

TRAJECTORY OPTIMIZATION OF SPACE PLANE USING GENETIC ALGORITHM COMBINED WITH GRADIENT METHOD

Nobuhiro Yokoyama

**Department of Aeronautics and Astronautics, University of Tokyo,
(Research Fellow of the Japan Society for the Promotion of Science)
7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan**

Keywords: *Genetic Algorithm, Sequential Quadratic Programming, Singular Perturbation, Guidance and Control Optimization, Space Plane*

Abstract

Although a large number of studies has been carried out on the trajectory optimization of space planes, the dynamics of the vehicle has usually been assumed as the point mass and the rigid body dynamics has not been covered at the stage of trajectory optimization. In a practical point of view, it is important to optimize the trajectory of a space plane with the rigid body assumption, since the optimal trajectory obtained with the point mass assumption might be difficult for the attitude control system to track, and the force generated by the attitude control might affect the flight trajectory. Therefore, this study aims to obtain the optimal trajectory with the rigid body assumption using effective optimization methods. In order to avoid the difficulty in direct optimization of the actual control variables, the guidance law and the feedback gains are optimized simultaneously. Furthermore, Genetic Algorithm (GA) is applied in order to obtain an appropriate initial guess of the optimal solution, and subsequently Sequential Quadratic Programming (SQP) is utilized to refine the solution. The effectiveness of the proposed methods is demonstrated.

1 Introduction

Trajectory optimization with minimum propellant consumption will play an important role to design the flight path of a space plane. In the studies of the trajectory optimization

problem of a space plane [1]-[4], its dynamics has been usually assumed as a point mass, and the attitude control as a rigid body has not been covered simultaneously. Moreover, the attitude control as a rigid body has usually been designed to track the optimal trajectory obtained with the point mass assumption. However, the optimal trajectory obtained with the point mass assumption is not necessarily optimal for the trajectory taking the rigid body dynamics into consideration. This is because of two main reasons. First, the optimal trajectory obtained with the point mass assumption might be difficult for the attitude control system to track. Second, the force generated by the attitude control such as trim drag might affect the flight trajectory.

Thus, in a practical point of view, it is important to optimize the trajectory with the rigid body assumption. However, it is usually difficult to solve this type of problem. This is mainly due to the substantially smaller time scale in the control variables of the rigid body dynamics, which results in high dimensionality with considerable nonlinearity in the solution domain and the difficulty in giving an appropriate initial guess of the solution.

Therefore, this study aims to obtain the optimal trajectory with the rigid body assumption using effective optimization methods. Taking the short time scale of the

actual control variables into consideration, the guidance law and the feedback gains are optimized simultaneously instead of optimizing the actual control variables. Furthermore, GA [5] is applied in order to obtain an appropriate initial guess of the optimal solution, and subsequently SQP [6] is utilized to refine the solution. The effectiveness of the proposed methods is demonstrated through the application for the ascent trajectory optimization problem on the equatorial plane.

2 Optimization Methods

2.1 Trajectory Optimization Problem

The trajectory optimization problem covered in this study is defined as follows.

Let $\mathbf{x}(t)$ be the state variables, $\mathbf{u}(t)$ be the control variables, and \mathbf{p} be the static unknown parameters. If t_0 or t_f are unknown, they are incorporated into \mathbf{p} . The system must satisfy the following constraints.

Initial conditions at time $t = t_0$:

$$\boldsymbol{\psi}_{E0}(\mathbf{x}(t_0), \mathbf{u}(t_0), \mathbf{p}, t_0) = \mathbf{0} \quad (1)$$

$$\boldsymbol{\psi}_{I0}(\mathbf{x}(t_0), \mathbf{u}(t_0), \mathbf{p}, t_0) \leq \mathbf{0} \quad (2)$$

State equations:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) \quad (3)$$

Path constraints at time $t \in [t_0, t_f]$:

$$\mathbf{C}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) = \mathbf{0} \quad (4)$$

$$\mathbf{S}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}) \leq \mathbf{0} \quad (5)$$

Terminal conditions at time $t = t_f$:

$$\boldsymbol{\psi}_{Ef}(\mathbf{x}(t_f), \mathbf{u}(t_f), \mathbf{p}, t_f) = \mathbf{0} \quad (6)$$

$$\boldsymbol{\psi}_{If}(\mathbf{x}(t_f), \mathbf{u}(t_f), \mathbf{p}, t_f) \leq \mathbf{0} \quad (7)$$

Under the above constraints, the control variables and undetermined parameters are optimized to minimize the following performance index

$$\phi(\mathbf{x}(t_f), \mathbf{u}(t_f), \mathbf{p}, t_f) \quad (8)$$

2.2 Direct Shooting Application

The trajectory optimization problem stated in the preceding section can be converted to a nonlinear programming problem (NLP). In this study, direct shooting application [7] is used as the conversion method.

Divide the time domain $[t_0, t_f]$ into N segments.

$$[t_0, t_1], [t_1, t_2], \dots, [t_{N-1}, t_N] \quad (t_N = t_f) \quad (9)$$

At the initial point of each segment, the state variables \mathbf{x}_i are allocated. Furthermore, dividing each segment into M sub-segments, the control variables \mathbf{u}_{ij} ($j=0, \dots, M$) at the nodes of each sub-segment are also allocated. Between the nodes of sub-segments, the control variables $\mathbf{u}(t)$ are determined by linear interpolation of \mathbf{u}_{ij} ($j=0, \dots, M$). Then, propagation of the trajectory from the initial point to the final point in each segment can be accomplished by integrating the state equations. In this study, 4th order Runge-Kutta-Gill method is used as the integration scheme. Thus, the total set of NLP variables \mathbf{X} is defined as

$$\mathbf{X} = [\mathbf{X}_0^T, \mathbf{X}_1^T, \dots, \mathbf{X}_{N-1}^T, \mathbf{p}^T]^T \quad (10)$$

$$\mathbf{X}_i = [\mathbf{x}_i^T, \mathbf{u}_{i0}^T, \dots, \mathbf{u}_{iM}^T]^T \quad (11)$$

Where \mathbf{p} are the parameters defined in the preceding section.

