Multi-Objective Skip Trajectory Optimization for Hypersonic Reentry Vehicles

*Sangjun Bae*1, Hyo-Sang Shin*2, Al Savvaris*3, Antonios Tsourdos*4*
**1-4School of Aerospace, Transport and Manufacturing*
Cranfield University, Cranfield, Bedfordshire, MK43 0AL, United Kingdom

Abstract
This paper focuses on multi-objective optimization of skip trajectory for a mission at a specific altitude such as information gathering. In this paper, we use a fuzzy satisfactory goal programming (FSGP) method for multi-objective optimization. We use the FSGP method to formulate a trajectory optimization problem that is subsequently solved using GPOPS-II. We use the NASA space shuttle model since its characteristics are representative. Using simulations, we show that the trajectories produced satisfy all the constraints and priority of objectives.

1. Introduction

Many recent studies have focused on reentry trajectory optimization for hypersonic reentry vehicles [1][2][3]. Although the reentry trajectory optimization problem has been studied for a long time, rather less attention has been paid to the skip trajectory optimization problem in the context of mission planning. The main purpose of this paper is to optimize a skip trajectory for hypersonic reentry vehicles given a mission in a specific altitude for example information gathering. The mission profile starts with first entry to the atmosphere and descend up to a specific altitude and gather certain information at the altitude and skip out of the atmosphere and back into the thermosphere with thrust. For this skip trajectory optimization problem thrust is needed to skip out again.
The objectives for reentry trajectory optimization are generally maximizing the cross range and minimizing heat flux. However in the skip trajectory optimization problem the objectives also include maximizing the kinetic energy and minimize the fuel consumption. This brings about the demand of multi-objective optimization problem because each objective function conflict each other.
There are many multi-objective optimization techniques including: weighted sum method, pareto optimality, goal programming among others [4]. In this paper, we use a fuzzy satisfactory goal programming (FSGP) method proposed by Hu and Li [5]. The advantage of FSGP method is that they allow the prioritization of objectives in advance and, thus, enabling the balance between optimality and priority.
Trajectory optimization determines the path and the corresponding control to a dynamic system that optimizes a specified objective function. Most problems are highly nonlinear and strongly coupled, hence, typically the trajectory optimization problem is solved using numerical methods. Numerical methods for trajectory optimization problem can be divided into two major classes, direct methods and indirect methods [6]. Within the family of direct methods, pseudo-spectral method has become popular because the co-state can be mapped from the Karush-Kuhn-Tucker multiplier of the Non-Linear Programming (NLP) problem [7][8]. In the pseudo-spectral method, the optimal control problem is discretized into a NLP problem by parameterizing the state and control using Chebychev or Lagrange polynomials, then collocating the dynamic at the points based on Gaussian quadrature. Gauss Pseudo-spectral Method (GPM), Radau Pseudo-spectral Method (RPM) and Lobatto Pseudo-spectral Method (LPM) are most well-developed pseudo-spectral methods. In this paper, we use a general purpose optimal control software GPOPS-II that adopts hp-adaptive RPM [9][10].
The rest of paper is organized as follows. Section 2 describes the skip trajectory optimization problem formulation based on FSGP. Section 3 describes numerical simulations and results. Section 4 gives conclusion.

2. Problem Formulation

2.1 Fuzzy Satisfactory Goal Programming Method

In a multi-objective problem, finding an optimal solution that satisfies all of the objectives simultaneously is hard because of the conflict between objective functions. To tackle this, in this research, we use the Fuzzy Satisfactory Goal Programming (FSGP) method devised for multi-objective optimization problems of Hu and Li [5]. This method defines objective functions using fuzzy membership functions representing the proximity to the goal value of the objective

Copyright © 2015 by First Author and Second Author. Published by the EUCASS association with permission.

functions. With this membership function we reformulate a goal programming problem including additional constraints for priority with a relaxed order regulating parameter. We transform the problem into a single objective problem. Which is solved using a numerical method, in our case is GPOPS-II.
The generic Fuzzy Goal Programming (FGP) formulation can be written as follows [11]

$$\begin{cases} \text{Find} & \mathbf{x}, \mathbf{u} \\ \text{s.t.} & f_i(\mathbf{x}, \mathbf{u}) \to f_i^*, \quad i = 1, \dots, m \\ & \mathbf{x}, \mathbf{u} \in G(\mathbf{x}, \mathbf{u}) \end{cases} \tag{1}$$

where f_i^* is the i^{th} goal value for the i^{th} objective function of f_i. The symbol "→" refers to the three different fuzzy relationship "$\lesssim$", "$\gtrsim$" and "$\cong$". The each symbol denotes that i^{th} objective is approximately less than or equal to, approximately more than or equal to and in the vicinity of the aspiration level, respectively. G are the system constraints. Then the triangle-like membership functions can be used for three fuzzy relationship. The membership function "$\lesssim$" is as follows [5]

$$\mu_{f_i}(\mathbf{x}, \mathbf{u}) = \begin{cases} 1, & f_i(\mathbf{x}, \mathbf{u}) \le f_i^* \\ 1 - \dfrac{f_i(\mathbf{x}, \mathbf{u}) - f_i^*}{f_i^{\max} - f_i^*}, & f_i^* \le f_i(\mathbf{x}, \mathbf{u}) \le f_i^{\max} \\ 0, & f_i^{\max} \le f_i(\mathbf{x}, \mathbf{u}) \end{cases} \tag{2}$$

with the tolerance interval of $(f_i^*, f_i^{\max})$.
The membership function "$\gtrsim$" is

$$\mu_{f_i}(\mathbf{x}, \mathbf{u}) = \begin{cases} 1, & f_i^* \le f_i(\mathbf{x}, \mathbf{u}) \\ 1 - \dfrac{f_i^* - f_i(\mathbf{x}, \mathbf{u})}{f_i^* - f_i^{\min}}, & f_i^{\min} \le f_i(\mathbf{x}, \mathbf{u}) \le f_i^* \\ 0, & f_i(\mathbf{x}, \mathbf{u}) \le f_i^{\min} \end{cases} \tag{3}$$

with the tolerance interval of $(f_i^{\min}, f_i^*)$.
Finally, the membership function "$\cong$" is

$$\mu_{f_i}(\mathbf{x}, \mathbf{u}) = \begin{cases} 0, & f_i(\mathbf{x}, \mathbf{u}) \le f_i^{\min} \\ 1 - \dfrac{f_i^* - f_i(\mathbf{x}, \mathbf{u})}{f_i^* - f_i^{\min}}, & f_i^{\min} \le f_i(\mathbf{x}, \mathbf{u}) \le f_i^* \\ 1, & f_i(\mathbf{x}, \mathbf{u}) = f_i^* \\ 1 - \dfrac{f_i(\mathbf{x}, \mathbf{u}) - f_i^*}{f_i^{\max} - f_i^*}, & f_i^* \le f_i(\mathbf{x}, \mathbf{u}) \le f_i^{\max} \\ 0, & f_i^{\max} \le f_i(\mathbf{x}, \mathbf{u}) \end{cases} \tag{4}$$

with the tolerance interval of $(f_i^{\min}, f_i^{\max})$.
The goal value f_i^*can be calculated by the each single objective optimization problem using GPOPS-II. Then we make a following payoff table.

