Proceedings of the 40th Chinese Control Conference
July 26-28, 2021, Shanghai, China

Landing Trajectory Generation using Gauss Pseudo-spectral Method

Lingxia Mu[1], Jichi Yu[1], Youmin Zhang[2], Lidong Zhang[3], Guo Xie[1], Yuan Zhang[1], Xianghong Xue[1]

1. Department of Automation and Information Engineering, Xi'an University of Technology, Xi'an, 710048 China
E-mail: mulingxia@xaut.edu.cn

2. Department of Mechanical, Industrial & Aerospace Engineering, Concordia University, Montreal, QC H3G 1M8 Canada
E-mail: ymzhang@encs.concordia.ca

3. Laboratory of ATM Avionics Technology, China Aeronautical Radio Electronics Research Institute, Shanghai, 201109, China
E-mail: zhang_lidong@careri.com

Abstract: The soft landing of the Mars lander is one of the key problems of Mars exploration. The lander's dynamic equation is nonlinear and is restricted by various constraints. Analytical solution is hard to be obtained. This paper focuses on planning the Mars landing trajectory by the Gauss pseudo-spectral method. The motion equations during the landing stage is firstly established. The trajectory optimization problem is then established with consideration to dynamic constraints, terminal state constraints, and landing site constraints. The control variables and state variables are discretized using Gauss pseudo-spectral method. Thereby a large-scale sparse nonlinear programming problem is formulated. The problem is then solved by SNOPT solver based on sequential quadratic programming algorithm. Simulation results show the efficiency of the proposed method.

Key Words: Mars soft landing, Trajectory planning, Gauss pseudo-spectral method

1 Introduction

The soft landing of the Mars lander plays an important role in Mars exploration. Landing trajectory is a basis for accurately arriving at the Mars [1]. Trajectory optimization problems are generally nonlinear optimal control problems. In the early days, the dynamic programming method in the optimal control theory is usually used to solve the nonlinear optimal control problem. Recently, due to the fast development of advanced computers, the research on trajectory optimization methods gradually turn to numerical methods [2-4].

Investigation to numerical trajectory optimization methods has always been a challenging task and attracted much attention. Betts [5] systematically summarized the early numerical algorithms for aircraft trajectory optimization, where direct and indirect methods are mainly discussed. Bryson [6] designed a steepest descent algorithm to solve the lift or drag profile of the minimum thermal load for the aircraft under acceleration constraints. This method has an important influence on the early trajectory optimization technology. Hull and Speyer [7] proposed an enhanced Lagrange method for calculating the maximum horizontal and maximum vertical range of the flight trajectory of an aircraft. This method can not only manually change the parameters to achieve a good estimate of the nominal trajectory, but also a penalty function is used to improve the convergence characteristics and prevent the weight from going to infinity. Pesch [8] summarized trajectory optimization algorithms in a series of papers. An overview of aircraft offline and online trajectory optimization is presented. Rao [9] analyzed and summarized the research progress of pseudo-spectral method. In [10], the principles, advantages and disadvantages and applications of common algorithms in aircraft trajectory optimization are analyzed, as well as the application prospects of various methods.

In this paper, a Gauss pseudo-spectral method is proposed to solve the Mars landing trajectory generation problem. First, the motion equations of the lander during the landing stage is firstly established and the trajectory planning problem is formulated. The algorithm design based on pseudo-spectral method are then presented. The time transformation is performed. After that, Lagrange interpolation is used to discretize state variables and control variables. The objective function composed of linear terminal constraints is constructed and discretized. A nonlinear programming (NLP) problem is thereby established. The SNOPT toolbox is used to solve the NLP problem. Finally, simulation results are analyzed.

2 Motion Equations of the Mars Lander

During the landing stage, the motion equations of the Mars lander are modelled as:

$$\dot{V} = -\frac{D}{m} - g\sin\theta \tag{1}$$

$$\dot{\theta} = \frac{L}{mV} - \left(\frac{g}{V} - \frac{V}{R+h}\right)\cos\theta \tag{2}$$

$$\dot{h} = V\sin\theta \tag{3}$$

$$\dot{x} = \frac{VR}{R+h}\cos\theta \tag{4}$$

where V,θ,h and x are the state variables representing lander's velocity, flight-path angle, altitude, and longitudinal range, respectively. D and L are drag and lift forces, which are determined by the angle-of-attack α, taken as a control variable, as well as by drag and lift coefficients (C_D, C_L), and dynamic pressure q.