In addition, the continuity condition for the state and control variables at the joint of segments must be satisfied, i.e.,

$$\mathbf{A}_i(\mathbf{X}_i, \mathbf{X}_{i+1}, \mathbf{p}) = \begin{bmatrix} \mathbf{x}(t_{i+1}) - \mathbf{x}_{i+1} \\ \mathbf{u}_{iM} - \mathbf{u}_{(i+1)0} \end{bmatrix} = \mathbf{0} \quad (12)$$

Where $\mathbf{x}(t_{i+1})$ denotes the state variables obtained by solving the following initial value problem of the state equation.

$$\begin{aligned} \mathbf{x}(t_i) &= \mathbf{x}_i \\ \dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \mathbf{p}), t \in [t_i, t_{i+1}] \end{aligned} \quad (13)$$

The performance index is evaluated at the terminal time t_f . In addition, path constraints are evaluated at the nodes of each sub-segment.

Thus, the trajectory optimization problem is converted into the following NLP.

$$\text{minimize: } F(\mathbf{X}) = \phi(\mathbf{X}_{N-1}, \mathbf{p}) \quad (14)$$

subject to :

$$\mathbf{G}(\mathbf{X}) = \begin{bmatrix} \psi_{E0}(\mathbf{X}_0, \mathbf{p}) \\ \mathbf{C}_{i \times M+j}(\mathbf{X}_i, \mathbf{X}_{i+1}, \mathbf{p}) \\ \mathbf{A}_i(\mathbf{X}_i, \mathbf{X}_{i+1}, \mathbf{p}) \\ \psi_{Ef}(\mathbf{X}_{N-1}, \mathbf{p}) \end{bmatrix} = \mathbf{0} \quad (15)$$

$$\mathbf{H}(\mathbf{X}) = \begin{bmatrix} \psi_{I0}(\mathbf{X}_0, \mathbf{p}) \\ \mathbf{S}_{i \times M+j}(\mathbf{X}_i, \mathbf{X}_{i+1}, \mathbf{p}) \\ \psi_{Ef}(\mathbf{X}_{N-1}, \mathbf{p}) \end{bmatrix} \leq \mathbf{0} \quad (16)$$

$$(0 \leq i \leq N-2, \quad 0 \leq j \leq M)$$

In the converted NLP, the time domain division and introduction of the additional NLP variables reduce the sensitivity of the ordinary differential equations to the NLP variables. Moreover, the Jacobian matrix of the constraint functions becomes sparse. In SQP approach, these characteristics in the converted NLP can enhance the robustness of the convergence and reduce the computational cost to obtain the gradient information. On the other hand, the time domain division is not carried out in GA approach (i.e., $N=1$), because the gradient information is not necessary and additional NLP variables as well as the continuity condition degrade the convergence property of GA.

2.3 Genetic Algorithm

The continuous parameter GA is utilized to select the initial solutions for the SQP. In this method, NLP variables \mathbf{X} are treated as the gene of each individual. The algorithm used in this study is as follows.

(Step1: Initialization)

The initial N_p population of the NLP variables is prepared at random.

(Step2: Crossover)

Multi-parental Unimodal Normal Distribution Crossover [5] (UNDX- m) is used as the crossover scheme. This scheme achieves efficient global search:

$m+2$ parents $\mathbf{X}_p^{(1)}, \dots, \mathbf{X}_p^{(m+2)}$ are selected at random. The median point of the first $m+1$ parents is defined as \mathbf{X}_G , i.e.,

$$\mathbf{X}_G = \frac{1}{m+1} \sum_{i=1}^{m+1} \mathbf{X}_p^{(i)} \quad (17)$$

The difference vectors of each parent are defined as

$$\hat{\mathbf{d}}^{(i)} = \mathbf{X}_p^{(i)} - \mathbf{X}_G \quad (i=1, \dots, m+2) \quad (18)$$

Let \hat{D} be the length of the component of $\hat{\mathbf{d}}^{(m+2)}$ orthogonal to $\hat{\mathbf{d}}^{(1)}, \dots, \hat{\mathbf{d}}^{(m)}$. Moreover, let $\hat{\mathbf{e}}^{(1)}, \dots, \hat{\mathbf{e}}^{(n-m)}$ be orthonormal basis of the subspace orthogonal to $\hat{\mathbf{d}}^{(1)}, \dots, \hat{\mathbf{d}}^{(m)}$. Generate children $\mathbf{X}_c^{(j)}$ ($j=1, \dots, N_c$) by following equation.

$$\mathbf{X}_c^{(j)} = \mathbf{X}_G + \sum_{i=1}^m w_i \hat{\mathbf{d}}^{(i)} + \hat{D} \sum_{i=1}^{n-m} v_i \hat{\mathbf{e}}^{(i)} \quad (19)$$

Where w_i, v_i are random numbers which conform to normal distribution with 0 mean and standard deviation of σ_w, σ_v respectively. The standard deviations of each distribution are set to the value recommended in [5].

$$\sigma_w = \frac{1}{\sqrt{m}}, \quad \sigma_v = \sqrt{\frac{3(m+1)}{2(n-m)(m+2)}} \quad (20)$$

(Step3: Selection)

In order to achieve the global search of the performance index and the feasibility search simultaneously, this study adopts a new selection method, which exploits the multiple criteria, i.e., the distance of the gene vectors, the performance index, and the following penalty function.

$$F_r(\mathbf{X}) = F(\mathbf{X}) + \tilde{r} \left[\sum_{i=1}^{m_E} |G_i(\mathbf{X})| + \sum_{i=1}^{m_I} \max[0, H_i(\mathbf{X})] \right] \quad (21)$$

Where \tilde{r} denotes the penalty parameter. m_E, m_I denote the number of NLP equality

and inequality constraints respectively. The selection method is as follows:

(Step3-1) Rank the created N_C children in ascending order on their penalty function. Set the rank parameter $i = 1$.

