

# Bounds for the Multislope Ski-Rental Problem\*\*\*

Hiroshi FUJIWARA<sup>†a)</sup>, Kei SHIBUSAWA<sup>†\*</sup>, Kouki YAMAMOTO<sup>†\*\*</sup>, *Nonmembers,*  
and Hiroaki YAMAMOTO<sup>†</sup>, *Member*

**SUMMARY** The multislope ski-rental problem is an online optimization problem that generalizes the classical ski-rental problem. The player is offered not only a buy and a rent options but also other options that charge both initial and per-time fees. The competitive ratio of the classical ski-rental problem is known to be 2. In contrast, the best known so far on the competitive ratio of the multislope ski-rental problem is an upper bound of 4 and a lower bound of 3.62. In this paper we consider a parametric version of the multislope ski-rental problem, regarding the number of options as a parameter. We prove an upper bound for the parametric problem which is strictly less than 4. Moreover, we give a simple recurrence relation that yields an equation having a lower bound value as its root.

**key words:** online algorithm, competitive analysis, online optimization, ski-rental problems

## 1. Introduction

Online optimization problems ask us to determine an output which optimizes some objective function, under the setting that an input arrives gradually [1]–[3]. The classical ski-rental problem [4] is often picked up as an example of an online optimization problem. Suppose that a player can buy a set of ski gear for \$500 and rent one for \$50. The task of the problem is to decide when to buy a set of ski gear after renting one some times, under the setting that the player is not sure how many times he/she is going skiing in the future. The objective is to minimize the total cost. It is known that the optimal strategy is to buy a set of ski gear at our 10th ski trip.

The multislope ski-rental problem [5] is a generalization of the classical ski-rental problem. In the multislope ski-rental problem, a player is offered multiple options, called *slopes*, for using a set of ski gear. Each slope is characterized by its initial fee and its per-time fee. The player can use a set of ski gear by paying the initial fee at the beginning, and then by paying the per-time fee every time he/she goes skiing. In a sense, to pay both the initial and per-time fees can be thought as to *lease* a set of ski gear.

The standard performance measure of strategies for on-

line optimization problems is the *competitive ratio* [2], defined here as the maximum ratio of the total cost incurred by a strategy to the minimum total cost with the number of times of skiing known in advance. It is an interesting research target to bound or, if possible, to identify the value of the competitive ratio of the optimal strategy for a problem, which is called simply the *competitive ratio of the problem*. The competitive ratio of the classical ski-rental problem was already identified as 2 [4]. (The strategy that appeared at the beginning indeed achieves this.)

In contrast, the exact value of the competitive ratio of the multiple ski-rental problem has not been known yet. What is known so far is an upper bound of 4 [6] and a lower bound of 3.62 [7]. In this paper, although we do not tighten the gap, we improve the upper bound for the case where the number of slopes is fixed.

### 1.1 Our Contribution

Our results are summarized as follows:

(I) We consider a parametric version of the multislope ski-rental problem with the number of slopes being a parameter  $(k+1)$ , and evaluate the performance of Bejerano et al.'s algorithm [6] that generates a strategy, with a slight modification. As a result, we establish an upper bound strictly less than 4 for the parametric problem. For example, 3.83 for  $k=4$ , 3.93 for  $k=5$ , and 3.97 for  $k=6$  (see Table 1 and Fig. 1). For each  $k$ , our upper bound value is smaller than  $4 - 2^{-k+1}$ .

(II) Fujiwara et al. showed lower bounds for the parametric multislope ski-rental problem [9]. To obtain a lower bound value involves solving a recurrence relation with an unknown and then finding a root of an equation. In this paper we simplify the recurrence relation by eliminating fractional terms. In the paper [9], it is conjectured that the lower bound value coincides with the competitive ratio of the parametric multislope ski-rental problem. If this is true, our simple recurrence relation will help for identification of the competitive ratio of the multislope ski-rental problem.

### 1.2 Related Work

The classical ski-rental problem [4] is a famous toy problem of online optimization problems [1]–[3]. According to the book [2], Rudolph picked up the classical ski-rental problem in his lecture in 1986. Karlin et al.'s paper [4] on snoopy

Manuscript received April 4, 2019.

Manuscript revised August 9, 2019.

Manuscript publicized November 25, 2019.

<sup>†</sup>The authors are with Department of Electrical and Computer Engineering, Shinshu University, Nagano-shi, 380–8553 Japan.

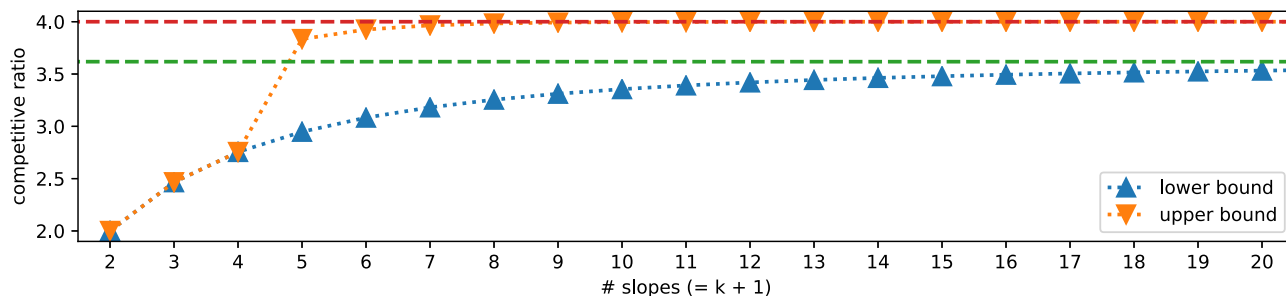
<sup>\*</sup>Presently, with JRC Engineering Co., Ltd.

<sup>\*\*</sup>Presently, with AISIN comCruise Co., Ltd.

<sup>\*\*\*</sup>This work was supported by KAKENHI (16K00033, 17K00013, and 17K00183).

a) E-mail: fujiwara@cs.shinshu-u.ac.jp

DOI: 10.1587/transinf.2019FCP0001



**Fig. 1** Lower and upper bounds on the competitive ratio of the  $(k + 1)$ -slope ski-rental problem. The dashed lines are at lower and upper bounds on the competitive ratio of the multislope ski-rental problem, in which the number of slopes is arbitrary.

