JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 36, No. 1, January–February 2013

Optimization of Time-Open Constrained Lambert Rendezvous Using Interval Analysis

Tong Chen*
Beihang University, 100191 Beijing, People's Republic of China
and
E. van Kampen,† H. Yu,‡ and Q. P. Chu§
Delft University of Technology, 2600 GB Delft, The Netherlands

DOI: 10.2514/1.56773

By defining a parking time and a transfer time, a time-open constrained Lambert rendezvous problem is proposed in this paper. The purpose of optimization is to minimize the velocity increment for the orbital rendezvous. The interval branch and bound optimization algorithm is introduced for solving a rendezvous optimization problem with strong nonlinearity and nonconvexity. A numerical example is undertaken to validate the interval optimization algorithm. The results indicate that although the global optimal solution for the time-open constrained Lambert rendezvous problem can't be found by using only the interval algorithm in the restricted hardware environment because of the inevitable influence of interval dependency and the complex calculation for the cost function, the interval algorithm can narrow the search space and split the feasible space into many small subspaces. Combining the gradient-based optimization method, an improved interval algorithm is used to find the minimal velocity increment for orbital rendezvous, which is better than the optimal solution using the genetic algorithm.

I. Introduction

THE rendezvous and docking mission can be divided into several major phases: launch, phasing, far-range rendezvous, close-range rendezvous, and mating. During the segment of phasing, not only launch injection errors for orbit inclination and right ascension of the ascending node need to be corrected, but also the difference of phase angle between the chaser and the target spacecraft is eliminated within a designed mission period [1]. The time and the total velocity increment are the two most important factors for rendezvous phasing. The two factors can be linked by Lambert's theorem, which was discussed by Battin [2].

Once the initial point, the aim point, and a specified time of transfer flight are given for the rendezvous phasing, the solution of the phasing transfer orbit is converted to the classical Lambert problem. Many methods have been developed to solve the classical Lambert problem [3–7]. The multiple-revolution Lambert problem was also discussed in [2,6,8–12]. Prussing [8] showed that given two points and a specified flight time, there are $2N_{\max} + 1$ transfer orbit solutions, where $N_{\max}$ is the maximum number of allowed revolutions. The minimum-cost, fixed-time transfer between two fixed points on coplanar circular orbits was also studied by Prussing [8]. Shen and Tsiotras [9] presented an algorithm for minimizing the velocity increment of fixed-time two-impulse transfer between two coplanar circular orbits. Only two of the multiple-revolution solutions need to be calculated and compared in Shen and Tsiotras's algorithm. Furthermore, Luo et al. [10] proposed a hybrid optimization approach for the multiple-impulse, multiple-revolution rendezvous problem. He et al. [11] extended the transverse-eccentricity-vector-based algorithm proposed by Avanzini [7] to solve the multiple-revolution Lambert problem. Zhang et al. [12] give a solution to the fixed-time multiple-revolution Lambert problem with constraints on the perigee and the apogee altitudes.

However, all the works just referenced are mainly focused on the fixed-time and fixed-point Lambert rendezvous problem, although the initial coasting time and the transfer time have been discussed in [9] when the target and the chaser orbits are coplanar and circular. To accord with the practical rendezvous operation much more, a constrained noncoplanar Lambert rendezvous optimization problem with time-open and angle-open is proposed in this paper. Given the initial position of a chaser spacecraft orbiting in a parking orbit and the initial position of a target spacecraft orbiting on a target orbit, the chaser starts the orbital maneuver to enter the transfer orbit after orbiting on the parking orbit for an initial coast period and then rendezvous with the target in the target orbit after orbiting on the transfer orbit for an transfer period. The purpose of optimization is to minimize the velocity increment for orbital rendezvous. This is an optimal problem with strong nonlinearity and nonconvexity. Obviously, linear gradient-based optimization methods are invalid to solve the nonlinear and nonconvex optimization problem. Therefore, a global nonlinear optimization method based on the theory of interval analysis [13–15] paper is introduced for solving the optimal Lambert rendezvous problem in this paper. Interval analysis was developed by Moore [13] in the 1960s to study error propagation. It describes the mathematical properties of intervals of numbers instead of regular numbers and it has shown to be an excellent tool for global nonlinear optimization in the areas of aeronautics and astronautics applications [16–20].

This paper is organized as follows. Firstly, the optimal model of time-open constrained Lambert rendezvous is presented. Secondly, the characteristics of the cost function are investigated using the numerical analysis method. Next, the interval branch and bound algorithm is introduced for solving the presented optimization problem. Then a numerical example is done to illustrate the interval optimization algorithm and the results of optimization of time-open constrained Lambert rendezvous are given.

II. Optimal Model of Time-Open Constrained Lambert Rendezvous

A. Description of Optimal Problem

As seen in Fig. 1, a chase spacecraft and a target spacecraft are orbiting on a parking orbit and a target orbit, respectively, which are

Received 25 October 2011; revision received 15 May 2012; accepted for publication 15 May 2012; published online 25 December 2012.

*Doctor, Lecturer, Spacecraft Guidance, Navigation and Control, Department of Guidance, Navigation and Control, School of Astronautics; chet@buaa.edu.cn

†Assistant Professor, Section of Control and Simulation, Faculty of Aerospace Engineering; E.vanKampen@tudelft.nl

‡Ph.D. Candidate, Section of Control and Simulation, Faculty of Aerospace Engineering; H.Yu@tudelft.nl

§Associate Professor, Section of Control and Simulation, Faculty of Aerospace Engineering; Q.P.Chu@tudelft.nl

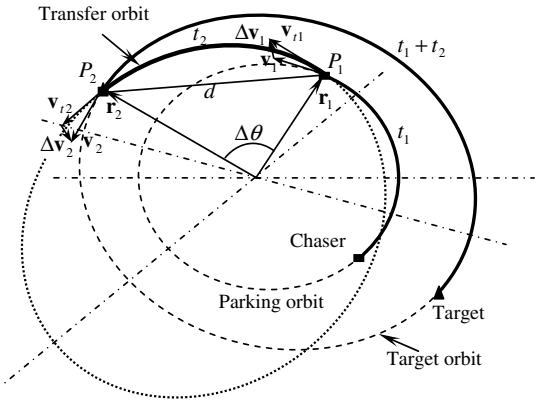

Fig. 1 Time-open impulsive orbital rendezvous.

supposed to be Keplerian orbits. The chaser implements the first impulse $\Delta\mathbf{v}_1$ at the point P_1 to enter the transfer orbit from the parking orbit. The initial coast period between the initial time t_0 and the first impulse is called the "parking time" and denoted by t_1. Then, the second impulse $\Delta\mathbf{v}_2$ is implemented at the point P_2 after the chaser has flown in the transfer orbit for t_2, which is called the "transfer time." The chaser and the target rendezvous at the point P_2. The solution of the transfer orbit from P_1 to P_2 is a standard Lambert problem. The transfer orbit varies with the unfixed transfer time t_2 and the positions of P_1 and P_2. Under the assumption of a Keplerianian two-body orbit, the positions of P_1 and P_2 are all functions of t_1 and t_2. Therefore, the two impulses $\Delta\mathbf{v}_1$ and $\Delta\mathbf{v}_2$ for rendezvous are determined by the parking time t_1 and the transfer time t_2 once the initial orbital elements of the chaser and target are given. An optimal problem of time-open Lambert rendezvous is summarized as finding (t_1, t_2) to minimize the total velocity increment for orbital rendezvous

$$J = \Delta v = |\Delta\mathbf{v}_1| + |\Delta\mathbf{v}_2| = J(t_1, t_2) \tag{1}$$

In an actual rendezvous mission, the perigee altitude of transfer orbit is usually required to be greater than a minimum value $h_{p\min}$ (lower bound, e.g., the atmosphere altitude $h_{p\min} = 200$ km). The semimajor axis of the transfer orbit is also required to be less than an upper bound $a_{t\max}$ to limit the energy. In addition, the total time for rendezvous needs to be limited by an upper bound $t_{\max}$ that is determined by the mission period. Therefore, the following constraints should be satisfied in this optimal problem

$$\begin{cases} a_t(1-e_t) \geq R_E + h_{p\min} \\ a_t \leq a_{t\max} \\ t_1 + t_2 \leq t_{\max} \end{cases} \tag{2}$$

where a_t is the semimajor axis of transfer orbit, e_t is the eccentricity of transfer orbit, and R_E is the radius of Earth.

