

The Multi-shop Ski Rental Problem

Lingqing Ai
IIIS, Tsinghua University
ailingqing@126.com

Xian Wu
IIIS, Tsinghua University
wuxian12@
mails.tsinghua.edu.cn

Lingxiao Huang
IIIS, Tsinghua University
huanglingxiao1990@126.com

Longbo Huang
IIIS, Tsinghua University
longbohuang@
tsinghua.edu.cn

Pingzhong Tang
IIIS, Tsinghua University
kenshin@tsinghua.edu.cn

Jian Li
IIIS, Tsinghua University
lijian83@tsinghua.edu.cn

ABSTRACT

We consider the *multi-shop ski rental* problem. This problem generalizes the classic ski rental problem to a multi-shop setting, in which each shop has different prices for renting and purchasing a pair of skis, and a *consumer* has to make decisions on when and where to buy. We are interested in the *optimal online (competitive-ratio minimizing) mixed strategy* from the consumer's perspective. For our problem in its basic form, we obtain exciting closed-form solutions and a linear time algorithm for computing them. We further demonstrate the generality of our approach by investigating three extensions of our basic problem, namely ones that consider costs incurred by entering a shop or switching to another shop. Our solutions to these problems suggest that the consumer must assign positive probability in *exactly one* shop at any buying time. Our results apply to many real-world applications, ranging from cost management in IaaS cloud to scheduling in distributed computing.

Categories and Subject Descriptors

G.3 [Probability and Statistics]: Distribution functions;
F.2 [Analysis of Algorithms and Problem of Complexity]: Miscellaneous

General Terms

Algorithms, Performance, Theory

Keywords

multi-shop ski rental, ski rental, optimal strategy, Nash equilibrium, online algorithm

1. INTRODUCTION

The ski rental problem (SR) is a dilemma faced by a consumer, who is uncertain about how many days she will ski

and has to trade off between buying and renting skis: once she buys the skis, she will enjoy the remaining days rent-free, but before that she must pay the daily renting cost. The literature is interested in investigating the *online optimal strategy* of the consumer. That is, a strategy that yields the lowest competitive ratio *without having any information of the future* (as is standard in the literature, competitive ratio is defined as the ratio between the cost yielded by the consumer's strategy and the cost yielded by the optimal strategy of a prophet, who foresees how many days the trip will last and design the optimal strategy accordingly). The ski rental problem and its variants constitute an important part of the online algorithm design literature from both theoretical and applied perspectives [9, 12, 16, 17, 18, 22].

In this paper, we consider the *multi-shop ski rental problem* (MSR), in which the consumer faces multiple shops that offer different renting and buying prices. She must choose one shop immediately after she arrives at the ski field and must rent or buy the skis *in that particular shop* since then. In other words, once she has chosen a shop, the only decision variable is when to buy the skis. Beyond the basic setting, we also propose three important extensions of MSR as below:

- MSR *with switching cost* (MSR-S): The consumer is allowed to switch from one shop to another and each switching costs her some constant amount of money.
- MSR *with entry fee* (MSR-E): Each shop requires some entry fee and the consumer *cannot* switch shops.
- MSR *with entry fee and switching* (MSR-ES): The consumer is able to switch from one shop to another, and she pays the entry fee as long as she enters any shop¹.

In all the settings above, the consumer's objective is to minimize the competitive ratio. In MSR and MSR-E, she has to consider two questions *at the very beginning*: (1) where should she rent or buy the skis (place), and (2) when should she buy the skis (timing)? While MSR-S and MSR-ES allow the consumer to switch shops and are thus more fine-grained than the previous two, in the sense that she is able to decide where to rent or buy the skis *at any time*. For example, it is among her options to rent in shop 1 on day 1, switch to shop 2 from day 2, and finally switch to shop 3 and then buys the skis.

¹For example, if she switches from shop 1 to shop 2, and then switches back to shop 1, she pays the entry fee of shop 1 twice and the entry fee of shop 2 once.

The multi-shop ski rental problem naturally extends the ski rental problem and allows heterogeneity in consumer’s options, a desirable feature that makes the ski rental problem a more general modeling framework for online algorithm design. Below, we present a few real world scenarios that can be modeled with the multi-shop ski rental problem.

1. Scheduling in distributed computing: A file is replicated and stored in different machines in the cluster. Some node undertaking some computing job needs data in the file during the execution. The node can either request the corresponding data block of the file from some selected machine whenever it needs to, which incurs some delay, or it can simply ask that machine to transmit the whole file beforehand, at the sacrifice of a longer delay at the beginning without any further waiting. When selecting the replicating machine, the scheduling node needs to consider the current bandwidth, read latency, etc. In this application, each replicating machine is considered as a shop, and renting corresponds to requesting for the data block on-demand while buying means to fetch the whole file beforehand.

2. Cost management in IaaS cloud: Multiple IaaS cloud vendors, such as Amazon EC2 [1], ElasticHosts [4] and Microsoft Windows Azure [6], offer different price options, which can be classified into two commitment levels: users pay for *on-demand* server instances at an hourly rate or make a one-time, upfront payment to host each instances for some duration (e.g., monthly or annually), during which users either use the instances for free [4], or enjoy a discount on renting the instance [1].

Consider an example in Table 1. Table 1 lists the pricing options for the instances with identical configurations offered by Amazon EC2 and ElasticHosts. Each pricing option can be considered as a shop in the multi-shop ski rental problem, where in the 1(3) year(s) term contract in Amazon EC2, the entry fee is the upfront payment and the hourly price is the renting price.

Vendor	Option	Upfront(\$)	Hourly(\$)
Amazon	On-Demand	0	0.145
	1 yr Term	161	0.09
	3 yr Term	243	0.079
ElasticHosts	1 mo Term	97.60	0
	1 yr Term	976.04	0

Table 1: Pricing Options of the ‘same’ instance in Amazon EC2 (c1.medium) and ElasticHosts (2400MHz cpu, 1792MHz memory, 350Gb storage and no data transfer).

3. Purchase decisions: A company offering high-resolution aerial or satellite map service chooses between Astrium [2] and DigitalGlobe [3]. It can either subscribe imagery from one company or exclusively occupy the service by ‘purchasing’ one satellite like what Google has done [19]. Similar applications include some person purchasing a SIM card from different telecommunication companies.

1.1 Related Work

The ski rental problem is first considered by Karlin *et al.* [14], and then studied by Karlin’s seminal paper [13] which proposes a randomized algorithm and gives a $\frac{e}{e-1}$ competitive ratio. Later researchers propose a few variants, including the Bahncard problem [9] and the TCP acknowledgment

problem [12]. A more recent work [15] analyzes the case in which the first or the second moment of the skiing days are known and gives an optimal online solution. However, all the aforementioned works deal with the case where a single consumer rents or buys the skis in one single shop. In their problems, the consumer only needs to decide when to buy. While in the multi-shop ski rental problem, the consumer has to make a two-fold decision (time and place). Closest to our work is the work by Lotker *et al.* [17], which considers the case where the consumer has multiple options in one shop, i.e., the multi-slop problem and their problem can be regarded as a special case of our problem by setting all the buying prices sufficiently large.

Research on “multiple consumers in one single shop” have been conducted from applied perspectives [16, 18, 22]. Lin *et al.* [16] investigate a dynamical ‘right-sizing’ strategy by turning off servers during periods of low load. Their model and lazy capacity provisioning algorithms closely tie to the ski rental problem. Lu *et al.* [18] derive the “dynamic provisioning techniques” to turn on or off the servers to minimize the energy consumption. They dispatch the servers so that each individual server is reduced to a standard ski rental problem. Wei *et al.* [22] propose an online algorithm to serve the time-varying demands at the minimum cost in IaaS cloud given one price option, in which the ‘consumers’ (servers) may be related to each other.