Table 1: Payoff table

	$\boldsymbol{f_1}$	$\boldsymbol{f_2}$	$\boldsymbol{f_3}$	$\cdots$	$\boldsymbol{f_m}$
max or min f_1	$f_1(\mathbf{x}_1^*, \mathbf{u}_1^*)$	$f_2(\mathbf{x}_1^*, \mathbf{u}_1^*)$	$f_3(\mathbf{x}_1^*, \mathbf{u}_1^*)$	$\cdots$	$f_m(\mathbf{x}_1^*, \mathbf{u}_1^*)$
max or min f_2	$f_1(\mathbf{x}_2^*, \mathbf{u}_2^*)$	$f_2(\mathbf{x}_2^*, \mathbf{u}_2^*)$	$f_3(\mathbf{x}_2^*, \mathbf{u}_2^*)$	$\cdots$	$f_m(\mathbf{x}_2^*, \mathbf{u}_2^*)$
max or min f_3	$f_1(\mathbf{x}_3^*, \mathbf{u}_3^*)$	$f_2(\mathbf{x}_3^*, \mathbf{u}_3^*)$	$f_3(\mathbf{x}_3^*, \mathbf{u}_3^*)$	$\cdots$	$f_m(\mathbf{x}_3^*, \mathbf{u}_3^*)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$

max or min f_m	$f_1(\mathbf{x}_m^*, \mathbf{u}_m^*)$	$f_2(\mathbf{x}_m^*, \mathbf{u}_m^*)$	$f_3(\mathbf{x}_m^*, \mathbf{u}_m^*)$	$\cdots$	$f_m(\mathbf{x}_m^*, \mathbf{u}_m^*)$

- $\mathbf{x}_j^*, \mathbf{u}_j^*$, ($j$=1,…,$m$) represents the optimal solution of the i^{th} single objective optimization.

Then we define the $f_i^{\min}$ and $f_i^{\max}$ as following

$$f_i^{\max} = \max_{j=1,\ldots,m} f_i\left(\mathbf{x}_j^*, \mathbf{u}_j^*\right), \qquad i = 1, \ldots, m. \tag{5}$$

$$f_i^{\min} = \min_{j=1,\ldots,m} f_i\left(\boldsymbol{x}_j^*, \boldsymbol{u}_j^*\right), \qquad i = 1, \ldots, m. \tag{6}$$

Now suppose that $f_s(\mathbf{x}, \mathbf{u})$ requires more high priority than $f_r(\mathbf{x}, \mathbf{u})$, ($s$, r = 1,…,m, $s \neq r$), we can describe priority conditions as follows

$$\mu_{f_r}(\mathbf{x}, \mathbf{u}) - \mu_{f_s}(\mathbf{x}, \mathbf{u}) \leq \kappa, \qquad s, r \in \{1, \ldots, m\}, \qquad s \neq r \tag{7}$$

where -1 ≤ κ ≤ 1. Note that κ makes the optimization problem more feasible than problems that are highly strict to satisfy the following inequality equation.

$$\mu_{f_r}(\mathbf{x}, \mathbf{u}) \leq \mu_{f_s}(\mathbf{x}, \mathbf{u}), \qquad s, r \in \{1, \ldots, m\}, \qquad s \neq r \tag{8}$$

When κ is less than or equal to 0, it means that the relaxed order of satisfactory degrees complies with the basic priorities. Otherwise, the requirements of priorities cannot be guaranteed [8].
For multi-objective optimization, now goal programming is applied. Goal programming is widely used in the multi-objective optimization problem. The basic goal programming problem is as follows [5]

$$\begin{cases} \min & \displaystyle\sum_{i=1}^{k} (p_i + n_i) \\ \text{s.t.} & f_i(z) + n_i - p_i = f_i^*, \qquad i = 1, \ldots, k \\ & n_i, p_i \geq 0, \qquad n_i, p_i = 0 \\ & z \in G \end{cases} \tag{9}$$

where p_i and n_i are positive and negative deviational variables respectively. For the FSGP method, the objectives in Eq. (9) are transformed. For the objective transformation the three fuzzy relations are represented in the GP method.

Table 2: New formulation of objectives and membership function by fuzzy relation

Fuzzy relation	$\lesssim$	$\gtrsim$	$\cong$
Formulation using GP	$f_i(\mathbf{x}, \mathbf{u}) - p_i = f_i^*$	$f_i(\mathbf{x}, \mathbf{u}) + n_i = f_i^*$	$f_i(\boldsymbol{x}, \boldsymbol{u}) + n_i - p_i = f_i^*$
Membership function	$\mu_{f_i}(\mathbf{x}, \mathbf{u}) = 1 - \dfrac{p_i}{f_i^{\max} - f_i^*}$	$\mu_{f_i}(\mathbf{x}, \mathbf{u}) = 1 - \dfrac{n_i}{f_i^* - f_i^{\min}}$	$\mu_{f_i}(\mathbf{x}, \mathbf{u}) = \begin{cases} 1 - \dfrac{p_i}{f_i^{\max} - f_i^*} \\ 1 - \dfrac{n_i}{f_i^* - f_i^{\min}} \end{cases}$

Suppose the following case

$$\begin{cases} \text{Find} & \mathbf{x},\mathbf{u} \\ \text{so as to satisfy} & f_i(\mathbf{x},\mathbf{u}) \gtrapprox f_i^*, \qquad i = 1,\dots,k_1 \\ & f_j(\mathbf{x},\mathbf{u}) \gtrapprox f_j^*, \qquad j = k_1,\dots,k_2 \\ & f_s(\mathbf{x},\mathbf{u}) \cong f_s^*, \qquad j = k_2,\dots,k \\ \text{subject to} & f_j(\mathbf{x},\mathbf{u}) \prec f_i(\mathbf{x},\mathbf{u}), f_s(\mathbf{x},\mathbf{u}) \prec f_j(\mathbf{x},\mathbf{u}), (\mathbf{x},\mathbf{u}) \in G \end{cases} \tag{10}$$

where the partial order symbol '$\prec$' denotes the priority relation and G is system constraints. After substituting equations in Table 2 into Eq. (10), we can get the generalized FSGP model as follows