**Table 1** Lower and upper bounds on the competitive ratio of the  $(k + 1)$ -slope ski-rental problem. Bold font indicates new bounds established in this paper.

# slopes (= $k + 1$ )	2	3	4	5	6	7	8	9	10	arbitrary
Upper bound	2 [4]	2.47 [8]	2.75 [8]	<b>3.83</b>	<b>3.93</b>	<b>3.97</b>	<b>3.98</b>	<b>3.99</b>	<b>4.00</b>	4 [6]
Lower bound	2 [4]	2.47 [8]	2.75 [8]	2.95 [9]	3.08 [9]	3.18 [9]	3.25 [9]	3.31 [9]	3.36 [9]	3.62 [7]

caching can be understood in the context of ski rental: The paper identifies the competitive ratio of the classical ski-rental problem as 2 (see Table 1 and Fig. 1). An easy introduction to the classical ski-rental problem can be found in Chapter 37 in the book [10]. The paper by Karlin [11] discusses various practical applications.

The multislope ski-rental problem [5] was first studied as Dynamic Power Management [12], [13], which we shall mention in Sect. 2.3. Augustine et al. [13] designed an algorithm that for a given instance of the multislope ski-rental problem, outputs the best possible strategy and its competitive ratio. The best bounds known so far on the competitive ratio of the multislope ski-rental problem are an upper bound of 4 [6] and a lower bound of  $\frac{5+\sqrt{5}}{2} \approx 3.62$  [7] (see Table 1 and Fig. 1). Although these two works aimed for other problems, the bounds straightforwardly apply to the multislope ski-rental problem.

Previous results on the parametric version that the number of slopes is  $k + 1$  are as follows (see Table 1 and Fig. 1): The competitive ratio of the problem each for  $k = 2$  and  $k = 3$  was identified by [8]: 2.47 for  $k = 2$  and 2.75 for  $k = 3$ . Fujiwara et al. [9] derived a lower bound for each value of  $k$ , though the lower bound value was not given in an explicit form.

The results we introduced above are all on a *deterministic* model, as well as our work throughout this paper, where every strategy makes decisions with a probability of one. For a randomized model, refer to, for example, [5], [14]–[16].

### 1.3 Note on Rounding

Throughout this paper numerical rounding is all done to the nearest value, differently from a convention that a lower bound is rounded down while an upper bound is rounded up. We nevertheless round to the nearest because for some

cases, the competitive ratio of the problem was already identified and we would like to display it as a single value (see Table 1). For example, a more accurate value for  $(\# \text{ slopes}) = 3$  is 2.46557. If we followed the conventional way, the upper and lower bounds would be displayed as 2.47 and 2.46, respectively, which do not look identical.

The upper bound of 4.00 for  $(\# \text{ slopes}) = 10$  in Table 1 is a result of rounding; its precise value is indeed strictly less than 4. We also mention that all numerical values in this paper can be calculated with an arbitrary precision.

## 2. Problem Statement

### 2.1 Instance of the $(k + 1)$ -Slope Ski-Rental Problem

For an integer  $k \geq 1$ , we refer to the multislope ski-rental problem having slopes  $0, 1, \dots, k$  as the  $(k + 1)$ -slope ski-rental problem. As long as the player continues to go skiing, the player has to stay some slope while paying per-time fees, or transition to another slope. (An option is called a slope because the total cost increases along the slope of the per-time fee.) The player is assumed to start from slope 0. We also assume that the number of times of skiing, denoted by  $t$ , is a nonnegative real number. Our notation basically follows [9].

An *instance* of the  $(k + 1)$ -slope ski-rental problem is a pair of vectors  $(\mathbf{r}, \mathbf{b})$ , where  $\mathbf{r}$  has  $k + 1$  entries and  $\mathbf{b}$  has  $k(k + 1)/2$  entries. For  $0 \leq i < j \leq k$ , the entry  $r_i$  denotes the per-time fee of slope  $i$ , and the entry  $b_{i,j}$  denotes the transition fee from slope  $i$  to slope  $j$ . (The transition fee is what we called the initial fee in Sect. 1.) We impose the following constraints:

$$r_0 = 1, r_k = 0, b_{0,k} = 1, \quad (1)$$

$$r_i > r_j \text{ for } 0 \leq i < j \leq k, \quad (2)$$

$$b_{l,j} - b_{l,i} \leq b_{i,j} \leq b_{l,j} \text{ for } 0 \leq l < i < j \leq k. \quad (3)$$

The constraints normalize so that the per-time and transition fees are all scaled down to between zero and one, without loss of generality. To apply a practical price system, multiply  $\mathbf{r}$  by the rental fee and multiply  $\mathbf{b}$  by the price of a set of ski gear. The constraint (1) fixes slope 0 to be a *rent* option and slope  $k$  to be a *buy* option with no per-time fee. Thus,  $k$  indicates the number of slopes other than the buy option, and the case of  $k = 1$  is equivalent to the classical ski-rental problem. (This is why we set the number of slopes as  $k + 1$ .) The constraint (2) says that a slope with a greater index is associated with a cheaper per-time fee. From this, we can assume that the player always transitions only to a slope with a greater index; the player cannot save cost by a backward transition. The left inequality in (3) is the triangular inequality: A direct transition from slope  $l$  to  $j$  is equal to or cheaper than a transition via another slope  $i$ . The right inequality in (3) says that a transition from slope  $i$  to  $j$  should be no cheaper than a transition from slope  $l < i$ .

## 2.2 Strategy and Competitiveness

At each occasion, the player at slope  $i$  either keeps staying at the same slope or transitions to a different slope  $j$  by paying  $b_{i,j}$ . This behavior can be described as a deterministic *strategy* of the player using a vector  $\mathbf{x}$  with  $k + 1$  entries. (Throughout this paper, we deal with only deterministic strategies.) Each entry  $x_i$  indicates the number of times of skiing the player has gone when the player transitions from slope  $i - 1$  to slope  $i$ . The sequence of the entries is assumed to be non-decreasing by the constraint (2). Since the player always starts from slope 0, we fix  $x_0 = 0$ . (Please do not confuse the notation of a strategy with the unknown  $x$  without a subscript that will appear in equations in Sects. 3 and 4.)

