JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 28, No. 1, January–February 2005

Modified Genetic Algorithm for Constrained Trajectory Optimization

Nobuhiro Yokoyama* and Shinji Suzuki†
University of Tokyo, Tokyo 113-8656, Japan

Although direct methods using sequential quadratic programming, such as direct collocation, have been widely applied to trajectory optimization problems, optimization results for these methods are sensitive to initial solutions in some cases. For the purpose of finding an appropriate initial solution to gradient-based direct trajectory optimization, a numerical trajectory optimization method using a real-coded genetic algorithm (GA) is considered. Because a typical GA solves unconstrained optimization problems, the penalty function has usually been used as the selection criterion of surviving individuals. However, the convergence properties of this conventional method are sensitive to the penalty parameter. Therefore, a new method that robustly achieves global optimization of the objective function and the feasibility search even with a large penalty parameter is proposed. The proposed method is applied to the constrained brachistochrone problem and a space plane reentry problem. Results indicate promising performance of the proposed method in providing appropriate initial solutions for gradient-based trajectory optimization.

Introduction

TRAJECTORY optimization plays an important role in the field of aerospace, because aircraft and spacecraft are often required to achieve maximum performance in severe environments. Thus, many numerical methods have been proposed and applied to trajectory optimization problems in the field of aerospace. Most of these methods utilize the gradient information of objective/constraint functions, and they can be categorized into two types, indirect or direct.[1] In the indirect methods, the two-point or multipoint boundary value problem of the state variables and the adjoint variables[2] is solved by gradient algorithms, such as the sequential conjugate gradient-restoration algorithm,[3] the modified quasilinearization algorithm,[4] and indirect multiple shooting.[5] In the direct methods, the trajectory optimization problem is converted into a nonlinear programming (NLP) problem and is solved by NLP techniques, such as sequential quadratic programming (SQP). Direct multiple shooting[6,7] and direct collocation (DC)[8–10] are well known as the direct methods. Compared to the indirect methods, the direct methods have advantages in terms of robustness of convergence and flexible applicability to practical complex problems. However, optimization results for the direct methods are still sensitive to initial solutions in some cases; i.e., a poor initial solution may converge on an infeasible solution or a local optimum in highly nonlinear or multimodal problems.

On the other hand, genetic algorithms (GAs), which are inspired by the evolution of biological systems, have been applied as a preliminary search method to find initial solutions to gradient-based optimization methods.[11–13] Although it is recognized that GAs are generally not computationally competitive against gradient-based methods,[1] they are useful in finding appropriate initial solutions for gradient-based methods due to their global search capability. This paper considers a numerical trajectory optimization method using a GA to find an appropriate initial solution for the gradient-based methods. We adopt a real-coded GA,[14] because it generally achieves efficient search by covering the continuity and the interdependency of optimized variables.

Because a typical GA solves unconstrained optimization problems, the penalty function has usually been used as the selection criterion for surviving individuals. However, the convergence of this conventional method is sensitive to the penalty parameter. If the penalty parameter is substantially larger than its optimum value, which cannot be estimated a priori, the individuals tend to prematurely converge on a local minimum. On the other hand, if the penalty parameter is smaller than its optimum value, the individuals probably cannot converge on a feasible region. Thus, some remedies have been applied in the selection process to eliminate the difficulty of the penalty function method for GAs. There are some methods that evaluate the Euclidean distance between individuals as part of the selection criteria. Typical examples are niche formation[15] and distance dependent alternation (DDA).[16] These methods avoid premature convergence by preserving diversity of population as much as possible. Nevertheless, their performance is dependent on the problem and still sensitive to the value of the penalty parameter. In Ref. 17, the penalty parameter is adaptively adjusted based on the population's mean and the best value of the penalty function, and successful results of the constrained optimizations are demonstrated. However, this method requires the introduction of additional parameters to update the penalty parameter. As with the penalty parameter, appropriate values of these additional parameters will depend on the problem. Moreover, this method excludes the objective function in the selection criterion by the time the population has at least one feasible individual. This exclusion may encourage the individuals to move toward a local minimum.

On the other hand, our paper proposes a new selection method using a penalty function with a fixed penalty parameter. In the proposed method, the selection of surviving individuals is carried out based on multiple criteria, that is, ranking by objective function, ranking by penalty function, and distances between individuals. In ranking by the objective function as well as the penalty function, the objective function is considered in the selection in all generations, while the individuals are encouraged to move toward a feasible region. These ranking strategies are similar to the Pareto ranking and the cost ranking adopted in COMOGA (constrained optimization by multiobjective genetic algorithms).[18] However, the proposed method aims to avoid premature convergence by evaluating the distances between individuals as one of the selection criteria. This strategy is not supported in COMOGA. Furthermore, unlike

Received 30 June 2003; presented as Paper 2003-5493 at the AIAA Guidance, Navigation, and Control Conference, Austin, TX, 11–14 August 2003; revision received 14 December 2003; accepted for publication 22 December 2003.