$$\begin{cases} \min & \frac{1}{m}\left[\sum_{i=1}^{m_1} \frac{p_i}{f_i^{\max} - f_i^*} + \sum_{rj=m_2+1}^{m_2} \frac{n_j}{f_j^* - f_j^{\min}} + \sum_{s=m_2+1}^{m} \left(\frac{n_s}{f_s^* - f_s^{\min}} + \frac{p_s}{f_s^{\max} - f_s^*}\right)\right] + \lambda \cdot \kappa \\ \text{s.t.} & f_i(\mathbf{x},\mathbf{u}) + n_i - p_i = f_i^*, \qquad i = 1,\dots,m_1 \\ & f_j(\mathbf{x},\mathbf{u}) + n_j - p_j = f_j^*, \qquad j = m_1+1,\dots,m_2 \\ & f_s(\mathbf{x},\mathbf{u}) + n_s - p_s = f_s^*, \qquad s = m_2+1,\dots,m \\ & p_i/(f_i^{\max} - f_i^*) - n_j/(f_j^* - f_j^{\min}) \le \kappa \\ & n_j/(f_j^* - f_j^{\min}) - (n_s/(f_s^* - f_s^{\min}) + p_s/(f_s^{\max} - f_s^*)) \le \kappa \\ & p_i \le f_i^{\max} - f_i^*, \qquad n_j \le f_j^* - f_j^{\min} \\ & n_s \le f_s^* - f_s^{\min}, \qquad p_s \le f_s^{\max} - f_s^* \\ & n_i, p_i, n_j, p_j, n_s, p_s \ge 0 \\ & n_i \cdot p_i = 0, \qquad n_j \cdot p_j = 0, \qquad n_s \cdot p_s = 0 \\ & -1 \le \kappa \le 1 \\ & (\mathbf{x},\mathbf{u}) \in G(\mathbf{x},\mathbf{u}) \end{cases} \tag{11}$$

where $n_i, p_i, n_j, p_j, n_s, p_s$ and κ are decision variables including the decision variables of the original problem $G(\mathbf{x},\mathbf{u})$. In the Eq. (11) the λ and κ reflects trade-off between optimality and priorities. λ is the regulating parameter. Increase of λ indicates the requirement of pre-emptive priorities will be reinforced. Otherwise, the deviation requirements will be weakened. If κ is more than zero, the priorities are not satisfied, then λ needs to be increased till κ is less than or equal to zero [5].

2.2 Trajectory Optimization Problem

Since the trajectory optimization problem consists of differential equations which is highly nonlinear and coupled, we cannot solve the problem directly. Generally optimal trajectories are generated using numerical methods. To solve the problem numerically, we have to transcribe the original problem into a finite dimensional NLP by parameterizing. The NLP problem is then solved using well known NLP solver. In this section, we explain a general form of trajectory optimization problem called continuous Bolza problem and Radau pseudospectral method which is for transcribe original problem into NLP.

2.2.1 Continuous Bolza Problem

A general trajectory optimization problem is posed as follows. The objective of the optimal control problem is to find the state and control that minimize or maximize an objective function subject to dynamic constraints, path constraints and boundary constraints. The general optimal control problem can be formulated in the Bolza form as follows. Minimize the cost functional [10][13]

$$J = \Phi\big(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f\big) + \int_{t_0}^{t_f} \mathbf{L}\big(\mathbf{x}(t), \mathbf{u}(t)\big)\mathrm{d}t, \qquad t \in [t_0, t_f], \tag{12}$$

subject to the constraints

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t)\big), \qquad t \in [t_0, t_f], \tag{13}$$

$$\mathbf{C}\big(\mathbf{x}(t),\mathbf{u}(t)\big) \le \mathbf{0}, \qquad t \in [t_0, t_f], \tag{14}$$

$$\phi\big(\mathbf{x}(t_0), t_0, \mathbf{x}(t_f), t_f\big) = \mathbf{0}, \tag{15}$$

where $\mathbf{x}(t) \in \mathbb{R}^{\mathrm{n}}$ is the state, $\mathbf{u}(t) \in \mathbb{R}^{\mathrm{m}}$ is the control and t is time. To simplify the problem, the optimal control problem on the time interval $t \in [t_0, t_f]$ can be divided into *K* mesh intervals, $t_0 < t_1 < t_2 < \cdots < t_K = t_f$. In each mesh interval *k*, $t \in [t_{k-1}, t_k]$, can be transformed into $\tau \in [-1,1]$ via the affine transformation

$$\tau = \frac{2t - (t_{k-1} + t_k)}{t_k - t_{k-1}} \tag{16}$$

and it is noted that

$$\frac{\mathrm{d}t}{\mathrm{d}\tau} = \frac{t_k - t_{k-1}}{2}, \qquad (k = 1, \dots, K). \tag{17}$$

Next, let $\mathbf{x}^{(k)}(\tau)$ and $\mathbf{u}^{(k)}(\tau)$ be the state and control in the k^{th} mesh interval. Then the original optimal control problem of Eqs. (12)–(15) can be reformulated in terms of the defined variable τ as follows. Minimize the cost functional

$$J = \Phi\big(\mathbf{x}^{(1)}(-1), t_0, \mathbf{x}^{(K)}(1), t_K\big) + \sum_{k=1}^{K} \frac{t_k - t_{k-1}}{2} \int_{-1}^{1} \mathbf{L}\big(\mathbf{x}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), \tau; t_{k-1}, t_k\big)\mathrm{d}\tau, \tag{18}$$

subject to the constraints

$$\frac{\mathrm{d}\mathbf{x}^{(k)}(\tau)}{\mathrm{d}\tau} = \frac{t_k - t_{k-1}}{2}\mathbf{f}\big(\mathbf{x}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), \tau; t_{k-1}, t_k\big), \qquad (k = 1, \dots, K), \tag{19}$$

$$\frac{t_k - t_{k-1}}{2}\mathbf{C}\big(\mathbf{x}^{(k)}(\tau), \mathbf{u}^{(k)}(\tau), \tau; t_{k-1}, t_k\big) \le \mathbf{0}, \qquad (k = 1, \dots, K), \tag{20}$$

$$\phi\big(\mathbf{x}^{(1)}(-1), t_0, \mathbf{x}^{(K)}(1), t_K\big) = \mathbf{0}, \tag{21}$$

and the interior point constraints

$$\mathbf{x}^{(k)}(1) - \mathbf{x}^{(K+1)}(-1) = \mathbf{0}, \qquad (k = 1, \dots, K - 1). \tag{22}$$