Note that the player may skip a slope, not just transitioning to the next slope. If the player transitions from slope  $i$  directly to  $j$  by skipping the slopes between, we define  $x_{i+1} = \dots = x_{j-1} = x_j$ . In order to describe which slopes are used, we define a relation of  $i < j$  if the player transitions from slope  $i$  to  $j$ .

Let the number of times of skiing  $t$  be such that  $x_i \leq t < x_{i+1}$ . By repeatedly staying and transitioning over slopes, the player according to strategy  $\mathbf{x}$  will have paid a cost of

$$ON(\mathbf{x}, t) := \sum_{0 \leq l < m \leq i} \{r_l(x_m - x_l) + b_{l,m}\} + r_i(t - x_i)$$

For the sake of evaluating performance of strategies, we consider an *optimal offline player* who behaves optimally with the value of  $t$  known. Due to the constraint (3), the optimal offline player will choose the best slope for him/her at the beginning and then keep staying there. The cost is written as

$$OPT(t) := \min_{0 \leq j \leq k} OFF_j(t),$$

where  $OFF_j(t) := b_{0,j} + r_j \cdot t$  represents the transition fee to slope  $j$  plus the cost of staying at slope  $j$ .

The standard performance measure of strategies for on-line optimization problems is the *competitive ratio* [2]. We say that the competitive ratio of strategy  $\mathbf{x}$  is  $c$ , if

$$ON(\mathbf{x}, t) \leq c \cdot OPT(t)$$

holds for all  $t \geq 0$ .

Next, we define the competitive ratio of the multislope ski-rental problem. Let strategy  $\bar{\mathbf{x}}$  be an optimal strategy for the multislope ski-rental problem. That is to say, the competitive ratio  $\bar{c}$  of strategy  $\bar{\mathbf{x}}$  is the minimum over all strategies. Then, the competitive ratio of the multislope ski-rental problem is  $\bar{c}$ . Note that these terminologies, the competitive ratio of a strategy and the competitive ratio of the problem, have different meanings in this way.

Please be careful also with the words *strategy* and *algorithm*: A strategy is, as we defined above, a schedule of when to transition to a new slope. On the other hand, algorithms in this paper are those which output a strategy based on an instance received as an input.

## 2.3 An Example and Applications

Consider an instance of the 3-slope ski-rental problem:  $(r_0, r_1, r_2) = (1, 0.3, 0)$  and  $(b_{0,1}, b_{0,2}, b_{1,2}) = (0.4, 1, 0.7)$ . As we introduced in Sect. 2.1, slope 0 is a rent option and slope 2 is a buy option. Slope 1 is a lease option in that the player has to both the initial and the per-time fees. Applying Augustine et al.'s algorithm [13], we get an optimal strategy of  $(\bar{x}_0, \bar{x}_1, \bar{x}_2) \approx (0, 0.41, 2)$  with  $0 < 1 < 2$  (i.e., it does not skip any slope).

We below illustrate applications of the multislope ski-rental problem using this numerical example.

**(A) Partial purchase of a set of ski gear:** In Sect. 2.1 we say that the player chooses an option to use the entire ski gear. This can be understood as a different setting: A set of gear consists of multiple components. The player is allowed to use his/her own components while renting the rest. A slope stands for what the player has already bought and therefore what should be rented. (Although the subject needs not to be ski rental any longer, we keep it for a while.)

Suppose that a set of ski gear consists of a ski wear and a pair of skis. Scaling the above example so that the entire ski gear costs \$500 and the rental fee of the entire ski gear is \$50 per time, we multiply  $\mathbf{r}$  by 50, multiply  $\mathbf{b}$  by 500, and then obtain  $(r'_0, r'_1, r'_2) = (50, 15, 0)$  and  $(b'_{0,1}, b'_{0,2}, b'_{1,2}) = (200, 500, 350)$ . This scaled instance can be understood as follows: After the player has bought a ski wear for \$200, the player can go skiing by renting only a pair of skis for \$30 per time. Furthermore, the player may buy a pair of skis for \$350. Note that if the player buys the entire ski gear at once, he/she saves  $200 + 350 - 500 = 50$  dollars.

The optimal strategy is scaled to  $(\bar{x}'_0, \bar{x}'_1, \bar{x}'_2) \approx (0, 4.1, 20)$ , which recommends the player: *Start by renting everything. Buy a ski wear at 4th ski trip. Then, buy a pair of skis at 20th ski trip.*

**(B) Dynamic Power Management:** Suppose that a mobile device has a energy-saving mode, say *Sleep*, and

we can set a strategy that specifies automatic transitions between modes according to the length of an idle period so far. Then, the multislope ski-rental problem is equivalent to minimization of the energy consumption during an idle period plus that for resuming the device [12]. Although the details are omitted here, the above example of an instance represents a specification of a device:

- Power for staying On: 1 ( $= r_0$ ),
- Power for staying Sleep: 0.3 ( $= r_1$ ),
- Power for staying Off: 0 ( $= r_2$ ),
- Energy for turning from On to Sleep and then to On: 0.4 ( $= b_{0,1}$ ),
- Energy for turning from On to Sleep, then to Off, and then to On: 1.1 ( $= b_{0,1} + b_{1,2}$ ),
- Energy for turning from On to Off and then to On: 1 ( $= b_{0,2}$ ).

If we interpret the time unit as hours, the optimal strategy says that: *When an idle period has lasted 24 minutes, then turn from On to Sleep. When an idle period has reached 2 hours in total, turn from Sleep to Off.*

### 3. Upper Bounds

#### 3.1 Algorithm

We study upper bounds of the competitive ratio of the  $(k+1)$ -slope ski-rental problem. The basis of our analysis is Bejerano et al.'s algorithm [6], which achieves a competitive ratio of 4. Applying this to the  $(k+1)$ -slope ski-rental problem, one can immediately have an upper bound of 4. In this section we focus on the performance of the algorithm for the  $(k+1)$ -slope case, with a little modification, and reveal that its competitive ratio is in fact strictly lower than 4.

Bejerano et al.'s algorithm is based on the so-called *doubling technique* (see [17] for example). Whenever the algorithm transitions to a new slope, it transitions so that the transition cost does not exceed a constant factor times the cost of the optimal offline player so far. Our algorithm DBL, defined below, inherits this property.