*Graduate Student (Research Fellow of the Japan Society for Promotion of Science), Department of Aeronautics and Astronautics, 7-3-1 Hongo, Bunkyo-ku. Student Member AIAA.

†Professor, Department of Aeronautics and Astronautics, 7-3-1 Hongo, Bunkyo-ku. Member AIAA.

the adaptive penalty method and COMOGA, the proposed method is free from the additional parameters that depend on problems to be solved.

Through application to the constrained brachistochrone problem, robust convergence of the proposed method is demonstrated. Furthermore, the proposed method is applied to a space plane reentry problem, and the effectiveness of the proposed method for a practical problem is confirmed.

Conversion to NLP Problem

Let $\boldsymbol{x}(t)$ be the state variables, let $\boldsymbol{u}(t)$ be the control variables, and let $\boldsymbol{p}$ be the static unknown parameters. The trajectory optimization problem covered in this study is defined as follows. Minimize the objective function,

$$J = \phi(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f), \boldsymbol{p}) \tag{1}$$

under the following constraints.

1) State equations:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}) \tag{2}$$

2) Initial conditions at time $t = t_0$:

$$\boldsymbol{\chi}_{0E}(\boldsymbol{x}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}) = \boldsymbol{0}, \qquad \boldsymbol{\chi}_{0I}(\boldsymbol{x}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}) \leq \boldsymbol{0} \tag{3}$$

3) Path constraints during the time interval $t \in [t_0, t_f]$:

$$\boldsymbol{S}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}) \leq \boldsymbol{0} \tag{4}$$

4) Terminal conditions at time $t = t_f$:

$$\boldsymbol{\chi}_{fE}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f), \boldsymbol{p}) = \boldsymbol{0}, \qquad \boldsymbol{\chi}_{fI}(\boldsymbol{x}(t_f), \boldsymbol{u}(t_f), \boldsymbol{p}) \leq \boldsymbol{0} \tag{5}$$

Note that initial state variables $\boldsymbol{x}_0$, initial time t_0, and terminal time t_f are incorporated into $\boldsymbol{p}$, if they are unknown.

The trajectory optimization problem is converted to an NLP problem in the following way. The time interval $[t_0, t_f]$ is divided into M intervals $[t_0, t_1], \ldots, [t_{M-1}, t_M]$ $(t_M = t_f)$ and the boundary time t_i of each interval is defined as follows:

$$t_i = t_{i+1} \cdot k_i^{1/i} (0 < k_i < 1, i = M-1, \ldots, 1) \tag{6}$$

where k_i $(i = 1, \ldots, M-1)$ denote the parameters that determine the length of the time intervals. It should be noted that if k_i $(i = 1, \ldots, M-1)$ are the uniform random numbers in the range $(0, 1)$, t_i $(i = 1, \ldots, M-1)$ are distributed uniformly in the range (t_0, t_M) with the proper order $t_0 < t_1 < \cdots < t_{M-1} < t_M$. The control variables $\boldsymbol{u}_i$ at each bound t_i $(i = 0, \ldots, M)$ and the time interval parameters k_i are regarded as the NLP variables. Then $\boldsymbol{u}(t)$ at arbitrary times are determined by the natural cubic spline interpolation of $\boldsymbol{u}_i$. In addition, $\boldsymbol{x}(t)$ at arbitrary times are obtained by numerical integration (fourth order Runge–Kutta method) of the state equations (2). Furthermore, path constraints (4) are evaluated at every step of the numerical integration t_j $(j = 0, \ldots, N, t_N = t_f)$.

Thus, the converted NLP problem is expressed as follows: for variables,

$$\boldsymbol{X} = \left[\boldsymbol{u}_0^T, \boldsymbol{u}_1^T, \ldots, \boldsymbol{u}_M^T, k_1, \ldots, k_{M-1}, \boldsymbol{p}^T\right]^T (\in \mathbb{R}^n) \tag{7}$$

minimize

$$F(\boldsymbol{X}) = \phi(\boldsymbol{x}(t_N), \boldsymbol{u}(t_N), \boldsymbol{p}) (\in \mathbb{R}) \tag{8}$$

subject to

$$\boldsymbol{G}(\boldsymbol{X}) = \begin{bmatrix} \boldsymbol{\chi}_{0E}(\boldsymbol{x}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}) \\ \boldsymbol{\chi}_{fE}(\boldsymbol{x}(t_N), \boldsymbol{u}(t_N), \boldsymbol{p}) \end{bmatrix} = \boldsymbol{0} (\in \mathbb{R}^{m_E}) \tag{9}$$

$$\boldsymbol{H}(\boldsymbol{X}) = \begin{bmatrix} \boldsymbol{\chi}_{01}(\boldsymbol{x}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}) \\ \boldsymbol{S}_0(\boldsymbol{x}(t_0), \boldsymbol{u}(t_0), \boldsymbol{p}) \\ \vdots \\ \boldsymbol{S}_N(\boldsymbol{x}(t_N), \boldsymbol{u}(t_N), \boldsymbol{p}) \\ \boldsymbol{\chi}_{f1}(\boldsymbol{x}(t_N), \boldsymbol{u}(t_N), \boldsymbol{p}) \end{bmatrix} \leq \boldsymbol{0} (\in \mathbb{R}^{m_I}) \tag{10}$$

This NLP formulation has the merit of a relatively small number of variables. Moreover, the addition of the time interval parameters k_i into the NLP variables makes it possible for the spline function to behave smoothly. In this study, a GA was applied to the NLP problem (7–10) and the GA solution was applied as the initial solution to a DC formulation of the problem.

