Journal of Guidance, Control, and Dynamics
Vol. 30, No. 1, January–February 2007

Hybrid Approach to Optimize a Rendezvous Phasing Strategy

Ya-Zhong Luo,* Hai-yang Li,† and Guo-Jin Tang‡
National University of Defense Technology, 410073 Changsha, People's Republic of China

DOI: 10.2514/1.20232

The design of a rendezvous phasing strategy can be formulated as a mixed integer nonlinear programming problem. A new hybrid approach combining a genetic algorithm with Newton's method is proposed for solving this problem. An integer-coded genetic algorithm is used to handle the discrete design variables, whereas Newton's method is applied to handle the continuous design variables. Three improvements are imposed on the hybrid approach to make it more efficient and robust. The first improvement is not to impose the exact analysis on the explicitly constraint-violated design variables. The second is to use a memory database to record the previously completed analysis, and the third is to renew the initial guess to Newton's method by the nearest one in the memory database. A two-day rendezvous phasing problem is used as an example. Results show that our hybrid approach is effective, efficient, and can find multiple solutions in a single run.

I. Introduction

Rendezvous and docking (RVD) is a key operational technology that is required for many missions involving more than one spacecraft. The RVD process consists of a series of orbital maneuvers and controlled trajectories, which successively bring an active vehicle (chaser) into the vicinity of, and eventually in contact with, a passive vehicle (target). A rendezvous mission can be divided into a number of major phases: launch, phasing, far range rendezvous, close range, and mating. The objective of the phasing segment is to reduce the phase angle between the chaser and the target spacecraft, by making use of the fact that a lower orbit has a shorter orbital period [1]. Figure 1 depicts the definition of phase angle and the initial conditions for the rendezvous phasing problem considered in our study. During the rendezvous phasing segment, launch injection errors on inclination and right ascension of the ascending node will successively be corrected. Phasing ends with the acquisition of either an "initial aim point," or with the achievement of a set of margins for position and velocity values at a certain range, called the "trajectory gate" or "entry gate." Fehse [1] explains the phase angle and entry gate in more detail. The phasing stage can take anywhere from 1–3 days (Mir-Progress, Shuttle-ISS) to two weeks (Zvezda-ISS) [1,2].

The design of a phasing strategy is an important part of mission planning for most practical rendezvous missions. A well-designed phasing strategy can save fuel and improve safety. There are two widely used phasing strategies. One is the Soyuz/Progress phasing strategy [1–3], which is based on the impulsive maneuvers that combine in-plane and out-of-plane components. Baranov [3] proposed a semianalytical design method for this type of phasing strategy using a linearized dynamic model. In contrast to the Soyuz/Progress, the phasing strategy of the space shuttle is based on impulsive maneuvers at special points (such as orbit apogee) at which desired orbit changes can be achieved using minimum propellant [1]. Little published research, however, has addressed the topic of optimizing a rendezvous phasing strategy using maneuvers at special points [4]. We formulate the phasing strategy design problem using special-points maneuvers as an optimization problem with mixed integer design variables and continuous design variables, a nonlinear objective function, and a set of linear and nonlinear equality and inequality constraints. An optimization problem of this form is known as a mixed integer nonlinear programming (MINLP) problem.

MINLP problems are among the most difficult optimization problems because of their combinatorial nature and the potential existence of multiple local minima. The general methods for solving such problems are the branch and bound method, simulated annealing (SA) method and genetic algorithms (GA) [5–7]. These methods do not require any gradient or Hessian information. However, to reach an optimal solution with a high degree of confidence, the aforementioned methods typically require a large number of function evaluations during the optimization search. Constrained problems generally require more function evaluations.

Other MINLP problems have been solved using GAs. For example, some structural design problems (e.g., spacecraft design problems) can be formulated as MINLP problems. Recently, some modifications have been proposed to make GAs better at solving those problems. Those modifications include local memory using multivariate approximation [6,7], fine-tuning [8], and hill-climbing strategies [8]. Similar to structural design problems, trajectory design problems have functions that are expensive to evaluate. However, the methods used to solve structural design problems are not directly applicable to trajectory design problems because the problems have different forms. For example, trajectory optimization problems often have terminal equality constraints [9–14], which make it very difficult to locate a feasible solution no matter which algorithm we use. For this reason, although GAs have demonstrated better global convergence ability than the classical algorithms, the successful application of GAs to trajectory design problems are always owing to the combination of GAs with gradient-based methods [11–14]. It is shown that a GA can just find an approximation solution to known rendezvous problems [10]. Hughes and McInnes [11], Crain et al. [12], and Yokoyama and Suzuki [13] all use a GA to find an approximate initial solution for the gradient-based methods to solve different trajectory problems such as space plane reentry, interplanetary transfer, etc., and Hartmann et al. [14] use a calculus-of-variations-based trajectory optimization algorithm to aid a Pareto genetic algorithm to find the feasible solution. In these studies only the continuous, real-valued design variables were involved. We could have used a pure GA to solve our problem, but that would not have been very efficient because there are three

Received 24 September 2005; revision received 20 March 2006; accepted for publication 4 May 2006.

*Corresponding Author, Ph.D. Candidate, College of Aerospace and Material Engineering, Department of Aerospace Science and Technology; yzluo@sohu.com. Student Member AIAA.