2.2.2 Radau Pseudospectral Method

Now we describe parameterization of the optimal control problem. In this paper, a general purpose optimal control software GPOPS-II developed at University of Florida is used for solving skip trajectory optimal control problem. RPM that is one of advanced variable order Gaussian quadrature collocation methods is implemented in GPOPS-II. In a RPM approximation of an optimal control problem, the infinite-dimensional optimal control problem is converted into a finite-dimensional NLP problem. In GPOPS-II it is possible to vary both the number or mesh interval and the degree of the approximating polynomial within each mesh interval. RPM uses the Lagrange polynomial approximation at a set of discrete Legendre-Gauss-Radau (LGR) collocation points. In each phase, the state can be approximated by the following polynomial

$$\mathbf{x}^{(k)}(\tau) \approx \mathbf{X}^{(k)}(\tau) = \sum_{j=0}^{N_k} \mathbf{X}_j^{(k)} L_j^{(k)}(\tau) \tag{23}$$

where $L_j^{(k)}(\tau)$ are defined as

$$L_j^{(k)}(\boldsymbol{\tau}) = \prod_{i=0, i\neq j}^{\boldsymbol{N_k}} \frac{\tau - \tau_i^{(k)}}{\tau_j^{(k)} - \tau_i^{(k)}}, j = 0,1, \dots, N_k \tag{24}$$

where $\tau \in [-1,1]$, $(\tau_1^{(k)}, \dots, \tau_{N_k}^{(k)})$ are the LGR collocation points in the k^{th} mesh interval and the final point $\tau_{N_k}^{(k)}$ is a non-collocation point. The control can be also approximated similarly as follows

$$\mathbf{u}^{(k)}(\tau) \approx \mathbf{U}^{(k)}(\tau) = \sum_{j=1}^{N_k} \mathbf{U}_j^{(k)} \tilde{L}_j^{(k)}(\tau) \tag{25}$$

where $\tilde{L}_j^{(k)}(\tau)$ are defined as

$$\tilde{L}_j^{(k)}(\boldsymbol{\tau}) = \prod_{i=1, i\neq j}^{\boldsymbol{N_k}} \frac{\tau - \tau_i^{(k)}}{\tau_j^{(k)} - \tau_i^{(k)}}, j = 1, \dots, N_k \tag{26}$$

The Lagrange polynomials Eq. (24) and Eq. (26) have the property

$$L_j^{(k)}\left(\tau_i^{(k)}\right) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \qquad \tilde{L}_j^{(k)}\left(\tau_i^{(k)}\right) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \tag{27}$$

Differentiating the expression in Eq. (23) with respect to τ produces

$$\frac{\mathrm{d}\mathbf{X}^{(k)}(\tau)}{\mathrm{d}\tau} \equiv \dot{\mathbf{X}}^{(k)}(\tau) = \sum_{j=0}^{N_k} \mathbf{X}_j^{(k)} \dot{L}_j^{(k)}(\tau) = \sum_{j=0}^{N_k} \mathbf{X}_j^{(k)} \boldsymbol{D}_{ij}^{(k)} \tag{28}$$

$$\boldsymbol{D}_{\boldsymbol{ij}}^{(\boldsymbol{k})} = \dot{\boldsymbol{L}}_{\boldsymbol{j}}^{(\boldsymbol{k})}\left(\tau_i^{(k)}\right), \qquad i = 1, \dots, N_k, j = 0,1, \dots, N_k \tag{29}$$

The dynamic constraints that are transcribed into algebraic constraints via the differential approximation matrix are as follows

$$\sum_{j=0}^{N_k} \boldsymbol{D}_{ij}^{(k)} \mathbf{X}_j^{(k)} = \frac{t_k - t_{k-1}}{2} \mathbf{f}\left(\mathbf{X}_i^{(k)}, \mathbf{U}_i^{(k)}, \mathrm{\tau}_i^{(k)}; t_{k-1}, t_k\right), \qquad i = 1, \dots, N_k \tag{30}$$

Furthermore, the inequality path constraints can be evaluated at the N_k LGR points in each mesh interval as follows

$$\mathbf{C}(\mathbf{X}_i^{(k)}, \mathbf{U}_i^{(k)}, \mathrm{\tau}_i^{(k)}, t_{k-1}, t_k) \leq 0, \qquad i = 1, \dots, N_k \tag{31}$$

Finally, the boundary condition can be rewritten at the N_k LGR points in each mesh interval as follows

$$\phi\left(\mathbf{X}_0^{(1)}, t_0, \mathbf{X}_{N_k}^{(K)}, t_K\right) = \mathbf{0} \tag{32}$$

The continuous-time cost functional can be constituted by the multi-interval at LGR points, resulting in

$$J \approx= \Phi\left(\mathbf{X}_0^{(1)}, t_0, \mathbf{X}_{N_k}^{(K)}, t_K\right) + \sum_{k=1}^{K} \sum_{j=0}^{N_k} \left(\frac{t_k - t_{k-1}}{2}\right) \omega_j \, \mathbf{L}\left(\mathbf{X}_j^{(k)}, \mathbf{U}_j^{(k)}, \mathrm{\tau}_j^{(k)}; t_{k-1}, t_k\right), \tag{33}$$

where ω_j represent the LGR weights. The NLP problem that arise from the RPM is then to minimize the objective function of Eq. (33) subject to the algebraic constraints of Eqs. (28) – (32). And then the existing nonlinear programming solver such as IPOPT [14] and SNOPT [15] can be applied to the NLP problem with *hp*-adaptive methods [10]. We leave out the details here.
Considering the whole step of the multi-objective skip trajectory optimization with priorities, the procedure can be summarized as follows
Step 1. Solve the each single-objective optimization problem.
Step 2. Problem formulation according to Eq. (11) based on the FSGP method.
Step 3. Solve the optimization problem that is reformulated in **Step 2** using GPOPS-II.
Step 4. If $\kappa > 0$, then go to **Step 2** and increase λ until $\kappa \leq 0$. If $\kappa \leq 0$, then stop.

2.3 Skip Trajectory Optimization Problem

We study the skip trajectory problem of hypersonic reentry vehicle. The mission of the skip trajectoty is to reach a specific altitude and back into thermosphere over the flight. The mission is considered as hard constraints. The objective is to determine the trajectory that is to maximize velocity and mass at the terminal point because to preserve kinetic energy and fuel makes vehicle possible to have more skipping. Now the dynamic constraints, path constraints, boundary constraints and objective functions are described.

2.3.1 Dynamic Constraints

We uses the physical characteristics of the NASA space shuttle [3][1]. The following hypothesis for the skip trajectory optimization are presented; 1) The earth is a uniform sphere and flat 2) Rotational dynamics are ignored 3) The vehicle is considered as a point 4) The sideslip angle is kept to zero 5) Drag and lift coefficients are only functions of the angle of attack. 6) The thrust model is a simple 1st order system. The reentry trajectory is calculated by three degree of freedom point-mass dynamics that are described by the following set of differential algebraic equations:

$$\dot{r} = v\, sin\, \gamma \tag{34}$$

$$\dot{\theta} = \frac{(v \sin\psi \cos\gamma)}{r\cos\phi} \tag{35}$$

$$\dot{\phi} = \frac{(v \cos\psi \cos\gamma)}{r} \tag{36}$$

$$\dot{v} = \frac{T\cos\alpha - D}{m} - g\sin\gamma \tag{37}$$

$$\dot{\gamma} = \frac{L\cos\sigma + T\sin\alpha}{mv} - \frac{g\cos\gamma}{v} + \frac{v\cos\gamma}{r} \tag{38}$$

$$\dot{\psi} = \frac{L\sin\sigma}{mv\cos\gamma} + \frac{v\cos\gamma\sin\psi\tan\phi}{r} \tag{39}$$

$$\dot{m} = -\frac{T}{\mathrm{I_{SP}}g} \tag{40}$$

where the state is $x = [r, \theta, \phi, v, \gamma, \psi, m]^T$, which are representing radial position (m), longitude (degree), latitude (degree), velocity (m/s), flight path angle (degree), heading angle (degree), and mass (kg) respectively. The control is angle of attack α (degree), bank angle σ (degree) and thrust T (N). Decision variables are angle of attack, bank angle and thrust. Here, D is the aerodynamic drag force, L is the aerodynamic lift force, $\mathrm{I_{SP}}$ is the specific impulse, g is the acceleration due to gravity at the Earth's surface.

