Aerospace Science and Technology 36 (2014) 114–124

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Low-thrust trajectory design with constrained particle swarm optimization

Jinjun Shan *, Yuan Ren

Department of Earth and Space Science and Engineering, York University, Toronto, Ontario, M3J 1P3, Canada

ARTICLE INFO

Article history:
Received 11 June 2013
Received in revised form 5 April 2014
Accepted 8 April 2014
Available online 15 April 2014

Keywords:
Low-thrust trajectory
Particle swarm optimization
Multidisciplinary design optimization
Solar electric propulsion

ABSTRACT

In this paper, combined with the direct approach, particle swarm optimization (PSO) is applied to low-thrust trajectory optimization problems. A double-loop trajectory optimization algorithm is developed. The outer loop of this algorithm is a modified PSO optimizer, which can deal with constrained optimization problems and avoid premature convergence. The function of the outer loop is generating a series of time histories of control, called particles, and driving the particles toward the optimal solution. The direct approach (fourth-order Runge–Kutta shooting/parallel shooting method) is adopted as the inner loop algorithm, whose main task is to correct the particles provided by the outer loop and ensure that all the constraints are satisfied. This algorithm has the global search feature of the PSO and the relative large radius of convergence of the direct approach. Its efficiency is substantiated by solving a fixed-time fuel-optimal transfer problem from an asteroid to the Earth. Furthermore, this algorithm can be considered to be a universal low-thrust optimizer, and it can easily be used to solve more complex trajectory optimization problems such as multi-swingby problem and multidisciplinary design optimization (MDO) problems.

© 2014 Elsevier Masson SAS. All rights reserved.

1. Introduction

Most successful trajectory optimization methods are gradient-based, mainly due to their computational efficiency. However, these algorithms are local in nature and it is easy for the solution to get trapped in a local minimum. To avoid this, global rather than local search must be performed. In recent years, interest in the application of evolutionary algorithms (EAs) to trajectory optimization has grown, substantially due to their global search capabilities. In Ref. [8] a genetic algorithm (GA) [11] was used to adjust the direction of low-thrust engine on discrete nodes directly. The GA can also be employed to optimize the time history of control parameterized by polynomials [28], or used to solve the discrete-continuous optimization problem [29]. In Ref. [30], differential evolution (DE) [23] and simulated annealing (SA) [5] are used to optimize low-thrust trajectories modeled as a series of impulses connected by conics. The evolution branching technique (EB) is widely applied in the initial design of the low-thrust trajectories [7,24].

Another type of EA, particle swarm optimization (PSO) [25], is considered in this work. The PSO was first introduced by Kennedy and Eberhart [13] based on observation and simulation of the social behavior of flocks of birds or schools of fish. In this algorithm the optimal solution is sought by moving a swarm of particles around in the search space according to simple mathematical rules. The movement of each particle is determined by its best known position and the best position achieved by the entire swarm. The algorithm is simple and can be implemented in a few lines of computer code. Moreover, it is gradient-free and can solve irregular optimization problems. These characters make PSO an easy-to-use algorithm for real-life problems such as the trajectory optimization [9,12,20], and multidisciplinary design optimization (MDO) [21]. However, the basic PSO algorithm is unable to deal with constrained optimization problems. Constraints are dealt via the penalty approach with quadratic penalty functions or barrier functions [19,21]. Choosing the value of the penalty factor is a difficult problem: if the penalty factor is high, the optimization algorithms usually get trapped in local optimal solutions. On the other hand, if the penalty factor is low, feasible solutions can be barely detected. The PSO used in this paper is modified. The velocities of particles which violate one or more constraints are updated by a modified formula [26], which causes the particles to move towards a feasible region. Additionally, the so called craziness operator is adopted to avoid premature convergence of the PSO.

* Corresponding author.
E-mail addresses: jjshan@yorku.ca (J. Shan), yren@yorku.ca (Y. Ren).

http://dx.doi.org/10.1016/j.ast.2014.04.004
1270-9638/© 2014 Elsevier Masson SAS. All rights reserved.