†Associate Professor, College of Aerospace and Material Engineering, Department of Aerospace Science and Technology.

‡Professor, College of Aerospace and Material Engineering, Department of Aerospace Science and Technology.

equality constraints and GAs are not very efficient at solving equality constraints. Newton's method is very efficient at solving equality constraints (assuming the initial guess is good), but it only works with real-valued variables. Therefore, we decided to use a hybrid method. Our hybrid method uses a GA to search through the discrete part of the design space (involving the four integer-valued variables), but it uses Newton's method to solve the three equality constraints to determine the three real-valued variables. To improve the efficiency and robustness of the hybrid approach, three improvements are investigated. We illustrate our approach and our improvements by solving an example two-day phasing problem.

II. Rendezvous Phasing Problem

Depending on the phase angle to the target at the end of the launch phase, the given time constraints for the total flight up to docking and the necessary correction of the orbit parameters after launch, there is a multitude of possible phasing strategies. These strategies include the following choices of orbit and maneuver types [1]: 1) forward/backward phasing, 2) circular/elliptic phasing orbits, 3) change of orbit height in the case of circular orbits, 4) change of apogee/perigee height in the case of elliptical orbits, and 5) lateral correction maneuvers for inclination and right ascension of the ascending node (RAAN) corrections.

In general, the impulse-based maneuver is adopted for phasing. Assume an impulse $\Delta \boldsymbol{v} = (\Delta v_S, \Delta v_T, \Delta v_W)^T$ is imposed, where Δv_S, Δv_T, Δv_W are the components in the radial, tangential, and normal direction, respectively. The changes in the orbit parameters of the chaser caused by $\Delta \boldsymbol{v}$ are given by the following equations [15]:

$$\begin{cases} \Delta a = \frac{2}{n\sqrt{1-e^2}}[e \sin f \Delta v_S + (1 + e\cos f)\Delta v_T] \\ \Delta e = \frac{\sqrt{1-e^2}}{na}[\sin f \Delta v_S + (\cos f + \cos E)\Delta v_T] \\ \Delta i = \frac{r\cos u}{na^2\sqrt{1-e^2}}\Delta v_W \\ \Delta \Omega = \frac{r \sin u}{na^2\sqrt{1-e^2}\sin i}\Delta v_W \\ \Delta \omega = \frac{\sqrt{1-e^2}}{nae}\left[-\cos f \Delta v_S + \left(1+\frac{r}{p}\right)\sin f \Delta v_T\right] - \cos i \Delta\Omega \\ \Delta M = \frac{3n}{2a}\Delta a - \frac{1-e^2}{nae}\left[-\left(\cos f - 2e\frac{r}{p}\right)\Delta v_S + \left(1+\frac{r}{p}\right)\sin f \Delta v_T\right] \end{cases} \tag{1}$$

where a is the semimajor axis, i is the inclination, e is the eccentricity, Ω is the right ascension of the ascending node, ω is the argument of perigee, f is the true anomaly, and $u = f + \omega$ is the argument of latitude. Let $\boldsymbol{E} = (a, i, e, \Omega, \omega, f)$ represent the classical elements. Through Eqs. (1), we can describe the potential phasing maneuvers in detail as follows:

1) In-plane adjustments: The phase angle can be adjusted by changing the semimajor axis, i.e., adjusting the height of perigee and apogee. According to Eqs. (1), the simple and fuel-efficient maneuver to change the height of apogee/perigee is at perigee/apogee ($f = 0/180$ deg), respectively.

2) Lateral correction maneuver for inclination: When the impulse is imposed at the ascending node ($u = 0$ deg) or the descending node ($u = 180$ deg), the single impulse Δv_W is able to correct i without influencing Ω.

3) Lateral correction maneuver for RAAN: When the impulse is imposed on the highest argument of latitude ($u = 90$ deg) or the lowest argument of latitude node ($u = 270$ deg), the single impulse Δv_W is able to correct Ω with very small influence on i.

4) Combination correction maneuver for inclination and RAAN: When $u = \arctan[\sin i(\Delta\Omega/\Delta i)]$, corrections to Ω and i can be completed simultaneously by a single Δv_W. However, the location of this maneuver cannot be easily determined, so it is difficult for practical application.

Because these maneuvers are all executed on special points of the trajectory (such as apogee/perigee, ascending/descending node, highest/lowest argument of latitude) we call them "special-point based maneuvers."

Reaching the entry gate requires the chaser to enter a circular coplanar orbit with the target, while the chaser is below and behind the target. There are many different combinations of special-point based maneuver for phasing strategies, which are defined by the launch orbit, entry gate, time of flight, and many other RVD parameters. Discussions and comparisons on different phasing strategies are beyond the scope of this study. To describe the optimization problem and our hybrid approach more conveniently, we first give the general framework of the phasing strategy considered in this study. There are four maneuvers. The first maneuver is imposed at the apogee to increase the height of perigee by ΔH_p, the second is imposed at the ascending node to correct i by an amount Δi, the third is imposed at the highest argument of latitude to change Ω by an amount $\Delta\Omega$, and the last is imposed at the apogee to circularize the orbit. The phasing strategy design problem is to determine the optimal locations and the magnitudes of these four impulses.

