $M = 3$ and $k_q = 1.9 \times 10^{-4} kg^{0.5} s^{0.15}/m^{1.15}$. The bounds on the heat rate, dynamic pressure, aerodynamic acceleration, bank angle and bank angle rate were taken to be $\dot{Q}_{MAX} = 70W/cm^2$, $q_{MAX} = 10kPa$, $A_{MAX} = 8g_e$, $\sigma_{MIN} = 30°$, $\sigma_{MAX} = 120°$, $\dot{\sigma}_{MIN} = -3\ deg/s$ and $\dot{\sigma}_{MAX} = 3\ deg/sec$. The initial conditions of the spacecraft were $h_0 = 125km$, $V_0 = 6km/s$, $\gamma = -11.5°$, and $x_0 = 0km$ and the final desired velocity $V^*_f = 540m/s$ and downrange $x^*_f = 935km$. The above values are taken with respect to Zhao and Li (2019). Moreover, constrained MPSP is an iterative algorithm and requires an initial guess of the control trajectory to begin with, a constant control input of $\sigma = 90°$ is passed as the initial guess for the simulations. Based on this control guess the state dynamics are propagated using the RK4 integration method, which are then interpolated at discrete control nodes to solve the optimization problem using the MPSP algorithm. The discrete number of nodes selected for the current simulation were $N = 70$. This process is carried out every iteration till the algorithm converges to the desired terminal conditions within a given threshold, while satisfying all the path constraints. For the current formulation, the iterations were stopped when the error in the terminal constraints $|V_f - V^*_f| < 2\ m/s$ and $|x_f - x^*_f| < 5\ m$. All the simulations were performed on MATLAB R2021a version, running on a Windows 10 desktop with an Intel i5-2400S CPU @ 2.50GHz with 8 GB RAM.

Table 1. Results: Constarined-MPSP and MPSP

Method	h_f (km)	V_f (m/s)	x_f (m)	t_f (s)	Itera-tions	Total Time(s)
Con.MPSP	11.16	539.4	934999	271.4	6	0.55
MPSP	12.64	539.8	934998	312.7	7	0.35

To measure the effectiveness of the constrained - MPSP algorithm, the flexible final time Mars entry guidance (MEG) problem with terminal constraints is first solved without considering the path constraints, with the entire bank angle range, i.e., $0° \leq \sigma \leq 180°$ available to the spacecraft. The corresponding results of the simulation are given in Table 1 and Figure 1. The path unconstrained MPSP is able to reach a maximum terminal altitude of 12.64 km with an error of 0.2 m/s and 2 m in the terminal velocity and downrange, respectively, therefore, satisfying the terminal constraints within the desired accuracy. The converged final time was equal to 312 seconds, and the algorithm took 7 iterations to converge with the total computation time of only 0.35 seconds. However, while achieving these objectives, one can observe from the figures (1c)-(1f) that the path constraints are clearly violated. The values of these control and path constraints, i.e., bank angle rate, heat rate, dynamic pressure, and aerodynamic load, reach a maximum value of $-14.48\ deg/sec$, 75.87 W/cm^2, 11.94 kPa, and $9.89g_e$ respectively.

The path-constrained problem is addressed next, and the results are shown in Figure 2 and Table 1. Using the proposed flexible final time constrained MPSP algorithm, all the path constraints are satisfied while achieving the terminal constraints. Figures (2d)-(2f) provide the comparison of the path constraints between the unconstrained and constrained MPSP algorithm. As observed, all path constraints remain within their respective bounds using the constrained MPSP algorithm with the maximum value of heat rate, dynamic pressure, and aerodynamic acceleration being 69.8 W/cm^2, 9.28 kPa, and 7.69 g_e respectively. To limit the control authority of the spacecraft, the available bank angle range is reduced from $\sigma \in [0, 180]^\circ$ to $\sigma \in [30, 120]^\circ$ and the bank angle rate should also not exceed ± 3 deg/s, the optimal bank angle profile and the corresponding bank angle rate satisfying these objectives are shown in Figures (2b) and (2c). The bank angle profile generated by the unconstrained case in Fig. 1b demands a rapid transition between 100 to 150 seconds from $\sigma = 180^\circ$ to $\sigma = 0^\circ$, resulting in a required bank angle rate of around 15 deg/s (Fig. 1c), which may lie outside the capability of a spacecraft. However, by introducing a constraint on bank angle rate, such rapid transitions could be avoided, as shown in Fig. (2b) and (2c). The path and control constraints present throughout the trajectory reduce the maximum terminal altitude from 12.64 km to 11.16 km. As it can be seen from Table 1, the terminal constraints are satisfied within the desired threshold with an error of 0.6 m/s and 1 m in the terminal velocity and downrange. The converged final time was $t_f = 271.4$ seconds which is substantially different from the converged final time of the unconstrained problem. Starting from a crude/non-optimal initial guess, the algorithm converges to the optimal solution in just 6 iterations owing to the fast convergence nature of the MPSP algorithm. The total computation time is found to be only 0.55 seconds, indicating the capability of the algorithm to solve the constrained MEG problem in real time. During closed form simulation, the crude initial guess is only passed at the initial time, and the computed optimal solution is used as the initial guess for the successive time steps. The algorithm only required 2 iterations or 185 ms on an average in subsequent time steps to converge to the optimal solution. Moreover, coding with low-level programming languages and implementing on dedicated processors can further reduce computational time. With increasing onboard capabilities of space-grade processors (Lovelly and George (2017)), the proposed guidance strategy is a promising contender for a Mars entry mission.

