COMP9602 Assignment 3

Zi-Shen Li *[1]

[1]*Quantum Information and Computation Initiative, Department of Computer Science, The University of Hong Kong, Hong Kong, China*

December 6, 2023

1 Analysis

The target of the problem is to find the best $\vec{a} = (a_0, a_1, a_2, \cdots, a_{T-1})$ to minimize the energy use of the whole process. Thus the natrual choice of the original optimization problem is to minimize the energy use of each time step, i.e.

$$E(\vec{a}) = \sum_{i=0}^{T-1} \phi(a_i), \tag{1.1}$$

where $\phi : \mathbb{R}_+ \mapsto \mathbb{R}_+$ is the energy profile function of a single time step. In the problem the function ϕ is defined to be a convex, increasing function, which also makes the function $E : \mathbb{R}_+^T \mapsto \mathbb{R}_+$ a convex function because for any $p \in [0, 1]$,

$$\begin{aligned} E\left[p\vec{a} + (1-p)\vec{b}\right] &= \sum_{i=0}^{T-1} \phi[pa_i + (1-p)b_i] && (1.2) \\ &\leq p\sum_{i=0}^{T-1} \phi(a_i) + (1-p)\sum_{i=0}^{T-1} \phi(b_i) && (1.3) \\ &= pE(\vec{a}) + (1-p)E(\vec{b}), && (1.4) \end{aligned}$$

where from Equation (1.2) to Equation (1.3) we have used the convexity of function ϕ.

The technical difficulty of the problem is its constraint. First, we directly write down the constraint in the original form

$$\begin{cases} a_i \in [0, a_{\max}], & \forall i \in \{0, 1, \cdots, T-1\}, \\ v_{i+1} = v_i + a_i - g, & \forall i \in \{0, 1, \cdots, T-1\}, \\ y_{i+1} = y_i + v_i, & \forall i \in \{0, 1, \cdots, T-1\}, \\ y_t \in [l_t, h_t], & \forall t \in \{0, 1, \cdots, T\}, \end{cases} \tag{1.5}$$

where g is the acceleration of gravity and $a_{\max}$ is the maximum vertical acceleration of the drone. By utilizing Equation (1.5), we can rewrite the formula for y_t explicitly in v_i

$$\begin{aligned} y_0 &= y_{\text{init}}, \\ y_t &= y_{t-1} + v_{t-1}, \\ y_{t-1} &= y_{t-2} + v_{t-2}, \\ &\cdots \\ y_1 &= y_0 + v_0, \end{aligned} \tag{1.6}$$

*zishen@connect.hku.hk

which can be further rewritten as

$$y_t = y_0 + \sum_{i=0}^{t-1} v_i,\ \forall t \in \{0, 1, \cdots, T\}. \tag{1.7}$$

In addition, we can also rewrite the formula for v_t explicitly in a_i

$$\begin{aligned}
v_t &= v_{t-1} + a_t - g, \\
v_{t-1} &= v_{t-2} + a_{t-1} - g, \\
&\cdots \\
v_1 &= v_0 + a_1 - g, \\
v_0 &= v_{-1} + a_0 - g, \\
v_{-1} &= 0,
\end{aligned} \tag{1.8}$$

which can be further rewritten as

$$v_t = \sum_{i=0}^{t} a_i - gt,\ \forall t \in \{0, 1, \cdots, T-1\}. \tag{1.9}$$

Combining Equations (1.7) and (1.9), we can rewrite y_t explicitly in a_i

$$y_t = y_0 + \sum_{i=0}^{t-1} \left(\sum_{j=0}^{i} a_j - gi \right),\ \forall t \in \{1, \cdots, T\}. \tag{1.10}$$

More specifically, for $t \in \{1, \cdots, T\}$, we have

$$y_t = y_0 + \sum \left\{ \begin{array}{l} a_0 \\ a_0 + a_1 - g \\ a_0 + a_1 + a_2 - 2g \\ \cdots \\ a_0 + a_1 + \cdots + a_{t-1} - (t-1)g \end{array} \right. . \tag{1.11}$$

The above formula can be further simplified as

$$\begin{aligned}
y_t &= y_0 + t a_0 + (t-1) a_1 + \cdots + a_{t-1} - \frac{t(t-1)}{2} g && (1.12) \\
&= y_0 + \sum_{j=0}^{t-1} (t-j) a_j - \frac{t(t-1)}{2} g,\ \forall t \in \{1, \cdots, T\}. && (1.13)
\end{aligned}$$

For any given y_t, we can rewrite Equation (1.13) in the form of linear map, i.e.