III. Optimization Problem

A. Design Variables

The goal of phasing strategy design is to design the four impulse vectors $\Delta \boldsymbol{v}_1$, $\Delta \boldsymbol{v}_2$, $\Delta \boldsymbol{v}_3$, and $\Delta \boldsymbol{v}_4$. Because special-point based maneuvers are used, the location of the impulse is completely determined by the number of revolutions of the chaser when the impulse occurs. The directions of the impulses are always the same (relative to the orbit), and the magnitudes of the impulses are completely determined by ΔH_p, Δi, and $\Delta\Omega$. Therefore the design variables are

$$\boldsymbol{x} = (N_1, N_2, N_3, N_4, \Delta H_p, \Delta i, \Delta \Omega) \tag{2}$$

where $N_i (i = 1, 2, 3, 4)$ represents the number of revolutions of the chaser at the time of maneuver number i.

B. Objective Function

The total characteristic velocity increment is chosen as the optimization performance index.

Minimize:

$$J = \sum_{i=1}^{4} |\Delta \boldsymbol{v}_i| \tag{3}$$

C. Constraints

At the end of phasing, the target and the chaser are required to be coplanar, a constraint that can be met by satisfying the following two equations:

$$f_1(\boldsymbol{x}) = i_{\text{tar}}(t_f) - i_{\text{cha}}(t_f) = 0 \tag{4}$$

$$f_2(\boldsymbol{x}) = \Omega_{\text{tar}}(t_f) - \Omega_{\text{cha}}(t_f) = 0 \tag{5}$$

The phase angle required at the end of phasing is

$$f_3(\boldsymbol{x}) = u_{\text{tar}}(t_f) - u_{\text{cha}}(t_f) - p_f = 0 \tag{6}$$

where p_f is the required phasing angle. p_f is defined by the relative distance between the chase and the target, which is set in advance.

For a practical rendezvous mission, each maneuver must occur during a communication window. The communication window constraint requires that when an impulse is imposed on the chaser, the chaser should have a direct communication with at least one ground station. The elevation angle between the spacecraft and the ground station is used to decide whether the spacecraft can have direct communication with the ground station [1]. The communication window constraint can be met by satisfying inequalities of the form

$$\psi_i[\boldsymbol{r}_{\mathrm{cha}}(t_i), \boldsymbol{p}] < 0, \qquad i = 1, 2, \cdots, 4 \tag{7}$$

where $\boldsymbol{p}$ is a set of design parameters of the RVD system required to evaluate the communication window constraints, which involves the positions of the ground stations.

According to practical control requirements, the time between two maneuvers should be long enough for determination of the orbit. Herein, it is required that the number of revolutions between two maneuvers should not be less than 2.

$$N_i - N_{i-1} \geq 2, \qquad i = 1, 2, 3, 4 \tag{8}$$

where N_0 represents the number of revolutions when the phasing mission is started. N_i is counted from the entry of the chaser. We define $t = 0$ and the number of revolutions $=0$ at entry. $N_0 = 2$ is adopted in this paper.

D. Trajectory Simulation Model

The general dynamic equation for describing a spacecraft with all perturbations is Cowell's formulation [15]:

$$\begin{cases} \frac{\mathrm{d}\boldsymbol{v}}{\mathrm{d}t} = \frac{\mu}{r^3}\boldsymbol{r} + \boldsymbol{a}_{\mathrm{nonspherical}} + \boldsymbol{a}_{\mathrm{drag}} + \boldsymbol{a}_{3-\mathrm{body}} + \boldsymbol{a}_{\mathrm{SR}} + \boldsymbol{a}_{\mathrm{thrust}} + \boldsymbol{a}_{\mathrm{other}} \\ \frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}t} = \boldsymbol{v} \end{cases} \tag{9}$$

where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the position vector and velocity vector, respectively, $\boldsymbol{a}_{\mathrm{nonspherical}}$ is the perturbation acceleration caused by the nonspherical portion of the mass distribution of the central body, $\boldsymbol{a}_{\mathrm{drag}}$ is the atmospheric drag perturbation acceleration, $\boldsymbol{a}_{3-\mathrm{body}}$ is the third-body perturbation acceleration including the sun's and the moon's, $\boldsymbol{a}_{\mathrm{SR}}$ is the solar-radiation pressure perturbation acceleration, $\boldsymbol{a}_{\mathrm{thrust}}$ is the thrust acceleration, and $\boldsymbol{a}_{\mathrm{other}}$ is the perturbation acceleration caused by many other forces including tides.

When an impulsive maneuver is applied at time t,

$$\begin{cases} \boldsymbol{r}(t+\tau) - \boldsymbol{r}(t) = 0 \\ \boldsymbol{v}(t+\tau) - \boldsymbol{v}(t) = \Delta\boldsymbol{v} \end{cases} \tag{10}$$

where $\Delta\boldsymbol{v}$ is the impulse and $\tau \to 0$.

Through Eqs. (9) and (10), we can solve a multiple-impulse rendezvous problem with accurate solution by numerically integrating both the chaser and target orbits, which also accounts for disturbing forces. An eighth-order Runge–Kutta method [15] is used, and the integration step is 30 s. The perturbations considered in our study include J2, J3, J4, and atmospheric drag.