$$D = C_D S q \tag{5}$$

*This work was supported by the National Natural Science Foundation of China (No. 61903297 and No. 61833013), Natural Science Foundation of Shaanxi Province (2019JQ-751), Natural Science Foundation of Shaanxi Provincial Department of Education (19JK0569), and the Natural Sciences and Engineering Research Council of Canada (Corresponding author: Youmin Zhang).

$$L = C_L S q \tag{6}$$

$$C_D = C_N \sin\alpha + C_A \cos\alpha \tag{7}$$

$$C_L = C_N \cos\alpha - C_A \sin\alpha \tag{8}$$

$$q = \frac{1}{2}\rho V^2 \tag{9}$$

The gravitational acceleration g is calculated by Newton's inverse-square force law

$$g = g_0 \left(\frac{R}{R+h}\right)^2 \tag{10}$$

In Eq. (9), ρ denotes the Mars atmospheric density, which is modelled as a piecewise function of altitude h [11]:

$$\rho = \begin{cases} 3.7\times10^{-4} e^{\left(-1.371\times10^{-4}\left(h-40\times10^3\right)\right)}, & 40km \le 130km \\ 2.8\times10^{-3} e^{\left(-1.012\times10^{-4}\left(h-20\times10^3\right)\right)}, & 20km < h \le 40km \\ 0.01417e^{\left(-0.8109\times10^{-4} h\right)}, & others \end{cases} \tag{11}$$

The value of other parameters are listed in Table 1.

Table 1 The value of parameters in the Lander's model

Parameters	Values	Physical meaning
C_A	1.6	Axial force coefficient
C_N	0	Normal force coefficient
R	3395km	Radius of Mars
m	2200kg	Vehicle mass
S	12.57m^2	Aeroshell reference area
g_0	3.716m/s^2	Gravitational acceleration at the Martian reference sphere

3 Problem Formulation

During the Mars landing phase, it is very important to ensure soft landing for the safe and reliability of Mars exploration task. Hence, the velocity and altitude of the vehicle are restricted at the landing process. When the vehicle reaches to the target altitude h_f, the velocity is subject to certain constraint V_f. In this study, the final landing altitude and velocity are set as $h_f = 7km$, $V_0 = 400m/s$, respectively, while the initial landing altitude and velocity are $h_0 = 125km$, $V_0 = 6000m/s$, respectively.

The trajectory planning problem can be formulated to be an optimal control problem. To this end, by defining the state vector be $z(t) \triangleq [V(t), \theta(t), h(t), x(t)]$, and the system input be $u(t) \triangleq \alpha(t)$, the model (Eqs. (1)-(4)) is firstly written in a general form of nonlinear differential equation

$$\dot{z}(t) = f\left(z(t), u(t), t\right) \quad t \in \left[t_0, t_f\right] \tag{12}$$

where t_0 and t_f are initial and final time of the concerned landing phase, respectively.

The boundary constraints are composed of initial and final constraints for state variables and control variables.

$$\begin{aligned} &V(t_0) = V_0, \quad \theta(t_0) = \theta_0, \quad h(t_0) = h_0, \quad x(t_0) = x_0 \\ &V(t_f) = V_f, \quad h(t_f) = h_f \end{aligned} \tag{13}$$

where $V_0 = 6000\, m/s$, $\theta_0 = -15°$, $h_0 = 125\, km$, $x_0 = 0\, km$, $V_f = 400\, m/s$, $h_f = 7\, km$.

The system input α is restricted between -15° and 15°, which can be formulated as

$$u \in U = \left\{u \in R^m : -15° \le u \le 15°\right\} \tag{14}$$

The objective function is formulated as a function of the most important final velocity, altitude, and time.

$$\min_{u(\tau), \tau\in[-1,1]; t_f} \left\{ \begin{aligned} &J = \Phi\left(x(-1), t_0, x(1), t_f\right) \\ &+ \frac{t_f - t_0}{2} \int_{-1}^{1} g\left(x(\tau), u(\tau), \tau; t_0, t_f\right) \end{aligned} \right\} \tag{15}$$

The Mars landing trajectory can be achieved by solving the optimal control problem (OCP) with differential equation constraints described by Eq. (12), boundary constraints denoted by Eq. (13), control input constraints represented by Eq. (14), and objective function described as Eq. (15).