B. Computation Model of Time-Open Two-Impulse Lambert Rendezvous

The initial orbital elements of the chaser and target are given as follows: the semimajor axis a_j, the eccentricity e_j, the right ascension of the ascending node Ω_j, the orbit inclination i_j, the argument of perigee ω_j, the true anomaly $\theta_{j0}(j = p, t)$, where the subscripts "p" and "t" refer to the parking orbit and the target orbit, respectively, and the subscript "0" refers to the initial time t_0. Only the true anomaly is with the subscript "0" because other orbital elements of the parking orbit and the target orbit are fixed under the assumption of a Keplerian two-body orbit.

From its initial position to the point P_1, the chaser has been orbiting on the parking orbit for time t_1. Therefore, the true anomaly of the chaser at the point P_1 on the parking orbit θ_{p1} is calculated by

$$\begin{cases} E_{p0} = 2a\ \tan[\tan(\theta_{p0}/2)\sqrt{(1-e_p)/(1+e_p)}] \\ M_{p0} = (E_{p0} - e_p\ \sin\ E_{p0}) \\ M_{p1} = M_{p0} + \sqrt{\mu/a_p^3}\cdot t_1 \\ M_{p1} = (E_{p1} - e_p\ \sin\ E_{p1}) \\ \theta_{p1} = 2a\ \tan[\tan(E_{p1}/2)\sqrt{(1+e_p)/(1-e_p)}] \end{cases} \tag{3}$$

where E is the eccentric anomaly, M is mean anomaly, and the subscript "1" refers to the point P_1. Note that the true anomaly θ_{p1} increases monotonically with the parking time t_1. And then, the position vector $\mathbf{r}_1$ and the velocity vector $\mathbf{v}_1$ of the chaser at the point P_1 on the parking orbit can be calculated according to the true anomaly θ_{p1} and other orbital elements of the parking orbit.

Also, from its initial position to the point P_2, the target has been orbiting on the target orbit for time $t_1 + t_2$. Therefore, the true anomaly of the target at the point P_2 on the target orbit θ_{p2}, which increases with the total time $t_1 + t_2$, is calculated similarly. The position vector $\mathbf{r}_2$ and the velocity vector $\mathbf{v}_2$ of the target at the point P_2 on the target orbit can be obtained according to the true anomaly of the target at the point P_2 on the target orbit θ_{p2} and other orbital elements of the target orbit.

Briefly, as the initial orbital elements of chaser and target are given, the position vectors $\mathbf{r}_1$, $\mathbf{r}_2$ and the velocity vectors $\mathbf{v}_1$, $\mathbf{v}_2$ are the functions of the parking time t_1 and the transfer time t_2:

$$\mathbf{r}_1 = \mathbf{R}(t_1), \qquad \mathbf{v}_1 = \mathbf{V}(t_1) \tag{4}$$

$$\mathbf{r}_2 = \mathbf{R}(t_1 + t_2), \qquad \mathbf{v}_2 = \mathbf{V}(t_1 + t_2) \tag{5}$$

Next, the transfer orbit is determined by the position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ and the transfer time t_2. The solution of the transfer orbit is a standard Lambert problem [2]. From Lagrange's formulation of the multiple-revolution Lambert problem, the relationship between the transfer time t_2 and the semimajor axis a_t of transfer orbit can be expressed as

$$t_2 = \sqrt{a_t^3/\mu}[2N\pi + (\alpha - \sin\ \alpha) - (\beta - \sin\ \beta)] \tag{6}$$

where μ is the gravitational constant, N is the number of allowed revolutions, and α and β are Lagrange's parameters defined by

$$\sin^2\alpha/2 = (r_1 + r_2 + d)/4a_t \tag{7}$$

$$\sin^2\beta/2 = (r_1 + r_2 - d)/4a_t \tag{8}$$

where d is the distance between the point P_1 and the point P_2. According to the geometric interpretation of the angles α and β [21], four distinct cases are obtained (Fig. 2):

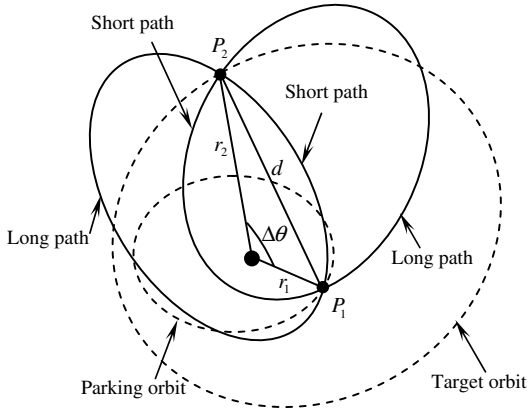

Fig. 2 Short path and long path for Lambert problem.

Case 1: If the transfer angle $\Delta\theta \le \pi$ and for a short path, then

$$\alpha = 2\text{arc sin}\sqrt{(r_1 + r_2 + d)/4a_t}, \qquad \alpha \in [0, \pi] \tag{9}$$

$$\beta = 2\text{arc sin}\sqrt{(r_1 + r_2 - d)/4a_t}, \qquad \beta \in [0, \pi] \tag{10}$$

Case 2: If the transfer angle $\Delta\theta \le \pi$ and for a long path, then

$$\alpha = 2\pi - 2\text{arc sin}\sqrt{(r_1 + r_2 + d)/4a_t}, \qquad \alpha \in [\pi, 2\pi] \tag{11}$$

$$\beta = 2\text{arc sin}\sqrt{(r_1 + r_2 - d)/4a_t}, \qquad \beta \in [0, \pi] \tag{12}$$

Case 3: If the transfer angle $\Delta\theta \ge \pi$ and for a short path, then

$$\alpha = 2\pi - 2\text{arc sin}\sqrt{(r_1 + r_2 + d)/4a_t}, \qquad \alpha \in [\pi, 2\pi] \tag{13}$$

$$\beta = -2\text{arc sin}\sqrt{(r_1 + r_2 - d)/4a_t}, \qquad \beta \in [-\pi, 0] \tag{14}$$

Case 4 If the transfer angle $\Delta\theta \ge \pi$ and for a long path, then

$$\alpha = 2\text{arc sin}\sqrt{(r_1 + r_2 + d)/4a_t}, \qquad \alpha \in [0, \pi] \tag{15}$$

$$\beta = -2\text{arc sin}\sqrt{(r_1 + r_2 - d)/4a_t}, \qquad \beta \in [-\pi, 0] \tag{16}$$

We can use Newton's iterative method to solve the transcendental Eq. (6) for the value a_t. For each $1 \le N \le N_{\max}$, there are two semimajor axes corresponding to a specified transfer time t_2 and they define two N-revolution transfer orbits. However, for $N = 0$, there is only one semimajor axis corresponding to a transfer time t_2 so that only one transfer orbit is determined. Therefore, for a specified transfer time t_2, there are $2N_{\max} + 1$ solutions for the multiple-revolution Lambert problem, where $N_{\max}$ is the maximum number of allowed revolutions for the specified transfer time t_2. The details about solutions of Lagrange's formulation [Eq. (6)] are given in [8].