IV. Hybrid Approach

A. Hybrid Scheme

Equations (2–8) form a nonlinear programming model for rendezvous phasing strategy design. This problem involves a mix of discrete and continuous design variables with equality constraints. In this paper, we propose a new hybrid approach that combines a GA and Newton's method. We divide the design variables into two parts: the discrete variables $\boldsymbol{x}_{\mathrm{dis}} = (N_1, N_2, N_3, N_4)$ and the continuous variables $\boldsymbol{x}_{\mathrm{con}} = (\Delta H_p, \Delta i, \Delta\Omega)$. The discrete variables are optimized by the GA. During the search on the discrete variables, for each $\boldsymbol{x}_{\mathrm{dis}}^{\mathrm{GA}}$ found by the GA, a system of nonlinear equations formulated as Eqs. (4–6) is solved by Newton's method with $\boldsymbol{x}_{\mathrm{con}}$ as the variables during iteration. The solution $\boldsymbol{x}_{\mathrm{con}}^*$ is obtained from Newton's method. Then, through $\boldsymbol{x}_{\mathrm{dis}}^{\mathrm{GA}}$ and $\boldsymbol{x}_{\mathrm{con}}^*$ the analyses of Eqs. (2–8) are completed and the objective function and constraint values can be obtained. The solution $\boldsymbol{x}_{\mathrm{dis}}^{\mathrm{GA-best}}$ obtained by the GA and its corresponding continuous variables $\boldsymbol{x}_{\mathrm{con}}^*$ are the final solution.

Our proposed hybrid approach effectively includes the advantages of GAs to handle discrete variables with global search ability and the high convergence rate of Newton's method to handle continuous variables and nonlinear equations. The conventional method for GAs to handle constraints is to use a penalty function. However, this method is not always effective, especially when equality constraints are included. Consequently, most successful applications of GAs to trajectory optimization use a hybrid approach that combines GAs with a gradient-based method [11–13]. Through our experiments, we found that the GA encounters great difficulty in locating a feasible solution to satisfy the equality constraints. In the hybrid approach, the introduction of Newton's method to handle the equality constraints enables efficient convergence to a feasible solution. Furthermore, to further improve the efficiency and robustness of the hybrid approach, some improved revisions are described in the next section.

B. Genetic Algorithm

1. Fitness Function

In general, GAs are used to solve unconstrained problems. Because our nonlinear programming problem is constrained, it is important to add the capability of dealing with the constraints in our GA. We decide to handle constraints by making use of a penalty function. For the general constrained problem

$$\begin{array}{ll} \min\ f(\boldsymbol{x}) & \\ \text{subject to} & \begin{array}{ll} g_i(\boldsymbol{x}) \leq 0, & i = 1, 2, \cdots, q \\ h_j(\boldsymbol{x}) = 0, & j = 1, 2, \cdots, m \end{array} \end{array} \tag{11}$$

we define the new penalized objective function $F(\boldsymbol{x}, M)$ as

$$F(\boldsymbol{x}, M) = f(\boldsymbol{x}) + M\left[\sum_{i=1}^{q} \max[0, g_i(\boldsymbol{x})] + \sum_{j=1}^{m} |h_j(\boldsymbol{x})|\right] \tag{12}$$

In the present work $M = 10,000$ is used as a penalty coefficient. In our programming model, Eq. (3) is the objective function, Eqs. (4–6) are the equality constraints represented by $h_j(\boldsymbol{x})$, and Eqs. (7) and (8) are the inequality constraints represented by $g_i(\boldsymbol{x})$.

Because tournament [16] is used as the selection operator in the GA, the fitness function is directly represented by Eq. (12).

2. Operations and Parameters

For the optimization of discrete variables in our GA, an integer-coded-type (in the design variables) is more effective than binary-coded-type and floating-point-coded-type. The one-point crossover model, one-point mutation model, and tournament selection model with elitist approach are adopted [16].

C. Newton's Method

Newton's method is one of the most widely applied iterative methods for solving systems of equations [17]. Finite-difference approximations are used to estimate the derivatives.

V. Improved Algorithms

Firstly, we name the hybrid approach described in the preceding section (without other improvements) "Algorithm 0."

A. Algorithm 0

For each $\boldsymbol{x}_{\mathrm{dis}}^{\mathrm{GA}}$ produced by GA.

Complete one exact analysis:

make the $\boldsymbol{x}_{\mathrm{con}}^0$ which is set in advance as the initial guess to Newton's method, solve Eqs. (4–6).

During the search process, the inequality constraints given in Eq. (8) are explicit functions of the design variables, which can be determined before the exact analysis. If the discrete variables from the GA ($\boldsymbol{x}_{\mathrm{dis}}^{\mathrm{GA}}$), violate the constraints in Eq. (8), then no more time should be spent performing an exact analysis. Therefore, the first improvement to our algorithm is to avoid exact analysis for the designs that violate the explicit constraints [Eq. (8)].

When $\boldsymbol{x}_{\mathrm{dis}}^{\mathrm{GA}}$ violates the constraints in Eq. (8), the penalized objective function is calculated as follows.

$$F(\boldsymbol{x}, M) = M\sum_{i=1}^{4} \min(N_i - N_{i-1} - 2, 0) \tag{13}$$

This algorithm is named "Algorithm 1: infeasible return," and it can be summarized as follows.