For describing our algorithm, we employ an additional notation of  $\{t_i\}_{i=1,2,\dots}$ . For each integer  $i \geq 0$ , denote by  $t_i$  the value of  $t$  that satisfies  $OFF_i(t) = OFF_{i+1}(t)$ . It is observed that if the number of times of skiing  $t$  satisfies  $t_{i-1} \leq t \leq t_i$ , then  $OPT(t) = OFF_i(t)$  holds. In other words, for such a case, the optimal offline player will choose slope  $i$  and keep staying there.

**Definition 1.** *Algorithm DBL receives an instance of the  $(k+1)$ -slope ski-rental problem with  $k \geq 1$ , and a real constant  $\alpha > 1$ . The algorithm successively determines a sequence of  $s_0, s_1, s_2, \dots$  which the player stays (i.e.,  $s_0 < s_1 < s_2 < \dots$ ) and a sequence of  $x_{s_0}, x_{s_1}, x_{s_2}, \dots$  as follows: Set  $s_0 = 0$  and  $x_{s_0} = 0$ . For each integer  $j \geq 0$ , while  $s_j < k$ , set  $s_{j+1} = \max\{i \mid b_{s_j,i} \leq \alpha \cdot OPT(t_{s_j}), i \leq k\}$  and  $x_{s_{j+1}} = t_{s_j}$ , as well as  $x_{s_{j+1}} = x_{s_j+2} = \dots = x_{s_{j+1}-1} = t_{s_j}$  for skipping the slopes in-between.*

We give a numerical example for the instance that we used in Sect. 2.3. Take  $\alpha = 2$  for example. We get  $s_1 = 2$  and then the iteration is over. Consequently, we have  $(x_0, x_1, x_2) \approx (0, 0.57, 0.57)$  with  $0 < 2$  (i.e., it skips slope 1).

The rule of determining  $s_{j+1}$  was originally  $s_{j+1} = \max\{i \mid b_{0,i} \leq \alpha \cdot OPT(t_{s_j}), i \leq k\}$  in Bejerano et al.'s algorithm. This is because Bejerano et al.'s algorithm was an algorithm for a so-called *investment* instance which satisfies  $b_{j,i} = b_{0,i}$  for  $0 < j < i \leq k$  (see [8] for example). In contrast, a general instance admits  $b_{j,i} \leq b_{0,i}$ . So, we have naturally replaced  $b_{0,i}$  with  $b_{s_j,i}$ . This change affects the strategy in such a way that it transitions to a slopes with a larger index earlier, which saves the cost of the player. Nevertheless, from the viewpoint of competitiveness, this change does not contribute in improvement. Indeed, the following arguments still hold true for the original algorithm.

#### 3.2 Analysis

The following several lemmas tell us the performance of the algorithm DBL. The flow of the arguments is the same as Sect. 3 of the paper [6]. Lemmas 1, 2, and 3 correspond to Lemmas 1 and 2, and Theorem 1 of that paper, respectively.

**Lemma 1.** *For each integer  $j \geq 1$ ,*

$$OPT(t_{s_j}) > \alpha \cdot OPT(t_{s_{j-1}}).$$

*Proof.* For each integer  $j \geq 1$ , the algorithm DBL sets  $s_j = \max\{i \mid b_{s_{j-1},i} \leq \alpha \cdot OPT(t_{s_{j-1}}), i \leq k\}$ , which implies that  $b_{s_{j-1},s_j} \leq \alpha \cdot OPT(t_{s_{j-1}})$  and  $b_{s_{j-1},s_{j+1}} > \alpha \cdot OPT(t_{s_{j-1}})$ . From definition of  $t_{s_j}$  and instances, we derive

$$\begin{aligned} OPT(t_{s_j}) &= b_{0,s_j} + r_{s_j} \cdot t_{s_j} \\ &= b_{0,s_{j+1}} + r_{s_{j+1}} \cdot t_{s_j} \\ &\geq b_{0,s_{j+1}} \\ &\geq b_{s_{j-1},s_{j+1}} \\ &> \alpha \cdot OPT(t_{s_{j-1}}). \end{aligned} \quad \square$$

The next lemma claims that the cost incurred for a slope is at most the cost of the optimal offline player so far. Here, the cost incurred for slope  $s_j$  consists of the transition cost from the previous slope  $s_{j-1}$  plus the sum of the per-time cost for staying slope  $s_j$ , which is the left-hand side of the inequality of Lemma 2.

**Lemma 2.** *For each integer  $j \geq 1$ ,*

$$b_{s_{j-1},s_j} + r_{s_j}(t_{s_j} - t_{s_{j-1}}) \leq OPT(t_{s_j}).$$

*Proof.* By definition of instances, it is derived that

$$\begin{aligned} b_{s_{j-1},s_j} + r_{s_j}(t_{s_j} - t_{s_{j-1}}) &\leq b_{s_{j-1},s_j} + r_{s_j} \cdot t_{s_j} \\ &\leq b_{0,s_j} + r_{s_j} \cdot t_{s_j} \\ &= OPT(t_{s_j}). \end{aligned} \quad \square$$

The following lemma gives a parametric performance

of the algorithm DBL. Unlike Theorem 1 of the paper [6], we fix the number of slopes to be  $k + 1$ . Since a matching bound is already given for each of the cases of  $k \leq 3$  by [4] and [8], we consider only  $k \geq 4$  in the rest of this section.