(Step3-2) Carry out the selection with respect to the rank i th child $\mathbf{X}_C^{(i)}$: If there are any parents $\mathbf{X}_P^{(j)}$ that satisfy both (22) and (23), replace the nearest parent $\mathbf{X}_P^{(j_n)}$ (i.e., $|\mathbf{X}_P^{(j_n)} - \mathbf{X}_C^{(i)}|$ is minimum amongst the parents which satisfy both (22) and (23).) with $\mathbf{X}_C^{(i)}$ and go to Step3-3.

$$F(\mathbf{X}_C^{(i)}) \leq F(\mathbf{X}_P^{(j)}) \quad (22)$$

$$F_r(\mathbf{X}_C^{(i)}) < F_r(\mathbf{X}_P^{(j)}) \quad (23)$$

If $i = N_C$ or there are no parents that satisfy (23), go to Step3-3. Otherwise, set $i \rightarrow i + 1$ and repeat Step3-2

(Step3-3) Rank the children newly in ascending order on performance index and set the rank parameter $i' = 1$.

(Step3-4) Carry out the selection with respect to the rank i' th child $\mathbf{X}_C^{(i')}$: If there are any parents $\mathbf{X}_P^{(j)}$ that satisfy (24), replace the nearest parent $\mathbf{X}_P^{(j_n)}$ with $\mathbf{X}_C^{(i')}$ and go to Step4.

$$F_r(\mathbf{X}_C^{(i')}) < F_r(\mathbf{X}_P^{(j)}) \quad (24)$$

If $i' = N_C$ or there are no parents that satisfy (24), go to Step4. Otherwise set $i' \rightarrow i' + 1$ and repeat Step3-4.

(Step4: Termination check)

If iteration number equals N_I (Step2-Step4 correspond to 1 iteration), terminate the algorithm. Otherwise, return to Step2.

Five parameters $N_P, N_C, N_I, m, \tilde{r}$ characterize the search performance of the GA. Especially, the penalty parameter \tilde{r} should be sufficiently large so as to assure that the solution converge on the feasible region.

2.4 Sequential Quadratic Programming

The algorithm of SQP [6] is as follows.

(Step1) Select an appropriate initial solution $\mathbf{X}^{(0)}$ and positive definite symmetric matrix $\mathbf{B}^{(0)}$. Set the parameter $k = 0$.

(Step2) Solve the quadratic programming problem (25)-(27) and obtain the variables $\mathbf{d}^{(k)}$ as well as Lagrange multipliers $\lambda_i^{(k+1)} (i = 1, \dots, m_E + m_I)$.

$$\text{minimize: } \nabla F(\mathbf{X}^{(k)}) + \frac{1}{2} \mathbf{d}^{(k)T} \mathbf{B}^{(k)} \mathbf{d}^{(k)} \quad (25)$$

$$\text{subject to: } \mathbf{G}(\mathbf{X}^{(k)}) + \nabla \mathbf{G}(\mathbf{X}^{(k)}) = \mathbf{0} \quad (26)$$

$$\mathbf{H}(\mathbf{X}^{(k)}) + \nabla \mathbf{H}(\mathbf{X}^{(k)}) \leq \mathbf{0} \quad (27)$$

(Step3) Carry out the one-dimensional search of the penalty function (21), i.e., find the scalar variable $\alpha^{(k)} \in (0, 1]$ that satisfies

$$\alpha^{(k)} = \arg \min [F_r(\mathbf{X}^{(k)} + \alpha^{(k)} \mathbf{d}^{(k)})] \quad (28)$$

Note that the penalty parameter \tilde{r} in (21) is determined by the following equation.

$$\tilde{r} = \rho + \max_{1 \leq i \leq m_E + m_I} |\lambda_i| \quad (29)$$

Where ρ denotes an appropriate positive parameter. Then, update the variables as follows.

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} + \alpha^{(k)} \mathbf{d}^{(k)} \quad (30)$$

(Step4) If the solution $\mathbf{X}^{(k+1)}$ satisfies the convergence condition, terminate the algorithm. Otherwise, update the matrix $\mathbf{B}^{(k)}$ to $\mathbf{B}^{(k+1)}$ by the Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula [6] and return to Step2.

Generally, it takes long time to calculate the gradient information $\nabla F(\mathbf{X}^{(k)})$, $\nabla \mathbf{G}(\mathbf{X}^{(k)})$, $\nabla \mathbf{H}(\mathbf{X}^{(k)})$ in Step2. However, the sparsity of the Jacobian matrix in the converted NLP makes it possible to exploit the sparse difference approach [8] and it results in a substantial reduction of calculation time.

2.5 Combinatorial Strategy

The best solution in terms of the penalty function at the final iteration of GA is given as the initial solution for SQP. Then SQP refines it. The merit of this method is that the solution in SQP can converge on the global minimum in the feasible region with high probability because of the reasonableness of the best GA solution.

3 Trajectory Optimization of a Space Plane

3.1 Statement of the Problem

An ascent of a space plane on the equatorial plane is considered. After taking off horizontally, the space plane ascends and accelerates. Above 90km altitude, the engine is cut off and the space plane starts coasting. At the apogee in the elliptic orbit, the engine is reignited and the space plane is placed on the circular orbit at an altitude of 400km.