If the transfer angle $\Delta\theta$ is generated by a counterclockwise rotation or a clockwise rotation from $\mathbf{r}_1$ to $\mathbf{r}_2$, the transfer angle $\Delta\theta$ is uniquely determined by the position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$. Therefore, either the cases of $\Delta\theta \le \pi$ or the cases of $\Delta\theta \ge \pi$ need to be calculated and $2N_{\max} + 1$ transfer orbits are determined. But in a general case, it is not certain whether it is better to choose the counterclockwise rotation transfer orbits (corresponding to $\Delta\theta < \pi$ in Fig. 2) or the clockwise rotation transfer orbits (corresponding to $\Delta\theta > \pi$ in Fig. 2). Therefore, the cases of $\Delta\theta \le \pi$ and $\Delta\theta \ge \pi$ must be calculated so that there are $4N_{\max} + 2$ solutions for Eq. (6).

Because the semimajor axis a_t of the transfer orbit and the angles α and β can be solved using Eqs. (6–8), other orbital elements of transfer orbit are calculated using the basic theory of Keplerian orbits (see Appendix). Therefore, the velocity vector $\mathbf{v}_{t1}$ of the chaser at the point P_1 in the transfer orbit and the velocity vector $\mathbf{v}_{t2}$ of the chaser at the point P_2 in the transfer orbit are obtained according to the orbital elements of the transfer orbit. Then, the two impulses are given by

$$\Delta\mathbf{v}_1 = \mathbf{v}_{t1} - \mathbf{v}_1 \tag{17}$$

$$\Delta\mathbf{v}_2 = \mathbf{v}_2 - \mathbf{v}_{t2} \tag{18}$$

The computation process of time-open two-impulse rendezvous can be summarized as three steps.

Step 1: Calculate the position vector $\mathbf{r}_1$ and the velocity vector $\mathbf{v}_1$ of the chaser at the point P_1 on the parking orbit according to the parking time t_1, calculate the position vector $\mathbf{r}_2$ and the velocity vector $\mathbf{v}_2$ of the target at the point P_2 on the target orbit according to the parking time t_1 and the transfer time t_2.

Step 2: Determine transfer orbits according to the position vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ and the transfer time t_2 using multiple-revolution Lambert solutions.

Step 3: Calculate the velocity vector $\mathbf{v}_{t1}$ of the chaser at the point P_1 on the transfer orbit and the velocity vector $\mathbf{v}_{t2}$ of the chaser at the point P_2 on the transfer orbit, and obtain the two impulses $\Delta\mathbf{v}_1$ and $\Delta\mathbf{v}_2$ for orbit rendezvous.

Finally, the total velocity increment Δv for orbital rendezvous can be calculated by Eqs. (1), (17), and (18).

According to these computation processes, it is certain that the total velocity increment Δv for orbital rendezvous is a function of the parking time t_1 and the transfer time t_2 if the initial orbital elements of the chaser and the target are given. However, it is emphasized that there are $4N_{\max} + 2$ transfer orbit solutions for one specified (t_1, t_2) for the multiple-revolution Lambert problem. Therefore, the cost function J, namely the total velocity increment Δv for orbital rendezvous, is a multiple-valued function of the optimal variables t_1 and t_2 in this optimal problem (Fig. 3). For each $1 \le N \le N_{\max}$, there are four values of the cost function corresponding to a given (t_1, t_2), and for $N = 0$, there are only two values of the cost function corresponding to a given (t_1, t_2). In fact, only the minimum value of the cost function in these $4N_{\max} + 2$ solutions, defined as $J_{\Delta v} = \min(J_1, J_2, \ldots, J_{4N_{\max}+2})$ with the corresponding number of revolution $N_{\Delta v}$, needs to be considered for minimizing the cost function.

III. Cost Function Characteristics

Clearly, using gradient-based optimization methods such as Matlab's "fmincon" and "fminsearch" can find the global minimum of the cost function as long as the cost function is convex in the whole searching space. For the cost function that is nonconvex, gradient-based optimization methods are in general not valid. A genetic algorithm (GA) is an effective method to solve the optimal problem with the convex or nonconvex cost function. However, GA is vulnerable to premature convergence so that the method does not guarantee the global optimal solution for an optimal problem with multiple local minima. Therefore, it is necessary to know the cost function characteristics for choosing the optimal method.

The convexity of the cost function is checked according to the second-order derivative of the cost function with respect to the optimal variables. But for the optimal problem presented in this paper, the second-order derivatives of the cost function with respect to t_1 and t_2 are not available because of the multiple-valued cost function of the variables t_1, t_2 and the existence of constraints. Therefore, the characteristics of the cost function are investigated using only numerical analysis.

For the numerical analysis example, initial orbit elements of the chaser and the target are given in Table 1. The chaser is assumed to be flying at a 2000-km altitude near-circular Earth orbit with small eccentricity and the target is flying almost at a geosynchronous orbit.

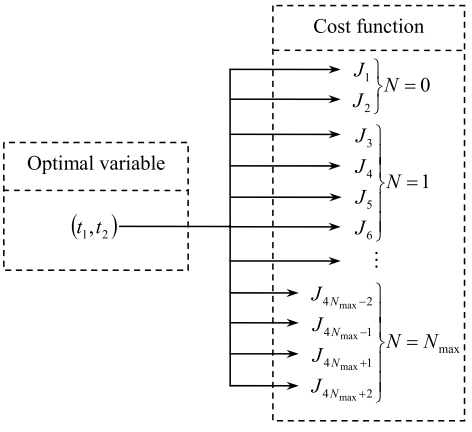

Fig. 3 Multiple valued cost function.

Table 1 Initial orbit elements of spacecraft

Orbit elements	Chaser	Target
Semimajor axis a, km	$R_E + 2000$	$R_E + 36,000$
Eccentricity e	0.002	0.0002
Orbit inclination i, deg	60.00	55
Right ascension of the ascending node Ω, deg	30.00	35
Argument of perigee ω, deg	0.00	−20
True anomaly θ, deg	0.00	30

But they are noncoplanar because of the different orbit inclination and right ascension of the ascending node. In addition, assume the upper bound of the total time for rendezvous $t_{\max} = 24$ h so that the total mission time is constrainted, the lower bound of perigee altitude of the transfer orbit $h_{p\min} = 200$ km, and the upper bound of the semimajor axis of transfer orbit $a_{t\max} = 100,000$ km.

The case without constraints of perigee altitude and semimajor axis of transfer orbit is investigated first. The cost function $J_{\Delta v}$ and corresponding contour are shown in Figs. 4 and 5, respectively. It is obvious that there are many local minima. As seen in Fig. 6, the maximum number of revolutions $N_{\Delta v}$ corresponding to $J_{\Delta v}$ is two and is available only in a very small region of (t_1, t_2). From Fig. 7 it can been seen that the maximum number of allowed revolutions $N_{\max}$ is also two for $t_1 \in (0, 24\text{ h})$ and $t_2 \in (0, 24\text{ h})$, and the corresponding region is also very small.

Numerical analysis is undertaken for the case with constraints of perigee altitude and semimajor axis of the transfer orbit. From Figs. 8 and 9, it is shown that not only are there many local minima of the cost function $J_{\Delta v}$, but also the cost function $J_{\Delta v}$ is discontinuous. The maximum number of revolutions $N_{\Delta v}$ corresponding to $J_{\Delta v}$ (Fig. 10) and the maximum number of allowed revolutions $N_{\max}$ (Fig. 11) reduce to one. The reason for a discontinuous cost function and reduction of $N_{\Delta v}$ and $N_{\max}$ is that the feasible region of (t_1, t_2) decreases under the effects of constraints. Therefore, minimizing the total velocity increment for time-open two-impulse rendezvous is an optimal problem with the discontinuous nonconvexcost function and multiple local minima. Using only gradient-based optimization methods can not find the global minimum with certainly in the whole region of (t_1, t_2). But from Fig. 12, in which the cost function over the intervals $[t_1] = [56382\text{ s}, 56562\text{ s}]$ and $[t_2] = [17883\text{ s}, 18063\text{ s}]$ is shown, it is worth noting that the local cost function $J_{\Delta v}$ is convex. If the whole region of (t_1, t_2) is split into many small subspaces and the convexity of the cost function is guaranteed for every subspace, it

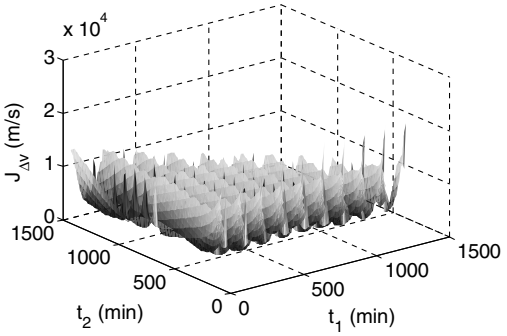

Fig. 4 Cost function $J_{\Delta v}$ without constraints.