Another line of work [7, 11] focusing on minimizing the cost in data centers or other cloud services assumes that the long-term workloads are stationary and thus can be predicted, and Guenter *et al.* [10] consider the cases of short predictions. However, for many real-world applications, future workloads can exhibit non-stationarity [20]. Other researchers [16, 18, 21, 22] that require *no priori* knowledge of the future workload minimize the cost *given one option is selected*. Our paper is orthogonal to theirs since we focus on how to select a price option.

1.2 Our Contributions

In this paper, we consider the multi-shop ski-rental problem and its extensions, in which there are multiple shops and the consumer must make two-fold decisions (time and place) to minimize the competitive ratio. We model each problem using a zero-sum game played by the consumer and nature. We simplify the strategy space of the consumer via *removal of strictly dominated strategies* and derive the form of the optimal mixed strategy of the consumer. We summarize the key contributions as follows:

1. For each of the problems, we prove that under the optimal mixed strategy of the consumer, the consumer only assigns positive buying probability to *exactly one* shop at any time. As the buying time increases, she follows the shop order in which the ratio between buying price and renting price is increasing. This order also holds in MSR-E and MSR-ES, where entry fee is involved.
2. We derive a novel, easy-to-implement *linear time* algorithm for computing the optimal strategy of the consumer, which drastically reduces the complexity of computing the solution to MSR.
3. For MSR-S, we prove that under the optimal mixed strategy, the consumer only needs to consider switching to another shop at the buying time, i.e., she will

never switch to another shop and continue renting. Moreover, we show that **MSR-S** can be reduced to an equivalent **MSR** problem with modified buying prices.

4. For **MSR-ES**, we prove that under the optimal mixed strategy, the consumer may switch to another shop either during the renting period or at the buying time, but she only follows some particular order of switching. Moreover, the number of times of switching is no more than n where n is the number of shops.
5. We characterize any action of the consumer in **MSR-ES** by proving that the action can be decoupled into a sequence of operations. We further show that each operation can be viewed as a virtual shop in **MSR-E** and in total, we create $O(n^2)$ ‘virtual’ shops of **MSR-E**. Therefore, **MSR-ES** can be reduced to **MSR-E** with minor modifications.

Due to limited space, most of the proofs are omitted and the readers can find them in our technical report [5].

2. BASIC PROBLEM

In the *multi-shop ski rental problem* (also **MSR**), a person goes skiing for an ambiguous time period. There are multiple shops providing skies either for rental or for buying. The person must choose one shop *as soon as* she arrives at the ski field, and she can decide whether or not to buy the skis in that particular shop at any time². Note that she cannot change the shop once she chooses one. The objective is to minimize the worst-case ratio between the amount she actually pays and the money she would have paid if she knew the duration of skiing in advance. We assume that there are n shops in total, denoted by $[n] \triangleq \{1, 2, 3, \dots, n\}$. Each shop j offers skis at a renting price of r_j dollars per unit time and at a buying price of b_j dollars. This problem is a natural extension of the classic ski rental problem (**SR**) and it is exactly **SR** when $n = 1$.

In **MSR**, it is clear that if there is a shop of which the rental and buying prices are both larger than those of another shop, it is always suboptimal to choose this shop. We assume that

$$0 < r_1 < r_2 < \dots < r_n \\ b_1 > b_2 > \dots > b_n > 0$$

We apply a game-theoretic approach for solving our problem. For the case of expressing the formulation, we assume that how long the consumer skis is determined by a player called *nature*. Therefore, there are two parties in the problem, and we focus on the optimal strategy of the consumer.

In the remainder of this section, we first formulate our problem as a zero-sum game, and simplify the strategy space in Lemma 1. Then, we combine Lemma 2-5, and fully characterize the optimal strategy of the consumer in Theorem 1. We show that optimally the consumer assigns positive buying probability to *exactly one* shop at any time. Moreover, the possible times the consumer buys the skis in a shop constitute a continuous interval. Thus, we can partition the optimal strategy of the consumer into different sub-intervals which relate to different shops, and the problem is reduced to how to find the optimal breakpoints. Based on Lemma 7 and 8, we develop a linear time algorithm for computing the optimal breakpoints and prove its correctness in Theorem 2.

² In this paper, we focus on the continuous time model.

2.1 Formulation

We first analyze the action set for both players in the game. For the consumer, we denote by j the index of the shop in which she rents or buys the skis. Let x be the time when she chooses to buy the skis, i.e., the consumer will rent the skis before x and buy at x if nature has not yet stopped her. The action of the consumer is thus represented by a pair (j, x) . Denote by Ψ_c the action set of the consumer:

$$\Psi_c \triangleq \{(j, x) : j \in [n], x \in [0, +\infty) \cup \{+\infty\}\}$$

where $x = +\infty$ means that the consumer always rents and never buys. Next, let y denote the time when nature stops the consumer from skiing. Thus, the action set of nature is

$$\Psi_n \triangleq \{y : y \in (0, +\infty) \cup \{+\infty\}\}$$

where $y = +\infty$ means that nature always allows the consumer to keep skiing. If $y = x$, we regard it as the case that right after the consumer buys the skis, nature stops her. Given the strategy profile $\langle (j, x), y \rangle$, let $c_j(x, y) \geq 0$ denote the cost paid by the consumer:

$$c_j(x, y) \triangleq \begin{cases} r_j y, & y < x \\ r_j x + b_j, & y \geq x \end{cases}$$

Now we define the strategy space for the consumer and nature. Let $\mathbf{p} \triangleq (p_1, \dots, p_n)$ be a mixed strategy represented by a vector of probability density functions. $p_j(x)$ is the density assigned to the strategy (j, x) for any $j = 1, \dots, n$ and $x \in [0, +\infty) \cup \{+\infty\}$. In this paper, we assume that for each point, either probability density function exists or it is probability mass.³ If $p_j(x)$ is probability mass, we regard $p_j(x)$ as $+\infty$ and define $p_{j,x} \triangleq \int_{x-}^x p_j(t) dt$ satisfying $p_{j,x} \in (0, 1]$. The strategy space \mathcal{P} of the consumer is as follows:⁴

$$\mathcal{P} = \left\{ \mathbf{p} : \begin{aligned} & \sum_{j=1}^n \int_0^{\infty} p_j(x) dx = 1, \\ & p_j(x) \geq 0, \forall x \in [0, +\infty) \cup \{+\infty\}, \forall j \in [n] \end{aligned} \right\}$$

Similarly, define $q(y)$ to be the probability density of nature choosing y and the strategy space \mathcal{Q} of nature is given by

$$\mathcal{Q} = \left\{ \mathbf{q} : \int_0^{\infty} q(y) dy = 1, q(y) \geq 0, \forall y \in (0, +\infty) \cup \{+\infty\} \right\}$$

When the consumer chooses the mixed strategy \mathbf{p} and nature chooses the stopping time y , the expected cost to the consumer is:

$$C(\mathbf{p}, y) \triangleq \sum_{j=1}^n C_j(p_j, y)$$

³In fact, our results can be extended to the case where in the strategy space the cumulative distribution function is not absolutely continuous and thus no probability density function exists.