$$g = \frac{\mu}{r^2} \tag{41}$$

$$L = 0.5\rho v^2 C_L S \tag{42}$$

$$D = 0.5\rho v^2 C_D S \tag{43}$$

where μ = 3.9856×10^{14} m^3/ s^2 is the gravitational parameter of the Earth, S = 249.91 m^2 is a reference area. ρ is the local atmospheric density with , C_L is the coefficient of lift and C_D is the coefficient of drag respectively.

$$\rho = \rho_0 \mathrm{e}^{-h/\mathrm{H}} \tag{44}$$

$$C_L = C_{L0} + C_{L1}\alpha \tag{45}$$

$$C_D = C_{D0} + C_{D1}\alpha + C_{D1}\alpha^2 \tag{46}$$

where ρ_0 = 1.2256 kg/m^3, H = 7254.24 m, C_{L0} = -0.2070, C_{L1} = 1.676, C_{D0} = 0.07854, C_{D1} = -0.3529, C_{D2} = 2.040 with α, the angle of attack, in radians [3].

2.3.2 Boundary Conditions and Path Constraints

We consider the typical reentry path constraints for the thermal, structure, and operational consideration include, such as heat flux constraint $\dot{Q}$, dynamic pressure constraint q and normal load factor n.

$$\dot{Q} = (h_0 + h_1\alpha + h_2\alpha^2 + h_3\alpha^3) \cdot C\rho^{0.5}v^{3.07} \le \dot{Q}_{\max} \tag{47}$$

$$q = 0.5\rho v^2 \le q_{\max} \tag{48}$$

$$n = \frac{\sqrt{L^2 + D^2}}{mg} \le n_{\max} \tag{49}$$

where h_0 = 1.0672181, h_1 = -1.9213774×10^{-2}, h_2 =2.1286289×10^{-4}, h_3 = -1.0117249×10^{-6}, C is coefficient of . The maximum value of each path constraints are $\dot{Q}_{\max}$ = 2269.7864 kW/m^2, $q_{\max}$ = 13406.4583 Pa and $n_{\max}$ = 2.5 [3].
In addition, we define the specific altitude constraints to carry out the mission which is to reach a specific altitude at specific time t_c and back into the thermosphere.

$$\begin{cases} r(t_c) = 51.880 \text{ km}, & t_0 < t_c < t_f \\ r(t_c) \le r(t) \le r(t_f), & \forall\, t \\ r(t_f) = 79.248 \text{ km} \end{cases} \tag{50}$$

The initial, terminal, minimum and maximum value of the state and control for skip trajectory optimization are describe as follows

Table 3: Equality and inequality constraints of state and control

	Initial time t_0	**Specific time t_c**	**Final time t_f**	**Minimum value**	**Maximum value**
r (m)	$r_e + 79\,248$	$r_e + 51\,880$	$r_e + 79\,248$	$r_e + 51\,880$	$r_e + 79\,248$
θ (deg)	0	free	free	-180	180
ϕ (deg)	0	free	free	-89	89
v (m/s)	7 802.88	free	free	1	Free
ψ (deg)	90	free	free	-180	180

γ (deg)	-1	free	free	-80	80
m (kg)	90 719	free	free	3 177	90 719
α (deg)	17.42	free	free	-10	30
σ (deg)	-75	free	free	-80	80
T (N)	0	free	free	0	2×10^6

- r_e= 6371203.92 m is radius of the Earth.

In Table 3 shows the equality and inequality constraints of state and control of the vehicle. Particularly specific time t_c ($t_0 < t_c < t_f$) is a constraints for reaching the altitude $r(t_c)$.

2.3.3 Optimization Objectives

Kinetic energy and thrust are important variables for skip trajectory optimization. Because to preserve the kinetic energy and fuel at the terminal point helps to keep more skipping within limited fuel quantity. Therefore the objective function is maximizing velocity and mass at the terminal point.

1) Maximizing the terminal mass

$$\max f_1 = m(t_f) \tag{51}$$

2) Maximizing the terminal velocity

$$\max f_2 = v\ (t_f) \tag{52}$$

We define the priority of objectives in accordance with the importance of the objectives. In this paper, we give high priority to the maximizing the terminal mass.

3. Numerical Simulation

In this section, we present the simulation results of the proposed multi-objective skip trajectory optimization problem. We formulate multi-objective problem to single-objective problem using FSGP method. Then the single-objective problem is solved using GPOPS-II [9]. GPOPS-II is a software that transcribe the infinite dimensional optimal control problem into finite dimensional NLP problem. Firstly, we solve the each optimal control problem using GPOPS-II and make a payoff table. Second, we reformulate the problem using FSGP method. Finally, we solve the reformulated problem using GPOPS-II.

3.1 Single-Objective Problem Simulation

Maximizing the terminal velocity and mass are used as object function. Constraints and parameters and of simulations are in section 2. To apply FSGP method we solve the each single objective skip trajectory optimization problem. Figure 1~2 show the each single objective optimization simulation results. In case of maximizing the terminal mass, the terminal velocity is relatively lower than that in case of maximizing the terminal velocity. In contrast maximizing the terminal velocity have lower mass at the terminal point. The simulation results indicate that two objective functions conflict with each other. So it is hard to find a solution which is satisfying two objective function.