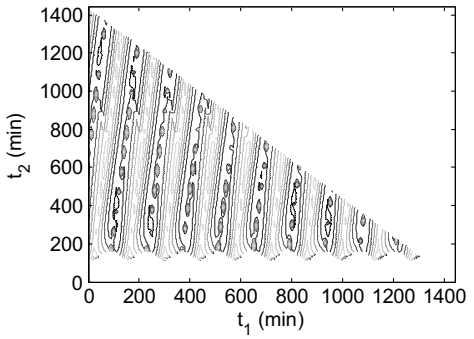

Fig. 5 Contour of cost function $J_{\Delta v}$ without constraints.

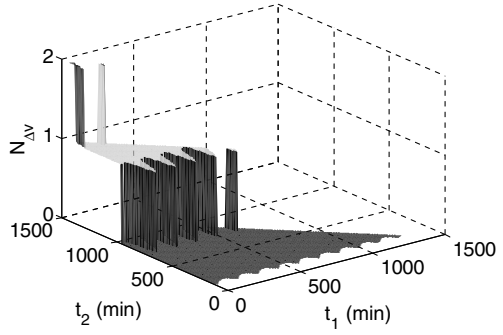

Fig. 6 Number of revolutions $N_{\Delta v}$ corresponding to $J_{\Delta v}$ without constraints.

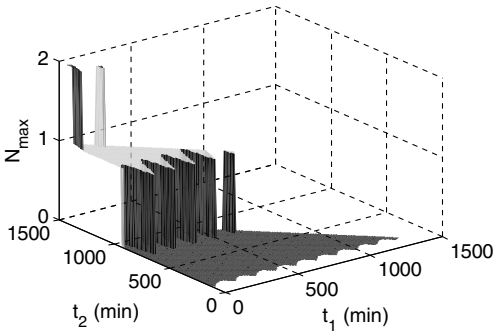

Fig. 7 Maximum number of allowed revolutions $N_{\max}$ without constraints.

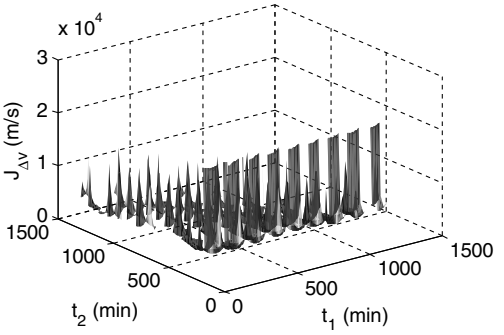

Fig. 8 Cost function $J_{\Delta v}$ with constraints.

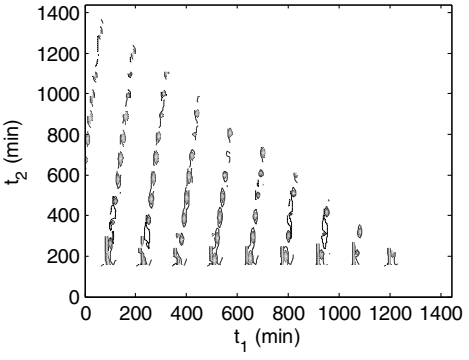

Fig. 9 Contour of cost function $J_{\Delta v}$ with constraints.

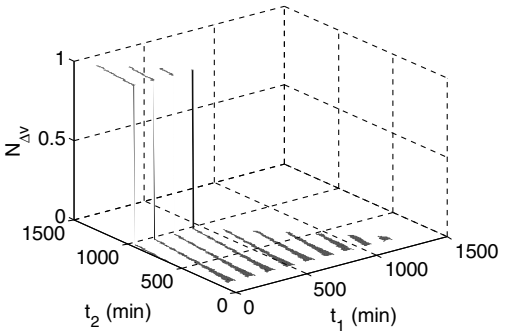

Fig. 10 Number of revolutions $N_{\Delta v}$ corresponding to $J_{\Delta v}$ with constraints.

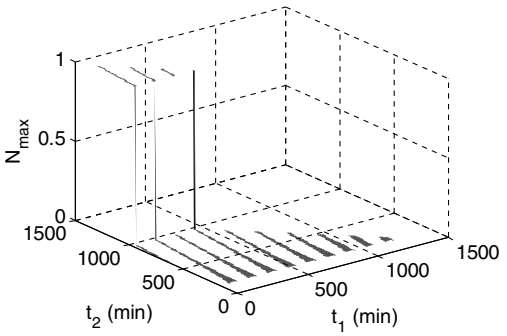

Fig. 11 Maximum number of allowed revolutions $N_{\max}$ with constraints.

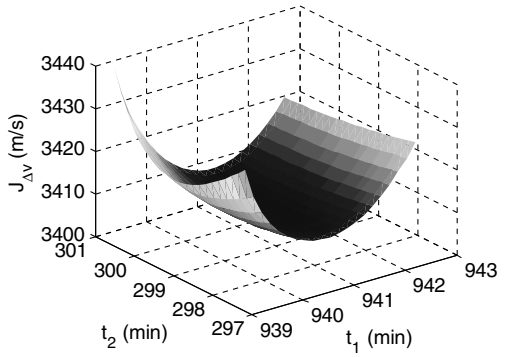

Fig. 12 Local cost function $J_{\Delta v}$ with constraints.

is feasible to find the local minimum for every section using gradient-based optimization methods so that the global minimum is obtained from these local minima.

IV. Interval Analysis Optimization Method

A. Basic Concept of Interval Analysis

Interval analysis is the theory dealing with interval numbers and the arithmetic operations on them [13,14]. An interval number $[x]$ is defined as an ordered pair of real numbers $[\underline{x}, \overline{x}]$ with $\underline{x} \le \overline{x}$, where $\underline{x}$ is the lower bound (or infimum) of the interval number $[x]$ and $\overline{x}$ is the upper bound (or supremum) of the interval number $[x]$. In the following text, interval variables and numbers will be denoted with square brackets.

Interval arithmetic contains the same operators as ordinary arithmetic such as the basic computational operations of addition, subtraction, multiplication, and division. The key point is that computing with intervals is computing with sets. Because all unary interval arithmetic definitions have the same general form, it is possible to summarize them neatly in one line [15]:

$$[x]\odot[y] = \{x \odot y : x \in [x], y \in [y]\}, \quad \text{where } \odot \text{can be} +, -, \times, \text{ or} / \tag{19}$$

Furthermore, functions of interval variables can be defined as by treating these as "unary operations"

$$f([x]) = \{f(x) : x \in [x]\} \tag{20}$$

Depending on n interval variables $[x_1], \ldots, [x_n]$, more general interval functions are defined as $f([x_1], ..., [x_n])$. The core of interval analysis is to use interval arithmetic to form an interval inclusion function $[f]([x_1], \ldots, [x_n])$ of any function $f(x_1, \ldots, x_n)$. This property of interval arithmetic follows from the inclusion function theorem given by Moore et al. [15].

Corollary: If $[f]$ is a rational interval function and an interval extension of f, then

$$f([x_1], \ldots, [x_n]) \subseteq [f]([x_1], \ldots, [x_n]) \tag{21}$$

That is, an interval value of $[f]$ contains the range of values of the corresponding real function f when the real arguments of f lie in the intervals shown.