⁴For convenience, we denote by $\int_a^b f(x) dx$ ($a < b$) the integral over $(a, b]$, except that when $a = 0$, the integral is over $[0, b]$.

in which

$$\begin{aligned} C_j(p_j, y) &\triangleq \int_0^\infty c_j(x, y)p_j(x)dx \\ &= \int_0^y (r_j x + b_j)p_j(x)dx + \int_y^\infty yr_j p_j(x)dx \end{aligned}$$

is the expected payment to shop j for all $j \in [n]$. Given the strategy profile $\langle \mathbf{p}, \mathbf{q} \rangle$, the competitive ratio is defined as:

$$R(\mathbf{p}, \mathbf{q}) \triangleq \int_0^\infty \frac{C(\mathbf{p}, y)}{\text{OPT}(y)} q(y) dy \quad (1)$$

Here $\text{OPT}(y)$ is the optimal offline cost and can be seen to have the following form:

$$\text{OPT}(y) = \begin{cases} r_1 y, & y \in (0, B] \\ b_n, & y > B \end{cases} \quad (2)$$

where B is defined as $B \triangleq \frac{b_n}{r_1}$.

Note that B is the dividing line between the minimum buying cost and the minimum renting cost. When $y < B$, the offline optimal is always to rent at the first shop, and when $y > B$, the offline optimal is to buy the skis at the last shop. We will show that B determines the effective action sets of the consumer and nature in section 2.1.1.

The objective of the consumer is to minimize the worst-case competitive ratio, i.e., to choose a strategy $\mathbf{p} \in \mathcal{P}$ that solves the problem

$$\begin{aligned} &\text{minimize} && \max_{y>0} \left\{ \frac{C(\mathbf{p}, y)}{\text{OPT}(y)} \right\} \\ &\text{subject to} && \mathbf{p} \in \mathcal{P} \end{aligned}$$

which is equivalent to the following:

$$\begin{aligned} &\text{minimize} && \lambda && (3) \\ &\text{subject to} && \frac{C(\mathbf{p}, y)}{r_1 y} \leq \lambda \\ &&& \sum_{j=1}^n \int_0^\infty p_j(x) dx = 1 \\ &&& p_j(x) \geq 0 \quad \forall x \in [0, +\infty) \cup \{+\infty\} \\ &&& \forall y \in (0, +\infty) \cup \{+\infty\}, \forall j \in [n] \end{aligned}$$

2.1.1 Simplifying the Zero-sum Game

In this section, we show that the game can be greatly simplified and the action set for both the consumer and nature can be reduced. Specifically, nature prefers the strategy $y = +\infty$ to any other strategy $y' > B$. For the consumer, for any $j \in [n]$, she prefers the strategy (j, B) to any other strategy (j, x') where $x' > B$.

LEMMA 1. *For nature, any strategy $y \in [B, +\infty)$ is dominated. While for the consumer, any strategy (j, x) is dominated, in which $x \in (B, +\infty) \cup \{+\infty\}$, $\forall j \in [n]$.*

PROOF. Recall the cost $c_j(x, y)$ is defined as follows:

$$c_j(x, y) = \begin{cases} r_j y, & y < x \\ r_j x + b_j, & y \geq x \end{cases}$$

Thus for any fixed (j, x) , $c_j(x, y)$ is a non-decreasing function of y . Further, from (2), we can see that the offline optimal

cost is unchanged when $y \geq B$. Thus, for any $y \geq B$ it holds that

$$\frac{c_j(x, y)}{b_n} \leq \lim_{y \rightarrow +\infty} \frac{c_j(x, y)}{b_n} = \frac{r_j x + b_j}{b_n}$$

Therefore, any strategy of nature that includes $y \geq B$ is dominated by the strategy of never stopping the consumer.

Now for the consumer, for any shop $j \in \{1, \dots, n\}$, and any $x' \in (B, +\infty) \cup \{+\infty\}$, it holds that

$$c_j(B, y) - c_j(x', y) \leq 0, \quad \forall y \in (0, B) \cup \{+\infty\}$$

Therefore, any strategy of the consumer that includes buying at time x' in any shop is dominated by the strategy of buying at B in the same shop. \square

From this lemma, the consumer's buying time is restricted in $[0, B]$. Note that for any (j, x) in which $x \in [0, B]$, it holds that

$$\frac{c_j(x, B)}{\text{OPT}(B)} = \frac{c_j(x, +\infty)}{\text{OPT}(+\infty)}$$

Therefore, the action set of nature Ψ_n can be reduced to $\Psi_n = \{y \in (0, B]\}$.

Similarly, in the strategy space of the consumer \mathcal{P} , nature \mathcal{Q} , the expected cost $C(\mathbf{p}, y)$ and the competitive ratio $R(\mathbf{p}, \mathbf{q})$, we can replace $+\infty$ by B .

Comments on B: recall that the boundary B is defined as $\frac{\min\{b_i\}}{\min\{r_i\}} = \frac{b_n}{r_1}$ in **MSR**, while this value is $\frac{b_j}{r_j}$ if shop j is the only shop in **SR**. For instance, if only shop n appears in **SR**, then the consumer will never consider to buy at any time $x > \frac{b_n}{r_n}$. However, in **MSR**, the consumer may want to put some positive possibility to the strategy of buying at time $x > \frac{b_n}{r_n}$ in shop n (since $r_1 < r_n$). The difference between these two cases is due to the fact that in **MSR**, the consumer has the global information of all the shops and the offline optimal is always to rent at shop 1 at the cost of r_1 per unit time until the total cost reaches the minimum buying price b_n , whereas in **SR**, the consumer always rents at the cost of $r_n \geq r_1$ per unit time until b_n .

With the above results, problem (3) can now be reduced to the following:

$$\text{minimize} \quad \lambda \quad (4)$$

$$\text{subject to} \quad \frac{C(\mathbf{p}, y)}{r_1 y} \leq \lambda \quad (4a)$$

$$\sum_{j=1}^n \int_0^B p_j(x) dx = 1 \quad (4b)$$

$$p_j(x) \geq 0 \quad (4c)$$

$$\forall x \in [0, B], \forall y \in (0, B], \forall j \in [n] \quad (4d)$$

We will show that the optimal strategy of the consumer results in exact equality in (4a) in the next subsection.

2.2 Optimal Strategy of the Consumer

In this subsection, we look into the optimal solution \mathbf{p}^* for (4). In short, \mathbf{p}^* yields the same expected utility for nature whenever nature chooses to stop. In other words, given \mathbf{p}^* , any pure strategy of nature yields the same utility for both the consumer and nature. Moreover, at any time x , the consumer assigns positive buying probability to exactly one of the shops, say shop j , and j is decreasing as x increases. Finally, we can see that for any shop j and the time interval

in which the consumer chooses to buy at shop j , the density function $p_j(x)$ is $\alpha_j e^{r_j x/b_j x}$ where α_j is some constant to be specified later.

We now state our first theorem that summarizes the structure of the optimal strategy.

THEOREM 1. *The optimal solution \mathbf{p}^* satisfies the following properties:*

(a) *There exists a constant λ , such that $\forall y \in (0, B]$,*

$$\frac{C(\mathbf{p}^*, y)}{r_1 y} = \lambda$$

(b) *There exist $n + 1$ breakpoints: d_1, d_2, \dots, d_{n+1} , such that $B = d_1 \geq d_2 \geq \dots \geq d_n \geq d_{n+1} = 0$, and $\forall j \in [n]$, we have*

$$p_j^*(x) = \begin{cases} \alpha_j e^{r_j x/b_j}, & x \in (d_{j+1}, d_j) \\ 0, & \text{otherwise} \end{cases}$$

in which α_j satisfies that

$$\alpha_j b_j e^{r_j d_j/b_j} = \alpha_{j-1} b_{j-1} e^{r_{j-1} d_j/b_{j-1}} \quad \forall j = 2, \dots, n$$

In the following, We will prove property (a) by Lemma 2, property (b) by Lemma 3-5. All proof details can be found in our technical report [5].