In this work, a double-loop low-thrust trajectory optimizer is developed. The outer loop is the modified PSO optimizer. In its first iteration it generates a series of time history of control randomly by utilizing conjugate equations in Pontryagin's maximum principle [4]. Here the time history of control is denoted by the initial values of the conjugate states. These sets of conjugate states, called particles, are then driven toward optimal positions or the feasible region in subsequent iterations. A direct approach (fourth-order Runge–Kutta shooting/parallel shooting method) is adopted in the inner loop, which is used to find the local optimal or feasible solution by refining the time histories of the control provided by the outer loop as much as possible. This algorithm possesses the global search behavior of the PSO and the relative large radius of convergence of the direct approach. Moreover, the optimization variables without any gradient information can be added into the outer loop directly enabling it to be easily adopted in multidisciplinary design optimization (MDO).

The rest of this paper is organized as follows. Section 2 is the problem statement. Section 3 presents the methodology, including the structure of the algorithm, the details of the outer loop and inner loop algorithms, application in MDO and parallelization strategies. Section 4 discusses the computational efficiency of the three different inner loop strategies, and presents an example of the low-thrust trajectory optimization from an asteroid to the Earth. Section 5 concludes this paper.

2. Problem statement

In celestial mechanics, Lambert's problem is defined as finding the trajectory $\mathbf{r}(t)$, which satisfies the two-body dynamical equations and the specified boundary conditions $\mathbf{r}(t_0)$ and $\mathbf{r}(t_f)$. For a spacecraft with solar electric propulsion (SEP) in a central force field, the similar problem is called the low-thrust Lambert problem [2], which can be summarized as an optimal control problem stated in Mayer form:

$$\max_{\mathbf{u}(t)\in(PWC[t_0,t_f])} J\big[u(t)\big], \qquad J\big[\mathbf{u}(t)\big]=m(t_f),$$
$$\mathbf{u}(t)=\big[\boldsymbol{\gamma}(t),T(t)\big]^T \tag{1}$$

subject to the differential equations

$$\begin{aligned}\dot{\mathbf{r}}&=\mathbf{v}\\ \dot{\mathbf{v}}&=-\frac{\mu}{r^3}\mathbf{r}+\frac{T}{m}\boldsymbol{\gamma}\\ \dot{m}&=-\frac{T}{g_0 I_{sp}}\end{aligned} \tag{2}$$

and the boundary conditions

$$\mathbf{r}(t_0)=\mathbf{r}_0, \qquad \mathbf{v}(t_0)=\mathbf{v}_0, \qquad m(t_0)=1 \tag{3}$$

$$\mathbf{r}(t_f)=\mathbf{r}_f, \qquad \mathbf{v}(t_f)=\mathbf{v}_f \tag{4}$$

In Eq. (1), $\mathbf{u}(t)$ denotes the time history of the control, which is a piecewise continuous function between the initial time t_0 and the final time t_f. $\mathbf{u}$ consists of two control variables, $\boldsymbol{\gamma}$ and T, where $\boldsymbol{\gamma}$ is a unit vector denoting the direction of thrust in the heliocentric inertial frame. It can be expressed in terms of two angles

$$\boldsymbol{\gamma}=[\cos\beta\cos\alpha,\ \cos\beta\sin\alpha,\ \sin\beta]^T \tag{5}$$