Therefore, we have a means for the finite evaluation of upper and lower bounds on the ranges of values of real rational functions. However, a property of interval arithmetic, called "interval dependency," is a crucial consideration when using interval computations. Two examples of interval arithmetic are shown:

$$[0, 1] - [0, 1] = [-1, 1] \tag{22}$$

$$[-1, 1]^2 = [0, 1], \quad \text{whereas } [-1, 1] \cdot [-1, 1] = [-1, 1] \tag{23}$$

These two equations show that in interval arithmetic $[x] - [x] \neq 0$ and $[x]^2 \neq [x] \cdot [x]$ but the interval of $[x] - [x]$ does contain 0 and the interval $[x] \cdot [x]$ does contain $[x]^2$. The reason for overestimation is that the two terms are considered as different interval variables and vary independently. It is also a major reason why simply replacing floating point computations by intervals is not likely to lead to satisfactory results.

For reducing the influence of interval dependency, van Kampen et al. [16] present the "Subdivisions and Refinements" technique, which can be used to compute arbitrarily sharp lower and upper bounds on the exact range of values of a wide variety of real-valued functions of n variables by subdividing the domain of arguments and taking the union of interval evaluations over the elements of the subdivision. This technique, also known as "splitting," converges at least linearly in the width of the pieces in the subdivision.

B. Global Optimization Using Interval Branch and Bound Algorithm

Because interval algorithm provides upper and lower bounds on ranges of values of function over sets, a global optimization algorithm without using derivatives, called interval branch and bound algorithm, is presented via this property of interval algorithms.

The inclusion function theorem can be used to compute guaranteed bounds on the cost function $J(t_1, t_2)$ for a given interval vector $([t_1], [t_2])$. However, as described in Sec. II, the iteration calculation is inevitable for solving the transcendental Eq. (6) to obtain the middle variable, namely, the semimajor axis a_t of transfer orbit. The complexity and dependency effect in the interval calculation will remarkably increase once the iteration calculation occurs. Therefore, to avoid the iteration interval calculation, the semimajor axis a_t of transfer orbit is taken as the optimal variable. The complete search space is defined by three interval variables $[t_1]$, $[t_2]$, and $[a_t]$, which can be combined into a single interval vector or box: $[\mathbf{x}] = ([t_1], [t_2], [a_t])^T$. Furthermore, by regarding the transcendental Eq. (6) as a constraint formulation, the optimization problem can now be written as

$$\begin{cases} \min J(\mathbf{X}), \qquad \forall \mathbf{X} \in [\mathbf{X}] \\ \text{subject to the contraints} \\ p_1(\mathbf{X}) = R_E + h_{p\min} - a_t(1 - e_t) \le 0 \\ p_2(\mathbf{X}) = a_t - a_{t\max} \le 0 \\ p_3(\mathbf{X}) = t_1 + t_2 - t_{\max} \le 0 \\ p_4(\mathbf{X}) = t_2 - \sqrt{a_t^3/\mu}[2N\pi + (\alpha - \sin\alpha) - (\beta - \sin\beta)] = 0 \end{cases} \tag{24}$$

Note that the expression of constraint formulation $p_4(\mathbf{X})$ is different in the different cases of the number of revolution N and the intervals of angles α and β. As shown in Eqs. (9–16), for a given N, there are four distinct cases: case 1 for$\alpha \in [0, \pi]$, $\beta \in [0, \pi]$, case 2 for $\alpha \in [\pi, 2\pi], \beta \in [0, \pi]$, case 3 for$\alpha \in [0, \pi], \beta \in [-\pi, 0]$, and case 4 for $\alpha \in [\pi, 2\pi]$, $\beta \in [-\pi, 0]$. The point is certainly infeasible only if the point is not satisfied the constraint formulation $p_4(\mathbf{X})$ for any case and any number of revolutions N.

Because it is guaranteed that the real cost function value J for any point in $[\mathbf{X}]$ lies in the cost function interval $[J]$, the information obtained from the upper bound sup $[J]$ can be used to check if the global minimum may lie in $[\mathbf{X}]$ or is guaranteed not to lie in box $[\mathbf{X}]$. The goal of the branch and bound algorithm is to efficiently remove subboxes from the initial search box, for which it is guaranteed that they do not contain the set $[\mathbf{X}^*]$, for which the cost function has a global minimum J^*. If $[\mathbf{X}_i]$ is assumed to be any subbox of $[\mathbf{X}]$, which may even be a point (degenerate box) in $[\mathbf{X}]$, there are two kinds of deletion tests on $[\mathbf{X}_i]$ made in the algorithm:

1) *Feasibility Test*: If any one or more of the following formulations is satisfied:

$$\inf([p_1]([\mathbf{X}_i])) > 0 \tag{25}$$

$$\inf([p_2]([\mathbf{X}_i])) > 0 \tag{26}$$

$$\inf([p_3]([\mathbf{X}_i])) > 0 \tag{27}$$

$$0 \notin [p_4]([\mathbf{X}_i]) \tag{28}$$

then $[\mathbf{X}_i]$ is certainly infeasible (contains no feasible points with satisfying all constraints), delete $[\mathbf{X}_i]$.

2) *Optimality Test*: Let J^*_{est} be the best estimation for the global minimum of the cost function. If $\inf([J]([\mathbf{X}_i])) > J^*_{\text{est}}$, then $[\mathbf{X}_i]$ cannot contain a feasible global minimum, and will be deleted. Usually, the global minimum estimation J^*_{est} can be the smallest upper bound in the cost function intervals over all current subboxes:

$$J^*_{\text{est}} = UJ^* = \min(\sup([J]([\mathbf{X}_i]))), \qquad i = 1, \ldots, n \tag{29}$$

Furthermore, J^*_{est} can be the cost function value J_f corresponding to any feasible point in $[\mathbf{X}]$ as long as J_f is lower than the current best estimation. In this paper, a combination of both global minimum estimations is used.

The full scheme of the interval branch and bound algorithm is given for the constrained optimization problem as follows. The first step is to define the initial search space as a multidimensional interval box $[\mathbf{X}]$. Next, the cost function is evaluated for this complete search space, resulting in $[J]$. This interval is likely to be very wide, but it is guaranteed that the global minimum is contained in it. The next step is to split the initial search space into subboxes. There are many possible ways to split an interval box, for example, bisecting in all dimensions, bisecting in the dimension with the widest interval, bisecting in the dimension that is set according to the sensitivity of optimal variables to the cost function, and so on. The same global minimum can be found using each splitting method. The type of splitting definitely influences the efficiency of the interval algorithm. But there is still not the theory to determine which method is best. Then, the feasibility test and the optimality test for the two new subboxes are done to delete the subbox that is not feasible or cannot contain the global minimum. The remaining subbox is put into the last list of subboxes.

Table 2 Results of optimization using GA

Number	t_1, s	t_2, s	J, m/s	N	α, β Case
1	56,473.67	17,977.29	3404.8614	0	Case 1
2	6607.67	24,222.40	3462.2279	0	Case 4
3	10,490.92	63,956.74	3413.3730	1	Case 4
4	56,473.67	17,974.65	3404.8602	0	Case 1
5	6607.67	24,222.40	3462.2279	0	Case 4
6	10,490.92	63,956.74	3413.3730	1	Case 4
7	56,473.67	17,977.29	3404.8614	0	Case 1
8	10,461.92	63,911.25	3414.2204	1	Case 4
9	56,478.95	18,000.36	3404.9872	0	Case 1
10	56,474.99	17,975.31	3404.8674	0	Case 1

The current best estimation of the cost function J^*_{est} can be updated if either the upper bound of the cost function corresponding to the added subbox or the cost function value J_f corresponding to the feasible point selected in the added subbox is lower than the current best estimation. When J^*_{est} is lowered, an optimality check is done again to see whether there are the boxes in the list with a lower bound on the cost function that is higher than the current best estimate. If so, then these boxes are removed from the list, thereby reducing the search space.