LEMMA 2. *$\forall y \in (0, B]$, \exists a constant λ , such that \mathbf{p}^* satisfies that*

$$\frac{C(\mathbf{p}^*, y)}{r_1 y} = \lambda \quad (5)$$

PROOF. (sketch) By reductio ad absurdum, we assume that $\frac{C(\mathbf{p}^*, y)}{r_1 y}$, which can be considered as a function with respect to y , is not a constant. Then we can find a valley in the graph of this function and there are 3 cases depending on the positions of the maximum points of the function:

- Maximum points only appear on the left side of the valley;
- Maximum points only appear on the right side of the valley;
- Maximum points appear on the both side of the valley.

Then we prove that there is a better strategy in each case which makes contradiction. \square

From the above lemma, the problem (4) is thus equivalent to the following:

$$\begin{aligned} & \text{minimize} && \lambda && (6) \\ & \text{subject to} && (5), (4b), (4c), (4d) \end{aligned}$$

Here are some intuitions of MSR: In the extreme case where the buying time x is sufficiently small, the consumer will prefer shop n than any other shops since b_n is the minimum buying price. As x increases, the renting cost weights more and the skier gradually chooses the shop with lower rent yet higher buying price. In the other extreme case when the skier decides to buy at time x close to B , shop 1 may be the best place since it has the lowest rent. Thus, in the optimal strategy, the interval $[0, B]$ may be partitioned into several sub-intervals. In each interval, the consumer only chooses to buy at one and only one shop. The following two lemmas formally show that the above intuitions are indeed the case.

LEMMA 3. *$\forall j \in [n]$, we have $p_j^*(0) < +\infty$, and $\forall x \in (0, B]$, $p_j^*(x) < \frac{2b_1 r_1}{b_n^2}$.*

LEMMA 4. *In the optimal strategy \mathbf{p}^* , there exists $n + 1$ breakpoints $B = d_1 \geq d_2 \geq \dots \geq d_{n+1} = 0$, which partition $[0, B]$ into n sub-intervals, such that $\forall j = 1, \dots, n$, $\forall x \in (d_{j+1}, d_j)$, $p_j^*(x) > 0$ and $p_i^*(x) = 0$ for any $i \neq j$.*

PROOF. (sketch) It suffices to show that $\forall x \in (0, B)$, $\forall \epsilon > 0$, if there exists some j such that $\int_{x-\epsilon}^x p_j^*(t) dt > 0$, then $\forall j' > j$, $x' \geq x$, we must have $\int_{x'}^{B} p_{j'}^*(t) dt = 0$. We use reductio ad absurdum to prove this proposition.

We first show that if there exists some $j' > j$, $x' > x$, $\epsilon > 0$ such that $\int_{x-\epsilon}^x p_j^*(t) dt > 0$, $\int_{x'}^{x'+\epsilon} p_{j'}^*(t) dt > 0$, then there exist 2 intervals $(x_1, x_1 + \theta) \subseteq (x - \epsilon, x)$ and $(x_2, x_2 + \theta) \subseteq (x', x' + \epsilon)$, such that

$$\int_0^{\epsilon_0} \min\{p_j^*(x_1 + \theta), p_{j'}^*(x_2 + \theta)\} d\theta > 0$$

We next move some suitable buying probabilities of $p_{j'}^*$ from $(x_2, x_2 + \theta)$ to $(x_1, x_1 + \theta)$ for shop j' , and correspondingly move some purchase probabilities of p_j^* from $(x_1, x_1 + \theta)$ to $(x_2, x_2 + \theta)$ for shop j . Then we obtain a new strategy \mathbf{p}^1 . We show that $\forall y \in (0, B]$, \mathbf{p}^1 is no worse than \mathbf{p}^* , and $\forall y \in (x_1, B]$, \mathbf{p}^1 is strictly better than \mathbf{p}^* , which makes a contradiction. \square

The lemma explicitly specifies the order of the shops in the optimal strategy: as x increases, the index of the shop where the consumer assigns positive density decreases. Based on this lemma, for any $j \in [n]$, $x \in (d_{j+1}, d_j)$, multiplying both sides of (5) by $r_1 y$, and taking twice derivatives, we have

$$b_i \frac{dp_j^*(x)}{dx} = r_j p_j^*(x) \quad \forall x \in (d_{j+1}, d_j) \quad (7)$$

Solving this differentiable equation, we obtain the optimal solutions as follows⁵:

$$p_j^*(x) = \begin{cases} \alpha_j e^{r_j x/b_j}, & x \in (d_{j+1}, d_j) \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

where α_j is some constant. The relationship between α_j and α_{j-1} is described in the following lemma:

LEMMA 5.

$$\alpha_j b_j e^{r_j d_j/b_j} = \alpha_{j-1} b_{j-1} e^{r_{j-1} d_j/b_{j-1}} \quad \forall j = 2, \dots, n \quad (9)$$

2.3 Computing the Optimal Strategy

In this section we propose a linear time algorithm to compute the optimal strategy for the consumer. First we show the relationship between the competitive ratio λ and α_1 by the following lemma:

LEMMA 6. *For any strategy \mathbf{p} which satisfies property (b) in Theorem 1, it holds that*

$$\frac{C(\mathbf{p}, y)}{r_1 y} = \alpha_1 \frac{b_1}{r_1} e^{\frac{r_1}{b_1} B}, \quad \forall y \in (0, B]$$

From the above lemma, we know that minimizing λ is equivalent to minimizing α_1 . Therefore, problem (6) is now equivalent to the following:

$$\text{minimize} \quad \alpha_1 \quad (10)$$

⁵Because $p_j(x)$ is finite, we say $p_j(d_i) = 0$ for all $i, j \in N$, which does not affect the expected cost at all.

$$\begin{aligned} \text{subject to } & \sum_{j=1}^n \alpha_j \frac{b_j}{r_j} \left(e^{\frac{r_j}{b_j} d_{j-1}} - e^{\frac{r_j}{b_j} d_j} \right) = 1 \quad (11) \\ & \alpha_j e^{r_j d_j / b_j} = \alpha_{j-1} e^{r_{j-1} d_j / b_{j-1}}, \forall j = 2, \dots, n \\ & \alpha_j > 0, \quad \forall j \in [n] \\ & B = d_1 \geq d_2 \geq \dots \geq d_n \geq d_{n+1} = 0 \end{aligned}$$

where (11) is computed directly from (4b).

In Theorem 1, if we know $(d_1, d_2, \dots, d_{n+1})$, then we can see that α_j is proportional to α_1 . Therefore, we can get a constant Ω_j such that $\Omega_j \alpha_1 = \int_{d_{j+1}}^{d_j} p_j(x) dx$ since the breakpoints are known. Finally we can get a constant $\Omega = \sum_{j=1}^n \Omega_j$ such that $\Omega \alpha_1 = \sum_{j=1}^n (\int_0^B p_j(x) dx)$. Using the fact that $\sum_{j=1}^n (\int_0^B p_j(x) dx) = 1$, we can easily solve α_1 , all the α_j and the whole problem.

Therefore, the computation of \mathbf{p} reduces to computing $\{d_1, d_2, \dots, d_{n+1}\}$. Notice that $d_1 \equiv B, d_{n+1} \equiv 0$.

In this case, we treat this problem from another perspective. We first fix α_1 to be 1. After that, without considering the constraint (11), we compute the optimal breakpoints $(d_1, d_2, \dots, d_{n+1})$ to maximize $\sum_{j=1}^n (\int_0^B p_j(x) dx)$. Denote the optimal value of this problem as Ω . We then normalize all the probability functions, i.e., reset all the α_j to be α_j / Ω . By Lemma 5, we know the ratio λ is proportional to α_1 , which is fixed at first and normalized at last. Hence, maximizing Ω is equivalent to minimizing λ . Notice that all the probability functions in the remainder of section 2.3 is unnormalized when $\alpha_1 = 1$.