Genetic Algorithm

The real-coded GA, in which the NLP variables vector $\boldsymbol{X}$ is regarded as the gene of each individual, is used in this study. The algorithm of this approach is outlined as follows.

Step 1: Initialization

The initial N_P population of the NLP variables is prepared at random. Each NLP variable is determined by the uniform random numbers. The infeasible solutions in which the numerical integration fails are not included in the initial N_P population.

Step 2: Crossover

Multiparental unimodal normal distribution crossover[14] (UNDX-m) is used as the crossover model. This model achieves an efficient global search. The algorithm of UNDX-m is as follows: $(m+2)$ parents $\boldsymbol{X}_P^{(1)}, \ldots, \boldsymbol{X}_P^{(m+2)}$ are selected at random. The median point of the first $(m+1)$ parents is defined as X_G; i.e.,

$$\boldsymbol{X}_G = \frac{1}{m+1} \sum_{i=1}^{m+1} \boldsymbol{X}_P^{(j)} \tag{11}$$

The difference vectors of each parent are defined as

$$\hat{\boldsymbol{d}}^{(j)} = \boldsymbol{X}_P^{(j)} - \boldsymbol{X}_G, \qquad (j = 1, \ldots, m+2) \tag{12}$$

Let $\hat{D}$ be the length of the component of $\hat{\boldsymbol{d}}^{(m+2)}$ orthogonal to $\hat{\boldsymbol{d}}^{(1)}, \ldots, \hat{\boldsymbol{d}}^{(m)}$. Moreover, let $\hat{\boldsymbol{e}}^{(1)}, \ldots, \hat{\boldsymbol{e}}^{(n-m)}$ be the orthonormal basis of the subspace orthogonal to $\hat{\boldsymbol{d}}^{(1)}, \ldots, \hat{\boldsymbol{d}}^{(m)}$. Generate children $\boldsymbol{X}_C^{(i)}$ $(i = 1, \ldots, N_C)$ by the equation

$$\boldsymbol{X}_C^{(i)} = \boldsymbol{X}_G + \sum_{j=1}^{m} w_j \hat{\boldsymbol{d}}^{(j)} + \hat{D} \sum_{j=1}^{n-m} v_j \hat{\boldsymbol{e}}^{(j)} \tag{13}$$

where w_j, v_j are random numbers that conform to the normal distribution with 0 mean and variance of σ_w^2, σ_v^2 respectively. σ_w, σ_v are specified by the value recommended in Ref. 14:

$$\sigma_w = \frac{1}{\sqrt{m}}, \qquad \sigma_v = \sqrt{\frac{3(m+1)}{2(n-m)(m+2)}} \tag{14}$$

Step 3: Selection

Select the surviving individuals from the $(m+2)$ parents and the N_C children based on some criteria. The details of the new selection method are proposed in the next section.

Step 4: Termination Check

If the generation number (step 2–step 4 correspond to one generation) equals the specified number N_G, terminate the algorithm. Otherwise, return to step 2. (End)

No mutation scheme is performed in the above GA, because UNDX-*m* crossover also plays the role of mutation, i.e., random perturbation of the created NLP vector.

Selection Method

In step 3 of the GA, a fitness function is usually used as a criterion of selection. In order to handle the constraints, a penalty function has been used frequently as part of the fitness function, of the form

$$F_r(\boldsymbol{X}) = F(\boldsymbol{X}) + rE(\boldsymbol{X}) \tag{15}$$

$$E(\boldsymbol{X}) = \sum_{i=1}^{m_E} |G_i(\boldsymbol{X})| + \sum_{i=1}^{m_I} \max[0, H_i(\boldsymbol{X})] \tag{16}$$

where r and $E(\boldsymbol{X})$ denote the penalty parameter and the constraint error, respectively. Because the optimal value of r cannot be estimated a priori, r is usually specified as an arbitrarily large value so that the global minimum of the penalty function (15) becomes that of the original NLP problem (7–10). However, the individuals tend to converge on a local minimum in the case of large r, because the contribution of the objective function to the penalty function with large r is small, especially in the early stages of evolution (i.e., when the constraint error of each individual is large).

Therefore, this study proposes a new selection method in which the objective function is considered in the selection at all stages of evolution. The algorithm is as follows.

Step 3.1

Rank the generated N_C children in ascending order on the penalty function. Set the rank parameter $i = 1$.