Splitting, evaluating, and discarding of subboxes loop constantly until the list of subboxes is empty. A stopping criteria is required to check if a given box $[\mathbf{X}_i]$ can be considered as a solution. If the width of box $[\mathbf{X}_i]$ is lower than the user defined threshold ε_x or if the width of $[J]([\mathbf{X}_i])$ is lower than the user defined threshold ε_J:

$$\begin{gathered} \max(w([\mathbf{X}_i])) \le \varepsilon_x \quad \text{or} \quad w([J]([\mathbf{X}_i])) \le \varepsilon_J \\ w([\cdot]) = \sup([\cdot]) - \inf([\cdot]) \end{gathered} \tag{30}$$

the box $[\mathbf{X}_i]$ will not be split again and it will be put in a list of solutions.

V. Numerical Results

The numerical optimization is done for the impulsive rendezvous example given in Sec. III. First, GA is used to find the minimum of total velocity increment for orbital rendezvous. Second, the interval optimization method is validated in the same optimization rendezvous example. All computations are carried out using Matlab 2008b M-files in the hardware environment of an Intel Core CPU i3 2.53 GHz, RAM DDR3 2 GB. The INTLAB toolbox by Rump is applied for the interval computations[¶].

A. Optimization Results Using Genetic Algorithm

GA evolves for 200 generations with a population size of 100 individuals, which are generated randomly in the search space $t_1 \in (0, 24\text{ h})$ and $t_2 \in (0, 24\text{ h})$. All initial parameters and constraint conditions have been given in Sec. III. The results for 10 times optimizations are shown in Table 2. We can see that the results for each optimization except numbers 2 and 5 are different, respectively. From Fig. 13, it is shown that the results for 10 times optimizations are distributed in three zones. The results of numbers 1, 4, 7, 9, and 10 are all for $N = 0$, case 1 about α and β and lie in one zone. The results of numbers 2 and 5 are the same completely, corresponding to $N = 0$, case 4 about α and β. The results of numbers 3, 6, and 8 are located in another zone and correspond to $N = 1$, case 4 about α and β. The maximal difference of the cost function in 10 times results reaches 57.37 m/s. The reason for dispersive optimization results is that there are multiple local minima in the presented optimal problem. It is only guaranteed that one local minimum can be found in each optimization using a GA.

[¶]Additional information from the Institute for Reliable Computing, Hamburg University of Technology is available online at http://www.ti3.tu-harburg.de/rump/intlab/ [accessed July 2011].

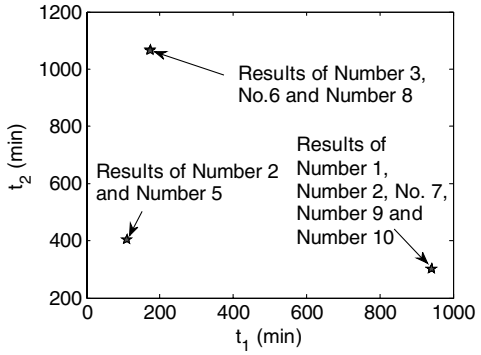

Fig. 13 Distribution of optimization solution using GA.

B. Optimization Results Using the Interval Algorithm

All initial parameters and constraint conditions given in Sec. III are still used in the interval optimization. The initial search space is defined as

$$[\mathbf{X}] = ([t_1], [t_2], [a_t])^T = \left([0, 24\text{ h}], [0, 24\text{ h}], \left[\frac{a_t(1-e_t) + R_E + h_{p\min}}{2}, 100000\text{ km}\right]\right)^T \tag{31}$$

The stopping criteria is $\varepsilon_J = 0.01$.

As mentioned in Sec. IVB, the constraint $p_4(\mathbf{X})$ is changed with differences of the number of revolution N and the case about angles α and β. In this paper, the interval optimization is respectively done for the different constraint function $p_4(\mathbf{X})$. Because the maximum number of allowed revolutions for this rendezvous example (Fig. 11) is $N_{\max} = 1$, there are 8 different constraint functions $p_4(\mathbf{X})$ for $N = 0$, case $1 \sim 4$ and $N = 1$, case $1 \sim 4$. Therefore, the interval optimization will be undertaken for eight times according the different constraint function $p_4(\mathbf{X})$.

Firstly, the results for $N = 0$, case 1 about α and β are given in Table 3. In the second column of Table 3, the first number is the number of subboxes in the current list before the feasibility check and the optimality check and the second number is that after the feasibility check and the optimality check. The number of subboxes in the next iteration always doubles because each subbox in the current list is split into two new subboxes in each iteration. The number of subboxes decreases after the feasibility check and the optimality check because the subboxes that are not feasible or cannot contain the global minimum are deleted in each iteration. Although the operations of splitting and discarding narrow the search space continually, the number of subboxes in the 36th iteration reaches 528,850 such that the calculation exceeds the computer's RAM utmost before subboxes reach the stopping criteria. The primary reason for the calculation overflow is the interval dependency. For example, the true width of the cost function over the intervals $[t_1] = [56382\text{ s}, 56562\text{ s}]$ and $[t_2] = [17883\text{ s}, 18063\text{ s}]$ is about 20 m/s (see Fig. 12), but the interval calculation result is $[J] = [161.63\text{ m/s}, 18221.48\text{ m/s}]$ over the same interval space. The interval width of the cost function is magnified greatly so that the evaluating of subbox is always overestimated and the subboxes in the list cannot be deleted accurately by the feasibility check and the optimality check. The interval dependency is inevitable in interval calculation. For this optimal problem, the influence of interval dependency is distinct because of the complex calculation for the cost function.

Because for this optimal problem, using the interval algorithm can only narrow the search space but not find the global minimum directly in the restricted hardware environment, the gradient-based optimization method is considered as the complementary of the interval algorithm.

The global minimum also cannot be found using only the gradient-based optimization method because of the nonconvex cost function over the whole search space. Matlab's "fmincon" is employed to find the optimal solution in the whole search space $t_1 \in (0, 24\text{ h})$ and $t_2 \in (0, 24\text{ h})$, but the result is "NaN." Therefore, an improved interval algorithm combining the gradient-based optimization method is presented. The interval algorithm is used to narrow the search space and split the search space into small subspaces in advance. Once the widths of interval variables $[t_1]$ and $[t_2]$ in all current subboxes are lower than 100 s, the interval calculation is stopped. And then, Matlab's "fmincon" is used to find the local minimum for each remained subbox whose middle point is taken as the initial search point.

The results for eight cases with different constraint functions $p_4(\mathbf{X})$ are shown in Table 4. The number of remaining subboxes after the interval calculation is listed in the second column. The minimum in all local minima found by Matlab's "fmincon" is listed in the fifth

Table 3 Results for $N = 0$, case 1 using interval algorithm

Number of iteration	Number of subboxes in list	Maximum width of $[t_1]$ in list, s	Maximum width of $[t_2]$ in list, s	Maximum width of $[a_t]$ in list, m
1	$1 \rightarrow 1$	86,400	86,400	755,332,331.004
2	$2 \rightarrow 2$	43,200	86,400	755,332,331.004
3	$4 \rightarrow 2$	43,200	43,200	755,332,331.004
4	$4 \rightarrow 4$	43,200	43,200	377,666,165.502
5	$8 \rightarrow 8$	21,600	43,200	377,666,165.502
6	$16 \rightarrow 14$	21,600	21,600	377,666,165.502
⋮	⋮	⋮	⋮	⋮
33	$199,400 \rightarrow 152,109$	21.09375	84.37500	73,762.9229496
34	$304,218 \rightarrow 233,189$	21.09375	42.18750	73,762.9229496
35	$466,378 \rightarrow 264425$	21.09375	42.18750	36,881.4614748
36	$528,850 \rightarrow$ Out of memory			

Table 4 Results of optimization using improved IA

Number	Number of remaining subboxes	t_1, s	t_2, s	J, m/s	N	Case about α and β
1	16,135	56,472.24	17,973.35	3404.8576	0	Case 1
2	47,464	6803.16	24,176.06	3517.8738	0	Case 2
3	17,406	56,340.89	18,042.04	3426.5936	0	Case 3
4	9223	6602.99	24,203.33	3462.1952	0	Case 4
5	7691	18,822.14	58,289.69	3628.7727	1	Case 1
6	23,751	10,608.18	63,774.49	3439.3996	1	Case 2
7	9786	18,211.32	59,478.23	3705.7524	1	Case 3
8	2254	10,486.64	63,940.40	3413.3147	1	Case 4

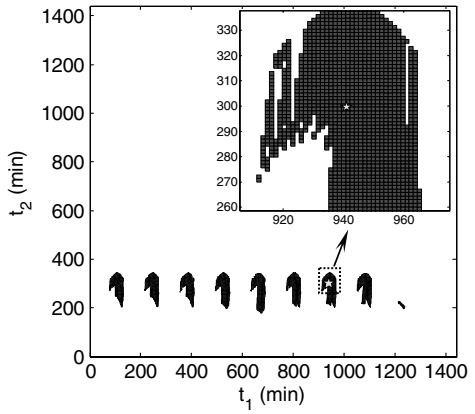

Fig. 14 Remaining subboxes for $N = 0$, case 1 about α and β.