where $\alpha\in[0,2\pi]$ and $\beta\in[-\pi/2,\pi/2]$. $T\in[0,T_{\max}]$ is the variable magnitude of the thrust. Eqs. (2) are the equations of motion. $\mathbf{r}$ and $\mathbf{v}$ are the position and the velocity of the spacecraft in the heliocentric inertial frame, respectively. m is the instantaneous mass of the spacecraft. I_{sp} is the specific impulse of the thruster, and $g_0=9.80665\ \mathrm{m/s^2}$ is the acceleration of gravity at sea level. The gravitational constant of the Sun μ is $1.32712440018\times10^{-11}\ \mathrm{km^3/s^2}$. In the low-thrust Lambert problem, the initial mass, initial states, and the final states are fixed. Eqs. (3), (4) are the boundary conditions at the initial and the final time. For convenience of computation, the length, time, and mass are nondimensionalized by using the astronomical unit ($\mathrm{AU}=1.4959787\times10^8$ km), astronomical time ($\mathrm{TU}=5.02264285645364\times10^6$ s), and the initial mass m_0. Accordingly, the dimensionless velocity, acceleration and force units are AU/TU, AU/TU^2 and $m_0\cdot$ AU/TU^2, respectively.

In this paper the optimal control problem is not solved by an indirect method, but an indirect approach is adopted to generate the initial guess of the control. From Pontryagin's minimum principle, the Hamiltonian of the dynamical system denoted by Eq. (2) is

$$H=\boldsymbol{\lambda}_r\cdot\mathbf{v}+\boldsymbol{\lambda}_v\left(-\frac{\mu}{r^3}\mathbf{r}+\frac{T}{m}\boldsymbol{\gamma}\right)-\lambda_m\frac{T}{g_0 I_{sp}} \tag{6}$$

where, $\boldsymbol{\lambda}_r$, $\boldsymbol{\lambda}_v$ and λ_m are the time-varying Lagrange multipliers. The optimal controls which minimize the Hamiltonian are

$$\begin{aligned}&\boldsymbol{\gamma}^*=-\frac{\boldsymbol{\lambda}_v}{\|\boldsymbol{\lambda}_v\|}\\ &T^*=0 \quad \text{if } H_T>0\\ &T^*=T_{\max} \quad \text{if } H_T<0\\ &0<T^*<T_{\max} \quad \text{if } H_T=0\end{aligned} \tag{7}$$

where H_T is switching function defined as

$$H_T=-\frac{\|\boldsymbol{\lambda}_v\|}{m}-\frac{\lambda_m}{g_0 I_{sp}} \tag{8}$$

Generally, H_T equals to zero only at finite isolated points, so the magnitude of T can only be chosen to be either zero or $T_{\max}$ (bang-bang control). The costate differential equations are given as

$$\begin{aligned}\dot{\boldsymbol{\lambda}}_r&=\boldsymbol{\lambda}_v\frac{\mu}{r^3}-\frac{3\boldsymbol{\lambda}_v^T\mathbf{r}}{r^5}\mathbf{r}\\ \dot{\boldsymbol{\lambda}}_v&=-\boldsymbol{\lambda}_r\\ \dot{\lambda}_m&=-\|\boldsymbol{\lambda}_v\|\frac{T}{m^2}\end{aligned} \tag{9}$$

The final mass $m(t_f)$ is free. From the transversality condition, the final value of the costate corresponding to the mass is

$$\lambda_m(t_f)=0 \tag{10}$$

Thus the optimal control problem has been transformed into a two-point boundary-value problem (TPBVP), which consists of the differential equations, Eqs. (2), (9), and boundary conditions, Eqs. (3), (4), (10). Generally, the TPBVP has more than one solution, each of them corresponding to a local optimal low-thrust transfer orbit.

3. Methodology

3.1. Algorithm structure

To find the global optimal solution of the low-thrust Lambert problem, a double-loop algorithm is used. The outer loop of this algorithm is a modified PSO, which provides the global behavior of the algorithm. The inner loop is a gradient-based algorithm (GBA), which is used to find the local optimal or feasible solutions near the initial guesses provided by the outer loop. Ideally, any gradient-based algorithm can be used as the inner loop algorithm. In this research, Runge–Kutta fourth-order shooting/parallel shooting methods are adopted. The structure of the algorithm is shown in Fig. 1. First, the initial swarm, consisting of NP

Download English Version:

https://daneshyari.com/en/article/1718072

Download Persian Version:

https://daneshyari.com/article/1718072

Daneshyari.com