In the following 2 sections 2.3.1 and 2.3.2, we show some intuitions and ideas of our algorithm about how to compute the breakpoints. In Section 2.3.3, we formally propose our algorithm and prove the optimality and complexity of our algorithm.

2.3.1 Computing d_n

To facilitate further calculations, we denote P_j to be the probability sum of shop j to shop n , i.e.,

$$P_j \triangleq \sum_{\tau=j}^n \left(\int_0^B p_\tau(x) dx \right) = \sum_{\tau=j}^n \left(\int_0^{d_j} p_\tau(x) dx \right)$$

Now we just need to maximize P_1 since by definition $P_1 = \Omega$. To compute some breakpoint d_j , we assume that all the breakpoints $\{d_i : i \neq j\}$ are fixed. Since breakpoints d_1, d_2, \dots, d_{j-1} are fixed, parameters $\alpha_1, \alpha_2, \dots, \alpha_{j-1}$ are constants. Therefore, $\sum_{\tau=1}^{j-2} (\int_0^B p_\tau(x) dx)$, part of the probability sum, is a constant and we just need to maximize the rest of the sum which is P_{j-1} .

First we consider how to compute $\arg \max_{d_n} P_{n-1}(d_n)$ when given d_1, \dots, d_{n-1} , where

$$P_{n-1}(d_n) = \alpha_n \int_0^{d_n} e^{\frac{r_n x}{b_n}} dx + \alpha_{n-1} \int_{d_n}^{d_{n-1}} e^{\frac{r_{n-1} x}{b_{n-1}}} dx$$

Notice that α_{n-1} is a constant but α_n depends on d_n . From Lemma 5 we know that:

$$\alpha_n = \alpha_{n-1} b_{n-1} e^{(r_{n-1}/b_{n-1} - r_n/b_n) d_n} / b_n$$

The following lemma shows the concavity of $P_{n-1}(d_n)$:

LEMMA 7. $P_{n-1}(d_n)$ is a strictly concave function.

Notice that $P'_{n-1}(d_n) > 0$ when $d_n = 0$. This implies that: if $d_n < d_{n-1}$, we must have $P'_{n-1}(d_n) = 0$ since it is concave.

Otherwise, $d_{n-1} = d_n$ which means $\forall x, p_{n-1}(x) = 0$, i.e., shop $n-1$ does not exist. Thus we can delete shop $n-1$ and view shop $n-2$ as shop $n-1$. Similarly, if $d_n < d_{n-2}$, $P'_{n-2}(d_n) = 0$; otherwise delete shop $n-2$ and treat shop $n-3$ as shop $n-1$. Repeat this procedure until we find some shop k , such that $d_n = d_{n-1} = \dots = d_{k+1} < d_k$. Then d_n should be the maximal point because of the concavity derived by Lemma 7, i.e.,

$$d_n = \frac{b_n}{r_n} \ln \left(\frac{b_k r_n - b_n r_k}{b_n (r_n - r_k)} \right)$$

Notice that d_n is always positive.

2.3.2 Computing d_j

Notice that d_n is unrelated to d_{n-1} if $d_n < d_{n-1}$. Therefore, we can work out all the breakpoints d_j in descending order of the subscript of d . Here we show how to obtain d_j after $d_n, d_{n-1}, \dots, d_{j+1}$.

If $j = n$, we just temporarily take

$$d_n = \frac{b_n}{r_n} \ln \left(\frac{b_{n-1} r_n - b_n r_{n-1}}{b_n (r_n - r_{n-1})} \right)$$

If $j \neq n$, our target becomes $\arg \max_{d_j} P_{j-1}(d_j)$. According to the definition, we have

$$P_{j-1}(d_j) = \alpha_j (D_j + \int_0^{d_j} e^{\frac{r_j x}{b_j}} dx) + \alpha_{j-1} \int_{d_j}^{d_{j-1}} e^{\frac{r_{j-1} x}{b_{j-1}}} dx$$

where

$$D_j \triangleq - \int_0^{d_{j+1}} e^{r_j x / b_j} dx + \sum_{\tau=j+1}^n \frac{\alpha_\tau}{\alpha_j} \int_{d_{\tau+1}}^{d_\tau} e^{r_\tau x / b_\tau} dx \geq 0$$

Notice that the breakpoints $d_n, d_{n-1}, \dots, d_{j+1}$ are fixed and we can compute α_τ / α_j by the following equations which is derived from Lemma 5:

$$\alpha_\tau = \alpha_{\tau-1} b_{\tau-1} e^{(r_{\tau-1}/b_{\tau-1} - r_\tau/b_\tau) d_\tau} / b_\tau, \forall \tau \in [n] \setminus [j]$$

Therefore, D_j is a constant.

It can be seen that we can compute D_j recursively, i.e.,

$$D_j = \frac{\alpha_{j+1}}{\alpha_j} (D_{j+1} + \int_0^{d_{j+1}} e^{\frac{r_{j+1} x}{b_{j+1}}} dx - \int_0^{d_{j+1}} e^{\frac{r_j x}{b_j}} dx)$$

Also note that α_{j-1} is a constant but α_j depends on d_j :

$$\alpha_j = \alpha_{j-1} b_{j-1} e^{(r_{j-1}/b_{j-1} - r_j/b_j) d_j} / b_j$$

The following lemma shows that $P_{j-1}(d_j)$ is a quasi-concave function:

LEMMA 8. If $D_j r_j / b_j \geq 1$, we always have $P'_{j-1}(d_j) < 0$; if $D_j r_j / b_j < 1$, $P''_{j-1}(d_j) < 0$, i.e., $P_{j-1}(d_j)$ is strictly concave.

Similarly with the computation of d_n , if $D_j r_j / b_j > 1$, then we always have $P'_{j-1}(d_j) < 0$ and the optimal d_j is d_{j+1} . Hence we delete shop j and treat shop $j-1$ as shop j . Then we need to recompute d_{j+1} and let $d_j = d_{j+1}$; otherwise it is concave and we temporarily get the maximal point:

$$d_j = \frac{b_j}{r_j} \ln \left(\frac{(b_{j-1} r_j - b_j r_{j-1})(1 - D_j r_j / b_j)}{b_j (r_j - r_{j-1})} \right)$$

Here if the temporary d_j is no larger than d_{j+1} , it means that the optimal solution is d_{j+1} because of the constraints

$d_{j+1} \leq d_j \leq d_{j-1}$. So we have $d_j = d_{j+1}$ which means that $\forall x, p_j(x) = 0$. Therefore, we delete shop j and treat shop $j-1$ as shop j . Then recompute d_{j+1} and temporarily skip d_j . At last we set $d_j = d_{j+1}$.

2.3.3 A Linear Time Algorithm

Now we are ready to show our algorithm for computing the optimal strategy of the consumer.

THEOREM 2. *There is an algorithm for computing the unique optimal strategy of the consumer. The time and space complexity of the algorithm are linear.*

We first show how to construct our algorithm, and analyze the correctness and the complexity of our algorithm later.