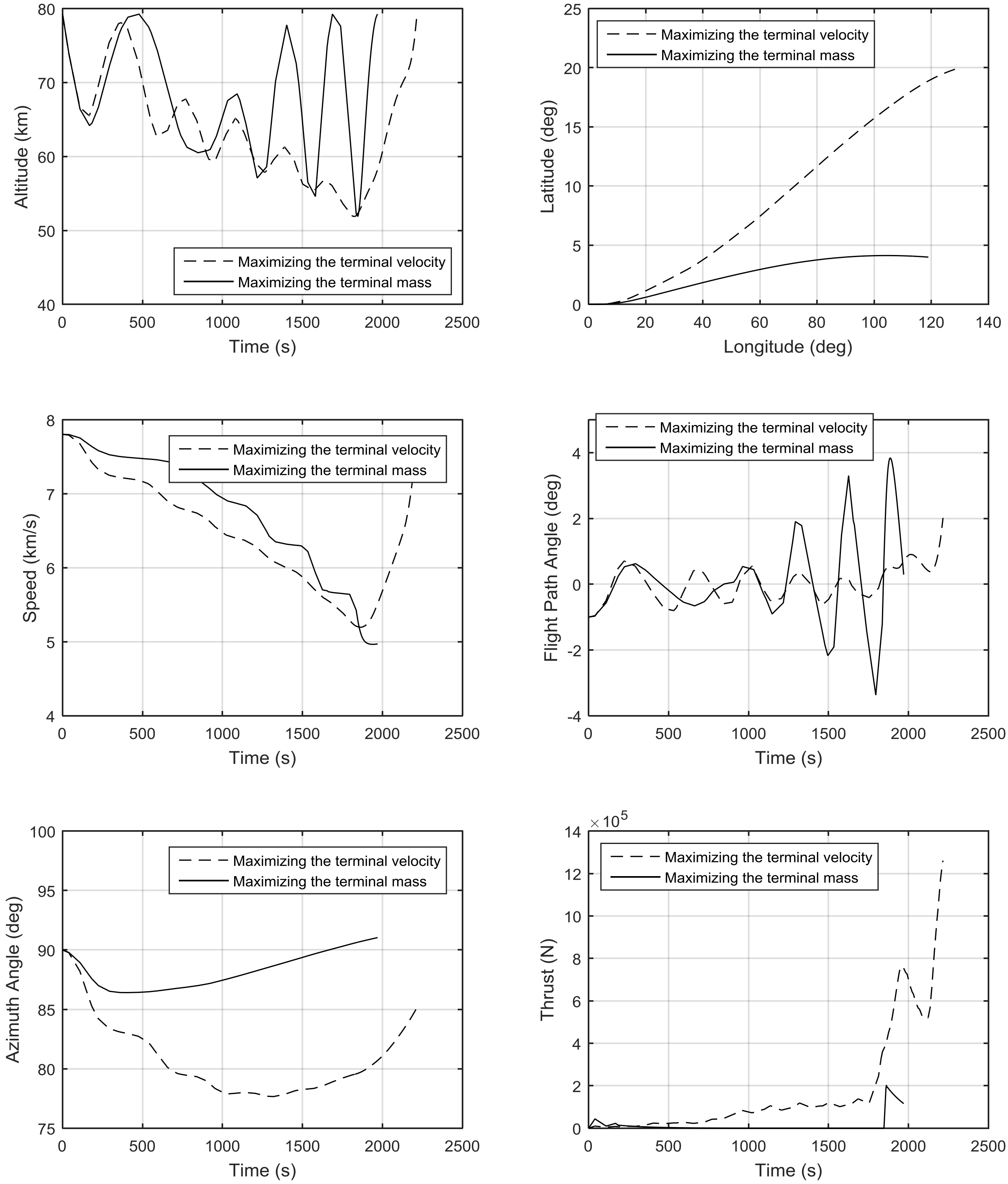

Figure 1: Single objective optimization result 1

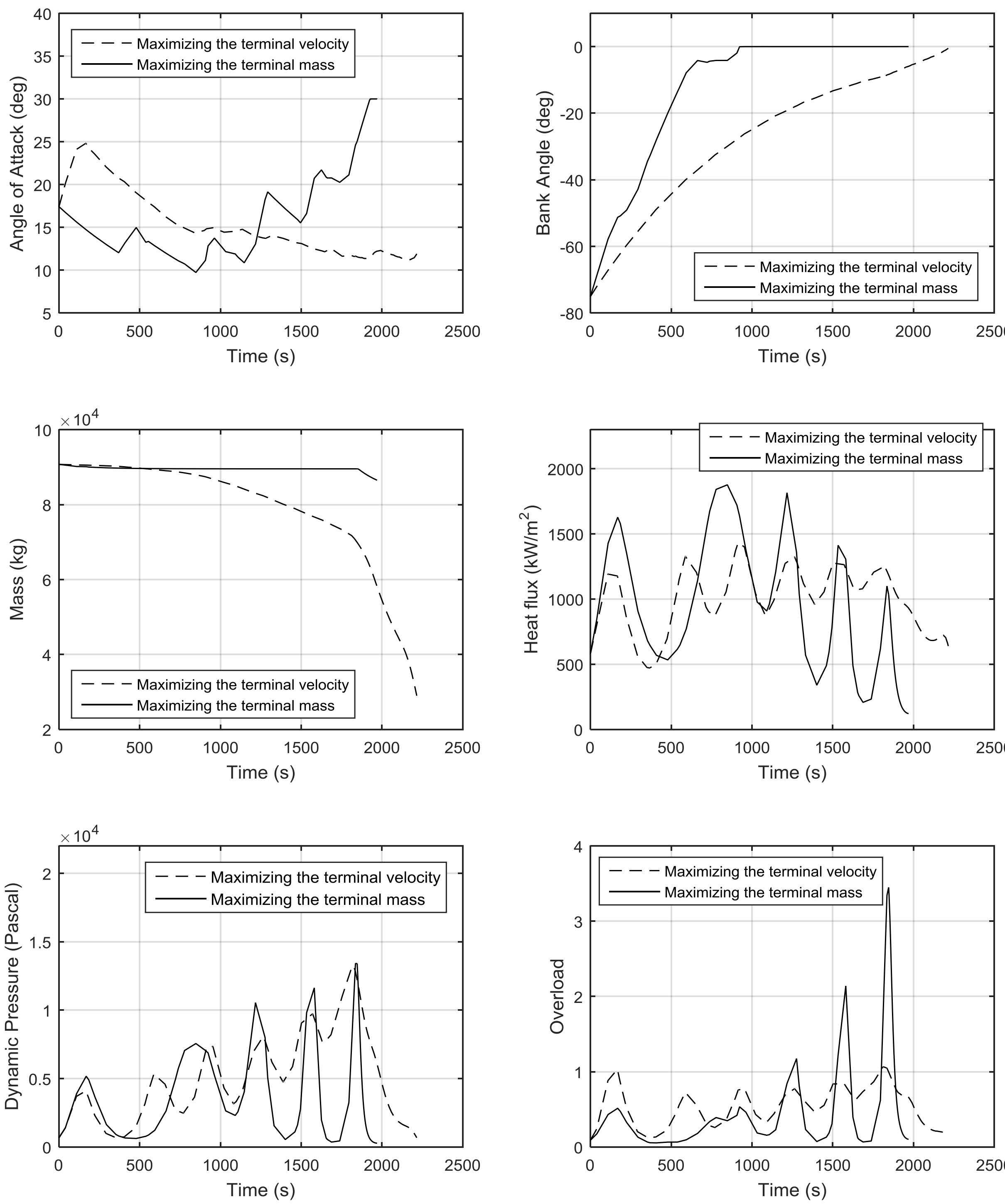

Figure 2: Single objective optimization result 2

3.1 Multi-Objective Problem Simulation

From the single-objective simulation results, we make a payoff Table 4. Now we transform multi-objective problem into single-objective problem with high priority of maximizing the terminal velocity using FSGP method based on Table 4.