The vehicle model used in this study is Single Stage To Orbit (SSTO), and its aerodynamic model is based on the winged-cone configuration [9] as shown in Fig. 1. Assume that the space plane is equipped with two engines, i.e., air turbo ramjet engine (AE) runs below Mach 6, and rocket engine (RE) runs over Mach 6.

Let us consider the two-dimensional rigid body dynamics of the space plane. The state variables are V : velocity, γ : flight path angle, h : altitude, Θ : pitch angle, Q : pitch rate, m : mass of the vehicle. The state equations are described as follows.

$$\dot{V} = \frac{T \cos(\alpha + \delta_T) - D}{m} + (r\omega^2 - \frac{\mu}{r^2}) \sin \gamma \quad (31)$$

$$\dot{\gamma} = \frac{1}{V} \left[\frac{L + T \sin(\alpha + \delta_T)}{m} + (r\omega^2 - \frac{\mu}{r^2}) \cos \gamma \right] + V \cos \gamma / r + 2\omega \quad (32)$$

$$\dot{h} = V \sin \gamma \quad (33)$$

$$\dot{\Theta} = Q + V \cos \gamma / r \quad (34)$$

$$\begin{aligned} \dot{Q} = I_{yy}^{-1} [& \tilde{M} - L(z_{CG} \sin \alpha + x_{CG} \cos \alpha) \\ & - D(x_{CG} \sin \alpha - z_{CG} \cos \alpha) \\ & + T\{(z_T - z_{CG}) \cos \delta_T + (x_T - x_{CG}) \sin \delta_T\}] \end{aligned} \quad (35)$$

$$\dot{m} = -T / (I_{sp} g_0) \quad (36)$$

Where γ : angle of attack, L : lift, D : drag, T : thrust, \tilde{M} : pitching moment, I_{sp} : specific impulse, r : distance from the center of the earth, δ_T : thrust angle, I_{yy} : moment of inertia, x_{CG}, z_{CG} : center of gravity, x_T, z_T : action point of thrust, ω : angular velocity of the rotation of the earth, μ : gravitational constant of the earth g_0 : acceleration of gravity at the surface of the earth.

Let q be the dynamic pressure, and S be the reference area of the vehicle. Then L, D, \tilde{M} are defined as follows.

$$L = qSC_L, D = qSC_D, \tilde{M} = qSC_m \quad (37)$$

Where C_L, C_D, C_m are given by the cubic polynomials of the elevon deflection angle δ_e .

$$\begin{aligned} C_L = C_{L0}(\alpha, M) + C_{L\delta_e}^{(1)}(\alpha, M)\delta_e \\ + C_{L\delta_e}^{(2)}(\alpha, M)\delta_e^2 + C_{L\delta_e}^{(3)}(\alpha, M)\delta_e^3 \end{aligned} \quad (38)$$

$$\begin{aligned} C_D = C_{D0}(\alpha, M) + C_{D\delta_e}^{(1)}(\alpha, M)\delta_e \\ + C_{D\delta_e}^{(2)}(\alpha, M)\delta_e^2 + C_{D\delta_e}^{(3)}(\alpha, M)\delta_e^3 \end{aligned} \quad (39)$$

$$\begin{aligned} C_m = C_{m0}(\alpha, M) + C_{mq}(\alpha, M)Qc / (2V) \\ + C_{m\delta_e}^{(1)}(\alpha, M)\delta_e + C_{m\delta_e}^{(2)}(\alpha, M)\delta_e^2 \\ + C_{m\delta_e}^{(3)}(\alpha, M)\delta_e^3 \end{aligned} \quad (40)$$

M denotes the Mach number. The aerodynamic effect of the canard deflection δ_c is also given by the cubic polynomials. The canard is deflected in conjunction with the elevon up to Mach 0.9 in the following way.

$$\delta_c = C_0 \delta_e, -0.5 \leq C_0 \leq 0.5 \quad (41)$$

Thus, the effect of the canard deflection is incorporated into (38)-(40) and C_0 is regarded as one of the static optimization parameter. The canard is stored in the body beyond Mach 0.9.

The maximum thrust and the specific impulse are given as follows.

$$T = \begin{cases} C_{AE}(h, M)S_{AE} & : \text{AE phase} \\ T_{RE} & : \text{RE phase} \end{cases} \quad (42)$$

$$Isp = \begin{cases} Isp_{AE}(h, M) & : \text{AE phase} \\ Isp_{RE} & : \text{RE phase} \end{cases} \quad (43)$$

Both engines hold the maximum thrust. The unit thrust $C_{TAE}(h, M)$ and the specific impulse $Isp_{AE}(h, M)$ are based on the data in [10].

I_{yy} and x_{CG} are given as the function of the remaining mass of LH₂ and LOX. For simplicity, z_{CG} is assumed zero throughout the flight. Other parameters are defined as shown in Table 1.

Thus, the following trajectory optimization problem is defined.