$$y_t = \vec{c}_t^{\,T} \vec{a} + b_t, \tag{1.14}$$

where $\vec{c}_t \in \mathbb{R}^T$ is a column vector with the $(j+1)$-th entry being $(t-j)$, $\forall j < t$ and 0 otherwise, and the second term b_t is equal to $y_0 - \frac{t(t-1)}{2} g$. Furthermore, we can vectorize Equation (1.14) to obtain

$$\vec{y} = C\vec{a} + \vec{b}, \tag{1.15}$$

where $\vec{y} = (y_1, y_2, \cdots, y_T)^T$, and C is a $T \times T$ matrix with the t-th row being $\vec{c}_t^{\,T}$. Note that the matrix C is a lower triangular matrix with all diagonal entries being 1. Thus the matrix C is invertible and its inverse matrix is also a lower triangular matrix with all diagonal entries being 1. Therefore, we can rewrite Equation (1.15) as

$$\vec{a} = C^{-1}(\vec{y} - \vec{b}). \tag{1.16}$$

The constraint in Equation (1.5) can be rewritten as

$$\begin{cases} 0 \preceq \vec{a} \preceq a_{\max}\mathbb{1}, \; \mathbb{1} := (1,1,1,\cdots,1)^T, \\ \vec{l} \preceq C\vec{a} + \vec{b} \preceq \vec{h}, \end{cases} \tag{1.17}$$

where $\preceq$ means element-wise inequality. $\vec{l}$ and $\vec{h}$ are lower and upper bounds of the height of the drone, respectively.

2 Solution

(a)

The original optimization problem can be rewritten as

$$\min_{\vec{a}} E(\vec{a}) = \sum_{i=0}^{T-1} \phi(a_i), \tag{2.1}$$

$$\text{s.t. } 0 \preceq \vec{a} \preceq a_{\max}\mathbb{1}, \tag{2.2}$$

$$\vec{l} \preceq C\vec{a} + \vec{b} \preceq \vec{h}. \tag{2.3}$$

Or equivalently, we can rewrite Equation (2.3) as

$$\vec{l} - \vec{b} \preceq C\vec{a} \preceq \vec{h} - \vec{b}. \tag{2.4}$$

We can show that the above constraints are convex. Considering $\vec{a}_1, \vec{a}_2 \in \mathbb{R}^T$ and $p \in \mathbb{R}$, we have

$$C(p\vec{a}_1 + (1-p)\vec{a}_2) = pC\vec{a}_1 + (1-p)C\vec{a}_2, \tag{2.5}$$

and we assume a_1, a_2 satisfy the constraints in Equation (2.4), i.e., $\vec{l}_1 - \vec{b} \preceq C\vec{a}_1 \preceq \vec{h}_1 - \vec{b}$ and $\vec{l}_2 - \vec{b} \preceq C\vec{a}_2 \preceq \vec{h}_2 - \vec{b}$. It is easy to see that the convex combination of $\vec{a}_1$ and $\vec{a}_2$ also satisfies the constraints, i.e.,

$$\vec{l}_1 - \vec{b} \preceq C(p\vec{a}_1 + (1-p)\vec{a}_2) \preceq \vec{h}_1 - \vec{b}. \tag{2.6}$$

The result still holds for the constraint in Equation (2.2) for the same reason.

As we shown in Section 1, the function $E(\vec{a})$ is convex. Thus we can conclude the optimization problem is a convex problem.

(b)

The energy profile ϕ is given by

$$\phi(a) = 1 + a + a^2 + a^3. \tag{2.7}$$

We aim to solve the optimization problem with ellipsoid method. The objective function can be written as

$$E(\vec{a}) = \sum_{i=0}^{T-1} \phi(a_i) = T + \sum_{i=0}^{T-1} a_i + \sum_{i=0}^{T-1} a_i^2 + \sum_{i=0}^{T-1} a_i^3, \tag{2.8}$$

which are differentiable. Thus the subgradient of $E(\vec{a})$ is the gradient of $E(\vec{a})$, i.e.,

$$\frac{\partial E}{\partial a_i} = 1 + 2a_i + 3a_i^2. \tag{2.9}$$

The gradient of the constrain in Equation (2.4) is also differentiable, i.e.,

$$\nabla_{\vec{a}}(C\vec{a}) = C^T. \tag{2.10}$$

For each constrain in Equation (1.14), we have $\forall t \in \{1, \cdots, T\}$,

$$f_{1t} = \vec{c}_t^T \vec{a} + b_t - h_t \le 0, \tag{2.11}$$

$$f_{2t} = l_t - \vec{c}_t^T \vec{a} - b_t \le 0, \tag{2.12}$$

where $b_t = y_0 - \frac{t(t-1)}{2} g$, and additionally, we have

$$f_{3t} = -a_{\max} + a_t \le 0, \tag{2.13}$$

$$f_{4t} = -a_t \le 0. \tag{2.14}$$

The gradient of the above two sets of constrains are

$$\nabla_{\vec{a}} f_{1t} = \vec{c}_t, \tag{2.15}$$

$$\nabla_{\vec{a}} f_{2t} = -\vec{c}_t, \tag{2.16}$$

$$\nabla_{\vec{a}} f_{3t} = \vec{e}_t, \tag{2.17}$$

$$\nabla_{\vec{a}} f_{4t} = -\vec{e}_t, \tag{2.18}$$

where $\vec{c}_t = (t, t-1, \cdots, 1, 0, \cdots, 0)^T$ is a column vector with the first t entries being $t, t-1, \cdots, 1$ and the rest being 0, and $\vec{e}_t$ is a column vector with the t-th entry being 1 and the rest being 0.