**Lemma 3.** *For each integer  $k \geq 4$ , the competitive ratio of the strategy generated by the algorithm DBL for the  $(k + 1)$ -slope ski-rental problem is at most  $\frac{\alpha^2 - \alpha^{-k+1}}{\alpha - 1}$ .*

*Proof.* Choose arbitrarily an instance  $(\mathbf{r}, \mathbf{b})$  of the  $(k + 1)$ -slope ski-rental problem, and the number of times of skiing  $\tau \geq 0$ . Suppose that when a player who obeys the strategy generated by the algorithm DBL has gone skiing  $\tau$  times, the player is at slope  $s_{n+1}$  with  $n \leq k - 1$ . (Recall that the last slope is slope  $k$ .) We denote by  $DBL(\tau)$  the cost paid by the player and bound it from above. The derivation involves several equalities and inequalities: It holds that  $r_{s_0} \cdot t_{s_0} = OPT(t_{s_0})$  since  $s_0 = 0$ . The cost paid before reaching slope  $s_{n+1}$  is bounded by Lemma 2, as well as  $b_{s_n, s_{n+1}} \leq b_{0, s_{n+1}} \leq \alpha \cdot OPT(t_{s_n})$ . By applying Lemma 1 repeatedly, we have for any  $j \geq 1$ ,  $OPT(t_{s_n}) > \alpha^{n-j} \cdot OPT(t_{s_j})$ . Besides, by definition of  $OPT$ ,  $r_{s_j}(\tau - t_{s_n}) = OPT(\tau) - OPT(t_{s_n})$  holds. Using these equalities and inequalities, we evaluate

$$\begin{aligned}
 DBL(\tau) &= r_{s_0} \cdot t_{s_0} + \sum_{j=1}^n \{b_{s_{j-1}, s_j} + r_{s_j}(t_{s_j} - t_{s_{j-1}})\} \\
 &\quad + b_{s_n, s_{n+1}} + r_{s_{n+1}}(\tau - t_{s_n}) \\
 &\leq OPT(t_{s_0}) + \sum_{j=1}^n OPT(t_{s_j}) + \alpha \cdot OPT(t_{s_n}) \\
 &\quad + (OPT(\tau) - OPT(t_{s_n})) \\
 &= \sum_{j=0}^n OPT(t_{s_j}) + \alpha \cdot OPT(t_{s_n}) \\
 &\quad + (OPT(\tau) - OPT(t_{s_n})) \\
 &\leq \sum_{j=0}^n \frac{1}{\alpha^{n-j}} \cdot OPT(t_{s_n}) + \alpha \cdot OPT(t_{s_n}) \\
 &\quad + (OPT(\tau) - OPT(t_{s_n})) \\
 &= \frac{\alpha - \alpha^{-n}}{\alpha - 1} \cdot OPT(t_{s_n}) + \alpha \cdot OPT(t_{s_n}) \\
 &\quad + (OPT(\tau) - OPT(t_{s_n})) \\
 &= \frac{\alpha^2 - \alpha^{-n}}{\alpha - 1} \cdot OPT(t_{s_n}) \\
 &\quad + (OPT(\tau) - OPT(t_{s_n})) \\
 &< \frac{\alpha^2 - \alpha^{-n}}{\alpha - 1} \cdot OPT(t_{s_n}) \\
 &\quad + \frac{\alpha^2 - \alpha^{-n}}{\alpha - 1} \cdot (OPT(\tau) - OPT(t_{s_n})) \\
 &= \frac{\alpha^2 - \alpha^{-n}}{\alpha - 1} \cdot OPT(\tau) \\
 &\leq \frac{\alpha^2 - \alpha^{-k+1}}{\alpha - 1} \cdot OPT(\tau),
 \end{aligned}$$

which means that the competitive ratio of the strategy is at

**Table 2** Behavior of the functions  $h'$ ,  $h$ ,  $H$ , and  $f$ .

$x$	1	$\frac{2k}{k+2}$	$\bar{\beta}$	$\bar{\alpha}$	2
$h'(x)$	-	-	0	+	+
$h(x)$	0	$\searrow$	-	$\nearrow$	$2^{k+1} + k$
$H(x)$	0	$\searrow$	-	$\nearrow$	0
$f(x)$		$\searrow$	$\searrow$	$\nearrow$	$\frac{\bar{\alpha}^2 - \bar{\alpha}^{-k+1}}{\bar{\alpha} - 1}$

most  $\frac{\alpha^2 - \alpha^{-k+1}}{\alpha - 1}$ .  $\square$

We then bound the value of  $\frac{\alpha^2 - \alpha^{-k+1}}{\alpha - 1}$  from above. In other words, we investigate how much the algorithm DBL improves by choosing a good value for  $\alpha$ .

**Lemma 4.** *Let  $k$  be an integer  $\geq 4$ , and  $\bar{\alpha}$  be a root of the equation  $x^{k+2} - 2x^{k+1} + kx - k + 1 = 0$  which is greater than one. The function  $f(x) = \frac{x^2 - x^{-k+1}}{x - 1}$  on the interval  $(1, \infty)$  achieves its minimum at  $x = \bar{\alpha}$ .*

*Proof.* We first prepare some functions. Differentiating  $f(x)$  we obtain

$$f'(x) = \frac{x^{k+2} - 2x^{k+1} + kx - k + 1}{x^k(x - 1)^2}.$$

We denote  $x^k(x - 1)^2 f'(x)$  by  $H(x)$ :

$$H(x) = x^{k+2} - 2x^{k+1} + kx - k + 1.$$

Let  $h(x)$  be the derivative of  $H(x)$ :

$$h(x) = (k + 2)x^{k+1} - 2(k + 1)x^k + k.$$

Moreover, differentiating  $h(x)$  we get

$$\begin{aligned}
 h'(x) &= (k + 2)(k + 1)x^k - 2k(k + 1)x^{k-1} \\
 &= (k + 2)(k + 1)x^{k-1} \left( x - \frac{2k}{k + 2} \right).
 \end{aligned} \tag{4}$$

Although the function  $f$  is not defined at  $x = 1$ , we consider  $H$ ,  $h$ , and  $h'$  with their domains including  $x = 1$ . Table 2 indicates the behavior of these functions, whose analysis is given below.

We begin by showing the following two facts: (A) The equation  $h(x) = 0$  has a unique root  $\bar{\beta}$  which is greater than one, and the root lies in the interval  $(\frac{2k}{k+2}, 2)$ . (B)  $h(x)$  is negative for  $1 < x < \bar{\beta}$  and positive for  $\bar{\beta} < x$ .

*Proof of (A):* The following three properties suffice to prove (A): (i) The function  $h$  increases monotonically on the interval  $(\frac{2k}{k+2}, 2)$ , since  $h'(x) > 0$  for  $x > \frac{2k}{k+2}$  by (4). (ii) It follows that  $h(\frac{2k}{k+2}) < 0$ , since the function  $h$  decreases monotonically on the interval  $1 < x < \frac{2k}{k+2}$ , which is derived by (4), and  $h(1) = (k + 2) - 2(k + 1) + k = 0$ . (iii)  $h(2) = 2^{k+1} + k > 0$ .