Since delete operations may be executed frequently in the algorithm, we use a linked list to store the shop info. Each shop is an element in this linked list and the shop index decreases when we traverse from the head to the tail. So the head is shop n and the tail is shop 1. Considering that the shops appear in the form of linked list in the algorithm, we rewrite some equations we may use in the algorithm:

$$\begin{aligned} D_j &= \frac{\alpha_{prev[j]}}{\alpha_j} (D_{prev[j]} + \int_0^{d_{prev[j]}} \exp(\frac{r_{prev[j]}x}{b_{prev[j]}}) dx) \\ &\quad - \int_0^{d_{prev[j]}} \exp(\frac{r_j x}{b_j}) dx \\ &= \frac{\alpha_{prev[j]}}{\alpha_j} D_{prev[j]} - \frac{b_j}{r_j} (\exp(\frac{r_j d_{prev[j]}}{b_j}) - 1) \\ &\quad + \frac{\alpha_{prev[j]} b_{prev[j]}}{\alpha_j r_{prev[j]}} (\exp(\frac{r_{prev[j]} d_{prev[j]}}{b_{prev[j]}}) - 1) \end{aligned} \quad (12)$$

Here $\frac{\alpha_{prev[j]}}{\alpha_j}$ is represented as follow:

$$\begin{aligned} \frac{\alpha_{prev[j]}}{\alpha_j} &= \frac{b_j}{b_{prev[j]}} e^{(\frac{r_j}{b_j} - \frac{r_{prev[j]}}{b_{prev[j]}}) d_{prev[j]}} \\ d_j &= \frac{b_j}{r_j} \ln\left(\frac{(b_{next[j]} r_j - b_j r_{next[j]}) (1 - D_j r_j / b_j)}{b_j (r_j - r_{next[j]})}\right) \end{aligned} \quad (13)$$

Here is the pseudocode of our algorithm.:

Algorithm 1 MSR Algorithm

```

1:  $D_n \leftarrow 0$ ;
2: for  $j \leftarrow 1$  to  $n$  do
3:    $next[j] \leftarrow j - 1$ ;
4:    $prev[j] \leftarrow j + 1$ ;
5: end for
6: for  $j \leftarrow n$  to 2 do
7:    $ComputingBP(j)$ ;
8: end for
9: for  $j \leftarrow n$  to 2 do
10:  if  $d_j \neq$  "decide later" then
11:    if  $d_j > B$  then
12:       $d_j \leftarrow B$ ;
13:    end if
14:  else
15:     $d_j \leftarrow d_{j+1}$ ;
16:  end if
17: end for

```

Though we may revise those breakpoints for many times when running the algorithm, it will still lead to the exact

Algorithm 2 Function $ComputingBP(j)$

```

1: if  $j \neq n$  then
2:   Update  $D_j$  according to (12);
3: end if
4: if  $D_j \geq b_j / r_j$  then
5:    $d_j \leftarrow$  "decide later";
6:    $next[prev[j]] \leftarrow next[j]$ ;
7:    $prev[next[j]] \leftarrow prev[j]$ ;
8:    $ComputingBP(prev[j])$ ;
9: else
10:  Compute  $d_j$  according to (13);
11:  if  $d_j \leq d_{j+1}$  then
12:     $d_j \leftarrow$  "decide later";
13:     $next[prev[j]] \leftarrow next[j]$ ;
14:     $prev[next[j]] \leftarrow prev[j]$ ;
15:     $ComputingBP(prev[j])$ ;
16:  end if
17: end if

```

optimal solution at the end. Since the feasible solution of (d_2, d_3, \dots, d_n) is convex and functions $P_{j-1}(\cdot)$ are always concave, we have the following properties for the optimal solution:

If $d_{j-1} > d_j$, $P'_{j-1}(d_j) \leq 0$; if $d_{j+1} < d_j$, $P'_{j-1}(d_j) \geq 0$.

So in our computation method, we delete a shop when and only when the shop should be deleted in the optimal solution. Notice that Line 5, 6, 7 and Line 15, 16, 17 are what we actually do when we say we delete shop j . We say a shop is *alive* if it has not been deleted. Based on the following lemma, we rigorously prove the correctness and complexity of this algorithm.

LEMMA 9. *After an invocation of $ComputingBP(j)$ is completed, the temporary breakpoints, whose indexes are less than or equal to j , $\mathbf{td} = (td_n, td_{n-1}, \dots, td_j)$ are identical with the optimal solution $\mathbf{d}^* = (d_n^*, d_{n-1}^*, \dots, d_j^*)$ if $d_j^* < d_{next[j]}^*$. And all the deletions are correct, i.e., once a shop j is deleted in the algorithm, d_j^* must be equal to d_{j+1}^* .*

PROOF. (sketch) This lemma is proved by inductions (an outer induction and an inner induction). In the outer induction, we prove the lemma for the invocation of $ComputingBP(j)$ under the 2 assumptions:

- all the deletions in the past is correct;
- $\forall \tau > j$, the lemma holds for $ComputingBP(\tau)$

In each step of the outer induction, we consider 2 cases:

- 1 one or more shops are deleted during this invocation;
- 2 no shop is deleted during this invocation.

For Case 1, we construct a sequence of vectors and prove the correctness of Case 1 based on its monotonicity. The monotonicity of the sequence is also proved by induction (the inner induction). And Case 2 is proved based on Case 1. \square

Here $\mathbf{td} = (td_n, td_{n-1}, \dots, td_j)$ are the temporary values of d_n, d_{n-1}, \dots, d_j just after this invocation, $d_n^*, d_{n-1}^*, \dots, d_j^*$ are the optimal breakpoints, and $next[\cdot]$ and $prev[\cdot]$ denote the current state of the linked list, not the eventual result.

PROOF. (Theorem 2) We first show the correctness of the algorithm. According to Lemma 9, we know that all the deletions are correct. Also, we know that $\forall j_1, j_2$ such that $1 < j_1 < j_2$, and that shop j_1 and shop j_2 are alive, $td_{j_1} > td_{j_2}$ when the algorithm terminates. There are 2 cases:

Case 1: $B = d_1^* > d_{prev[1]}^*$, the final solution is the unique optimal solution by Lemma 9.

Case 2: $B = d_1^* = d_{prev[1]}^*$. Similar to the proof of Case 1 in Lemma 9, the solution of the alive breakpoints $\mathbf{td}^* = (td_n^*, td_{next[n]}^*, \dots, td_{prev[1]}^*, d_1^*)$, satisfying that $\forall \tau \in [n], td_\tau^* = \min\{td_\tau, d_1^*\}$, is the unique optimal solution.

Next, we analyze the complexity of the algorithm. Obviously, the space complexity is $O(n)$. For the time complexity, we just need to prove that this algorithm invokes the function *ComputingBP* for $O(n)$ times. The main function invokes *ComputingBP* for $O(n)$ times. And notice that a shop is deleted once *ComputingBP* is invoked recursively by *ComputingBP*. The algorithm can delete $O(n)$ shops at most, therefore the total number of the invocations is $O(n)$. \square

2.4 Optimal Strategy of Nature

Now we briefly show the optimal strategy \mathbf{q}^* of nature in the following theorem:

THEOREM 3. *Suppose that the probability for nature of choosing $y = B$ (actually $y = +\infty$) is q_B^* . The optimal strategy \mathbf{q}^* of nature satisfies the following properties:*

- (a) $\forall j > 1$, we have $\int_{d_{j-1}^*}^{d_j^*} q^*(y) dy = 0$.
- (b) $q^*(y) = \beta_j y e^{-\frac{r_j}{b_j} y}$, $y \in (d_{j+1}^*, d_j^*)$.
- (c) $\beta_1 = (q_B r_1 / B b_1) e^{\frac{r_1}{b_1} B}$.
- (d) $\frac{b_i}{r_j} \beta_j e^{-\frac{r_j}{b_j} d_j^*} = \frac{b_{j-1}}{r_{j-1}} \beta_{j-1} e^{-\frac{r_{j-1}}{b_{j-1}} d_j^*}$, $j = 2, 3, \dots, n$.

