4

Finite-Thrust Trajectory Optimization Using a Combination of Gauss Pseudospectral Method and Genetic Algorithm

Qibo Peng
College of Aerospace and Material Engineering,
National University of Defense Technology, Changsha,
China

1. Introduction

Finite-thrust propulsion is now widely used in space missions, such as lunar or mars descent, interplanetary transfer, spacecraft rendezvous, etc.. The finite-thrust optimal control problem is qualitatively different from the impulsive case as there are now no integrable arcs and the control itself, must be modeled and determined. Optimizing finite-thrust trajectory is a challenging problem due to the existence of long powered arcs. Therefore, obtaining optimal trajectory is sometimes tedious and time consuming.

Finite-thrust trajectory optimization has been studied by many researchers. Traditional optimization method for this problem is indirect optimization method, which rely on solving the necessary conditions derived from the Pontryagin et al. minimum principle. For example, the Pontryagin maximum principle is applied to the problem of optimal thrust programming for the least fuel consumption of the lunar soft landing in Ref. [1]. But with the increasing of the applicable extension, the model complicate, the various uncertainties and the strict requirement to the system, the indirect optimization method is faced with the more and more challenge. In recent years, direct solution methods have been used extensively in a variety of trajectory optimization problems, such as shooting methods and collocation methods. Cho [2] applied shooting method to the lunar soft-landing problem; Huang [3] proposed a hybrid strategy combining genetic algorithms (GA) and SQP to optimize the lunar landing trajectory; Pourtakdoust [4] used direct collocation method to solve the three-dimensional optimal orbital transfer for thrust-limited spacecraft. Luo [5] proposed a hybrid strategy to optimize the rendezvous phasing trajectory, and the discrete variables are solved by integer-coded GA.

To solve the problem, an optimization method combined a collocation method-Gauss Pseudospectral Method (GPM) and Genetic Algorithm (GA) is proposed in this chapter. Firstly, both the control and state variables are discretized at the nodes of discretization. Different from the traditional collocation schemes, piecewise-continuous polynomials such as linear or cubic splines are used as the interpolating polynomials over each time segment, the Lagrange interpolating polynomials are used to approximate the state and control in GPM. By using GPM, the continuous time optimal control problem is converted into a nonlinear programming problem (NLP). Then GA is employed to solve this NLP. The results of a numerical simulation verified the validity of the proposed optimization method. Results also indicate that the method has good performance on accuracy and fast convergence.

2. Problem statement for finite-thrust trajectory optimization

A general problem statement for finite-thrust trajectory optimization can be stated as follows [6]: determine the optimal transfer time t_f and optimal control variable $\mathbf{u}(t)$, $0 < t < t_f$, that minimize the performance index

$$J = \Phi\left(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f\right) + \int_{t_0}^{t_f} L(\mathbf{x}(t), \mathbf{u}(t), t)dt \tag{1}$$

subject to the dynamic equations

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \quad t \in [t_0, t_f] \tag{2}$$

where $\mathbf{x}(t)$ is the state variables, $\mathbf{x}(t) \in \mathbb{R}^n$, t_0 and t_f are the initial and terminal time.

The boundary conditions

$$\phi(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f) = 0 \tag{3}$$

and the path constraints

$$\mathbf{C}(\mathbf{x}(t), \mathbf{u}(t), t) \le 0 \tag{4}$$

3. Optimization method

To solve the complex finite-thrust optimization problem with strict constraints, an optimization method is proposed in this section. It is a combination of a collocation method-GPM and GA. Here the GPM is used to transforming the optimal control problem to a NLP, and then GA is employed to solving the resulting NLP. The detailed optimization method is given as follows [7].

3.1 NLP construction by GPM

We will give the detailed method how to transforming the finite-thrust optimal control problem to a NLP. The problem formulation will be presented in this section.

The Gauss pseudospectral method, like Legendre and Chebyshev methods, is based on approximating the state and control trajectories using interpolating polynomials. In the case of the GPM, the Lagrange interpolating polynomials are used to approximate the state and control. By using GPM, the continuous-time optimal control problem is converted into a NLP. The GPM for the powered descent control problem is summarized as follows [8][9].