*Proof of (B):* We have seen that the function  $h(x)$  is monotonically decreasing for  $1 < x < \frac{2k}{k+2}$  and monotonically increasing for  $\frac{2k}{k+2} < x$ . Together with the uniqueness of the root of  $h(x) = 0$  and the fact (A), we conclude that  $h(x) < 0$  for  $1 < x < \bar{\beta}$  and  $h(x) > 0$  for  $\bar{\beta} < x$ .

We then prove another fact: (C) The equation  $H(x) = 0$  has a unique root  $\bar{\alpha}$  which is greater than one, and the root lies in the interval  $(\bar{\beta}, 2)$ .

*Proof of (C):* The fact (B) says that the function  $H(x)$  decreases monotonically for  $1 < x < \bar{\beta}$  and increases monotonically for  $\bar{\beta} < x$ . Since  $H(1) = 0$ ,  $H(\bar{\beta}) < 0$  should hold. We also have  $H(2) = k + 1 > 0$ . Hence, the equation  $H(x) = 0$  has a unique root in the interval  $(\bar{\beta}, 2)$ .

We are finally ready to prove the lemma. The fact (C) with  $H(1) = 0$  says that  $H(x) < 0$  for  $1 < x < \bar{\alpha}$  and  $H(x) > 0$  for  $\bar{\alpha} < x$ . By definition of  $H$ , the signs of  $H(x)$  and  $f'(x)$  coincide for  $x > 1$ . We then know that the function  $f(x)$  decreases monotonically for  $1 < x < \bar{\alpha}$  and increases monotonically for  $\bar{\alpha} < x$ . Thus, the function  $f(x)$  on the interval  $(1, \infty)$  achieves its minimum at  $x = \bar{\alpha}$ .  $\square$

Lemmas 3 and 4 prove our main theorem on upper bounds.

**Theorem 1.** *For each integer  $k \geq 4$ , the competitive ratio of the  $(k + 1)$ -slope ski-rental problem is at most  $f(\bar{\alpha}) = \frac{\bar{\alpha}^2 - \bar{\alpha}^{-k+1}}{\bar{\alpha} - 1}$ , where  $\bar{\alpha}$  be a root of the equation  $x^{k+2} - 2x^{k+1} + kx - k + 1 = 0$  which is greater than one.*

See Table 1 for numerical values. For example, our upper bound is 3.83 for the 5-slope ski-rental problem and is 3.93 for the 6-slope ski-rental problem, each of which improves 4. We conclude this section by adding that for any  $k$ , our upper bound is strictly less than 4.

**Lemma 5.** *Let  $f$  and  $\bar{\alpha}$  be as defined in Lemma 4 and Theorem 1. It holds that  $f(\bar{\alpha}) < 4 - 2^{-k+1}$ .*

*Proof.* The proof of Lemma 4 says that the function  $f(x)$  increases monotonically for  $\bar{\alpha} < x$ . Thus, we have  $f(\bar{\alpha}) < f(2) = 4 - 2^{-k+1}$ .  $\square$

**Corollary 1.** *For each integer  $k \geq 4$ , the competitive ratio of the  $(k + 1)$ -slope ski-rental problem is at most  $4 - 2^{-k+1}$ .*

#### 4. The Equations Related to Lower Bounds

In this section we consider equations with a root that is a lower bound for the  $(k + 1)$ -slope ski-rental problem, not improving lower bounds themselves. We present a simpler construction of a sequence of equations.

Fujiwara et al. [9] gave lower bounds on the competitive ratio of the  $(k + 1)$ -slope ski-rental problem, extending the analysis by Damaschke [7]. For example, the maximum real root ( $\approx 2.47$ ) of the equation  $x^3 - 4x^2 + 5x - 3 = 0$  is a lower bound for the 3-slope ski-rental problem, and the maximum real root ( $\approx 2.75$ ) of the equation  $x^3 - 5x^2 + 8x - 5 = 0$  is a lower bound for the 4-slope ski-rental problem. The following theorem states their claim in a general form.

**Theorem 2.** ([9]) *For each integer  $k \geq 2$ , the competitive ratio of the  $(k + 1)$ -slope ski-rental problem is at least the maximum of real roots of the equation  $q_k(x) - x = 0$  which lie between 2 and  $\frac{5+\sqrt{5}}{2}$ , where  $\{q_i(x)\}_{i=1,2,\dots}$  is defined as*

$$q_1(x) = \frac{x}{x-1} \quad (5)$$

and

$$q_{i+1}(x) = \frac{x^3 - x^2 + x \cdot q_i(x)}{x^3 - x^2 - (x-1)^2 \cdot q_i(x)} \quad (6)$$

for  $i \geq 1$ .

Table 1 gives numerical values of lower bounds. The procedure in the paper [9] describes how to obtain an instance of the  $(k + 1)$ -slope ski-rental problem for which any strategy has a competitive ratio which is at least the lower bound value. In this paper we omit the construction of instances and concentrate on equations that have the lower bound values as roots.

It is known that the lower bound values for  $k = 2$  and  $k = 3$  coincide with the competitive ratios of the  $(k + 1)$ -slope ski-rental problem, respectively [8]. In other words, these lower bounds match upper bounds. Observing this fact, the paper [9] conjectures as follows:

**Conjecture 1.** ([9]) *For each integer  $k \geq 2$ , the competitive ratio of the  $(k + 1)$ -slope ski-rental problem is equal to the maximum of real roots of the equation  $q_k(x) - x = 0$  which lie between 2 and  $\frac{5+\sqrt{5}}{2}$ .*

The equation  $q_k(x) - x = 0$  itself is not an equation that leads us immediately to the value of a lower bound. In general, the equation includes irrelevant factors, also in its denominator. To get a lower bound value, we first have to cancel such irrelevant factors. (The two equations that appeared at the beginning of this section are presented after simplification.)

We investigate the recurrence relation in Theorem 2 and consequently find a simpler recurrence relation on equations that directly yield the same lower bound values as roots.