PROOF. (sketch) When the consumer chooses a pure strategy (j, x) and nature chooses \mathbf{q}^* , the competitive ratio is as follows:

$$R_j(x, \mathbf{q}^*) \triangleq \int_0^x \frac{r_j}{r_1} q^*(y) dy + \int_x^B \frac{r_j x + b_j}{r_1 y} q^*(y) dy$$

We know that \mathbf{p}^* and \mathbf{q}^* reach a Nash equilibrium with a competitive ratio λ and that $p_j^*(x) > 0$ when $x \in (d_{j+1}^*, d_j^*)$. Therefore, $R_j(x, \mathbf{q}^*) = \lambda$ when $x \in (d_{j+1}^*, d_j^*)$ and the 4 properties of \mathbf{q}^* can be easily derived from it. \square

Recall that \mathbf{d}^* is computed by Algorithm 1. Thus $\forall j \in [n]$, we can compute β_j / q_B and we can obtain $1/q_B$ by computing the sum $\sum_{\tau=1}^n \int_{d_\tau}^{d_{\tau+1}} q(y) / q_B dy$. Since the sum of probability is 1, i.e., $\int_0^B q(y) dy = 1$, it is not hard to work out \mathbf{q}^* after normalization.

2.5 Including Switching Cost

In this subsection, we consider an extension, the *multi-shop ski rental with switching cost problem* (also MSR-S), in which we allow the consumer switches from one shop to another, but with some extra fee to be paid. At any time, suppose the consumer chooses to switch from shop i to shop j , she has to pay for an extra switching cost $c_{ij} \geq 0$. If there exists some $i \neq j$, such that $c_{ij} = +\infty$, then the consumer cannot *directly* switch from shop i to shop j . Consider the following 2 cases:

- If the consumer is allowed to switch from shop to shop freely, i.e. the switching cost is always 0, she will optimally rent at the shop with the lowest renting price and buy at the shop with the lowest buying price. All that she concerns is when to buy the skies. Thus, this problem (MSR-S) is reduced to the basic ski rental problem (SR).
- If the switching cost is always $+\infty$, she will never switch to another shop and the MSR-S becomes MSR.

We will prove that the consumer never switches between shops even in MSR-S later.

The settings of MSR-S can be viewed as a directed graph $G = (V, A)$, where $V = \{1, \dots, n\}$, and $A = \{(i, j) : c_{ij} < +\infty\}$. Each arc $(i, j) \in A$ has a cost c_{ij} . We define a path $\mathbf{p} \subseteq G$ as a sequence of arcs. Define the cost of \mathbf{p} as the summation of the costs of all arcs on \mathbf{p} . Note that if the consumer is allowed to switch from shop i to shop j ($i \neq j$), there must be a path \mathbf{p} which starts at i and ends at j .

It is clear that if the consumer decides to switch from shop i to shop j ($(i, j) \in A$) at any time, she will choose the shortest path from i to j in the graph G . Denote the cost of the shortest path from i to j by c_{ij}^* . We obtain that $c_{ij}^* \leq c_{ij}$, for any $(i, j) \in A$. Moreover, for any different $i, j, k \in V$ such that $(i, j), (j, k), (i, k) \in A$, we have:

$$c_{ik}^* \leq c_{ij}^* + c_{jk}^* \quad (14)$$

Comparing to MSR, MSR-S has a much richer action set for the consumer, where the consumer is able to choose where to rent and for how long to rent at that shop.

Although we allow the consumer to switch from shop to shop as many times as she wants, the following lemma shows that the consumer will never choose to switch to another shop and continue renting, i.e. the only moment that the consumer will switch is exactly when she buys the skis.

LEMMA 10. *Any strategy which includes switching from one shop to another shop and continually renting the skis is dominated.*

PROOF. Given an action ψ , we assume that the buying time in ψ is x and there exists a switching operation strictly before x . Now we need to prove that ψ is dominated.

We assume that the first switching operation in ψ is from shop i to shop j at time x_0 ($0 \leq x_0 < x$). If there are more than one switching operations at time x_0 , they can be considered as a big switching operation. But the switching cost is the sum of all the switching cost at time x_0 . We assume that the last switching operation at x_0 is from some shop to shop k . If the switching cost at x_0 is larger than c_{ik}^* , then obviously it is dominated. But if not, then there will be 2 cases.

Case 1 : when $r_i < r_k$,

Denote by x_1 the switching time just after x_0 . But if switching only happens at time x_0 in ψ , we let x_1 to be x . It can be seen that $x_0 < x_1$. Since $r_i \leq r_k$, in action ψ it is better for the consumer to move the switching operation(s) from time x_0 to time x_1 . It can be verified that the cost remains the same when $y < x_0$ and decreases when $y \geq x_0$. Thus, ψ is dominated.

Case 2 : when $r_i \geq r_k$,

Consider another action ψ^1 : the consumer enters shop k at the very beginning and does not execute any switching operation at time x_0 , but she acts the same as ψ after time

x_0 . It can be verified that the cost does not increase when $y < x_0$ and decreases when $y \geq x_0$. Thus, it is dominated by ψ^1 . \square

This lemma significantly reduces the action set of the consumer to the same one as MSR. Thus, if the consumer considers to switch to another shop at the buying time. Further, she chooses a shop such that the sum of the buying cost and the switching cost (if any) is minimized. Once the consumer decides to switch for buying, she will switch at most once, since the buying cost only increases otherwise. Therefore, for any strategy, suppose s is the shop in which the consumer rents the skis right before the buying time, we define the buying price of s as follows:

$$b'_s = \min\{b_s, b_j + c_{sj}, \forall j \neq s\}$$

Observe that once s is settled, b'_s is settled. As a result, MSR-S is reduced to MSR, in which for any shop j , the rent is still r_j per unit time while the buying price b'_j is $\min\{b_j, \min_{i \neq j}\{b_i + c_{ji}\}\}$.

3. SKI RENTAL WITH ENTRY FEE

In this section, we discuss another extension of MSR, the *multi-shop ski rental with entry fee included problem* (MSR-E). In this problem, all the settings are the same to those of MSR, except that each shop has an entry fee. Once the consumer enters a shop, she pays for the entry fee of this shop and cannot switch to another shop. Our goal is to minimize the worst case competitive ratio. Notice that MSR can be viewed as a special case of MSR-E in which the entry fee of each shop is zero.

We introduce this problem not only as an extension of MSR, but more importantly, as a necessary step to solve a more general extension, the (MSR-ES) problem in next section. We will show that MSR-ES can be converted into MSR-E with minor modifications.

3.1 Single Shop Ski Rental with Entry Fee

We start by briefly introducing the special case of MSR-E when $n = 1$. The entry fee, renting price and buying price are supposed to be $a \geq 0$, $r > 0$ and $b > 0$. Without loss of generality, we assume that $r = 1$.

It can be verified that

- (i) Using dominance, the buying time of the consumer $x \in [0, b]$, and the stopping time chosen by nature $y \in (0, b]$.
- (ii) For all $y \in (0, b]$, the ratio is a constant if the consumer chooses the optimal mixed strategy.
- (iii) No probability mass appears in $(0, b]$.

By calculation, we obtain the following optimal mixed strategy:

- The probability that the consumer buys at time $x = 0$ is $p_0 = a / ((a + b)e - b)$.
- The probability density function that the consumer buys at time $x \in (0, b]$ is $p(x) = \frac{\exp(x/b)}{b(e - \frac{b}{a+b})}$.
- The competitive ratio is $\frac{e}{e - \frac{b}{a+b}}$.

Note that the biggest difference from MSR is that p_0 may be probability mass, which means that the consumer may have non-zero probability to buy at the initial time.

3.2 Analysis of MSR-E

In this problem, assume that there are n shops in total. For any shop $j \in [n]$, the entry fee, renting price and buying price of shop j are $a_j \geq 0$, $r_j > 0$ and $b_j > 0$, respectively. Similar to the procedures in MSR, we use a tuple (j, x) in which $j \in [n]$, $x \in [0, +\infty) \cup \{+\infty\}$ to denote an action for the consumer and a number $y \in (0, +\infty) \cup \{+\infty\}$ for nature.