2.1 (c)

The codes used to solve the problem are provided in the link .

Instance 1

The first instance is a trivial case where the optimal solution is letting the drone turn off its engine during the whole process. The result and required informations are provided in Figure 1. The optimal objective function value provided by the programe is (keep 12 significant digits)

$$E(\vec{a}^*) \approx 2.00054077480. \tag{2.19}$$

Instance 2

The second instance is a more complicated case where the drone needs to tuning its engine to manipulate its height. The result and required informations are provided in Figure 2. The optimal objective function value provided by the programe is (keep 12 significant digits)

$$E(\vec{a}^*) \approx 35951.8168589. \tag{2.20}$$

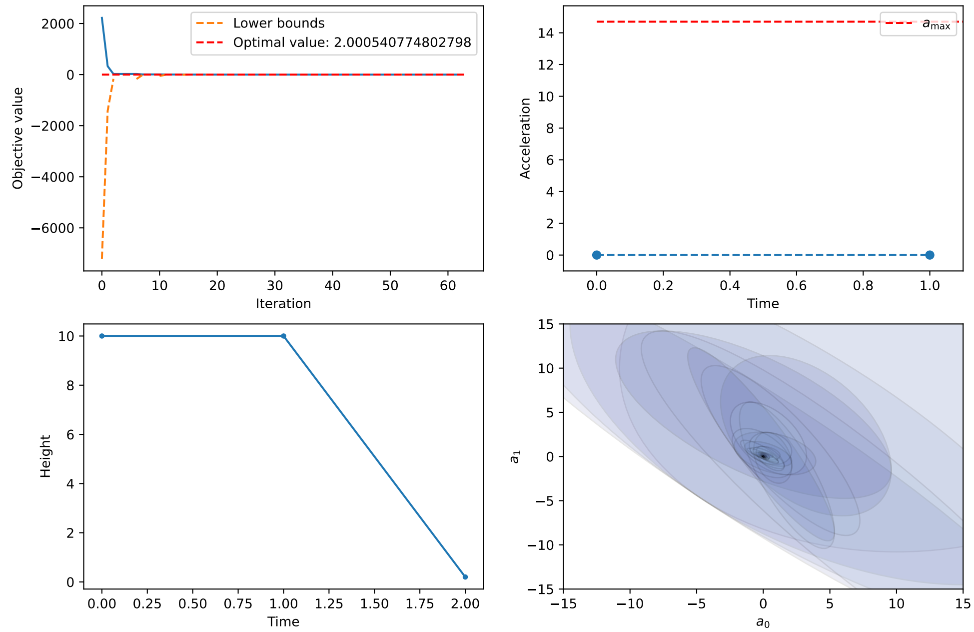

Figure 1: **Instance 1**. The drone turns off its engine and undergoes a parabolic trajectory. Lower-right figure shows evolution of the ellipsoid during the optimization process.

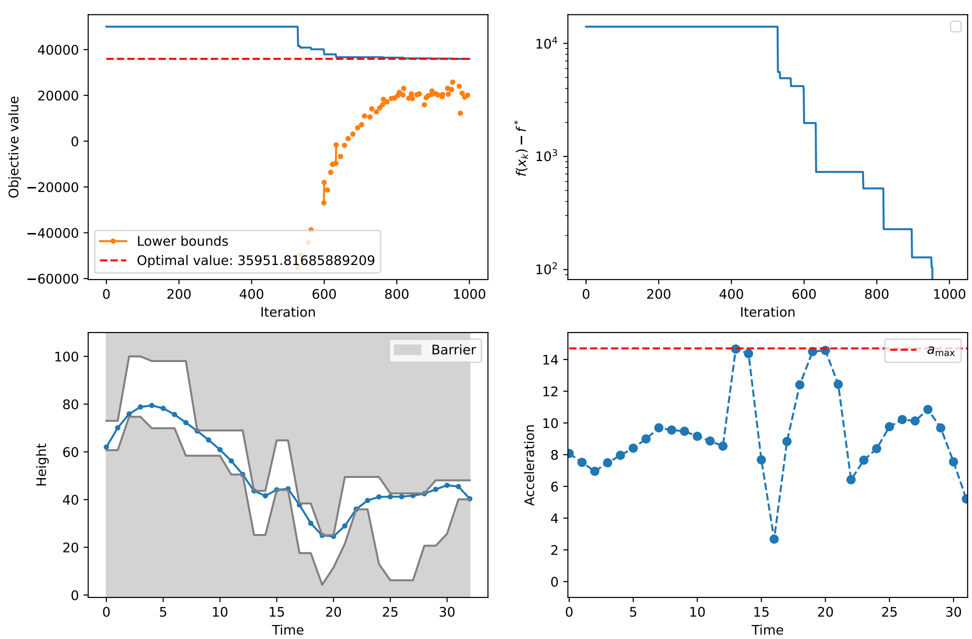

Figure 2: **Instance 2**. The drone avoids obstacles through a series of complex operations. Upper-right figure shows how the objective function converges in the log scale.