B. Algorithm 1: Infeasible Return

For each x_{dis}^{GA} produced by GA.

Determine whether it satisfies the inequality constraints described as Eqs. (8).

If it satisfies, perform exact analysis:

make the x_{con}^{0} which is set in advance as the initial guess to Newton's method, solve Eqs. (4–6).

Else

Calculate the penalized objective function by Eq. (13)

When the mechanics of the GA operators are examined, we observe that the diversity of a population tends to decrease as the algorithm runs longer. The objective function and constraint values for the same chromosomes are recalculated repeatedly, especially towards the end of the optimization process [4,5]. In our hybrid approach one exact analysis needs one solution of the system of nonlinear equations, and one analysis of the system needs a whole flight-time integration, so it has a high computation time cost. In the standard GA, a new population may contain designs that have already been encountered in the previous generations, especially towards the end of the optimization process. If previously calculated objective function and constraint values can be efficiently saved and retrieved, computation time will decrease significantly. The memory procedure eliminates the possibility of repeating an analysis that could be expensive [6,7]. A memory database is adopted in our study to overcome repeated exact analyses. Algorithm 2 shows the pseudocode of the objective function evaluation with the aid of the memory database.

After the genetic operations create a new generation of designs, Algorithm 2 searches in the memory database to try to look for the same design points. If the same design is found in the database, the objective function value is retrieved from the memory database without conducting an analysis. Otherwise, the objective function and constraint values are obtained based on an exact analysis. The new design and its objective function and constraint values are then inserted in the memory database as a new item. Our memory database is to store not only the discrete variables and the objective function and constraint values, but also the continuous variables corresponding to the discrete variables. The latter will contribute to the convergence of Newton's method, which will be discussed later.

C. Algorithm 2: Infeasible Return + Memory with the Initial Guess Fixed

For each x_{dis}^{GA} produced by GA.

Determine whether it satisfies the inequality constraints described as Eqs. (8).

If satisfies

Evaluation of objective function using memory database

Search for the given design x_{dis}^{GA} in the memory database

if found then

Get the objective function value from the memory database;

Else

Perform exact analysis: make the x_{con}^{0} which is set in advance as the initial guess to Newton's method, solve Eqs. (4–6).

Else

Calculate the objective function by Eq. (13).

In our hybrid approach, one exact analysis requires one solution to a system of nonlinear equations Eqs. (4–6). As is well known, Newton's method is highly sensitive to the initial guess. Moreover, during the process of the hybrid approach, the systems of nonlinear equations may be different from each other because of different x_{dis}^{GA}. Therefore it is more difficult to choose a good initial guess for Newton's method. The improper choice of initial guess will induce convergence failure, which was seen in our experiment (in the next section). To overcome this problem, the initial guess to Newton's method is renewed by the nearest one to x_{dis}^{GA} in the memory database. The algorithm with initial guess renewed is the improved edition of Algorithm 2, so it is named "Algorithm 3: infeasible return and memory with initial guess renewed."

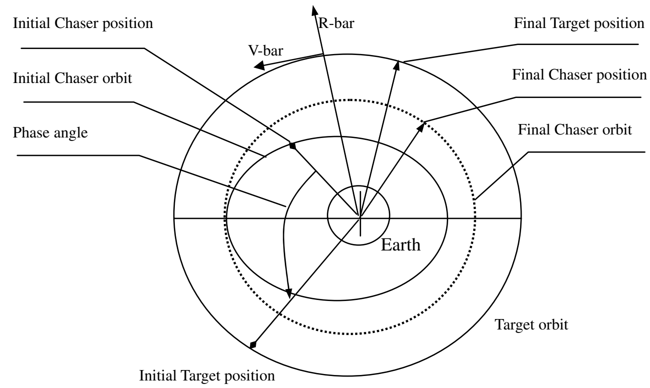

Fig. 1 Definition of phase angle; initial and final conditions for rendezvous phasing problem.

D. Algorithm 3: Infeasible Return and Memory with Initial Guess Renewed

For each x_{dis}^{GA} produced by GA.

Determine whether it satisfies the inequality constraints described as Eqs. (12).

If satisfies

Evaluation of objective function using memory database

Search for the given design x_{dis}^{GA} in the memory database.

if found then

Get the objective function value from the memory database

Else

Find the point nearest to x_{dis}^{GA} in the memory database, where the distance is defined as follows:

$$d = \|x_{dis}^{GA} - x_{dis}^{i}\|$$

and 2-norm is used. Assume $x_{int}^{i_0}$ is found, make $x_{con}^{i_0}$ which is according with $x_{dis}^{i_0}$ as the initial guess to Newton's method. And complete perform exact analysis, i.e., solve Eqs. (4–6).

Else

Calculate the objective function using Eq. (13)

VI. Results

In this section, we optimize a two-day phasing mission by our hybrid approach. The total number of revolutions is 27.