**Definition 2.** Define  $\{y_i(x)\}_{i=0,1,\dots}$  by

$$\begin{aligned} y_0(x) &= x - 1, \\ y_1(x) &= x - 2, \\ y_2(x) &= x^3 - 4x^2 + 5x - 3, \\ y_3(x) &= x^3 - 5x^2 + 8x - 5, \end{aligned}$$

and for  $k \geq 2$ ,

$$y_{k+2}(x) = (x^2 - 3x + 3) \cdot y_k(x) - (x-1)^2 \cdot y_{k-2}(x). \quad (7)$$

Obviously,  $y_i(x)$  never becomes fractional. We next state that solving  $y_k(x) = 0$  is sufficient to obtain a lower bound value, instead of solving  $q_k(x) - x = 0$ .

**Lemma 6.** *For each integer  $k \geq 2$ , if  $y_{k-2}(x) = 0$  and  $x > 1$ , then  $y_k(x) \neq 0$ .*

*Proof.* We show the lemma by induction on  $k$ . It is easy to prove the cases of  $k = 2$  and  $k = 3$ : Indeed,  $x = 1$  is a unique

root of  $y_0(x) = 0$ , but  $y_2(1) = -1$ .  $x = 2$  is a unique root of  $y_1(x) = 0$ , but  $y_3(2) = 1$ .

Assume that for an integer  $i \geq 2$  and a real  $x$  such that  $y_{i-2}(x) = 0$  and  $x > 1$ ,  $y_i(x) \neq 0$  holds. Let  $\bar{x}$  be a real number that satisfies  $y_i(\bar{x}) = 0$  and  $\bar{x} > 1$ . Applying (7), we have

$$\begin{aligned} y_{i+2}(\bar{x}) &= (\bar{x}^2 - 3\bar{x} + 3) \cdot y_i(\bar{x}) - (\bar{x} - 1)^2 \cdot y_{i-2}(\bar{x}) \\ &= -(\bar{x} - 1)^2 \cdot y_{i-2}(\bar{x}). \end{aligned}$$

We know that  $y_{i-2}(\bar{x}) \neq 0$  should hold true. Otherwise, it is a contradiction to the assumption. Moreover, since  $\bar{x} > 1$ ,  $(\bar{x} - 1)^2$  is positive. Hence,  $y_{i+2}(\bar{x}) \neq 0$  holds. The lemma is thus proved by induction.  $\square$

**Lemma 7.** For each integer  $k \geq 2$ ,

$$(x - 1)^2(q_k(x) - x) \cdot y_{k-2}(x) = -x \cdot y_k(x). \quad (8)$$

*Proof.* Again we show the lemma by induction on  $k$ . We first see the lemma holds for  $k = 2$ . Using (5) and (6) with  $i = 1$ , we get

$$q_2(x) - x = -\frac{x(x^3 - 4x^2 + 5x - 3)}{(x - 1)^3} = -\frac{x \cdot y_2(x)}{(x - 1)^3}.$$

Together with  $y_0(x) = x - 1$ , we derive

$$\begin{aligned} (x - 1)^2(q_2(x) - x) \cdot y_0(x) &= -\frac{x \cdot y_2(x)}{(x - 1)} \cdot (x - 1) \\ &= -x \cdot y_2(x), \end{aligned}$$

which shows the lemma for  $k = 2$ .

Similarly, we see that the lemma holds for  $k = 3$  from

$$q_3(x) - x = -\frac{x(x^3 - 5x^2 + 8x - 5)}{(x - 1)^2(x - 2)} = -\frac{x \cdot y_3(x)}{(x - 1)^2(x - 2)}$$

and

$$\begin{aligned} (x - 1)^2(q_3(x) - x) \cdot y_1(x) &= -\frac{x \cdot y_3(x)}{(x - 2)} \cdot (x - 2) \\ &= -x \cdot y_3(x). \end{aligned}$$

Assume that for an integer  $i \geq 2$ ,

$$(x - 1)^2(q_i(x) - x) \cdot y_{i-2}(x) = -x \cdot y_i(x). \quad (9)$$

Incrementing the indices of (6), we obtain

$$q_{i+2}(x) - x = \frac{x^3 - x^2 + x \cdot q_{i+1}(x)}{x^3 - x^2 - (x - 1)^2 \cdot q_{i+1}(x)}.$$

Plugging (6) into the right-hand side again,

$$q_{i+2}(x) - x = -\frac{x}{(x - 1)^2} \cdot \left\{ (x^2 - 3x + 3) + \frac{x}{q_i(x) - x} \right\}.$$

Multiplying the both sides by  $(x - 1)^2 \cdot y_i(x)$ , we get

$$(x - 1)^2(q_{i+2}(x) - x) \cdot y_i(x)$$

$$= -x \cdot \left\{ (x^2 - 3x + 3) \cdot y_i(x) + \frac{x \cdot y_i(x)}{q_i(x) - x} \right\}.$$

The factor  $q_i(x) - x$  in the right-hand side can be canceled by using (9) as:

$$\begin{aligned} (x - 1)^2(q_{i+2}(x) - x) \cdot y_i(x) \\ = -x \cdot \left\{ (x^2 - 3x + 3) \cdot y_i(x) - (x - 1)^2 \cdot y_{i-2}(x) \right\}. \end{aligned}$$

Finally, applying the recurrence relation (7), we have

$$(x - 1)^2(q_{i+2}(x) - x) \cdot y_i(x) = -x \cdot y_{i+2}(x),$$

which is the statement of the lemma for  $k = i + 2$ . The lemma is thus proved by induction.  $\square$

**Lemma 8.** For each integer  $k \geq 2$  and any real  $x > 1$ ,  $q_k(x) - x = 0$  if and only if  $y_k(x) = 0$ .

*Proof.* Lemma 6 says that  $y_{k-2}(x) = 0$  and  $y_k(x) = 0$  do not hold true for the same  $x$ . To apply this fact to (8) of Lemma 7 implies the lemma.  $\square$

Our main theorem on lower bounds follows from Lemma 8.

**Theorem 3.** For each integer  $k \geq 2$ , the competitive ratio of the  $(k + 1)$ -slope ski-rental problem is at least the maximum of real roots of the equation  $y_k(x) = 0$  which lie between 2 and  $\frac{5+\sqrt{5}}{2}$ .

Conjecture 1, together with Theorem 3, says that the competitive ratio of the  $(k + 1)$ -slope ski-rental problem is equal to the root of the equation  $y_k(x) = 0$ . If this is true, the equation  $y_k(x) = 0$  should be related to some intrinsic characteristics behind the problem.