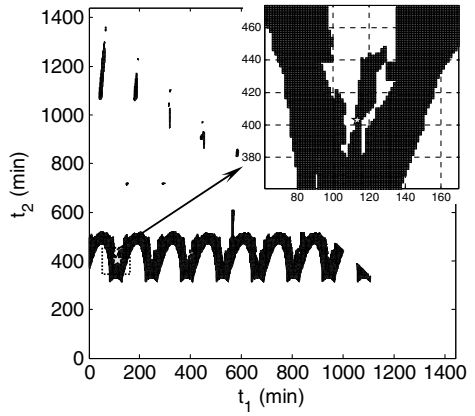

Fig. 15 Remaining subboxes for $N = 0$, case 2 about α and β.

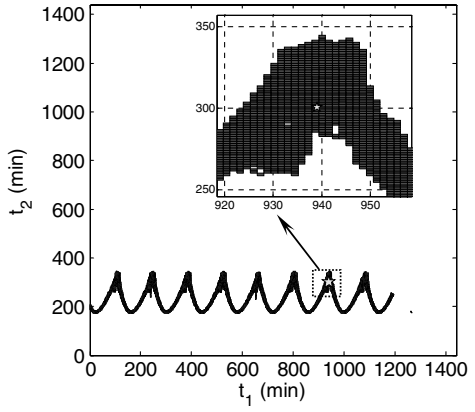

Fig. 16 Remaining subboxes for $N = 0$, case 3 about α and β.

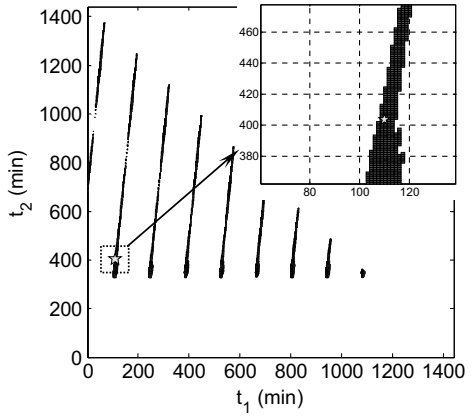

Fig. 17 Remaining subboxes for $N = 0$, case 4 about α and β.

column. The remaining subboxes after the interval calculation for different cases are shown in Figs. 14–21, where the stars denote the optimal solutions. We can see that the optimal solution for $N = 0$, case 1 about α and β by the improved interval algorithm is close to and better than the solutions of numbers 1, 4, 7, 9, and 10 by GA, the optimal solution for $N = 0$, case 4 about α and β by the improved interval algorithm is close to and better the solutions of numbers 2 and 5 by the GA, and the optimal solution for $N = 1$, case 4 about α and β by the improved interval algorithm is close to and better the solutions of numbers 3, 6, and 8 by the GA. Also, optimal solutions for other cases are found by the improved interval algorithm. Although it still cannot be guaranteed that the optimal solution found by the improved interval algorithm is the global minimum over the whole search space because the convexity of each remaining subbox cannot be guaranteed, the improved interval algorithm is more rigorous than the GA and can be used to find the better local optimal solution.

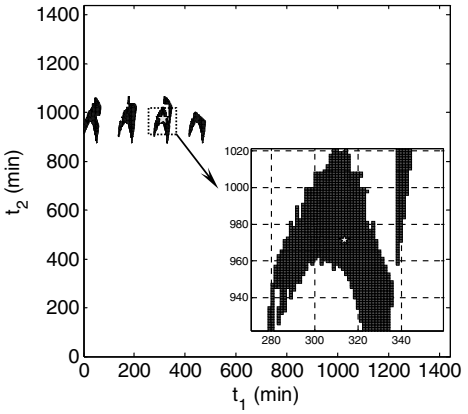

Fig. 18 Remaining subboxes for $N = 1$, case 1 about α and β.

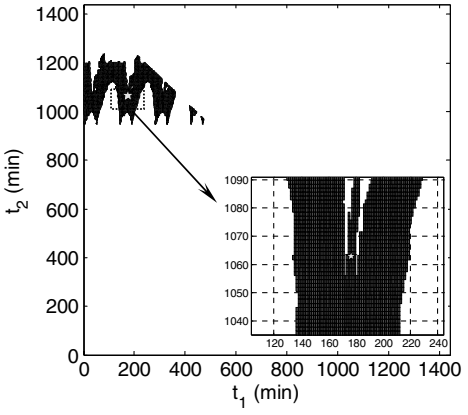

Fig. 19 Remaining subboxes for $N = 1$, case 2 about α and β.

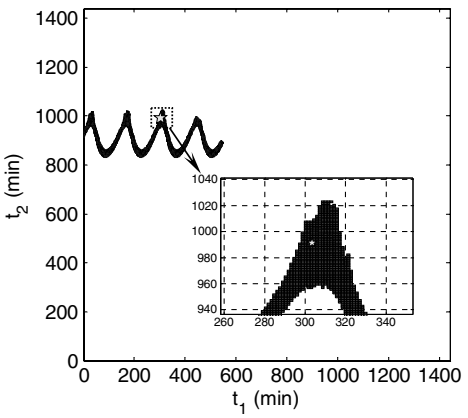

Fig. 20 Remaining subboxes for $N = 1$, case 3 about α and β.

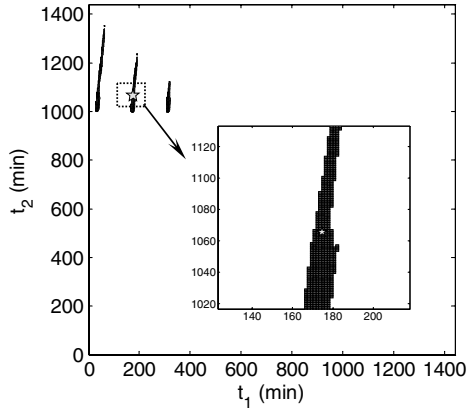

Fig. 21 Remaining subboxes for $N = 1$, case 4 about α and β.

VI. Conclusions

A time-open constrained Lambert rendezvous optimization problem is presented in this paper. The interval optimization method is introduced to solve this optimal problem with a discontinuous nonconvex cost function and multiple local minima.

The results of a numerical example illustrate that although the global minimum for the present optimal problem can't be found by using only the interval algorithm in the restricted hardware environment because of the inevitable influence of interval dependency and the complex calculation for the cost function, the interval algorithm can narrow the search space and split the feasible space into many small subspaces. Although the cost function is definitely nonconvex over the whole search space, the possibility of convexity of the cost function over a small subspace increases greatly by the interval splitting. Therefore, combining the gradient-based optimization method, an improved interval algorithm is attempted. The initial search space is split and narrowed using the interval algorithm and then the gradient-based optimization method is used to find the local minimum for each remained subspace. The minimal velocity increment for orbital rendezvous from the improved interval algorithm is better than that from the genetic algorithm, though the global minimum is not still guaranteed.