4 Trajectory Planning Algorithm Design

In this paper, Mars landing trajectory planning algorithm based on the Gauss pseudo-spectral method is developed. Firstly, the original time region is scaled to the standard range of [-1,1]. Then, the state and input variable are discretized using Lagrange interpolation method. Meanwhile, the objective function is discretized. After that, a nonlinear programming problem (NLP) can be formulated. The SNOPT toolbox [13] is introduced to solve the formulated NLP problem.

4.1 Time range scaling

The time range of the original OCP problem is $\left[t_0, t_f\right]$. By introducing a new variable τ which lies in the range of [-1,1], then the time is scaled by:

$$t = \frac{t_f - t_0}{2}\tau + \frac{t_f + t_0}{2} \tag{16}$$

Correspondingly, the original OCP problem is reformulated as:

$$\min_{u(\tau), \tau\in[-1,1]; t_f} \left\{J = \Phi\left(x(-1), t_0, x(1), t_f\right)\right\} \tag{17}$$

$$s.t \quad \frac{dx}{d\tau} = \frac{t_f - t_0}{2} f\left(x(\tau), u(\tau), \tau; t_0, t_f\right), \ \tau \in [-1,1] \tag{18}$$

$$E\left(x(-1), t_0, x(-1), t_f\right) = 0, \tag{19}$$

$$C\left(x(\tau), u(\tau), \tau; t_0, t_f\right) \le 0, \quad \tau \in [-1,1] \tag{20}$$

4.2 Variable discretization

In this paper, Lagrange interpolation is used to discretize the continuous variable in the pseudo-spectral method. To this end, considering N+1 points, i.e., $\left\{(\tau_1, y_1), (\tau_2, y_2), \ldots, (\tau_{N+1}, y_{N+1})\right\}$, Nth-order polynomial interpolation can be obtained as:

$$y(\tau) \approx Y(\tau) = \sum_{i=1}^{N+1} \hat{L}_i(\tau) y_i \tag{21}$$

where $Y(\tau)$ is Lagrange polynomial, and $\hat{L}_i(\tau)$ is the basis function described as:

$$\hat{L}_i(\tau) = \prod_{j=1, j\ne i}^{N+1} \frac{\tau - \tau_j}{\tau_i - \tau_j} \tag{22}$$

with the feature shown below

$$\hat{L}_i\left(\tau_j\right)=\delta_{ij}=\begin{cases}1, & i=j\\ 0, & i\neq j\end{cases} \tag{23}$$

At the discrete points, the Lagrange polynomial meets the following relationship:

$$Y\left(\tau_i\right)=y_i,\ i=1,2,\ldots,N+1 \tag{24}$$

The Runge phenomenon might take place when using the equal-space distributed points as the discrete point. To avoid this phenomenon, Gauss discrete points are used in this paper, where the distribution of the points is non-equal spaced. The Gauss discrete points are achieved by computing the zeros of Legendre polynomial. *N*th-order Legendre polynomial is defined as [12]:

$$P_N\left(\tau\right)=\frac{1}{2^N N!}\frac{d^N}{d\tau^N}\left[\left(\tau^2-1\right)^N\right] \tag{25}$$

To compute the zeros of Eq. (25), the following recursive method is used

$$\begin{aligned}&P_0\left(x\right)=1\\ &P_1\left(x\right)=x\\ &\left(n+1\right)P_{n+1}\left(x\right)=\left(2n+1\right)xP_n\left(x\right)-nP_{n-1}\left(x\right),\quad n\geq 1\end{aligned} \tag{26}$$

N zeros can be obtained for the *N*th-order Legendre polynomial. If N is larger than 45, non-real zeros can be shown, hence, N is set less than 45 in this paper. The obtained discrete points within the range of [-1,1] are defined as $\tau_i,\ i=1,2,\ldots,N$, which is known as LG collocation. In addition, by adding the initial point $\tau_0=-1$ and final point $\tau_{N+1}=\tau_f=1$, *N*+2 nodes are obtained totally, i.e., $\left\{\tau_0,\tau_1,\ldots,\tau_N,\tau_{N+1}\right\}$. If N is set as N = 3, 5, 10, 30, respectively and by solving Eq. (25), the obtained nodes are presented in Fig. 1. As can be observed from Fig. 2, the distribution of LG nodes at the two end points is denser, while the distribution of LG nodes at the middle zero point is relatively sparse, which can effectively suppress the Runge phenomenon.