First, the original time interval $t \in \left[t_0, t_f\right]$ is transformed in the time interval $t \in [-1,1]$ via the affine transformation:

$$t = \frac{t_f - t_0}{2}\bar{\tau} + \frac{t_f + t_0}{2} \tag{5}$$

The cost function, constraints, and boundary conditions can be given in terms of $\bar{\tau}$. Then the state is approximated using a basis of *N+1* Lagrange interpolating polynomials L,

$$\mathbf{x}(\bar{\tau}) \approx \mathbf{X}(\bar{\tau}) = \sum_{i=0}^{N} L_i(\bar{\tau})\mathbf{X}(\bar{\tau}_i) \tag{6}$$

where $L_i(\bar{\tau})\ (i = 0,\cdots,N)$ are defined as

$$L_i(\bar{\tau}) = \prod_{j=0, j\neq i}^{N} \frac{\bar{\tau} - \bar{\tau}_j}{\bar{\tau}_i - \bar{\tau}_j} \tag{7}$$

Additionally, the control is approximated using a basis of N Lagrange interpolating polynomials $\tilde{L}_i(\bar{\tau}), (i = 1,\cdots,N)$ as

$$\mathbf{u}(\bar{\tau}) \approx \mathbf{U}(\bar{\tau}) = \sum_{i=1}^{N} \tilde{L}_i(\bar{\tau})\mathbf{U}(\bar{\tau}_i) \tag{8}$$

where

$$\tilde{L}_i(\bar{\tau}) = \prod_{j=1, j\neq i}^{N} \frac{\bar{\tau} - \bar{\tau}_j}{\bar{\tau}_i - \bar{\tau}_j} \tag{9}$$

Differentiating Eq. (6), we obtain

$$\dot{\mathbf{x}}(\bar{\tau}_k) \approx \dot{\mathbf{X}}(\bar{\tau}_k) = \sum_{i=0}^{N} \dot{L}_i(\bar{\tau}_k)\mathbf{X}(\bar{\tau}_i) = \sum_{i=0}^{N} D_{ki}(\bar{\tau}_k)\mathbf{X}(\bar{\tau}_i) \tag{10}$$

where D_{ki} ($D \in \mathbb{R}^{N\times(N+1)}$) is known as differentiation matrix. In the GPM, the dynamics are collocated at the N Legendre-Gauss (LG) points $\bar{\tau}_k (k = 1,\cdots,N)$. The derivative of each Lagrange polynomial at the LG points can be represented in a differential approximation matrix $D \in \mathbb{R}^{N\times(N+1)}$. The elements of the differential approximation matrix are determined offline as follows:

$$D_{ki} = \dot{L}_i(\bar{\tau}_k) = \begin{cases} \dfrac{(1+\bar{\tau}_k)\dot{P}_N(\bar{\tau}_k) + P_N(\bar{\tau}_k)}{(\bar{\tau}_k - \bar{\tau}_i)\left[(1+\bar{\tau}_i)\dot{P}_N(\bar{\tau}_i) + P_N(\bar{\tau}_i)\right]}, & i \neq k \\ \dfrac{(1+\bar{\tau}_i)\ddot{P}_N(\bar{\tau}_i) + 2\dot{P}_N(\bar{\tau}_i)}{2\left[(1+\bar{\tau}_i)\dot{P}_N(\bar{\tau}_i) + P_N(\bar{\tau}_i)\right]}, & i = k \end{cases} \tag{11}$$

where $k = 1,\cdots,N$ and $i = 0,\cdots,N$. The dynamic constraint equation $\dot{\mathbf{X}}(\bar{\tau}) = \mathbf{f}\left(\mathbf{X}(\bar{\tau}),\mathbf{U}(\bar{\tau});t_0,t_f\right)$ is transcribed into an algebraic constraint using the differential approximation matrix as follows:

$$\sum_{i=0}^{N} D_{ki}\mathbf{X}(\overline{\tau}_i) - \frac{t_f - t_0}{2}\mathbf{f}\left(\mathbf{X}(\overline{\tau}_k), \mathbf{U}(\overline{\tau}_k), \overline{\tau}_k; t_0, t_f\right) = 0 \tag{12}$$

In addition, $X_0 \equiv X(-1)$ and X_f is defined using the Gauss quadrature given by

$$\mathbf{X}(\overline{\tau}_f) \equiv \mathbf{X}(\overline{\tau}_0) + \frac{t_f - t_0}{2}\sum_{k=1}^{N} w_k \mathbf{f}(\mathbf{X}(\overline{\tau}_k), \mathbf{U}(\overline{\tau}_k), \overline{\tau}, t_0, t_f) \tag{13}$$

where w_k are the Gauss weights.

The continuous cost function $J = \Phi\left(\mathbf{X}_0, t_0, \mathbf{X}_f, t_f\right) + \int_{t_0}^{t_f} g(\mathbf{X}, \mathbf{U})\,dt$ is approximated as

$$J = \Phi\left(\mathbf{X}_0, t_0, \mathbf{X}_f, t_f\right) + \frac{t_f - t_0}{2}\sum_{k=1}^{N} w_k g\left(\mathbf{X}_k, \mathbf{U}_k, \overline{\tau}_k; t_0, t_f\right) \tag{14}$$

The boundary constraints are also discretized at the LG points as

$$\phi\left(\mathbf{X}_0, t_0, \mathbf{X}_f, t_f\right) = 0 \tag{15}$$

Furthermore, the path constraints are evaluated at the LG points as

$$C\left(\mathbf{X}_k, \mathbf{U}_k, \overline{\tau}_k; t_0, t_f\right) \le 0 \quad (k = 1, \cdots, N) \tag{16}$$

The cost function in Eq. (14) and the algebraic constraints in Eqs. (12), (13), (15) and (16) define an NLP whose solution is an approximate solution to the continuous Mayer problem. Finally, it is noted that discontinuities in the state or control can be handled efficiently by dividing the trajectory into phases, where the dynamics are transcribed within each phase and then connected together by additional phase interface constraints.