In general, the interval algorithm is a feasible method for minimizing the velocity increment for time-open constrained Lambert rendezvous. It is notable that the interval algorithm can be used in the nonconvex optimal problem. But for a complex nonlinear optimal problem, the excessive computation load of the interval algorithm is inevitable to find the global optimal solution. This is the primary drawback of the interval algorithm.

Appendix: Solution of the Lambert Transfer Orbit

Once the semimajor axis a_t of the transfer orbit and the angles α and β are determined, other orbital elements of the transfer orbit can be solved as following steps.

Step 1: Calculate the eccentricity of the transfer orbit e_t

$$\gamma_1 = (1 - r_1/a_t)\cot(\alpha - \beta) - (1 - r_2/a_t)\csc(\alpha - \beta) \quad \text{(A1)}$$

$$\gamma_2 = 1 - r_1/a_t \quad \text{(A2)}$$

$$E_{t1} = \text{a tan } 2(\gamma_1, \gamma_2) \quad \text{(A3)}$$

$$e_t = (1 - r_1/a_t)/\cos E_{t1} \quad \text{(A4)}$$

where E_{t1} is the eccentric anomaly of point P_1 in the transfer orbit.

Step 2: Calculate the true anomaly of point P_1 on the transfer orbit θ_{t1} and the true anomaly of point P_2 on the transfer orbit θ_{t2}

$$E_{t2} = E_{t1} + (\alpha - \beta) \quad \text{(A5)}$$

$$\theta_{t1} = 2\text{a tan}\left(\sqrt{\frac{1+e_t}{1-e_t}} \cdot \tan\frac{E_{t1}}{2}\right) \quad \text{(A6)}$$

$$\theta_{t2} = 2\text{a tan}\left(\sqrt{\frac{1+e_t}{1-e_t}} \cdot \tan\frac{E_{t2}}{2}\right) \quad \text{(A7)}$$

where E_{t2} is the eccentric anomaly of point P_2 on the transfer orbit.

Step 3: Calculate the orbit inclination i_t and the right ascension of the ascending node Ω_t of the transfer orbit

$$\mathbf{H} = \begin{cases} \mathbf{r}_1 \times \mathbf{r}_2, & \Delta\theta \le \pi \\ \mathbf{r}_2 \times \mathbf{r}_1, & \Delta\theta \ge \pi \end{cases} \quad \text{(A8)}$$

$$i_t = \text{a tan } 2(\sqrt{H_x^2 + H_y^2}, H_z) \quad \text{(A9)}$$

$$\Omega_t = \text{a tan } 2(H_x, -H_y) \quad \text{(A10)}$$

Step 4: Calculate the argument of perigee of the transfer orbit ω_t

$$u_{t1} = \text{a tan } 2(-r_{1x}\cos i_t \sin\Omega_t + r_{1y}\cos i_t \cos\Omega_t + r_{1z}\sin i_t, r_{1x}\cos\Omega_t + r_{1y}\sin\Omega_t) \quad \text{(A11)}$$

$$\omega_t = u_{t1} - \theta_{t1} \quad \text{(A12)}$$

where u_{t1} is the argument of latitude of point P_1 on the transfer orbit, and r_{1x}, r_{1y}, and r_{1z} are three-axis components of position vector $\mathbf{r}_1$ on the inertial frame.

Acknowledgments

The authors thank the associate editor, Robert Melton, and the reviewers for their helpful suggestions and revisions to improve this paper.

References

[1] Fehse, W., *Automated Rendezvous and Docking of Spacecraft*, Cambridge Univ. Press, London, 2003, pp. 8–28.

[2] Battin, R. H., *An Introduction to the Mathematics and Methods of Astrodynamics*, Revised ed., AIAA Education Series, AIAA, Reston, VA, 1999, pp. 277–279, 295–342.

[3] Lancaster, E. R., Blanchard, R. C., and Devaneyj, R. A., "A Note on Lambert's Theorem," *Journal of Spacecraft and Rockets*, Vol. 3, No. 9, Sept. 1966, pp. 1436–1438. doi:10.2514/3.28673

[4] Battin, R. H., and Fill, T. J., "Extension of Gauss Method for the Solution of Kepler's Equation," *Journal of Guidance and Control*, Vol. 2, No. 3, 1979, pp. 190–195. doi:10.2514/3.55860

[5] Battin, R. H., and Vaughan, R. M., "An Elegant Lambert Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 7, No. 6, 1984, pp. 662–670. doi:10.2514/3.19910

[6] Gooding, R. H., "A Procedure for the Solution of Lambert's Orbital Boundary-Value," *Celestial Mechanics and Dynamical Astronomy*, Vol. 48, No. 2, 1990, pp. 145–165. doi:10.1007/BF00049511

[7] Avanzini, G., "A Simple Lambert Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 6, 2008, pp. 1587–1594. doi:10.2514/1.36426

[8] Prussing, J. E., "A Class of Optimal Two-Impulse Rendezvous Using Multiple-Revolution Lambert Solutions," *Advances in the Astronautical Sciences*, Vol. 106, Univelt, San Diego, CA, 2000, pp. 17–39.

[9] Shen, H., and Tsiotras, P., "Optimal Two-Impulse Rendezvous Using Multiple-Revolution Lambert Solutions," *Journal of Guidance,*

Control, and Dynamics, Vol. 26, No. 1, 2003, pp. 50–61.
doi:10.2514/2.5014

[10] Luo, Y. Z., Tang, G. J., Lei, Y. J., and Li, H. Y., "Optimization of Multiple-Impulse Multiple-Revolution Rendezvous Phasing Maneuvers," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 4, 2007, pp. 946–952.
doi:10.2514/1.25620

[11] He, Q., Li, J., and Han, C., "Multiple-Revolution Solutions of the Transverse-Eccentricity-Based Lambert Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 1, 2010, pp. 265–268.
doi:10.2514/1.45041

[12] Zhang, G., Mortari, D., and Zhou, D., "Constrained Multiple-Revolution Lambert's Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 6, 2010, pp. 1779–1786.
doi:10.2514/1.49683

[13] Moore, R. E., *Interval Analysis*, Prentice–Hall, Upper Saddle River, NJ, 1966, pp. 8–25; Chap. 1.

[14] Hickey, T., Ju, Q., and van Emden, M., "Interval Arithmetic: From Principles to Implementation," *Journal of the Association for Computing Machinery*, Vol. 48, No. 5, 2001, pp. 1038–1068.
doi:10.1145/502102.502106

[15] Moore, R. E., Kearfott, R. B., and Cloud, M. J., *Introduction of Interval Analysis*, Society for Industrial and Applied Mathematics, Philadelphia, 2009, pp. 45–49, 55–57; Chap. 5.

[16] van Kampen, E., Chu, Q. P., Mulder, J. A., and van Emden, M. H., "Nonlinear Aircraft Trim Using Interval Analysis," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2007-6766, Aug. 2007.

[17] Juliana, S., Chu, Q. P., and Mulder, J. A., "Reentry Flight Clearance Using Interval Analysis," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, 2008, pp. 1295–1307.
doi:10.2514/1.34091

[18] de Weerdt, E., Chu, Q. P., and Mulder, J. A., "Global Fuel Optimization for Constrained Spacecraft Formation Rotations," *AIAA Guidance, Navigation, and Control Conference*, AIAA 2009-6287, Aug. 2009.

[19] van Kampen, E., Chu, Q. P., and Mulder, J. A., "Optimization of Spacecraft Rendezvous and Docking Using Interval Analysis," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2010-7596, Aug. 2010.

[20] van Kampen, E., Zaal, P. M. T., de Weerdt, E., Chu, Q. P., and Mulder, J. A., "Optimization of Human Perception Modeling Using Interval Analysis," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 1, 2010, pp. 42–52.
doi:10.2514/1.46318

[21] Prussing, J. E., "Geometrical Interpretation of the Angles α and β in Lambert's Problem," *Journal of Guidance, Control, and Dynamics*, Vol. 2, No. 5, 1979, pp. 442–443.
doi:10.2514/3.55905