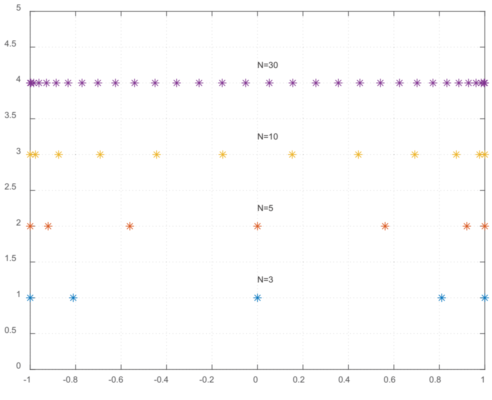

Fig. 1 LG collocation when *N*=3, 5, 10, 30, respectively

Based on the selected nodes, the system states and control inputs can be discretized. Firstly, by taking *N*+2 LG nodes and performing Lagrange interpolation on the state variables at the first *N*+1 LG nodes $\left\{\tau_0,\tau_1,\ldots,\tau_N\right\}$, one obtains:

$$x\left(\tau\right)\approx X\left(\tau\right)=\sum_{i=0}^{N}L_i\left(\tau\right)X\left(\tau_i\right)=\sum_{i=0}^{N}L_i\left(\tau\right)X_i \tag{27}$$

$$L_i\left(\tau\right)=\prod_{j=0,j\neq i}^{N}\frac{\tau-\tau_j}{\tau_i-\tau_j}=\frac{b\left(\tau\right)}{\left(\tau-\tau_i\right)\dot{b}\left(\tau_i\right)} \tag{28}$$

where

$$b\left(\tau\right)=\prod_{i=0}^{N}\left(\tau-\tau_i\right) \tag{29}$$

Then, by performing Lagrange interpolation on the control inputs at the N LG nodes $\left\{\tau_1,\ldots,\tau_N\right\}$, one gets:

$$u\left(\tau\right)\approx U\left(\tau\right)=\sum_{i=0}^{N}\tilde{L}_i\left(\tau\right)U\left(\tau_i\right)=\sum_{i=0}^{N}\tilde{L}_i\left(\tau\right)U_i \tag{30}$$

$$\tilde{L}_i\left(\tau\right)=\prod_{j=1,j\neq i}^{N}\frac{\tau-\tau_j}{\tau_i-\tau_j} \tag{31}$$

4.3 Constraints and objective function reformulation

By differentiating Eq. (27), the derivative of the state variables can be obtained as:

$$\dot{x}\left(\tau\right)\approx\dot{X}\left(\tau\right)=\sum_{i=0}^{N}\dot{L}_i\left(\tau\right)X_i \tag{32}$$

Substituting LG nodes into Eq. (32), i.e., letting $\tau=\tau_N$, one can obtain

$$\dot{x}\left(\tau_k\right)\approx\dot{X}\left(\tau_k\right)=\sum_{i=0}^{N}\dot{L}_i\left(\tau_k\right)X_i=\sum_{i=0}^{N}D_{k,i}X_i \tag{33}$$

$$D_{k,i}=\dot{L}_i\left(\tau_k\right) \tag{34}$$

Using Eqs. (28)-(29), $D_{k,i}$ can be obtained as:

$$D_{k,i}=\dot{L}_i\left(\tau_k\right)=\begin{cases}\dfrac{\dot{b}\left(\tau_k\right)}{\dot{b}\left(\tau_i\right)\left(\tau_k-\tau_i\right)}, & k\neq i\\ \dfrac{\ddot{b}\left(\tau\right)}{2\dot{b}\left(\tau_i\right)}, & k=i\end{cases} \tag{35}$$

where $i=0,1,\cdots,N;k=1,2,\cdots,N$. $D_{k,i}$ is a $N\times\left(N+1\right)$ matrix, which is called the state differential matrix. Now, the constraints on state derivate as described by Eq. (18) are transformed as:

$$\sum_{i=0}^{N}D_{k,i}X_i=\frac{t_f-t_0}{2}f\left(X_k,U_k,\tau_k;t_0,t_f\right),\ k=1,2,...,N \tag{36}$$

By this means, the original differential constraint is replaced by the algebraic constraints Eq. (36). The state differential matrix $D_{k,i}$ is usually known for a specific system. Hence, the left side of the equation is the linear combination of state variables.