3.2 NLP solution by GA

Many methods can be used to solve the NLP, such as the steepest descent algorithm, hill-climbing algorithm, evolution algorithm and so on. Here genetic algorithm is employed to solve the optimization problem due to its excellent performance on global searching and the convenience to realize by computer.

Genetic algorithms are search procedures based on the mechanics of natural genetics. All natural species survive by adapting themselves to the environment. Genetic algorithm search combines a Darwinian survival-of-the-fittest concept to eliminate unfit characteristics and utilizes random information exchange, with exploitation of knowledge contained in old solutions, to effect a search mechanism with power and speed. In using genetic algorithms, the usual goal is to find solutions that are closer to the globally optimal point. This technique has gained popularity in the recent years as a robust optimization tool for variety of problems in engineering, science, economics, finance, etc.

A simple genetic algorithm is composed of three operators: (1) selection, (2) crossover, and (3) mutation. Selection is a process where an old string is carried through into a new population J depending on the performance index values. Due to this move, strings with better fitness values get larger numbers of copies in the next generation. Selecting good strings for this operation can be implemented in many different ways. In conjunction with the selection procedures, the good strings can either be allowed to change (pure selection) or retained in to the next evolution (elite selection). A simple crossover follows selection in three steps. First, the newly selected strings are paired together at random. Second, an integer position "n" along every pair of strings is selected uniformly at random. Finally, based on a probability of crossover, the paired strings undergo crossing over at the integer position "n" along the string. This results in new pairs of strings that are created by swapping all the characters between characters 1 and "n" inclusively. Although the crossover operation is a randomized event, when combined with selection it becomes an effective means of exchanging information and combining portions of good quality solutions. Selection and crossover give GA most of their search power. The third operator, mutation, is simply an occasional random alteration of a string position (based on probability of mutation). In a binary code, this involves changing a 1 to a 0 and vice versa. The mutation operator helps in avoiding the possibility of mistaking a local minimum for a global minimum. When mutation is used sparingly (about one mutation per thousand bit transfers) with selection and crossover, it improves the global nature of the genetic algorithm search. [10]

By using GPM, the optimal control problem was transcribed to a NLP by parameterizing the state and control using global polynomials and collocating the differential-algebraic equations using nodes obtained from a Gaussian quadrature. The dispersed state and control variables at LG points should be optimized using a parameter optimization technique. Here GA can be easily used as a parameter optimization technique for solving the problem.

4. Numerical examples: Lunar powered descent trajectory optimization

In this section, a trajectory optimization problem for lunar powered descent is presented to verify the validity of the proposed optimization method.

To make the optimization problem easier to solve, the dynamical system considered in most of previous studies on lunar powered descent is a two dimensional dynamics [7][11]. The descent trajectory of the lunar lander is assumed to remain in a vertical plane without any provision for possible lateral movements. However, for standard trajectory design or simulation before launching the rocket, the error can not be ignored. To obtain more accurate results and demonstrate the validity of this method to solve complex optimization problem, a three dimensional descent dynamics with high precision is established in this paper, and many strict constraints is given.

Here, we give the simple formulation of the optimization.

4.1 Problem formulation

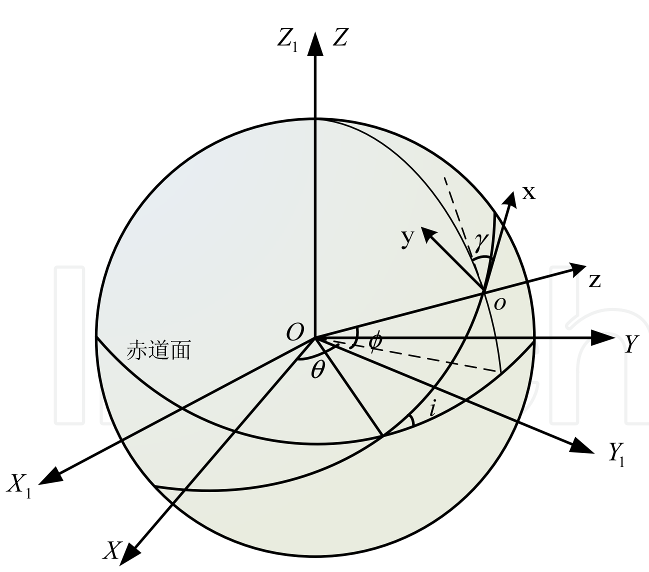
4.1.1 Dynamics equations

The lunar lander is in a circular orbit with an initial altitude H_0. A Hohmann transfer orbit is used to decrease the altitude from H_0 to the pericynthion (altitude 15km). From here the powered descent begins. The powered-descent phase of the lunar-landing mission is initiated at or near the pericynthion of the free descent orbit and finishes near the lunar surface (about altitude 2km). It is a continuous thrust maneuver of the duration of several minutes. The largest part of fuel is consumed during this phase [12].