Step 3.2

Carry out the selection with respect to the ith child $\boldsymbol{X}_C^{(i)}$: if there are some parents $\boldsymbol{X}_P^{(j)}$ that satisfy both

$$F\left(\boldsymbol{X}_C^{(i)}\right) \le F\left(\boldsymbol{X}_P^{(j)}\right) \tag{17}$$

$$F_r\left(\boldsymbol{X}_C^{(i)}\right) \le F_r\left(\boldsymbol{X}_P^{(j)}\right) \tag{18}$$

replace the nearest parent $\boldsymbol{X}_P^{(n)}$ with $\boldsymbol{X}_C^{(i)}$ [i.e., $\|\boldsymbol{X}_P^{(n)} - \boldsymbol{X}_C^{(i)}\|_2$ is minimum among the parents that satisfy both requirements (17) and (18)]. If the replacement was performed, or $i = N_C$, or there are no parents that satisfy (18), go to step 3.3. Otherwise, set $i \to i + 1$ and repeat step 3.2.

Step 3.3

Newly rank the children in ascending order on the objective function and set the rank parameter $k = 1$.

Step 3.4

Carry out the selection with respect to the kth child $\boldsymbol{X}_C^{(k)}$: if there are some parents $\boldsymbol{X}_P^{(j)}$ that satisfy

$$F_r\left(\boldsymbol{X}_C^{(k)}\right) < F_r\left(\boldsymbol{X}_P^{(j)}\right) \tag{19}$$

replace the nearest parent $\boldsymbol{X}_P^{(n)}$ with $\boldsymbol{X}_C^{(k)}$. If the replacement was performed, or $k = N_C$, go to step 4. Otherwise, set $k \to k + 1$ and repeat step 3.4.

In this method, decrease of the penalty function is required in the form of inequalities (18) and (19) for every replacement. Therefore, in the case where the penalty parameter is sufficiently large, convergence of the individuals on the feasible region is guaranteed. In steps 3.1 and 3.2, decrease of the objective function is also imposed for replacement, and it causes the objective function to be strongly reflected in the selection even at the early stage of evolution. However, evolution with only steps 3.1 and 3.2 can stagnate in the case where there are infeasible solutions that give an objective function smaller than the global optimum in the feasible region, because increase of the objective function is not allowed in steps 3.1 and 3.2. To remedy this drawback, increase of the objective function is allowed in steps 3.3 and 3.4 if necessary, and the increase is kept as small as possible by ranking the children on the objective function. Furthermore, diversity of the population is kept as great as possible by prioritizing the nearer parent as the target for replacement.

Application to Simple Test Problem

The GA proposed in the preceding section was applied to the constrained brachistochrone problem[2,10] (Fig. 1). This simple and well-known problem is defined as follows:

1) The state equations (path angle γ is the control variable) are

$$\dot{x} = v \cos\gamma \tag{20}$$

$$\dot{y} = v \sin\gamma \tag{21}$$

$$\dot{v} = g \sin\gamma \tag{22}$$

2) The initial conditions at fixed time $t = t_0 (= 0)$ are

$$x(t_0) = y(t_0) = v(t_0) = 0 \tag{23}$$

3) The path constraints are

$$y \le x/2 + h \tag{24}$$

4) The terminal conditions at unknown time $t = t_f (\ge 0)$ are

$$x(t_f) = l \tag{25}$$

5) The objective function to be minimized is

$$J = t_f \sqrt{g/(\pi l)} \tag{26}$$

where the parameters were specified as

$$l = 1, \qquad h = 0.1, \qquad g = 1 \tag{27}$$

The number of divided time intervals for the control variable $\gamma(t)$ was $M = 10$, and the number of steps in the numerical integration was $N = 100$. The GA parameters and the range of the initial GA population were specified by the following values:

$$N_P = 100, \qquad N_C = 50, \qquad N_G = 2 \times 10^4, \qquad m = 4 \tag{28}$$

$$0 \le \gamma \le 90\ [\text{deg}], \qquad 0 \le t_f \le 2\sqrt{\pi l/g} \tag{29}$$

The number of individuals N_P and the crossover number N_C were set to be several times larger than the number of NLP variables, considering the compromise between the global search performance and the rate of convergence. By monitoring the results of a trial run, the maximum generation number N_G was determined as the value for which further improvement of the average of the penalty function could be regarded as negligible.

The convergence properties of the proposed GA were compared with those of a conventional real-coded GA, which adopts $-F_r(\boldsymbol{X})$ as the fitness function and uses DDA[16] as the selection method in step 3. Both GAs were applied to the constrained brachistochrone

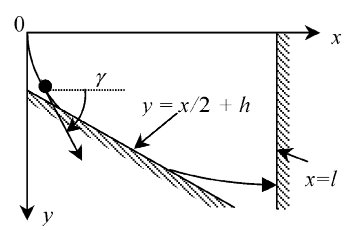


Fig. 1 Constrained brachistochrone problem.