The initial conditions are

$$t_0 = 0$$

$$E_{tar} = (6720.140\ \text{km}, 42\ \text{deg}, 0, 169.286\ \text{deg}, 0, 245\ \text{deg})$$

$$E_{cha} = (6638.140\ \text{km}, 42.2\ \text{deg}, 0.009039, 169.686\ \text{deg}, 120\ \text{deg}, 1\ \text{deg})$$

Table 1 GA parameters used in experiments

Parameter	Value
Coded-type	Integer coded
Population size	50
Maximum number of generations	10
Selection type	Tournament (elitist)
Scale of tournament	3
Crossover type	One-point
Probability of crossover, p_c	0.92
Mutation type	One-point
Probability of mutation, p_m	0.10

Table 2 Search space and initial guess of design variables

Variable	Min	Max	Initial guess
N_1	3	11	——
N_2	9	17	——
N_3	14	22	——
N_4	18	26	——
ΔH_p, km	——	——	45.000
Δi, deg	——	——	−0.100
$\Delta\Omega$, deg	——	——	−0.080

Table 3 All solutions for this two-day phasing maneuver

Solution index	Design variables							Δv, m/s
	N_1	N_2	N_3	N_4	ΔH_p, km	Δi, deg	$\Delta\Omega$, deg	
1	6	14	17	26	53.657	−0.194	−0.139	70.869
2	6	14	17	22	39.277	−0.194	−0.136	70.894
3	6	14	16	26	53.657	−0.194	−0.137	70.793
4	6	14	16	22	39.310	−0.194	−0.135	70.757
5	10	14	17	26	67.100	−0.194	−0.134	70.794
6	10	14	17	22	52.400	−0.194	−0.136	70.845
7	10	14	16	22	52.398	−0.194	−0.135	70.742
8	10	14	16	26	67.121	−0.194	−0.138	70.755

The error of RAAN between the target and the chaser is 0.4 deg, the error of inclination is 0.2 deg, and the phase angle is 128 deg. The initial conditions are depicted in Fig. 1.

The terminal conditions are $t_f = 140,000$ s, the chaser is below the target by 25 km (R-bar direction) and behind the target by 75 km (V-bar direction). The terminal conditions are depicted in Fig. 1.

The GA parameters used in our experiments are provided in Table 1. Table 2 contains the search space and initial guess for all design variables. Ten dependent runs for each algorithm are executed. From our experiments, eight feasible solutions for this phasing strategy design problem have been observed, and they are listed in Table 3. Considering the computation error caused by integration and iteration, all solutions listed in Table 3 are optimal solutions in terms of their objective function values. Table 4 lists the details of the orbit maneuvers corresponding to the first solution listed in Table 3. The statistics of the results from different algorithms are listed in Table 5 for comparison in terms of time cost and convergence probability. Also the average number of generations for locating one feasible solution and the average number of solutions obtained in one run are provided in Table 5. The "probability of convergence" listed in Table 5 represents the success rate of the algorithm in locating a feasible solution. If a solution satisfies the inequality constraints as described by Eqs. (7) and (8) and the equality constraints as described by Eqs. (4–6) with an error of $1e-3$, it is regarded as a feasible solution.

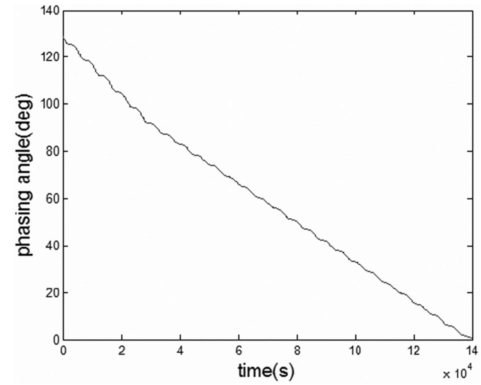

Fig. 2 Time histories of phasing angle.

Figures 2–5 show the trajectory characteristics according to the first solution in Table 3. Figure 2 shows the time history of the phase angle. Figures 3 and 4 show the relative distance and the relative velocity, respectively. Figure 5 shows how the chaser approaches the target in the V-bar direction and R-bar direction. (The coordinate system used herein is a target local orbital frame described as in Fig. 1, the center of the system is the center of mass of target, V-bar direction is named the orbital velocity vector $\boldsymbol{V}$, and the R-bar is named the radius vector $\boldsymbol{R}$.)

Through Table 5, by comparing the time cost of Algorithm 1 and Algorithm 0, it is calculated that "the infeasible return" improvement reduces the time cost by 20%. Comparing Algorithm 2 with Algorithm 1, we find that the time cost is further reduced by 59.5% with Guess 1 and 71.0% with Guess 2. The reduction in run time for Algorithm 2 is owing to its "memory database" improvement. Although the introduction of "initial guess renew" improvement for Algorithm 3 does not reduce the time cost very much compared with Algorithm 2 with "Guess 2," the convergence rate has been increased from 60% from 100%. These proposed improvements on the hybrid

Table 4 Orbit maneuvers corresponding with the first solution listed in Table 3

Maneuver index	Time, s	Number of revolutions	Velocity increment, m/s	Location of maneuver	Communication requirement satisfied?	Maneuver task
1	29,595.7	6	15.714	Apogee	Satisfy	Increase the height of perigee
2	73,691.0	14	26.128	Ascending node	Satisfy	Correction to inclination
3	85,857.5	17	12.561	The highest argument of latitude	Satisfy	Correction to RAAN
4	137,882.1	26	16.466	Perigee	Satisfy	Orbit circulization