The following frame of reference is established to describe the powered descent maneuver.

1. Moon Centred Inertial (MCI) coordinate system $O-X_1Y_1Z_1$
   The Origin is at the center of the moon; OX_1 axis is along the direction of the Moon's revolution, and OY_1 axis is pointing at the ascending node of the Moon's orbit relative to the equator.
2. Moon Centred Fixed (MCF) coordinate system $O-XYZ$
   The Origin is at the center of the moon; OX axis is along the direction of the Moon's revolution, and OY axis is in the Moon's equator plane, pointing at the Sinus-Medii.
3. Orbit coordinate system $o-xyz$
   The Origin is at the center of gravity of the lunar lander; ox axis is along the radial direction, oy axis is along the direction of the horizontal velocity at initial state of powered-descent.

It is assumed that the moon has a homogeneous gravity field and a constant rotation angular velocity. The coordinate systems and defined parameters are shown in Fig. 1.



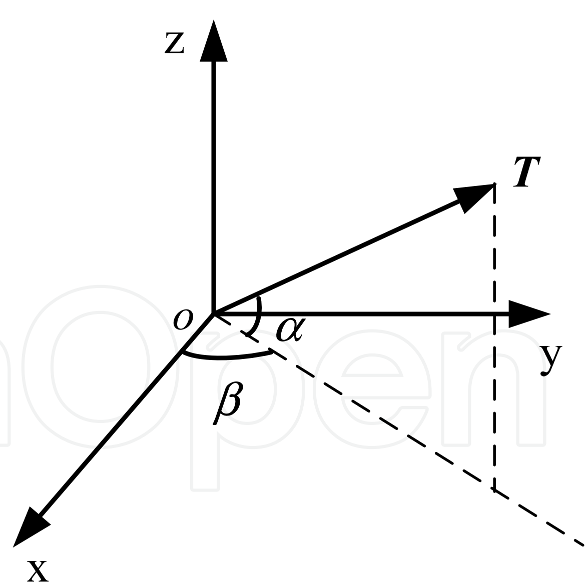


Fig. 1. Coordinate systems

The dynamics equations for lunar powered descent can be denoted as follow.

$$\begin{cases} \dot{r} = w \\ \dot{\theta} = \dfrac{u\sin\gamma - v\cos\gamma}{r\cos\phi} \\ \dot{\phi} = \dfrac{u\cos\gamma + v\sin\gamma}{r} \\ \dot{u} = \dfrac{T\cos\alpha\cos\beta}{m} - \dfrac{uw}{r} + \dfrac{uv}{r}\tan\phi\sin\gamma - \dfrac{v^2}{r}\tan\phi\cos\gamma \\ \quad -2w\omega\cos\phi\sin\gamma + 2v\omega\sin\phi - r\omega^2\sin\phi\cos\phi\cos\gamma \\ \dot{v} = \dfrac{T\cos\alpha\sin\beta}{m} - \dfrac{vw}{r} + \dfrac{uv}{r}\tan\phi\cos\gamma - \dfrac{u^2}{r}\tan\phi\sin\gamma \\ \quad +2w\omega\cos\phi\cos\gamma - 2u\omega\cos\phi - r\omega^2\sin\phi\cos\phi\sin\gamma \\ \dot{w} = \dfrac{T\sin\alpha}{m} - \dfrac{\mu_L}{r^2} + \dfrac{u^2+v^2}{r} - 2v\omega\cos\phi\cos\gamma + 2u\omega\cos\phi\sin\gamma + r\omega^2\cos^2\phi \\ \dot{m} = \dfrac{T}{I_{sp}g_0} \end{cases} \tag{17}$$

where thrust size T and thrust direction angle α, β are control variables for the dynamics. Other parameters nomenclature is given in appendix.

4.1.2 Objectives

The aim of optimal trajectory design is to minimize the amount of fuel required to perform a free end-time descent from the given initial state to the given terminal state. The objective function is:

$$\tilde{J} = \int_0^{t_f} \dot{m}dt = \int_0^{t_f} \frac{T}{I_{sp}g_0}dt = \frac{T}{I_{sp}g_0}t_f \tag{18}$$

The magnitude of thrust T is defined as a constant here, so the objective is to minimize the total powered descent time. The objective function can therefore be expressed as

$$J = t_f \tag{19}$$

4.1.3 Constraints

Firstly the boundary conditions including position and velocity constraints of lunar lander at initial time t_0 and final time t_f are considered.