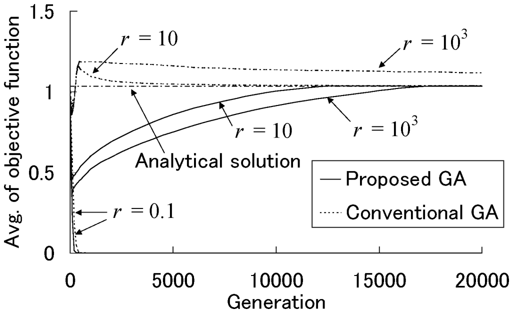


Fig. 2 Population average of objective function.

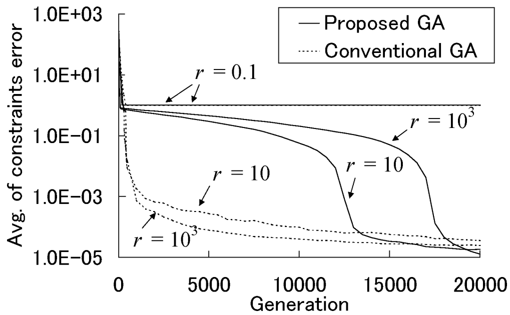


Fig. 3 Population average of constraints error.

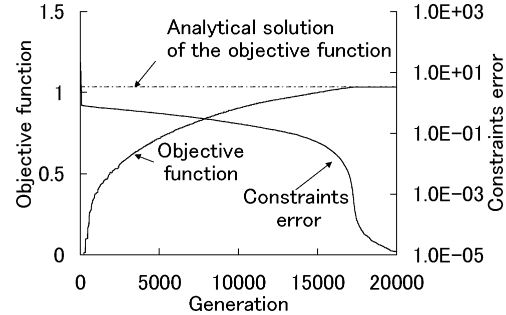
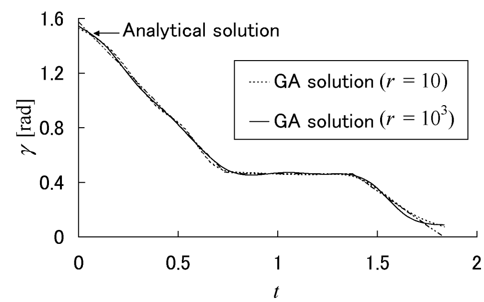
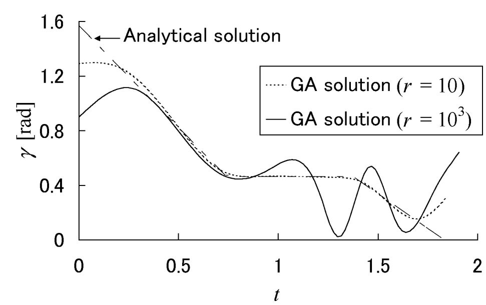


Fig. 4 Objective function and constraints error of best individual (proposed GA, $r = 10^3$).



a) Proposed GA



b) Conventional GA

Fig. 5 Control variable $\gamma(t)$ of the GA solution.

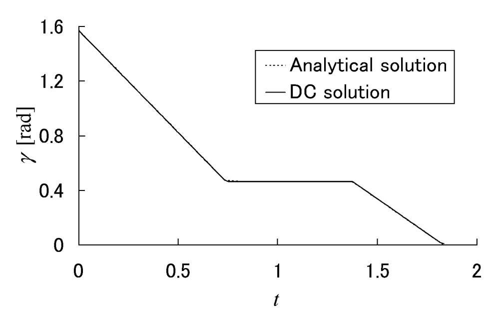


Fig. 6 Control variable $\gamma(t)$ of the DC solution.

problem under three conditions, $r = 10^3$ (too large), $r = 10$ (nearly optimal), and $r = 0.1$ (too small).

Figures 2 and 3 show the population average of the objective function and that of the constraints error for each generation. As can be seen, the conventional GA with both $r = 10$ and $r = 10^3$ showed rapid increase of the objective function in conjunction with rapid decrease of the constraints error. Although the objective function with $r = 10$ approached the analytical minimum asymptotically thereafter, the objective function with $r = 10^3$ prematurely converged on the local minimum. In addition, the individuals with $r = 0.1$ converged on an infeasible solution $[t_f, x(t_f)] = (0, 0)$. Although the conventional GA with DDA selection is regarded as robust to highly nonlinear problems,[16] these results indicate that it is sensitive to the choice of the penalty parameter.

On the other hand, in the proposed GA, with both $r = 10$ and $r = 10^3$, the objective function increased gradually after the rapid decrease and approached the analytical minimum asymptotically. Although the constraints error showed slow decrease until the objective function reached the analytical minimum, it decreased significantly thereafter and eventually became smaller than that of the conventional GA. These results indicate that the proposed GA robustly achieved the global search of the objective function and the feasibility search even with a large penalty parameter. It can also be seen that the proposed GA with $r = 0.1$ showed convergence on the infeasible solution $[t_f, x(t_f)] = (0, 0)$. Thus, even in the proposed GA, the penalty parameter must be larger than optimum.