Table 4: Payoff table

	$\boldsymbol{f}_{\text{mass}}$	$\boldsymbol{f}_{\text{vel}}$
max f_{mass}	86502	4970
max f_{vel}	28971	7803

$$\begin{cases} \min & \frac{1}{m}\left[\frac{n_1}{86502-28971}+\frac{n_2}{7803-4970}\right]+\lambda\cdot\kappa \\ \text{s.t.} & m(t_f)+n_1-p_1=86502 \\ & v(t_f)+n_2-p_2=7803 \\ & \frac{n_1}{86502-28971}-\frac{n_2}{7803-4970}\leq\kappa \\ & n_1\leq 86502-28971 \\ & n_2\leq 7803-4970 \\ & n_1,p_1,n_2,p_2\geq 0 \\ & n_1\cdot p_1=0,\qquad n_2\cdot p_2=0 \\ & -1\leq\kappa\leq 1 \\ & (\mathbf{x},\mathbf{u})\in G \end{cases} \tag{53}$$

where n_1, p_1, n_2, p_2 and κ are additional decision variables. λ is the regulating parameter that can trade-off between optimality and priority. Increasing λ strengthen the priority. κ reflects trade-off between optimality and priorities. If κ is less than or equal to zero, priority is satisfied. On the other hand, if κ is more than zero, priority is not satisfied. So, we increase λ until κ is less than or equal to zero.

After solving Eq. (53) with regulating parameter λ, we can get the κ and satisfactory degree for each objective functions as represented in Table 5. When $\lambda = 0.1$, the velocity satisfactory degree is higher than mass satisfactory degree and κ is bigger than zero, so the priorities are not met. We increase λ until κ is less than or equal to zero. In case of $\lambda = 0.2$ and $\lambda = 0.3$ the order of priority is satisfied in accordance with κ. In particular, the satisfactory degree of objective function of maximizing the terminal velocity can always be optimized. So user should find marginal λ which satisfy the priority. The corresponding simulation results for each λ are shown in Figure 3~4.

Mass satisfactory degree satisfies the perspective goal value for all λ which means this objective can always be optimized enough with this approach. Speed of the vehicle shown in Figure 3 decreases slightly with λ because increasing λ improve priority. The path constraints shown in Figure 4 are satisfying the range of permission.

Table 5: Satisfactory degree and $\boldsymbol{\kappa}$ by the $\boldsymbol{\lambda}$

$\boldsymbol{\lambda}$	$\boldsymbol{\kappa}$	$\boldsymbol{\mu}_{\text{mass}}$	$\boldsymbol{\mu}_{\text{vel}}$
0.1	0.1097	0.2695	0.3791
0.2	-0.6989	1.0000	0.3011
0.3	-0.7044	1.0000	0.2956