The constraints at the initial time are

$$\theta(\tau_0) = \theta_0, \quad \phi(\tau_0) = \phi_0, \quad r(\tau_0) = R_L + h_0 \tag{20}$$

$$u(\tau_0) = V_0 - (R_L + h_0)\omega\cos\phi\sin\gamma, \quad v(\tau_0) = (R_L + h_0)\omega\cos\phi\cos\gamma, \quad w(\tau_0) = 0 \tag{21}$$

The constraints at the final time are

$$\theta\left(\tau_f\right)=\theta_f,\ \ \phi\left(\tau_f\right)=\phi_f,\ \ r\left(\tau_f\right)=R_L+h_f \tag{22}$$

$$u\left(\tau_f\right)=0,\ \ v\left(\tau_f\right)=0,\ \ w\left(\tau_f\right)=0 \tag{23}$$

Then constrained by the propulsion system, the thrust direction angle should be subject to

$$\alpha\in[\alpha_{\min},\alpha_{\max}],\ \ \beta\in[\beta_{\min},\beta_{\max}] \tag{24}$$

where $\alpha_{\min}$, $\alpha_{\max}$, $\beta_{\min}$ and $\beta_{\max}$ are the boundary of thrust direction angle.

4.2 Simulation example

Here a test case scenario is given to validate the optimization method.

The initial and final conditions - treated as boundary conditions by the optimization algorithm - are given by Equation (25)

$$\begin{cases} h_0=15km \\ \theta_0=0^\circ \\ \phi_0=90^\circ \\ u_0=1.6943km/s \\ v_0=0 \\ w_0=0 \end{cases} \qquad \begin{cases} h_f=2km \\ \theta_f=5^\circ \\ \phi_f=76^\circ \\ u_f=0 \\ v_f=0 \\ w_f=0 \end{cases} \tag{25}$$

The constraints of thrust direction angle are set as follows:

$$\alpha\in[-50^\circ,50^\circ],\ \ \beta\in[150^\circ,220^\circ] \tag{26}$$

The values of the other parameters used in this scenario are summarized here:

$$\begin{array}{ll} T=45kN, & m_0=15t \\ R_L=1738km, & i_0=90^\circ \\ I_{sp}=365s, & \omega=2.6617\times10^{-6}rad/s \end{array} \tag{27}$$

Here taking the LG points N=50, and GPM-GA is employed to solve the optimization problem. The results show that the optimal flight time for lunar landing is 472.74s, and require a fuel mass of 5947.2kg. The trajectory of lunar lander is shown in Fig. 2, where the result of methodology outlined in this paper, are compared to the indirect method (Pontryagin's maximum principle). As can be see, the two methods yield practically the same results.

The velocity of lunar lander in the orbit coordinate system $o-xyz$ is shown in Fig. 3, while the time history of thrust direction angle during landing is shown in Fig. 4.