Figure 4 shows the objective function and the constraints error of the best individual (i.e., the individual whose objective function is the best among the population) in the case $r = 10^3$. As can seen, both the objective function and the constraints error of the best individual showed almost the same shift tendency as those of the population average. Thus, a large number of generations are required even to make the best individual converge on a reasonable solution. This computational expense is a drawback of the proposed GA, but it results from the prevention of the hasty reduction of the constraints error to successfully attain the global optimum of the objective function.

The path angle of the best individual in the final generation under each condition is shown in Fig. 5. The solutions with $r = 0.1$ are not shown in Fig. 5, because they converged on $t_f = 0$. As can be seen, whereas the solution of the conventional GA with $r = 10^3$ obviously converged on the local solution, the other three solutions (the solution of the conventional GA with $r = 10$ and the solutions of the proposed GA with $r = 10$ and $r = 10^3$) are close to the analytical solution. In addition, the solutions of the proposed GA are closer to the analytical solution than that of the conventional GA with $r = 10$.

In the next phase, giving the best individual of the proposed GA with $r = 10^3$ as the initial solution, DC with SQP[9] was carried out. In DC, the number of nodes was specified as 100 and trapezoidal integration was employed. The path angle and the trajectory of the DC solution are shown in Figs. 6 and 7, respectively. It can be seen that the DC solution is substantially close to the analytical solution, and we can confirm that finding an appropriate initial solution by the proposed GA worked.

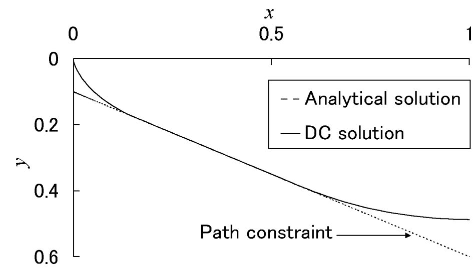


Fig. 7 Trajectory of the DC solution.

Optimization of Space Plane's Reentry Trajectory

Let us consider a space plane's reentry trajectory problem. The goal of the optimization is to choose an angle of attack and a bank angle such that the final cross range is maximized. This problem is highly nonlinear and it is difficult to find an appropriate initial solution without enough knowledge of the physics.

The simulated space plane was assumed to be a horizontally taking off, horizontally landing single stage to orbit (SSTO), and its aerodynamic coefficients C_L, C_D were based on the NAL zeroth configuration.[19] Approximating the dynamics of the space plane as a point mass model, the problem was defined as follows.

1) State equations:

$$\dot{V} = -\frac{D}{m} - \frac{\mu}{R^2}\sin\gamma \tag{30}$$

$$\dot{h} = V\sin\gamma \tag{31}$$

$$\dot{\gamma} = \left(\frac{V}{R} - \frac{\mu}{R^2 V}\right)\cos\gamma + \frac{L\cos\sigma}{mV} \tag{32}$$

$$\dot{\psi} = -\frac{V\cos\gamma\sin\psi\tan\phi}{R} + \frac{L\sin\sigma}{mV\cos\gamma} \tag{33}$$

$$\dot{\phi} = \frac{V\cos\gamma\sin\psi}{R} \tag{34}$$

where the Earth was assumed to be a sphere and its rotation was neglected. The variables are defined as follows: V, velocity; h, altitude; R, geocentric distance; γ, flight-path angle; ψ, azimuth; ϕ, latitude; α, angle of attack; σ, bank angle; m, mass (83.6×10^3 [kg]); L, lift ($= \rho V^2 C_L S/2$); D, drag ($= \rho V^2 C_D S/2$); S, wing area ($= 534.6$ [m^2]); ρ, atmospheric density ($= 1.225\exp(-1.5 \times 10^{-4}h)$[kg/m^2]; and μ, gravitational constant of the Earth.

2) Initial conditions at fixed time $t = t_0 = 0$[s]:

$$V(t_0) = 7800 \text{ [m/s]}, \qquad h(t_0) = 80 \times 10^3 \text{ [m]}$$

$$\gamma(t_0) = -1 \text{ [deg]}, \qquad \psi(t_0) = 90 \text{ [deg]}$$

$$\phi(t_0) = 0 \text{ [deg]} \tag{35}$$

3) Path constraints:

$$0 \le \alpha \le 40 \text{ [deg]}, \qquad -90 \le \sigma \le 90 \text{ [deg]} \tag{36}$$

$$h \ge 0 \text{ [m]} \tag{37}$$

$$n_{LF} = \frac{(L\cos\alpha + D\sin\alpha)}{(mg_0)} \le 2.5 \tag{38}$$

$$q = \frac{\rho V^2}{2} \le 5.0 \times 10^4 \text{ [Pa]} \tag{39}$$

$$\dot{Q} = 6.463 \times 10^{-5}\sqrt{\rho}V^{3.07} \le 3.0 \times 10^5 \text{ [W/m}^2\text{]} \tag{40}$$

where n_{LF}, q, and $\dot{Q}$ denote the load factor, the dynamic pressure, and the heating rate at the stagnation point, respectively.