In the Gauss pseudo-spectral method, Gauss integral formula is used for the integral term in the objective function, by which the highest integration accuracy can be achieved. This is also one of the advantages of the Gauss pseudo-spectral method. According to the Gauss integral rule, the integral part in Eq. (15) is written as:

$$\begin{aligned}&\int_{-1}^{1}g\left(x\left(\tau\right),u\left(\tau\right),\tau;t_0,t_f\right)d\tau\\ &\approx\sum_{k=1}^{N}w_k g\left(x\left(\tau_k\right),u\left(\tau_k\right),\tau_k;t_0,t_f\right)\\ &=\sum_{k=1}^{N}w_k g\left(X_k,U_k,\tau_k;t_0,t_f\right)\end{aligned} \tag{37}$$

where w_k, $k=1,2,...,N$ is the weight of Gauss integration, which is calculated by

$$w_k = \frac{1}{\left(1-\tau_k^2\right)\left(\dot{P}_N\left(\tau_k\right)\right)^2}, \qquad k=1,2,...,N \qquad (38)$$

At last, the trajectory constraint is reformulated as

$$C\left(X_k,U_k,\tau_k;t_0,t_f\right)\le 0, \qquad k=1,2,...,N \qquad (39)$$

According to the Gauss integral formula, the terminal state variable can be calculated by the following formula

$$\begin{aligned} X_f &= X_{N+1} \\ &= X_0 + \frac{t_f-t_0}{2}\int_{\tau_0}^{\tau_f} f\left(x(\tau),u(\tau),\tau;t_0,t_f\right)d\tau \\ &\approx X_0 + \frac{t_f-t_0}{2}\sum_{k=1}^{N} w_k f\left(x(\tau_k),u(\tau_k),\tau_k;t_0,t_f\right) \\ &= X_0 + \frac{t_f-t_0}{2}\sum_{k=1}^{N} w_k f\left(X_k,U_k,\tau_k;t_0,t_f\right) \end{aligned} \qquad (40)$$

$$\begin{aligned} 0 &= X_f - X_0 + \frac{t_f-t_0}{2}\sum_{k=1}^{N} w_k f\left(x(\tau_k),u(\tau_k),\tau_k;t_0,t_f\right) \\ &= X_f - X_0 - \sum_{k=1}^{N} w_k \sum_{k=1}^{N} D_{k,i} X_i \\ &= X_f - X_0 - \sum_{k=1}^{N} X_i \sum_{k=1}^{N} w_k D_{k,i} \end{aligned} \qquad (41)$$

Finally, the terminal constraint is discretized as

$$X_f - X_0 - \sum_{k=1}^{N} X_i \sum_{k=1}^{N} w_k D_{k,i} = 0 \qquad (42)$$

The terminal constraints are now described by a linear algebra equation.

4.4 Nonlinear programming problem formulation and solver

The decision variable of the trajectory optimization problem is composed of the terminal time t_f , the state variable z_i and the control variable u_i at each LG node. The decision variable vector is constructed as:

$$K = \left[t_f, z_0^T, u_0^T, \dots, z_N^T, u_{cN}^T\right] \qquad (43)$$

The objective function is constructed by the linear combination of the terminal constraints of V_f and h_f .

$$\begin{aligned} \min_{u(\tau),\tau\in[0,1];t_f} J &= \left(V_f - V_0 - \sum_{k=1}^{N} V_i \sum_{k=1}^{N} w_k D_{k,i}\right) \\ &+ \left(h_f - h_0 - \sum_{k=1}^{N} h_i \sum_{k=1}^{N} w_k D_{k,i}\right) \end{aligned} \qquad (44)$$

$$\text{s.}t \sum_{i=0}^{N} D_{k,i} Z_i = \frac{t_f-t_0}{2} f\left(Z_k,U_k,\tau_k;t_0,t_f\right), k=1,2,...,N \quad (45)$$

$$u \in U = \left\{u \in R^m : -15° \le u \le 15°\right\}$$

(46)

$$z_0 = [6000,\ -15°,\ 125000,\ 0] \qquad (47)$$

$$V_0 = 6000,\ V_f = 400,\ h_0 = 125000,\ h_f = 7000 \quad (48)$$

Finally, SNOPT solver is used to solve the above NLP problem. The flow chart of the solving principle is depicted in Fig. 2, which can divided in following five steps.