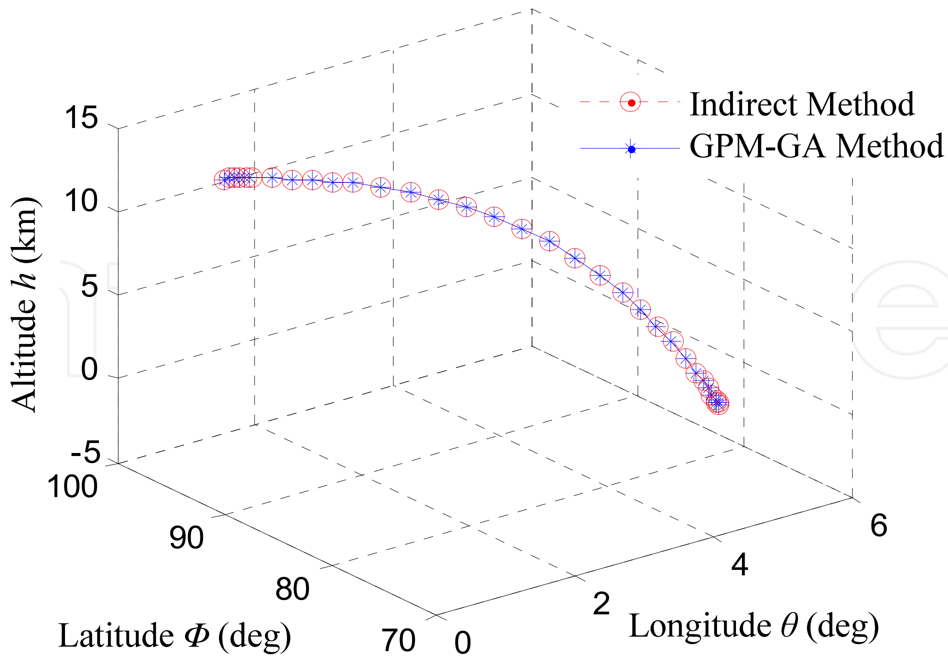


Fig. 2. The descent trajectory of lunar lander

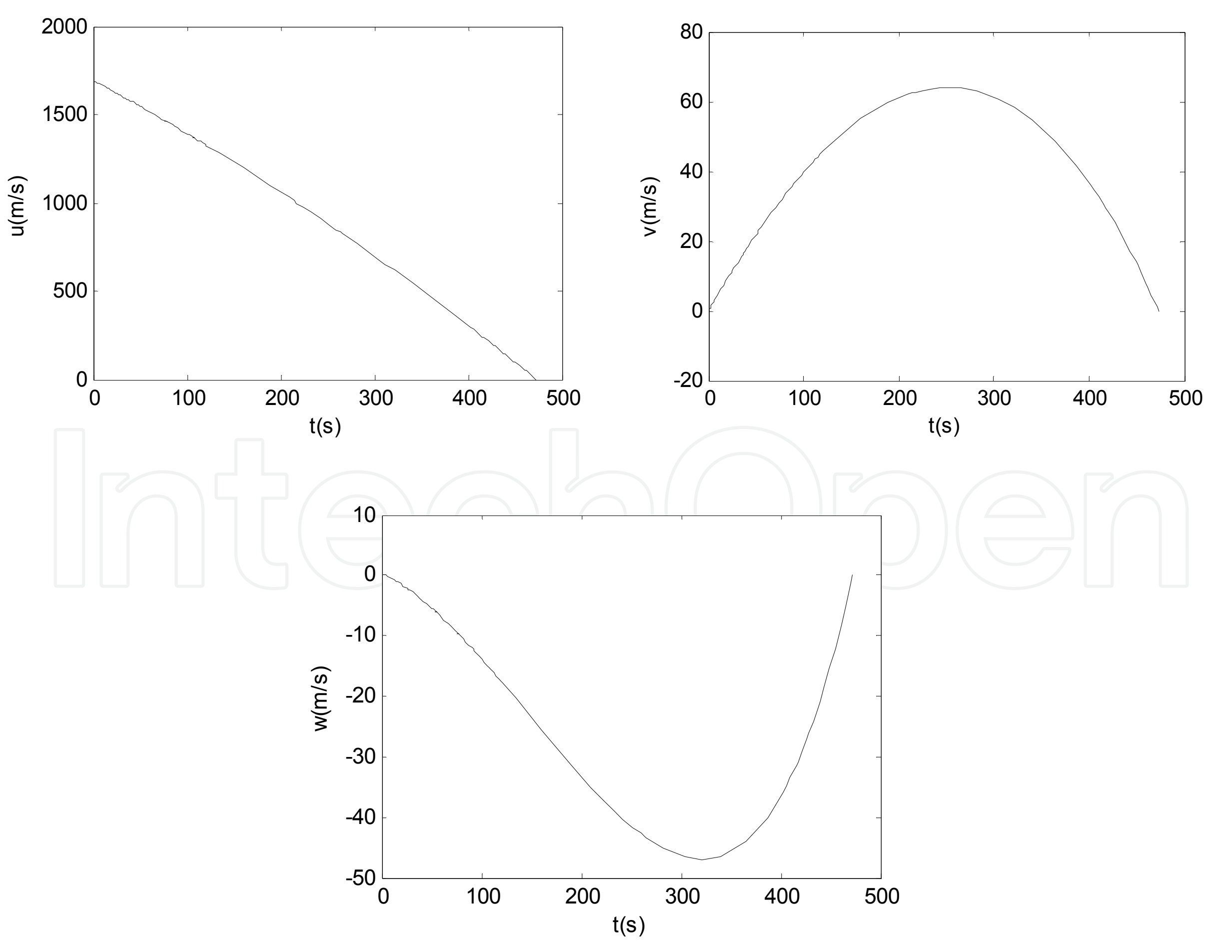


Fig. 3. Time history of velocity

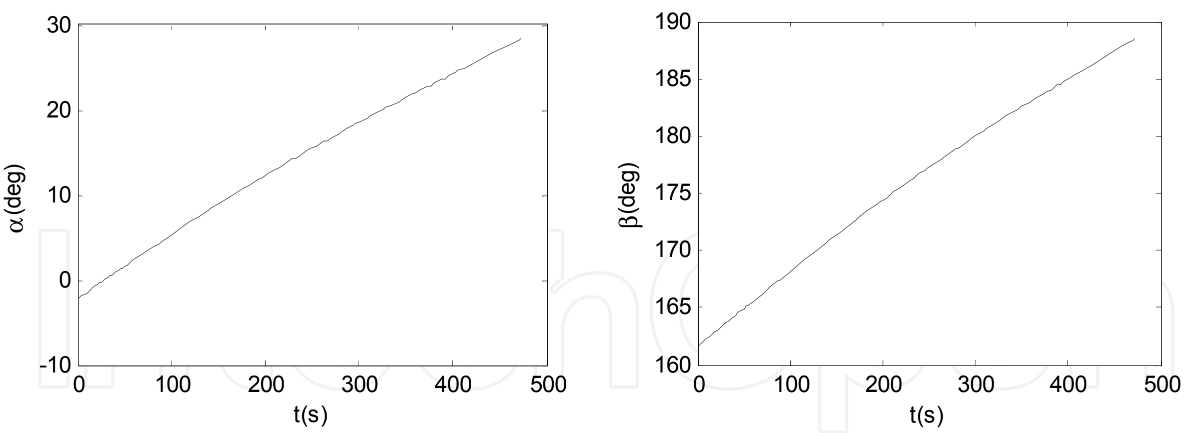


Fig. 4. Time history of thrust direction angle

The simulation results indicate that the GPM-GA optimization algorithm has high accuracy, and the error with results solved by indirect method is very small. What's more, the calculation will converge rapidly even when the initial values for GPM are chosen at random in the bound. Less than 2 minutes are needed for a result to be obtained on a PC with a 3.0GHz/Pentium 4 CPU. However, if using the traditional method such as direct shooting method to solve the optimal descent trajectory, the program can only be converged when the initial guess is closed to optimal values, and the calculation time is longer. For example, more than 20 minutes are needed for calculation with the method in reference [3].

4.3 Results analysis

4.3.1 Demonstration of computational feasibility

The feasibility of the computational solution can be validated by comparing the results to the propagated states via a separate ODE Runge-Kutta propagator. By interpolating the values of the control function, $\mathbf{u}(t_i)$, at the discretization time points and then integrating the differential dynamical equations 17, via MATLAB's ode45 solver, a comparison of error norms can be made with the results of methodology outlined in this paper. Results showed that powered descent trajectory dose satisfy the end-point conditions within an input constraint.

4.3.2 Demonstration of computational optimality

To demonstrate the necessary conditions needed for optimality the first step requires the formulation of the Hamiltonian [13]

$$H(\lambda,x,u,t)=L(x,u,t)+\lambda^T f(x,u,t) \tag{28}$$