4) The terminal conditions at unknown time $t = t_f$ [s] are

$$V(t_f) = 760 \text{ [m/s]}, \qquad h(t_f) = 24.4 \times 10^3 \text{ [m]}$$

$$\gamma(t_f) = -5 \text{ [deg]} \tag{41}$$

This terminal condition is referred to as the terminal area energy management interface.

5) Objective function to be minimized:

$$J = -\phi(t_f) \tag{42}$$

The minimization of the objective function (42) is equivalent to the maximization of the cross range.

The number of divided time intervals for the control variables $\alpha(t), \sigma(t)$, was $M = 15$, and the number of steps in the numerical integration was $N = 100$. In addition, the scale of the objective function and the constraints were coordinated by multiplying by appropriate values. The GA parameters and the range of the initial GA population were specified by the following quantities:

$$N_P = 500, \qquad N_C = 200, \qquad N_G = 5 \times 10^4$$

$$m = 4, \qquad r = 100 \tag{43}$$

$$0 \le \alpha \le 40 \text{ [deg]}, \qquad 0 \le \sigma \le 90 \text{ [deg]}$$

$$1000 \le t_f \le 4000 \text{ [s]} \tag{44}$$

Considering the high nonlinearity of the problem, the number of individuals N_P and the crossover number N_C were several times larger than those used in the constrained brachistchrone problem. The final generation number N_G was determined in the same way as that in the constrained brachistchrone problem. Furthermore, the penalty parameter r was specified by an arbitrarily large value.

The best individual at the final generation of the proposed GA was provided as an initial solution to DC with SQP, and DC minimized the solution further. In DC, the number of nodes was specified as 100 and trapezoidal integration was employed.

Figure 8 shows the time histories of the angle of attack α and the heating rate $\dot{Q}$. As can be seen, the angle of attack was constrained by the heating rate boundary for a while, and then it was approximately kept to 12 deg, which gives maximum lift-to-drag ratio for

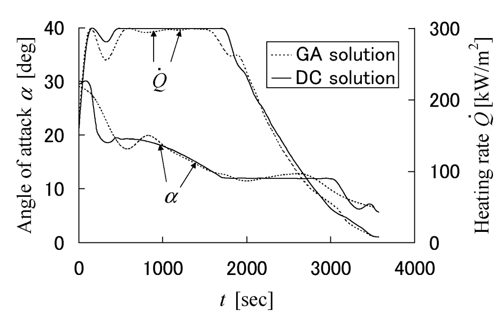


Fig. 8 Control variable $\alpha(t)$ and heating rate history of the obtained solutions.

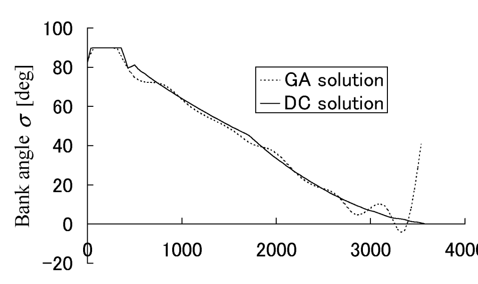


Fig. 9 Control variable $\sigma(t)$ of the obtained solutions.

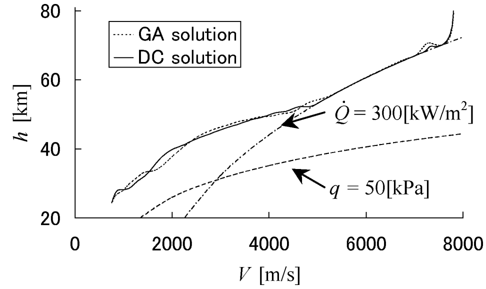


Fig. 10 The h–V trajectories of the obtained solutions.

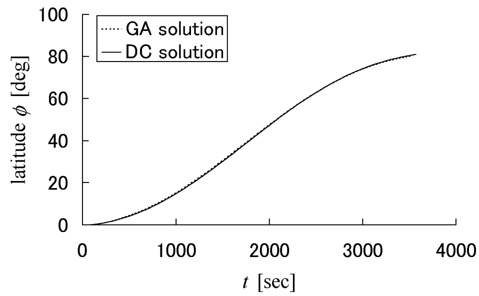


Fig. 11 Control variable $\sigma(t)$ of the obtained solutions.

the simulated model at hypersonic speed. The time history of bank angle σ, the h–V trajectory, and the time history of latitude ϕ are shown in Figs. 9, 10, and 11, respectively. It can be seen that the GA solution and the DC solution are very close and they both have the same properties of the shuttle's maximum cross-range trajectory.[10] These results indicate that the proposed GA is reliable in finding an initial solution for gradient-based trajectory optimization. Finally, it should be noted that some knowledge of the problem is required in the end to judge the physical reasonableness of the GA solution just as we have done in this example (i.e., checking the trajectory against the shuttle's maximum cross-range trajectory).