Initial conditions at time $t = 0$:

$$V(0) = 150 \text{ [m/s]} \quad (44)$$

$$h(0) = 0 \text{ [km]} \quad (45)$$

$$\dot{\gamma}(0) \geq 0 \text{ [rad/s]} \quad (46)$$

$$Q(0) = 0 \text{ [rad/s]} \quad (47)$$

$$m(0) = 1.5 \times 10^5 \text{ [kg]} \quad (48)$$

Path constraints:

$$h \geq 0 \text{ [m]} \quad (49)$$

$$-1 \leq \alpha \leq 12 \text{ [deg]} \quad (50)$$

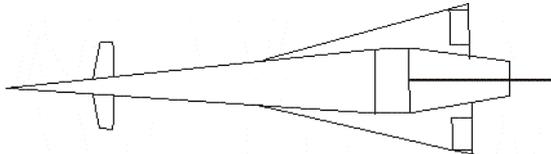


Fig. 1. Winged-cone configuration

Table 1. Design parameters

Parameter	Value
Reference area S	334.7 [m ²]
Fan area of AE S_{AE}	4.8 [m ²]
Max thrust of RE T_{RE}	1.03×10^3 [kN]
Specific impulse of RE Isp_{RE}	450 [sec]

$$q \leq 5.0 \times 10^4 \text{ [Pa]} \quad (51)$$

$$n_{LF} = (L \cos \alpha + D \sin \alpha) / (mg_0) \leq 2.5 \quad (52)$$

$$-20 \leq \delta_e \leq 20 \text{ [deg]}, \quad -20 \leq \delta_T \leq 20 \text{ [deg]} \quad (53)$$

Terminal conditions at time $t = t_f$:

$$h(t_f) \geq 90 \text{ [km]} \quad (54)$$

$$\gamma(t_f) \geq 0 \text{ [deg]} \quad (55)$$

$$H_a(h(t_f), V(t_f), \gamma(t_f)) = 400 \text{ [km]} \quad (56)$$

Where H_a denotes the apogee altitude of the elliptic orbit.

The performance index to be minimized is

$$\phi = -m(t_f) \exp \left[-\frac{\Delta V(h(t_f), V(t_f), \gamma(t_f))}{Isp_{RE} g_0} \right] \quad (57)$$

Where ΔV denotes the incremental velocity of the apogee boost. Note that minimizing the above performance index is equivalent to minimizing the total propellant consumption.

3.2 Simultaneous Optimization Method of Guidance Law and Feedback Gains

It is unrealistic to adopt the actual control variables δ_e, δ_T as the optimization variables u_{ij} , since the short time scale of these variables might result in a large number of time domain division. Therefore, feedback control optimization strategy is adopted in order to avoid the difficulty. In this strategy, the guidance law and the feedback gains are treated as the optimization variables instead of actual control variables δ_e, δ_T . Consequently, δ_e, δ_T are determined by the feedback control law.

Since we focus on the planar motion of the space plane and the engines hold maximum thrust, the time histories of ascent and acceleration can be determined by a guidance law with 1 degree of freedom. Thus, for simplicity, the time history of the reference altitude $h_c(t)$ is adopted as the guidance law and the actual altitude $h(t)$ is considered as the output of the control system. As a feedback control law, inverse PD control with singular

perturbation [11] is utilized in order to handle the high nonlinearity of the state equations. The outline is as follows:

Let us categorize Θ, α, Q as the slow state variables and other state variables as the fast state variables. Then the time scale separation for $\Theta, \alpha, \delta_e, \delta_T$ is described in the following way.

$$\Theta = \bar{\Theta} + \Delta\Theta, \alpha = \bar{\alpha} + \Delta\alpha = \bar{\alpha} + \Delta\Theta \quad (58)$$

$$\delta_e = \bar{\delta}_e + \Delta\delta_e, \delta_T = \bar{\delta}_T + \Delta\delta_T \quad (59)$$

By differentiating (33) with respect to time t , following relational expression can be obtained.

$$\begin{aligned} \dot{h} = & \left[qS \{ C_L(\alpha, \delta_e) \cos \gamma - C_D(\alpha, \delta_e) \sin \gamma \} \right. \\ & \left. + T \sin(\gamma + \alpha + \delta_T) \right] / m + (r\omega^2 - \mu/r^2) \quad (60) \\ & + V \cos \gamma (2\omega + V \cos \gamma / r) \end{aligned}$$

Furthermore, by differentiating (34) in the same way and linearizing the derived equation with respect to δ_e and δ_T , following relational expression can be obtained.

$$\ddot{\Theta} = A_0 + A_e \delta_e + A_T \delta_T \quad (61)$$

$$\begin{aligned} A_0 = & -2V \sin \gamma (r\omega + V \cos \gamma) / r^2 + qScC_{m0} / Iyy \\ & - qSC_{L0} \left[\sin \gamma / (mr) + (x_{CG} \cos \alpha + z_{CG} \sin \alpha) / Iyy \right] \\ & - qSC_{D0} \left[\cos \gamma / (mr) + (x_{CG} \sin \alpha - z_{CG} \cos \alpha) / Iyy \right] \\ & + T \left[\cos(\gamma + \alpha) / (mr) + (z_T - z_{CG}) / Iyy \right] \quad (62) \end{aligned}$$

$$\begin{aligned} A_e = & qScC_{m\delta_e}^{(1)} / Iyy \\ & - qSC_{L\delta_e}^{(1)} \left[\sin \gamma / (mr) + (x_{CG} \cos \alpha + z_{CG} \sin \alpha) / Iyy \right] \\ & - qSC_{D\delta_e}^{(1)} \left[\cos \gamma / (mr) + (x_{CG} \sin \alpha - z_{CG} \cos \alpha) / Iyy \right] \quad (63) \end{aligned}$$

$$A_T = T \left[(x_T - x_{CG}) / Iyy - \sin(\gamma + \alpha) / (mr) \right] \quad (64)$$

Then, the control inputs $\bar{\alpha}, \bar{\delta}_e, \bar{\delta}_T$ for the slow state variables can be obtained from the nonlinear inverse dynamics equations as follows.

$$\begin{aligned} U_1 = & \left[qS \{ C_L(\bar{\alpha}, \bar{\delta}_e) \cos \gamma - C_D(\bar{\alpha}, \bar{\delta}_e) \sin \gamma \} \right. \\ & \left. + T \sin(\gamma + \bar{\alpha} + \bar{\delta}_T) \right] / m + (r\omega^2 - \mu/r^2) \quad (65) \\ & + V \cos \gamma (2\omega + V \cos \gamma / r) \end{aligned}$$

$$0 = A_0(\bar{\alpha}) + A_e(\bar{\alpha})\bar{\delta}_e + A_T(\bar{\alpha})\bar{\delta}_T \quad (66)$$

$$\bar{\delta}_T = 0 : t \leq t_c, \quad \bar{\delta}_e = 0 : t_c < t \quad (67)$$

Where U_1 is the feedback input expressed as

$$U_1 = k_{p1}(h_c - h) - k_{d1}\dot{h} \quad (68)$$

t_c is a static optimization parameter and corresponds to the time at which the active control input is changed. If obtained $\bar{\alpha}, \bar{\delta}_e, \bar{\delta}_T$ violate the constraints (50), (53), they are rounded to the boundary values in (50), (53).