Table 5 Comparison of different algorithms

Type of algorithm	Algorithm 0	Algorithm 1	Algorithm 2		Algorithm 3
Algorithm descriptions	Original	Infeasible return	Infeasible return Memory Initial guess fixed		Infeasible return Memory Initial guess renew
			Guess 1[b]	Guess 2[c]	
Time cost, min[a]	134.8	110.6	44.8	32.1	32.0
Probability of convergence	60%	60%	40%	60%	100%
Average number of generations for a feasible solution	4.7	4.7	4.7	4.7	4.5
Average number of solutions obtained in a single run	2	2	2	2.3	2.2

[a]All runs are ended with a maximum number of generation of GA and completed in a Dell Dimension 8300 computer CPU 2.8 GHz.
[b]$\Delta H_p = 35.000$ km, $\Delta i = -0.06$ deg, $\Delta\Omega = -0.03$ deg.
[c]Data provided in Table 2. Guess 2 is closer to the solution than Guess 1.

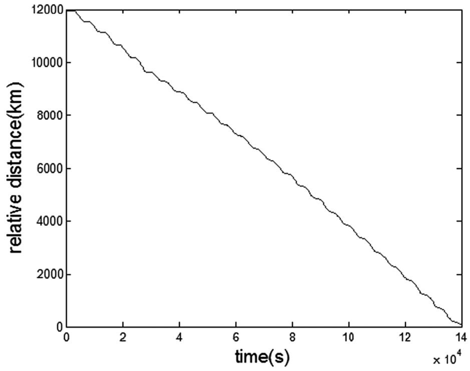

Fig. 3 Time histories of relative distance.

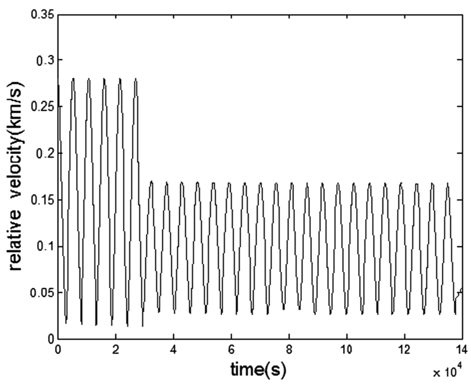

Fig. 4 Time histories of relative velocity.

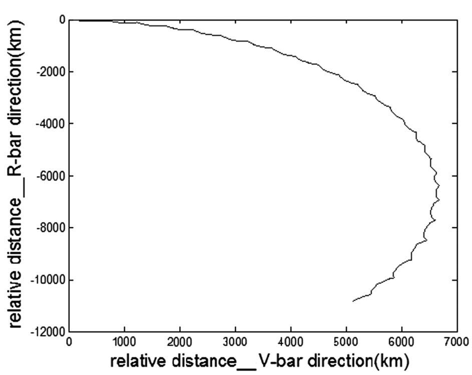

Fig. 5 The chaser approaches the target in V-bar direction and R-bar direction.

approach have been observed to increase the efficiency and convergence ability greatly. The average number of generations of the GA for obtaining a feasible solution is less than five, which shows that our hybrid approach has a very quick convergence rate, and its time cost is acceptable for practical rendezvous mission planning.

Furthermore, our hybrid approach can locate all solutions in several runs and a single run can locate at least two solutions. One distinguished characteristic of GAs is their population-based search mechanics, and this peculiarity makes GAs able to locate multiple solutions in a single run. Although the model of the GA used in our study is very simple, its ability to locate multiple solutions has been demonstrated. If the GA is revised elaborately through the adjustment of operators and parameters and introduction of other GA operators (such as the fitness share function, etc.), our hybrid approach can locate four and even more solutions in one run. Future work will concentrate on revising our hybrid approach to locate all eight solutions in one run.

In our hybrid approach, a GA is used to search the discrete variables. The easiest-to-implement method for discrete search is the exhaustive searching method, which is computationally expensive but will always find the global solution(s). The exhaustive searching method for the phasing strategy design problem is provided in the Appendix. There are 3081 combinations of Ns to check. From Table 5, the average number of combinations of Ns searched by the hybrid approach for a solution is about $4.5 \times 50/(110.6/32.1) = 65$. By using the memory database, the execution of the exhaustive search method is about 12 h, and the average time for Algorithm 3 to locate a solution is about 0.3 h. The hybrid approach has greatly improved the computation efficiency (by a factor of nearly 40). All of the global solutions for this phasing strategy design problem are located by executing the exhaustive search method; they are the same as those provided by Table 3. Thus, the global convergence ability of the hybrid approach is confirmed (for the example problem we considered).

In this paper, only a rendezvous scenario is considered and we find that 10 generations and a population size of 50 are enough for the simple GA to locate a solution for this test case. Other rendezvous scenarios have been examined by the hybrid approach to further test the effectiveness of the proposed approach. The hybrid approach is effective in solving other complex rendezvous scenarios; however, it does not guarantee a 100% success rate for some test cases. We find the convergence failure is mainly caused by the simple model of the GA we used, as the convergence problem is always existing for the simple GA. Through our experiments, if the maximum number of generations and the population size are increased and the operators and parameters of the GA are adjusted, the convergence failure can always be overcome.