where $L(\cdot)$ is the Lagrange cost, and $f(\cdot)$ is the vector field for the right hand side of the differential dynamical equations.

The objective function is defined as

$$J(x,u,t)=-m\left(t_f\right) \tag{29}$$

therefore

$$L(\cdot) = 0 \tag{30}$$

and

$$H(\cdot) = \lambda^T f(x,u) \tag{31}$$

It can be shown that $H(\cdot)$ is constant with respect to time, with boundary conditions

$$H\left(t_f\right) = -\frac{\partial \Phi\left(x_f, t_f\right)}{\partial t_f} \tag{32}$$

To determine the final value of the Hamiltonian, the Endpoint Lagrangian, given as,

$$\begin{aligned}\Phi\left(x_f, t_f\right) = &-m\left(t_f\right) + \mu_1\left(r_f - r\left(t_f\right)\right) + \mu_2\left(\theta_f - \theta\left(t_f\right)\right) \\ &+ \mu_3\left(\phi_f - \phi\left(t_f\right)\right) + \mu_4\left(u_f - u\left(t_f\right)\right) \\ &+ \mu_5\left(v_f - v\left(t_f\right)\right) + \mu_6\left(w_f - w\left(t_f\right)\right)\end{aligned} \tag{33}$$

is substituted into the Hamiltonian Value Condition:

$$H\left(t_f\right) = 0 \tag{34}$$

where, $\mu_i, i = 1, \ldots 6$ is Lagrange multipliers.

This indicates that the Hamiltonian should be 0 for all the time in this problem. The Hamiltonian from the optimization solution in this paper is almost 0 with respect to time, and it can be used to verify that the numerical results satisfy the necessary Karush-Kuhn-Tuhker (KKT) conditions for optimality.

5. Conclusion

An optimization algorithm GPM-GA method is presented to solve the optimal finite-thrust trajectory with an input constraint in the paper. The results of a numerical simulation verified the validity of the proposed optimization method. The results indicate that the method can provide good performance on accuracy and fast convergence. It is expected that this novel optimization algorithm can be used to solve the similar optimization problems.