By subtracting (66) from (61), following equation can be obtained.

$$\begin{aligned} \Delta\ddot{\Theta} = & \{ A_0(\alpha) - A_0(\bar{\alpha}) \} \\ & + \{ A_e(\alpha) - A_e(\bar{\alpha}) \} \bar{\delta}_e + A_e(\alpha) \Delta\delta_e \quad (69) \\ & + \{ A_T(\alpha) - A_T(\bar{\alpha}) \} \bar{\delta}_T + A_T(\alpha) \Delta\delta_T \end{aligned}$$

Then the control inputs $\Delta\delta_e, \Delta\delta_T$ for the fast state variables can be obtained from the linear inverse dynamics equations as follows.

$$\begin{aligned} U_2 = & \{ A_0(\alpha) - A_0(\bar{\alpha}) \} \\ & + \{ A_e(\alpha) - A_e(\bar{\alpha}) \} \bar{\delta}_e + A_e(\alpha) \Delta\delta_e \quad (70) \\ & + \{ A_T(\alpha) - A_T(\bar{\alpha}) \} \bar{\delta}_T + A_T(\alpha) \Delta\delta_T \end{aligned}$$

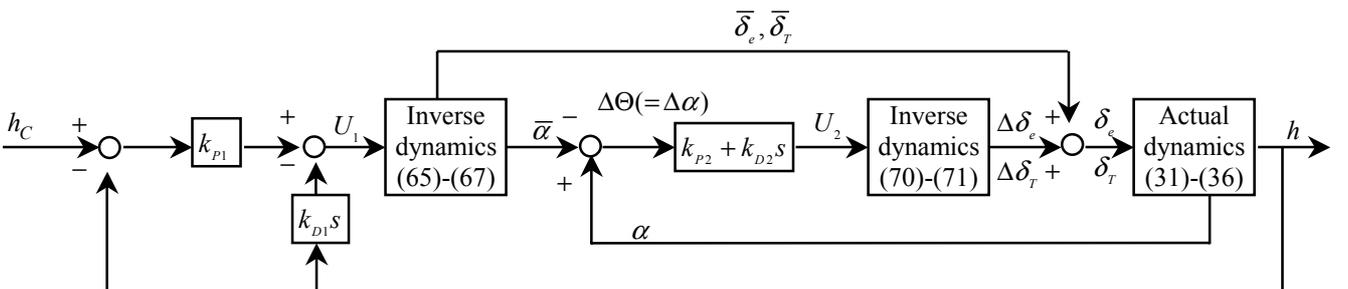


Fig. 2. Block diagram of the control system

$$\Delta\delta_T = 0 : t \leq t_C, \quad \Delta\delta_e = 0 : t_C < t \quad (71)$$

Where U_2 is the feedback input expressed as

$$U_2 = -k_{D2}\Delta\dot{\Theta} - k_{P2}\Delta\Theta \quad (72)$$

Consequently, δ_e, δ_T are obtained from (59). If obtained δ_e, δ_T violate the constraints (53), they are rounded to the boundary values in (53). Figure 2 shows the block diagram of the control system.

As stated above, the reference altitude h_C and the feedback gains $k_{P1}, k_{D1}, k_{P2}, k_{D2}$ are treated as the control variables \mathbf{u}_{ij} in the direct shooting application, i.e.,

$$\mathbf{u}_{ij} = [h_C(t_{ij}), k_{P1}(t_{ij}), k_{D1}(t_{ij}), k_{P2}(t_{ij}), k_{D2}(t_{ij})]^T \quad (73)$$

Furthermore, the following constraint with respect to the tracking ability of the control system is additionally imposed as the path constraint.

$$(h_C - h)^2 \leq 1.0 [\text{km}^2] \quad (74)$$

3.3 Assumptions of the Calculation

The parameters used in GA were determined as $N_p = 200$, $N_C = 40$, $N_I = 50000$, $m = 4$, $\tilde{r} = 100$. All the NLP constraints were divided by appropriate values and transformed into dimensionless values. In addition, the initial population of GA was generated randomly in the following range.

$$0 \leq h_C(t) \leq 100 [\text{km}], -0.5 \leq C_0 \leq 0.5 \quad (75)$$

$$10^{-3} \leq k_{P1}(t), k_{D1}(t), k_{P2}(t), k_{D2}(t) \leq 10^3 \quad (76)$$

$$300 \leq t_C \leq 600 [\text{sec}], 500 \leq t_f \leq 1000 [\text{sec}] \quad (77)$$

In order to handle an uncertainty of the scale of the gains effectively, genetic expression of the feedback gains in GA were given as exponential expression, e.g.,

$$k_{P1}(t) = 10^{6z_{P1}(t)-3}, 0 \leq z_{P1}(t) \leq 1 \quad (78)$$

Thus, $z_{P1}(t)$ was regarded as the optimization variable instead of $k_{P1}(t)$ itself. The same conversion was applied in the genetic expression of $k_{D1}(t), k_{P2}(t), k_{D2}(t)$. Furthermore, gains in GA were divided into only 4 sub-segments to reduce the computational cost. The numbers of the sub-segments of $h_C(t)$ and path constraints in GA were $M = 20$, and the number of segments and sub-segments in SQP were $N = 20$, $M = 1$ respectively.