We make some observations on  $\{y_i(x)\}_{i=1,2,\dots}$ , though we are not sure that they are helpful. One is that the constant term of  $y_i(x)$ , with its sign flipped, forms the Fibonacci sequence (see [18] for example). This can be shown by the recurrence relation (7). Another observation is that  $\{y_i(x)\}_{i=1,2,\dots}$  does not consist of only irreducible polynomials over the field of rationals. In fact, a counterexample is  $y_7(x) = x^7 - 11x^6 + 51x^5 - 132x^4 + 210x^3 - 209x^2 + 123x - 34 = (x - 2)(x^6 - 9x^5 + 33x^4 - 66x^3 + 78x^2 - 53x + 17)$ .

## 5. Concluding Remarks

As far as we observe results of some computational experiments implementing the algorithm DBL, the value of the competitive ratio of the  $(k + 1)$ -slope ski-rental problem does not seem close to the upper bound. That is to say, our analysis is loose. There is room for improvement of the upper bound by a more detailed analysis.

For any  $k$ , the equation  $y_k(x) = 0$  can be easily obtained by our recurrence relation. It is interesting to investigate the relation between roots for different  $k$  from an algebraic viewpoint.

## References

- [1] A. Fiat and G.J. Woeginger, eds., *Online Algorithms, The State of the Art* (the book grown out of a Dagstuhl Seminar, June 1996), LNCS, vol.1442, Springer, 1998.
- [2] A. Borodin and R. El-Yaniv, *Online Computation and Competitive Analysis*, Cambridge University Press, 1998.
- [3] D. Komm, *An Introduction to Online Computation: Determinism, Randomization, Advice*, 1st ed., Springer, 2016.
- [4] A.R. Karlin, M.S. Manasse, L. Rudolph, and D.D. Sleator, “Competitive snoopy caching,” *Algorithmica*, vol.3, no.1-4, pp.77–119, 1988.
- [5] Z. Lotker, B. Patt-Shamir, and D. Rawitz, “Rent, lease, or buy: Randomized algorithms for multislope ski rental,” *SIAM J. Discrete Math.*, vol.26, no.2, pp.718–736, 2012.
- [6] Y. Bejerano, I. Cidon, and J. Naor, “Dynamic session management for static and mobile users: a competitive on-line algorithmic approach,” *Proc. DIAL-M '00*, pp.65–74, 2000.
- [7] P. Damaschke, “Nearly optimal strategies for special cases of on-line capital investment,” *Theor. Comput. Sci.*, vol.302, no.1-3, pp.35–44, 2003.
- [8] H. Fujiwara, T. Kitano, and T. Fujito, “On the best possible competitive ratio for the multislope ski-rental problem,” *J. Comb. Optim.*, vol.31, no.2, pp.463–490, 2016.
- [9] H. Fujiwara, Y. Konno, and T. Fujito, “Analysis of lower bounds for the multislope ski-rental problem,” *IEICE Trans. Fundamentals*, vol.E97-A, no.6, pp.1200–1205, 2014.
- [10] B. Vöcking, H. Alt, M. Dietzfelbinger, R. Reischuk, C. Scheideler, H. Vollmer, and D. Wagner, eds., *Algorithms Unplugged*, Springer, 2011.
- [11] A.R. Karlin, “On the performance of competitive algorithms in practice,” *Online Algorithms, LNCS*, vol.1442, pp.373–384, Springer, 1998.
- [12] S. Irani, S. Shukla, and R. Gupta, “Online strategies for dynamic power management in systems with multiple power-saving states,” *ACM Trans. Embed. Comput. Syst.*, vol.2, no.3, pp.325–346, 2003.
- [13] J. Augustine, S. Irani, and C. Swamy, “Optimal power-down strategies,” *SIAM J. Comput.*, vol.37, no.5, pp.1499–1516, 2008.
- [14] A.R. Karlin, M.S. Manasse, L.A. McGeogh, and S. Owicki, “Competitive randomized algorithms for nonuniform problems,” *Algorithmica*, vol.11, no.6, pp.542–571, 1994.
- [15] A. Levi and B. Patt-Shamir, “Non-additive two-option ski rental,” *Theor. Comput. Sci.*, vol.584, pp.42–52, 2015.
- [16] Y. Azar, I.R. Cohen, and A. Roytman, “Online lower bounds via duality,” *Proc. SODA '17*, pp.1038–1050, 2017.
- [17] Y. Azar, “On-line load balancing,” in Fiat and Woeginger, pp.178–195.
- [18] J. Matousek and J. Nešetřil, *Invitation to Discrete Mathematics*, Oxford University Press, New York, NY, USA, 1998.



ing associate professor at University of Electronic Science and Technology, China, in 2012. His research interests include analysis of algorithms, online optimization, and functional optimization.

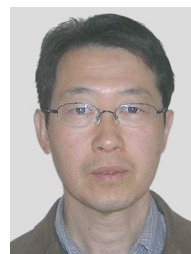
**Hiroshi Fujiwara** received his B.E., Master of Informatics, and Ph.D. in Informatics degrees from Kyoto University, in 2001, 2003, and 2006, respectively. Since 2014 he has been an associate professor in Department of Electrical and Computer Engineering at Shinshu University, Japan, after working at Kwansei Gakuin University and Toyohashi University of Technology. During his career, he worked as a research assistant at University of Freiburg, Germany, from 2004 to 2005, and worked as a visit-



**Kei Shibusawa** received his B.E. in Computer Science and Engineering from Shinshu University in 2018. He joined JRC Engineering Co., Ltd. in 2018. His research interests include analysis of online algorithms for ski-rental problems.



**Kouki Yamamoto** received his B.E. in Computer Science and Engineering from Shinshu University in 2018. He joined AISIN com-Cruise Co., Ltd. in 2018. His research interests include analysis of online algorithms for ski-rental problems.



**Hiroaki Yamamoto** received the B.E. degree in Information Engineering from Shinshu University in 1980, and the M.E. and Ph.D. degrees in Information Engineering from Tohoku University in 1982 and 1985, respectively. In 1988, he joined Shinshu University, and is currently a professor at the same university. His research interests include algorithm theory, automata theory, information security, and information retrieval. He is a member of the IEICE and the IPSJ.