Step 1: Set decision variable $K = \left[t_f, z_0^T, u_0^T, \dots, z_N^T, u_{cN}^T\right]$. Firstly, the initial value and limitation of the decision variable are set.

Step 2: Substituting the discrete decision variables into the Gauss pseudo-spectral algorithm, the discrete objective function, constraint function, and state constraints are obtained. The objective function and the constraint vector are formed into a matrix, and then it is calculated to obtain the Jacobian matrix.

Step 3: The sequential quadratic algorithm based SNOPT toolbox is used to solve the problem, an one-dimensional search is performed to obtain the next iteration point.

Step 4: Judge whether the iteration point has reached the convergence accuracy or the maximum number of iterations. If not, execute Step 3, otherwise continue.

Step 5: The value obtained after the appropriate iterations is regarded as the optimal solution.

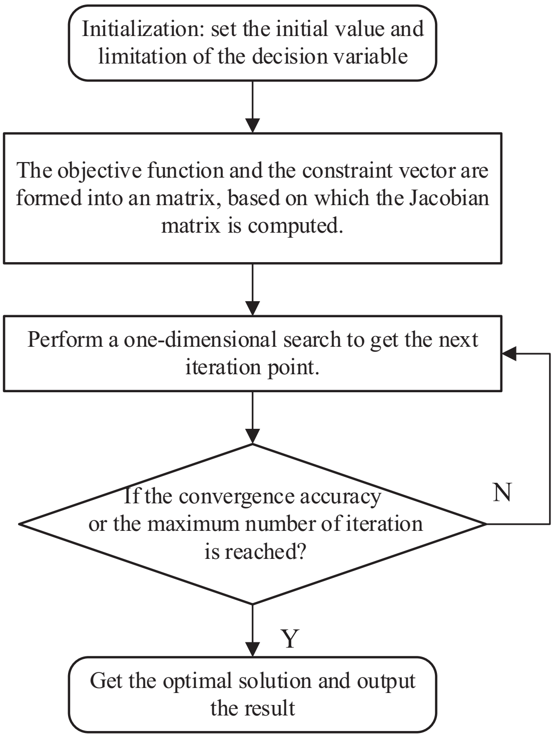
Initialization: set the initial value and limitation of the decision variable
The objective function and the constraint vector are formed into an matrix, based on which the Jacobian matrix is computed.
Perform a one-dimensional search to get the next iteration point.
If the convergence accuracy or the maximum number of iteration is reached?
N
Y
Get the optimal solution and output the result

Fig. 2 Flow chart of the optimal solution procedure

5 Simulation Results and Analysis

In the simulation, the initial state of the Mars Lander is $z_0 = [6000m/s, -15°, 125km, 0km]$. The terminal states are restricted at $z_{N1} = 400m/s$, $z_{N3} = 7km$. The control input α is ranged within $[-15°, 15°]$. The number of discrete points is set as $N = 30$.

The iterative time of the proposed landing trajectory planning method based on pseudo-spectral method is 3.47 s. The obtained trajectory results are shown in Figs. 3-6. The total time to complete the landing mission for the Mars lander is about 102 s. From Fig. 3, it can be observed that the velocity of the lander decreases steadily at first 60 s, and decreases sharply in the next 40 s. Fig. 4 presents the flight path angle history during the landing mission. It is in a reasonable range. In Fig. 5, the history of altitude is given, which shows that in the last 20 s the speed of the altitude drop slows down. Fig. 6 shows the corresponding longitudinal range of the lander.