5. CONCLUSION

This paper presents an optimal guidance scheme for the Mars atmospheric entry phase using the MPSP algorithm accounting for applicable state and control constraints. The constrained nonlinear optimal control problem is converted into an equivalent quadratic optimization problem with linear equality and inequality constraints and solved using the interior point method. The necessary algebra to account for additional terminal cost and flexibility in the final time update is also derived. The computed optimal bank angle profile is able to maintain all the path and control constraints within their respective bounds while satisfying the terminal constraints within the desired accuracy and maximizing the terminal deploy altitude. The additional constraint on the bank angle rate helped generate bank angle profiles that can be tracked by the spacecraft more favorably. Starting from a non-optimal initial guess the algorithm only required six iterations to converge to the optimal solution with a total computation time of less than 600 ms, which is expected to reduce further substantially once (a) it is coded in a low-level language and (b) only a finite iterations are carried out per time step. This clearly demonstrates that the proposed flexible final time constrained MPSP can be used as an effective guidance algorithm for a Mars entry mission.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the lab facility and research grants received from Indian Institute of Science, Bangalore (IE/RERE-21-0516 and IE/RERE-22-0505) to carryout this research.

REFERENCES

Braun, R. and Manning, R. (2006). Mars exploration entry, descent and landing challenges. In *2006 IEEE Aerospace Conference*.

Jacob, G.L., Neeler, G., and Ramanan, R.V. (2014). Mars entry mission bank profile optimization. *Journal of Guidance, Control, and Dynamics*, 37(4), 1305–1316.

Li, S. and Jiang, X. (2014). Review and prospect of guidance and control for mars atmospheric entry. *Progress in Aerospace Sciences*, 69, 40–57.

Li, S. and Peng, Y. (2011). Mars entry trajectory optimization using doc and dcnlp. *Advances in Space Research*, 47(3), 440–452.

Lovelly, T.M. and George, A.D. (2017). Comparative analysis of present and future space-grade processors with device metrics. *Journal of Aerospace Information Systems*, 14(3), 184–197.

Maity, A., Padhi, R., Mallaram, S., and Manickavasagam, M. (2012). Mpsp guidance of a solid motor propelled launch vehicle for a hypersonic mission. In *AIAA Guidance, Navigation, and Control Conference*.

Mendeck, G.F. and Craig McGrew, L. (2014). Entry guidance design and postflight performance for 2011 mars science laboratory mission. *Journal of Spacecraft and Rockets*, 51(4), 1094–1105.

Mondal, S. and Padhi, R. (2020). Constrained Quasi-Spectral MPSP With Application to High-Precision Missile Guidance With Path Constraints. *Journal of Dynamic Systems, Measurement, and Control*, 143(3).

Oza, H.B. and Padhi, R. (2012). Impact-angle-constrained suboptimal model predictive static programming guidance of air-to-ground missiles. *Journal of Guidance, Control, and Dynamics*, 35(1), 153–164.

Steinfeldt, B.A., Grant, M.J., Matz, D.A., Braun, R.D., and Barton, G.H. (2010). Guidance, navigation, and control system performance trades for mars pinpoint landing. *Journal of Spacecraft and Rockets*, 47(1).

Zhao, J. and Li, S. (2019). Mars atmospheric entry trajectory optimization with maximum parachute deployment altitude using adaptive mesh refinement. *Acta Astronautica*, 160, 401–413.

Zheng, Y., Cui, H., and Ai, Y. (2017). Indirect trajectory optimization for mars entry with maximum terminal altitude. *Journal of Spacecraft and Rockets*, 54(5).