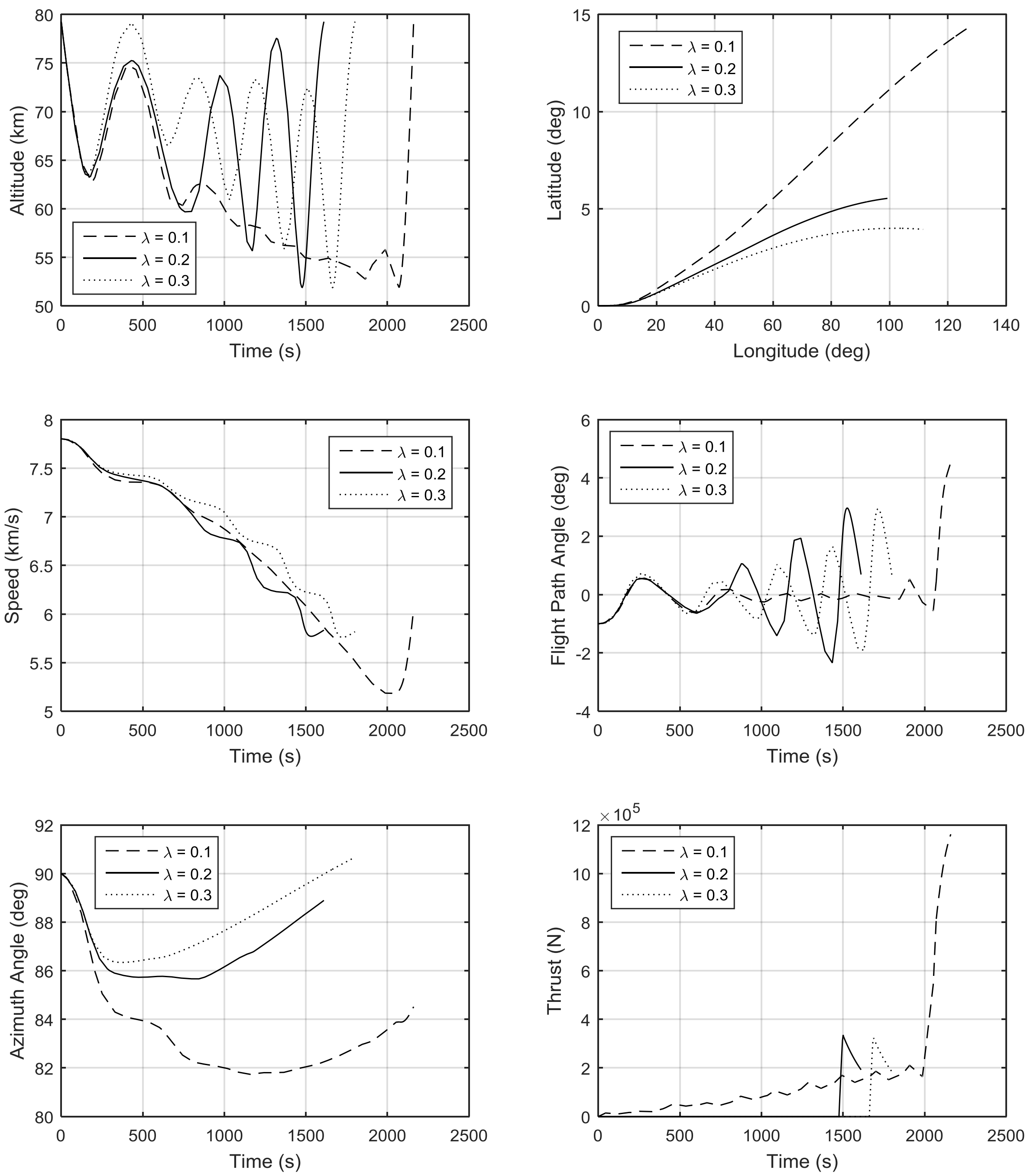

Figure 3: Multi-objective optimization result 1

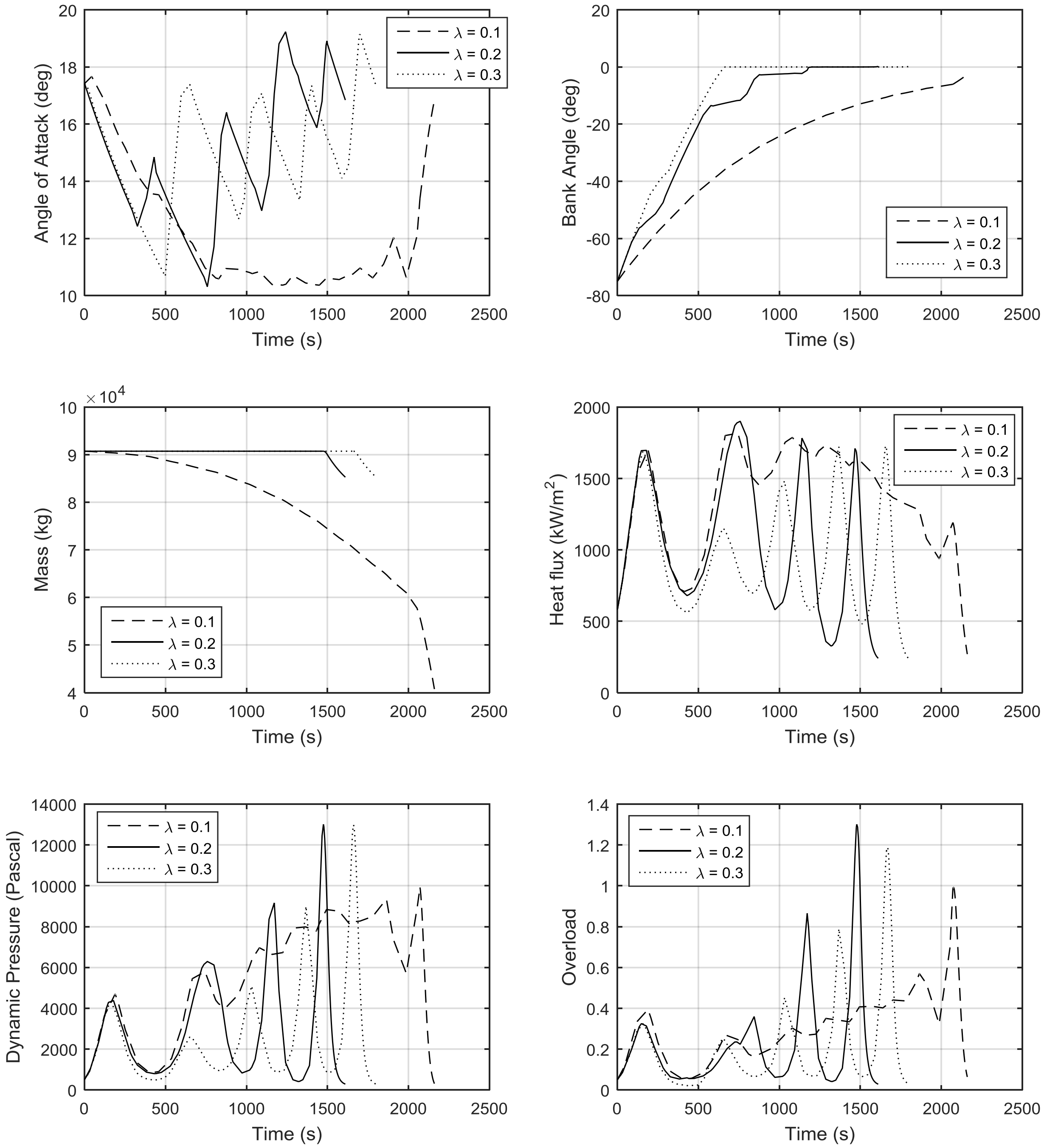

Figure 4: Multi-objective optimization result 2

4. Conclusion

In this paper, we demonstrate the skip trajectory optimization problem. Particularly, we set the specific altitude as a constraint to perform a mission during the skip trajectory. Maximizing terminal mass and velocity are considered as objective functions since kinetic energy and fuel consumption are the most important part. We give high priority to the maximizing terminal mass. To solve the multi-objective optimization we reformulate the problem using FSGP method, then use GPOPS-II because GPOPS-II implement pseudo-spectral method that has the co-state which can be mapped from the Karush-Kuhn-Tucker multiplier of the NLP problem. The key features of the results show that relationship between two objective function are very sensitive and conflicts each other. Future work includes improving the thrust and specifying the mission. For example, to specify waypoint and flight path angle to take a picture of specific area.

References

[1] T. Bauer, J. Betts, W. Hallman, W. Huffman, and K. Zondervan, "Solving the optimal control problem using a nonlinear programming technique. II - Optimal Shuttle ascent trajectories," *Astrodynamics Conference*. 1984.

[2] P. Lu, "Entry Guidance: A Unified Method," *J. Guid. Control. Dyn.*, vol. 37, no. 3, pp. 713–728, 2014.

[3] B. Tian and Q. Zong, "Optimal guidance for reentry vehicles based on indirect Legendre pseudospectral method," *Acta Astronaut.*, vol. 68, no. 7–8, pp. 1176–1184, 2011.

[4] R. T. Marler and J. S. Arora, "Survey of multi-objective optimization methods for engineering," *Struct. Multidiscip. Optim.*, vol. 26, no. 6, pp. 369–395, Apr. 2004.

[5] C. Hu and S. Li, "Enhanced Interactive Satisfying Optimization Approach To Multiple Objective Optimization With Preemptive Priorities," *Int. J. Inf. Technol. Decis. Mak.*, vol. 05, no. 1, pp. 47–63, 2006.

[6] J. T. Betts, "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21. pp. 193–207, 1998.

[7] D. Garg, W. W. Hager, and A. V. Rao, "Pseudospectral methods for solving infinite-horizon optimal control problems," *Automatica*, vol. 47, no. 4, pp. 829–837, 2011.

[8] P. Williams, "Application of Pseudospectral Methods for Receding Horizon Control," *J. Guid. Control. Dyn.*, vol. 27, no. 2, pp. 310–314, 2004.

[9] M. A. Patterson and A. V. Rao, "GPOPS-II," *ACM Trans. Math. Softw.*, vol. 41, no. 1, pp. 1–37, Oct. 2014.

[10] A. V. Rao, D. a. Benson, C. Darby, M. a. Patterson, C. Francolin, I. Sanders, and G. T. Huntington, "Algorithm 902," *ACM Trans. Math. Softw.*, vol. 37, no. 2, pp. 1–39, 2010.

[11] B. Re and Z. La, "Decision-Making in a Fuzzy Environment," *Management Science*, vol. 17, no. 4. 1970.

[12] C.-F. Hu and Y. Xin, "Reentry trajectory optimization for hypersonic vehicles using fuzzy satisfactory goal programming method," *Int. J. Autom. Comput.*, vol. 12, no. 2, pp. 171–181, 2014.

[13] H. Waschl, I. Kolmanovsky, M. Steinbuch, and L. del Re, *Optimization and Optimal Control in Automotive Systems*. 2014.

[14] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, Apr. 2005.

[15] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM J. Optim.*, vol. 12, no. 4, pp. 979–1006, Jan. 2002.