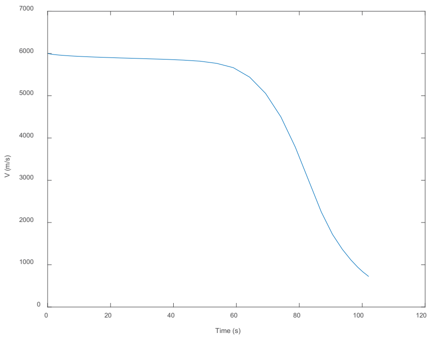

Fig. 3 The history of landing velocity

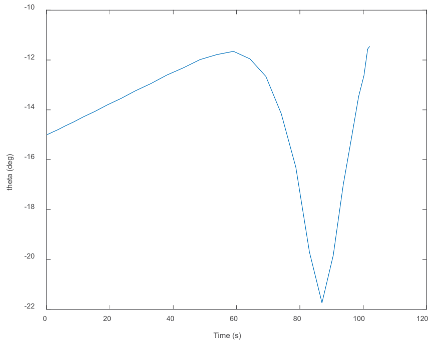

Fig. 4 The history of the flight-path angle

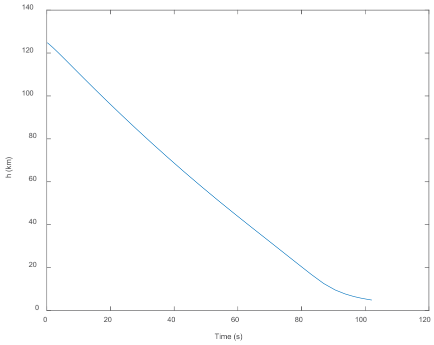

Fig. 5 The history of the altitude

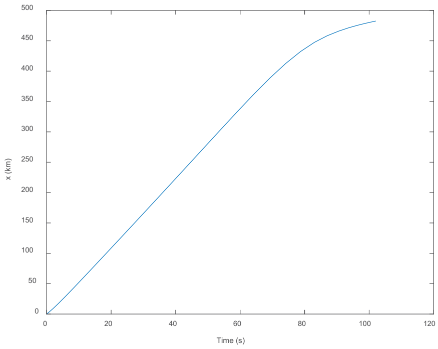

Fig. 6 The history of the longitudinal range

6 Conclusions

In this paper, an optimal control-based Mars landing trajectory generation problem is studied. The analytical solution is hard to obtain for such a problem with highly nonlinear differential equation constraints. Hence, direct method using the Gauss pseudo-spectral technique is proposed to solve the trajectory planning problem numerically. Simulation results show that the planning time is acceptable and the obtained trajectory can satisfy the specified landing requirement.

References

[1] Kluever C A, Entry guidance performance for Mars precision landing. *Journal of Guidance, Control, and Dynamics*, 2008, 31: 1537-1544.

[2] Reynolds T, Malyuta D, Mesbahi M, et al. A real-time algorithm for non-convex powered descent guidance. *AIAA SciTech 2020 Forum*. 2020: 0844

[3] Liu X, Shen, Lu P. Entry trajectory optimization by second-order cone programming. *Journal of Guidance, Control, and Dynamics*, 2016, 39(2): 227-241.

[4] Lu P, Xue S. Rapid generation of accurate entry landing footprints. *Journal of Guidance, Control, and Dynamics*, 2010, 33(3): 756-767.

[5] Betts J T. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control and Dynamics*, 1998, 21(2): 193-206.

[6] Bryson Jr A E, Desai M N, Hoffman W C. Energy-state approximation in performance optimization of supersonic aircraft. *Journal of Aircraft*, 1969, 6(6): 481-488.

[7] Hull D G, Speyer J L. Optimal reentry and plane-change trajectories. *Journal of the Astronautical Sciences*, 1982, 30(2): 117-130.

[8] Pesch H J. Offline and online computation of optimal trajectories in the aerospace field. *Applied Mathematics in Aerospace Science and Engineering*, Springer, Boston, MA, 1994: 165-220.

[9] Rao A V. Survey of numerical methods for optimal control. *2009 AAS/AIAA Astrodynamics Specialist Conference*, AAS Paper 09-334, Pittsburgh, PA, August 10-13, 2009.

[10] Huang G Q, Lu Y P, Nan Y. A survey of numerical algorithms for trajectory optimization of flight vehicles. *Sci China Tech Sci*, 2012, 55: 2538-2560.

[11] Kozynchenko A I. Analysis of predictive entry guidance for a Mars lander under high model uncertainties. *Acta Astronautica*, 2010, 68(1): 121-132

[12] Benson D A, Huntington G T, Thorvaldsen T P, et al. Direct Trajectory Optimization and Costate Estimation via an Orthogonal Collocation Method. *Journal of Guidance, Control and Dynamics*, 2006, 29(6): 1435-1440.

[13] Gill P E, Murray W, Saunders M A, SNOPT: An SQP algorithm for large-scale constrained optimization, *SIAM Review*, 2005, 47(1): 99-131.