VII. Conclusions

A new hybrid approach combining an integer-coded genetic algorithm with Newton's method is proposed to solve the rendezvous phasing strategy design problem with mixed discrete-continuous variables. The hybrid approach is applied to a two-day rendezvous phasing problem. By comparing with the exhaustive searching method, the hybrid approach is confirmed with the global convergence ability and to reduce the time cost by a factor of approximately 40. Three improvements on the hybrid approach including "infeasible return" strategy, memory for the discrete design variables, and renewing the initial guess for Newton's method are found to effectively improve the computation efficiency and make the proposed hybrid approach more robust. One distinguished advantage of the hybrid approach is that it can locate more than one solution in a single run. The hybrid approach provides a quick and efficient tool for the mission plan of rendezvous and docking.

Appendix: Pseudocode for the Exhaustive Searching Method

for $N1 = \min N1$ to $\max N1$
for $N2 = \max(N1 + 2, \min N2)$ to $\max N2$
for $N3 = \max(N2 + 2, \min N3)$ to $\max N3$
for $N4 = \max(N3 + 2, \min N4)$ to $\max N4$
Use Newton's method to determine ΔHp, Δi, and $\Delta\Omega$
Calculate all constraint and objective functions
If the solution is feasible and the objective function is the best seen store $N1$, $N2$, $N3$, and $N4$
end
end
end
end

The values of $\max N1$, $\min N1$, $\max N2$, etc., are the same as provided in Table 2.

Acknowledgments

The first author would like to thank Troy McConaghy and Ph.D. candidate Chit Hong Yam of Purdue University for their very insightful suggestions and improving the English of this paper.

References

[1] Fehse, W., *Automated Rendezvous and Docking of Spacecraft*, Cambridge Univ. Press, London, 2003, pp. 12–13, 441–449.

[2] Labourdette, P., and Baranov, A. A., "Strategies for On-Orbit Rendezvous Circling Mars," *Advances in the Astronautical Sciences*, Vol. 109, Pt. 2, 2002, pp. 1351–1368.

[3] Baranov, A. A., "An Algorithm for Calculating Parameters of Multi-Orbit Maneuvers in Remote Guidance," *Cosmic Research (Translation of Kosmicheskie Issledovaniya)*, Vol. 28, No. 1, 1990, pp. 61–67.

[4] Hooper, H. L., and Herrnstein, J. R., "Active Rendezvous Between a Low-Earth Orbit User Spacecraft and the Space Transportation System (STS) Shuttle," *Flight Mechanics Estimation Theory Symposium*, NASA Conference Publication 2050, 1989, pp. 381–401.

[5] Huang, M. W., and Arora, J. S., "Engineering Optimization with Discrete Variables," *Journal of Engineering and Applied Sciences*, Vol. 42, No. 3, 1995, pp. 1475–1485.

[6] Gantovnik, V. B., Anderson-Cook, C. M., Grdal, Z., and Watson, L. T., "A Genetic Algorithm with Memory For Mixed Discrete-Continuous Design Optimization," AIAA Paper 2002-1221, April 2002.

[7] Gantovnik, V. B., Gurdal, Z., Watson, L. T., and Anderson-Cook, C. M., "Genetic Algorithm for Mixed Integer Nonlinear Programming Problems Using Separate Constraint Approximations," *AIAA Journal*, Vol. 43, No. 8, 2005, pp. 1844–1849.

[8] Ndiritu, J. G., and Daniell, T. M., "Improved Genetic Algorithm for Continuous and Mixed Discrete-Continuous Optimization," *Engineering Optimization*, Vol. 31, No. 5, 1999, pp. 589–614.

[9] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[10] Kim, Y. H., and Spencer, D. B., "Optimal Spacecraft Rendezvous Using Genetic Algorithms," *Journal of Spacecraft and Rockets*, Vol. 39, No. 6, 2002, pp. 859–865.

[11] Hughes, G. W., and McInnes, C. R., "Solar Sail Hybrid Trajectory Optimization for Non-Keplerian Orbit Transfers," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 3, 2002, pp. 602–604.

[12] Crain, T., Bishop, R. H., and Fowler, W., "Interplanetary Flyby Mission Optimization Using a Hybrid Global-Local Search Method," *Journal of Spacecraft and Rockets*, Vol. 37, No. 4, 2000, pp. 468–474.

[13] Yokoyama, N., and Suzuki, S., "Modified Genetic Algorithm for Constrained Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 1, 2005, pp. 139–144.

[14] Hartmann, J. W., Coverstone-Carroll, V. L., and Williams, S. N., "Optimal Interplanetary Spacecraft Trajectories via a Pareto Genetic Algorithm," *Journal of the Astronautical Sciences*, Vol. 46, No. 3, 1998, pp. 267–282.

[15] Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, 2nd ed., Microscosm Press, El Segundo, CA, 2001, pp. 497–499, 592.

[16] Goldberg, D. E., Korb, B., and Deb, K., "Messy Genetic Algorithms: Motivation, Analysis, and First Results," *Complex Systems*, Vol. 3, No. 5, 1989, pp. 493–530.

[17] Ortega, J., and Rheinboldt, W., *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970, pp. 185–195.