6. Appendix

Nomenclature

m	mass of lunar lander	**r**	position vector of lunar lander
m_0	initial mass of lunar lander	t	flight time
I_{sp}	propulsion system's specific impulse	u , v , w	velocity of lunar lander in the orbit

g_0	gravitational acceleration on the Earth surface ($9.81 m / s^2$)	γ	angle between axis oy and north direction of the moon
μ_L	gravity constant of the moon	θ , ϕ	longitude and latitude
$\boldsymbol{\omega}$	rotation angular velocity of the moon	$\mathbf{T}$	thrust
h	altitude from lunar surface of lunar lander	α , β	thrust direction angle

7. References

[1] Meditch J. S. On the Problem of Optimal Thrust Programming for a Lunar Soft Landing, IEEE Transaction Automatic Control, 1964, pp.477-484.

[2] Cho D. H., Jeong B., Lee D., et al. Optimal Perilune Altitude of Lunar Landing Trajectory, Int'l Journal of Aeronautical & Space Sciences, 10 (2009), pp.67-74.

[3] Huang W. D., Wang W. and Xi X. N. Overall Mission Trajectory Optimization for Manned Lunar Landing Mission Using a Hybrid Algorithm, the 2nd Information Engineering and Computer Science (ICIECS), Wuhan, China, December 2010.

[4] Pourtakdoust S. H., Jalali M. A. Thrust-Limited Optimal Three-Dimensional Spacecraft Trajectory, AIAA-95-3325-CP, 1995, pp.1395-1404.

[5] Luo Y. Z., Li H. Y., Tang G. J. Hybrid approach to optimize a rendezvous phasing strategy, Journal of Guidance, Control and Dynamics, 30 (1) (2007), pp.185-191.

[6] Fahroo F., Ross I. M. Direct Trajectory Optimization by a Chebyshev Pseudospectral Method, Journal of Guidance, Control and Dynamics, 25 (1) (2002), pp.160-166.

[7] Peng Q. B., Li H. Y., Shen H. X. Rapid Lunar Exact-Landing Trajectory Optimization Via Gauss Pseudospectral method, Journal of Astronautics (in Chinese), 31 (4) (2010), pp.1012-1016.

[8] Houacine M. and Khardi S. Gauss Pseudospectral Method for Less Noise and Fuel Consumption from Aircraft Operations, Journal of Aircraft, 47 (2010), pp.2152-2159.

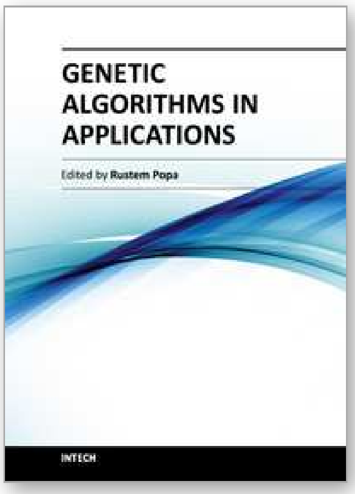
[9] Huntington G. T. Optimal Reconfiguration of Spacecraft Formations Using the Gauss Pseudospectral Method, Journal of Guidance, Control, and Dynamics, 31 (2008), pp.689-698.

[10] KrishnaKumar K. Genetic Algorithms - A Robust Optimization Tool, the 31st Aerospace Sciences Meeting & Exhibit, Reno, Nevada, January 1993.

[11] Tu L. H., Yuan J. P., Luo J. J., et al. Lunar Soft Landing Rapid Trajectory Optimization Using Direct Collocation Method and Nonlinear Programming, the 2nd International Conference on Space Information Technology, 2007.

[12] Uchiyama K.: Guidance Law for Lunar Lander with Input Constraint, AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, South Carolina, USA, August 2007.

[13] Doman D. B., Bollino K. P., Ross I. M. Nonlinear Feedback Control for Rapid, On-Line Trajectory Optimization of Reentry Vehicles, AFRL-VA-WP-TP-2006-307, December 2005.



Genetic Algorithms in Applications
Edited by Dr. Rustem Popa

ISBN 978-953-51-0400-1
Hard cover, 328 pages
Publisher InTech
Published online 21, March, 2012
Published in print edition March, 2012

Genetic Algorithms (GAs) are one of several techniques in the family of Evolutionary Algorithms - algorithms that search for solutions to optimization problems by "evolving" better and better solutions. Genetic Algorithms have been applied in science, engineering, business and social sciences. This book consists of 16 chapters organized into five sections. The first section deals with some applications in automatic control, the second section contains several applications in scheduling of resources, and the third section introduces some applications in electrical and electronics engineering. The next section illustrates some examples of character recognition and multi-criteria classification, and the last one deals with trading systems. These evolutionary techniques may be useful to engineers and scientists in various fields of specialization, who need some optimization techniques in their work and who may be using Genetic Algorithms in their applications for the first time. These applications may be useful to many other people who are getting familiar with the subject of Genetic Algorithms.