3.4 Results and Discussion

Under the above assumptions of the problem, combinatorial optimization by GA and SQP was performed.

Table 2 shows the residuals of the constraints as well as the performance indices in GA and SQP. As can be seen, the initial solution obtained by GA was nearly feasible, and SQP refined it in terms of both the residuals of the constraints and performance index. In addition, the static parameters in the optimal solution were $C_0 = -0.09734$, $t_C = 370.9[\text{sec}]$, and $t_f = 717.9[\text{sec}]$. For comparison, some solutions included in the initial population of GA were given as the initial solution for SQP, and subsequently SQP calculation was performed. As a result, little improvement neither in the performance index nor in the feasibility could be achieved in all cases. From this results, it is obvious that using GA as a method to give an initial solution for SQP is effective in this type of highly nonlinear problem.

In Fig. 3, the $V-h$ trajectory of the optimal solution by SQP is shown. It can be seen that the $V-h$ trajectory in the AE phase is

Table 2. Optimization results

	Performance index	Residual of the constraints
Mean value in the initial population of GA	-14.11	1.969×10^2
Best solution in the final iteration of GA	-28.80	8.288×10^{-2}
Obtained solution by SQP	-29.04	1.301×10^{-5}

closely along the dynamic pressure boundary and the RE is cut off at the boundary altitude of 90 [km]. Since these results are often seen in the optimal solution for the point mass assumption, the optimality of the obtained solution is recognizable.

Figure 4 shows the time histories of the reference and actual altitude in the optimal solution by SQP. The mean error of the altitude was 415.6 [m] and the tracking constraint (74) was completely satisfied throughout the flight. Figure 5 shows the time histories of the reference and actual angle of attack. As can be seen, almost sufficient tracking could be achieved throughout the flight, although there was a difference in the tracking ability between the control by the elevon and by the thrust angle. This difference may be because the effect of the higher order terms of δ_e , which were neglected in obtaining (61), was larger than that of δ_T . The time histories of δ_e , δ_T are shown in Fig. 6. Except for the initial stage of the control by thrust angle, δ_e , δ_T were within the range between -10 [deg] and 10 [deg].

Optimized feedback gains are shown in Fig. 7. It can be seen that the magnitude of the gains for the slow state variables is substantially smaller than that for the fast state variables. This is due to the slow response of the altitude h . Larger k_{p1}, k_{d1} would induce the saturation or oscillation of $\bar{\alpha}$, which degrade the tracking ability and performance index. On the other hand, k_{p2}, k_{d2} are comparatively large because of the fast response of the angle of attack α .

In the next place, let us decompose the acceleration \dot{V} , $\dot{V}\gamma$ in the following way.

$$\dot{V} = (\dot{V})_0 + (\dot{V})_\delta, V\dot{\gamma} = (V\dot{\gamma})_0 + (V\dot{\gamma})_\delta \quad (79)$$

Where $(\dot{V})_0, (V\dot{\gamma})_0$ denote the acceleration without the effect of control inputs δ_e, δ_T , and $(\dot{V})_\delta, (V\dot{\gamma})_\delta$ denote the incremental acceleration by the effect of control inputs. Figure 8 and Figure 9 show the time histories of \dot{V} vs. $(\dot{V})_0$ and $V\dot{\gamma}$ vs. $(V\dot{\gamma})_0$ respectively. It can be seen that there is a slight difference between \dot{V} and