Conclusions

To provide an appropriate initial solution to gradient-based direct trajectory optimization, this study proposes a new selection method for a real-coded genetic algorithm (GA). Through application to the constrained brachistochrone problem, it was demonstrated that the proposed GA robustly performed global search of the objective function and feasibility search even with a fixed large penalty parameter. Furthermore, the proposed GA was applied to the optimization of a space plane's reentry trajectory problem, which had a complexity of practical interest. It was observed that the solution of the proposed GA approached the vicinity of the optimal solution, and it was reliable enough as an initial solution to gradient-based trajectory optimization.

Acknowledgment

This study was supported by a Grant-in-Aid for Fellows of the Japan Society for the Promotion of Science.

References

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

[2]Bryson, A. E., Jr., and Ho, Y. C., *Applied Optimal Control*, Hemisphere, Washington, DC, 1975, pp. 42–127.

[3]Wu, A. K., and Miele, A., "Sequential Conjugate Gradient-Restoration Algorithm for Optimal Control Problems with Non-Differential Constraints and General Boundary Conditions, Part 1," *Optimal Control Applications and Methods*, Vol. 1, No. 1, 1980, pp. 69–88.

[4]Gonzalez, S., and Rodriguez, S., "Modified Quasiilnearization Algorithm for Optimal Control Problems with Nondifferential Constraints and General Boundary Conditions," *Journal of Optimization Theory and Applications*, Vol. 50, No. 1, 1986, pp. 109–128.

[5]Pesch, H. J., "Offline and Online Computation of Optimal Trajectories in the Aerospace Field," *Applied Mathematics in Aerospace Sciences and Engineering*, Plenum, New York, 1994, pp. 165–220.

[6]Jänsch, C., Schnepper, K., and Well, K. H., "Multi-Phase Trajectory Optimization Methods with Applications to Hypersonic Vehicles," *Applied Mathematics in Aerospace Sciences and Engineering*, Plenum, New York, 1994, pp. 133–164.

[7]Brauer, G. L., Cornick, D. E., and Stevenson, R., "Capabilities and Applications of the Program to Optimize Simulated Trajectories (POST)," NASA CR-2770, Feb. 1977.

[8]Hargraves, C. R., and Paris, S. W., "Direct Trajectory Optimization Using Nonlinear Programming and Collocation," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 4, 1987, pp. 338–342.

[9]Betts, J. T., and Huffman, W. P., "Path-Constrained Optimization Using Sparse Sequential Quadratic Programming," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 1, 1993, pp. 59–68.

[10]Betts, J. T., *Practical Methods for Optimal Control Using Nonlinear Programming*, Society for Industrial and Applied Mathematics, Philadelphia, 2001, pp. 61–175.

[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]Takae, Y., Ishikawa, Y., and Yamaguchi, K., "Application of Genetic Algorithm to Optimal Flight Path Problem," *Journal of the Japan Society for Aeronautical and Space Sciences,* Vol. 44, No. 514, 1996, pp. 637–645.

[13]Gage, P. J., Braun, R. D., and Kroo, I. M., "Interplanetary Trajectory Optimization Using a Genetic Algorithm," *Journal of the Astronautical Sciences*, Vol. 43, No. 1, 1995, pp. 59–75.

[14]Kita, H., Ono, I., and Kobayashi, S., "Multi-Parental Extension of the Unimodal Normal Distribution Crossover for Real-Coded Genetic Algorithms," *Transactions of the Society of Instrument and Control Engineers*, Vol. 36, No. 10, 2000, pp. 875–883.

[15]Deb, K., and Goldberg, D. E., "An Investigation of Niche and Species Formation in Genetic Function Optimization," *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, 1989, pp. 42–50.

[16]Takahashi, O., Kita, H., and Kobayashi, S., "A Distance Dependent Alternation Model for Real-Coded Genetic Algorithms," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, 1999, pp. 619–624.

[17]Nanakorn, P., and Meesomklin, K., "An Adaptive Penalty Function in Genetic Algorithms for Structural Design Optimization," *Computers and Structures*, Vol. 79, No. 29, 30, 2001, pp. 2527–2539.

[18]Surry, P. D., and Radcliffe, N. J., "The COMOGA Method: Constrained Optimization by Multi-Objective Genetic Algorithms," *Control and Cybernetics*, Vol. 26, No. 3, 1997, pp. 391–412.

[19]Nomura, S., Hozumi, K., Kawamoto, I., and Miyamoto, Y., "Experimental Studies on Aerodynamic Characteristics of SSTO Vehicle at Subsonic to Hypersonic Speeds," *Proceedings of the 16th International Symposium on Space Technology and Science*, Japan Society for Aeronautical and Space Sciences, Tokyo, 1988, pp. 1547–1554.