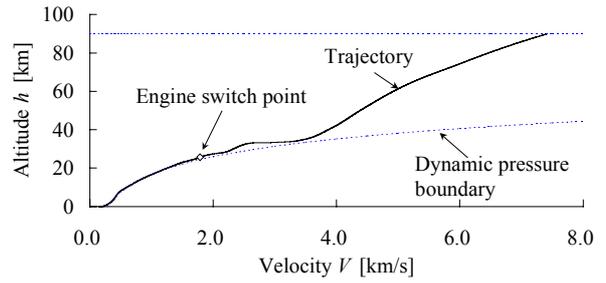


Fig. 3. $V-h$ trajectory

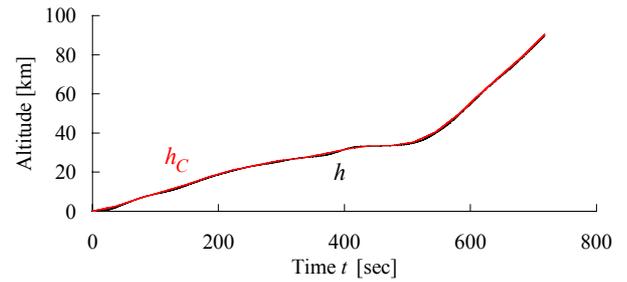


Fig. 4. Time histories of the reference and actual altitude

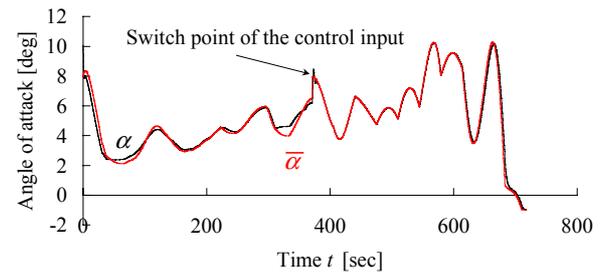


Fig. 5. Time histories of the reference and actual angle of attack

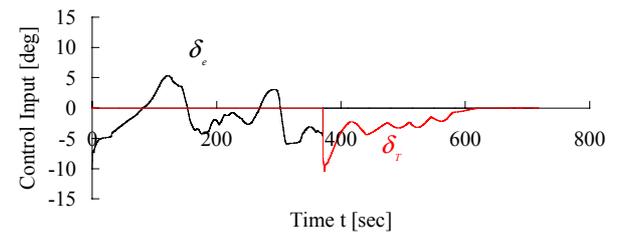


Fig. 6. Time histories of the control inputs

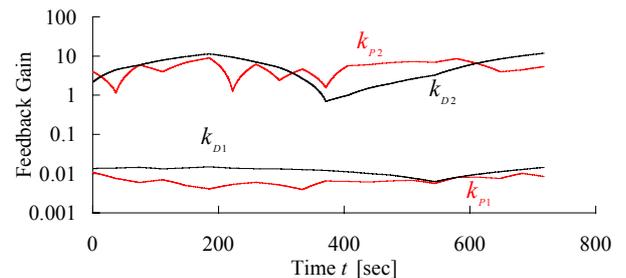


Fig. 7. Time histories of the feedback gains

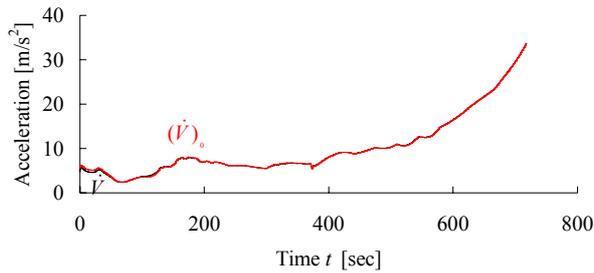


Fig. 8. Comparison of the acceleration \dot{V} and $(\dot{V})_0$

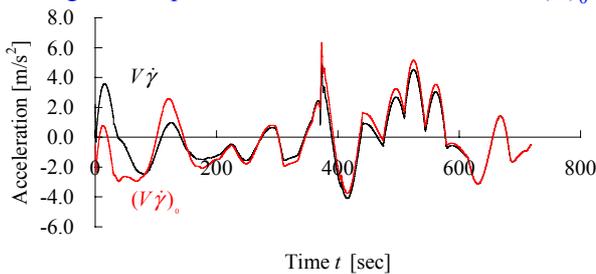


Fig. 9. Comparison of the acceleration $V\dot{\gamma}$ and $(V\dot{\gamma})_0$

$(\dot{V})_0$ except for the short period at the vicinity of take off, and the contribution of the control inputs to the acceleration \dot{V} is slight. On the other hand, there is a substantial contribution of control inputs to $V\dot{\gamma}$. It appears that the optimal trajectory should be determined taking account of this contribution of the control inputs to the path angle. Then, the results in the conventional trajectory optimization with the point mass assumption might have to be corrected, since it does not cover the above stated contribution. In contrast, the obtained trajectory in this study has an advantage in terms of its coverage of the above stated contribution.

4 Conclusions

The trajectory optimization of a space plane with the rigid body assumption was covered in this study. In order to handle the short time scale of the actual control inputs, the guidance law and the feedback control system using singular perturbation were optimized simultaneously. Furthermore, taking the difficulty in giving an appropriate initial solution into account, GA was applied to obtain the initial solution and SQP was subsequently applied to refine it. Consequently, the

reasonable optimal solution, which achieves minimization of the propellant consumption and sufficient tracking ability of the control system, was obtained. In addition, it was cleared that the obtained trajectory has an advantage in terms of its coverage of the substantial contribution of the control inputs to the path angle.

References

- [1] Hargraves CR and Paris SW. Direct trajectory optimization using nonlinear programming and collocation. *Journal of Guidance, Control, and Dynamics*, Vol. 10, No.4, pp.338-342, 1987.
- [2] Lu P. Inverse dynamics approach to trajectory optimization for an aerospace plane. *Journal of Guidance, Control, and Dynamics*, Vol.16, No.4, pp.726-732, 1993.
- [3] Kato K. *Space plane* (in Japanese). 1st edition, University of Tokyo Press, 1989.
- [4] Tsuchiya T and Suzuki S. Computational method of optimal control problem using mathematical programming (1st report) Introduction of sensitivity differential equations (in Japanese). *Transactions of the Japan Society for Aeronautical and Space Sciences*, Vol.45, No.519, pp.231-237, 1997.
- [5] Kita H, Ono I, and Kobayashi S. Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms (in Japanese). *Transactions of the Society of Instrument and Control Engineers*, Vol.36, No.10, pp.875-883, 2000.
- [6] Ibaragi T and Fukushima M. *FORTAN77 optimization programming* (in Japanese). 1st edition, Iwanami Shoten, 1991.
- [7] Betts JT. Survey of numerical methods for trajectory optimization. *Journal of Guidance, Control, and Dynamics*, Vol. 21, No.2, pp.193-207, 1998.
- [8] Betts JT and William PH. Application of sparse nonlinear programming to trajectory optimization. *Journal of Guidance, Control, and Dynamics*, Vol. 15, No.1, pp.198-206, 1992.
- [9] Shaughnessy JD, Pinckney SZ, McMinn JD, Cruz IC, and Kelley ML. Hypersonic vehicle simulation model: winged-cone configuration. *NASA TM-102610*, 1990.
- [10] Kobayashi H. Integrated design optimization of TSTO system (in Japanese). *Doctoral Dissertation, University of Tokyo*, 2001.
- [11] Sunasawa S and Ohta H. Nonlinear flight control for a reentry vehicle using inverse dynamics transformation (in Japanese). *Transactions of the Japan Society for Aeronautical and Space Sciences*, Vol.45, No.516, pp.52-61, 1997.