Without loss of generality, in this problem, we assume that

- $r_1 \leq r_2 \leq \dots \leq r_n$;
- $\forall i, j, a_i < a_j + b_j$;
- $\forall i < j, a_i > a_j$ or $a_i + b_i > a_j + b_j$.

The second condition is because that shop i is dominated by shop j if $a_i \geq a_j + b_j$. For the third condition, we know $r_i \leq r_j$ since $i < j$. So shop j is dominated by shop i if we also have $a_i \leq a_j$ and $a_i + b_i \leq a_j + b_j$.

Denote B as follows:

$$\begin{aligned} & \text{minimize} && B \\ & \text{subject to} && \forall i, a_i + Br_i \geq \min_j(a_j + b_j) \end{aligned}$$

Similar to Lemma 1 in MSR, we reduce the action sets for both players by the following lemma:

LEMMA 11. *For nature, any action $y \in [B, +\infty)$ is dominated. For the consumer, any action (j, x) is dominated, in which $x \in (B, +\infty) \cup \{+\infty\}$, $j \in [n]$.*

Similar to MSR, the consumer's action set is reduced to buying time $x \in [0, B]$, and nature's action set is reduced to $\{y \in (0, B]\}$.

The strategy spaces for both the consumer and the nature are identical to those in MSR. Similarly, we use \mathbf{p} to denote a mixed strategy of the consumer and \mathbf{p}^* to denote the optimal mixed strategy. If the consumer chooses mixed strategy \mathbf{p} and nature chooses y , we denote the cost function as follows:

$$\begin{aligned} C(\mathbf{p}, y) &= \sum_{j \in [n]} \left(\int_0^y (a_j + r_j x + b_j) p_j(x) dx \right. \\ &\quad \left. + \int_y^B (a_j + r_j y) p_j(x) dx \right) \end{aligned} \quad (15)$$

We define $\text{OPT}(y)$ as the offline optimal strategy when nature chooses the action y , i.e.

$$\text{OPT}(y) = \min_j \{a_j + r_j y\}, \quad y \in (0, B]$$

By [8], we can compute the function $\text{OPT}(y)$ in linear time. The objective of the consumer is $\min_{\mathbf{p}} \max_y \frac{C(\mathbf{p}, y)}{\text{OPT}(y)}$.

Similar to MSR, we give the following lemmas:

LEMMA 12. *For the optimal strategy \mathbf{p}^* of the consumer, $\frac{C(\mathbf{p}^*, y)}{\text{OPT}(y)}$ is a constant for any $y \in (0, B]$.*

LEMMA 13. $\forall x \in (0, B], p_j^*(x) < +\infty$.

The problem is formalized as follows:

$$\text{minimize} \quad \lambda \quad (16)$$

$$\text{subject to} \quad \frac{C(\mathbf{p}, y)}{\text{OPT}(y)} = \lambda, \quad \forall y \in (0, B] \quad (16a)$$

$$\sum_{j=1}^n \int_0^B p_j(x) dx = 1 \quad (16b)$$

$$p_j(x) \geq 0, \quad \forall x \in [0, B] \quad (16c)$$

Note that there may be probability mass at $x = 0$. For Problem 16, we find that the optimal mixed strategy \mathbf{p}^* of the consumer is also segmented.

LEMMA 14. *In the optimal mixed strategy \mathbf{p}^* , there exists $n + 1$ breakpoints $B = d_1 \geq d_2 \geq \dots \geq d_{n+1} = 0$, which partition $[0, B]$ into n sub-intervals, such that $\forall j = 1, \dots, n$, $\forall x \in (d_{j+1}, d_j)$, $p_j^*(x) > 0$ and $p_i^*(x) = 0$ for any $i \neq j$.*

Remark: Though we prove the form of the optimal solution to MSR-E, computing the analytical solution is very challenging. The point mass at $x = 0$ makes this problem much more difficult than MSR, because one needs to guarantee the nonnegativity of this probability mass. Moreover, the non differentiable points of $\text{OPT}(y)$, which we call the offline breakpoints, complicate the probability density function form in each segment (d_{j+1}, d_j) . In fact, we can obtain the exact analytical optimal solution when $n = 2$.

4. SKI RENTAL WITH ENTRY FEE AND SWITCHING

Now we introduce the last extension of MSR, the *entry fee included, switching allowed problem* (MSR-ES). All the settings are identical to those of MSR-E, except that the consumer is allowed to switch at any time. When a consumer enters or switches to a shop, she pays the entry fee of the shop. For instance, if a consumer enters shop 1 at first, then switches from shop 1 to shop 2 and returns to shop 1 at last, she pays the entry fee of shop 1 twice and the entry fee of shop 2 once.

Similar to MSR-E, there exist n shops. The entry fee, renting price, buying price of shop j are denoted by $a_j \geq 0, r_j > 0, b_j > 0$, respectively. Without loss of generality, we assume that

- $r_1 \leq r_2 \leq \dots \leq r_n$;
- $\forall i, j, a_i < a_j + b_j$;
- $\forall i < j, a_i > a_j$ or $a_i + b_i > a_j + b_j$.

As in the MSR-S case, if there exists $i, j \in [n]$ such that $b_i > a_j + b_j$, then instead of buying in shop i , the consumer will switch from shop i to shop j to buy skis⁶. This is equivalent to setting b_i to be $\min_{j \neq i} \{a_j + b_j\}$. Therefore, without loss of generality, we assume that

- $\forall i, j, b_i \leq a_j + b_j$.

In the remainder of this section, we first define the action set and formulate our problem. Then, we show that the strategy space can be reduced by Lemma 15, 16 and 17. In Lemma 18, we show that for each switching operation, the operations of the consumer before or after the switching is not important. The only things we care about are when the switching happens, and which shops the switching operation relates to. Thus, we can construct a virtual shop for each switching operation, and (nearly) reduce this MSR-ES problem to MSR-E. Finally, we show that MSR-ES has the similar nice properties as MSR-E which we have known in Lemma 13 and 14.

⁶This phenomenon is called “switching for buying”.

4.1 Notations and Analysis of MSR-ES

4.1.1 Reduced Strategy Space

The action set for nature is $\{y > 0\}$, defined as before. To represent the action set formally, we firstly introduce the operation tuple $\sigma = (i, j, x)$ to denote the switching operation of switching from shop i to shop j at time x . For special cases, $(0, j, 0)$ denotes the entering operation that the consumer enters shop j at the very beginning; $(i, 0, x)$ denotes the buying operation that the consumer buys at shop i at time x . An operation tuple can also be represented as (j, x) for short, denoting the switching operation to shop j at time x if $j > 0$, the entering operation if $x = 0$, and the buying operation at time x if $j = 0$.

Then, an action ψ is expressed as a sequence (may be infinite) of the operation tuples:

$$\psi = \{(j_0, x_0), (j_1, x_1), (j_2, x_2), \dots\}$$

satisfying that

- $0 = x_0 \leq x_1 \leq x_2 \leq \dots$;
- if there exists $x \geq 0$ such that $(0, x) \in \psi$, it is the last element in ψ .

or the full form with the same constraints:

$$\psi = \{(0, j_0, x_0), (j_0, j_1, x_1), (j_1, j_2, x_2), \dots\}$$

Similar to other extensions, we reduce the action set. In this model, the definition of B is the same as those of MSR-E:

$$\begin{aligned} & \text{minimize} && B \\ & \text{subject to} && \forall i, a_i + Br_i \geq \min_j (a_j + b_j) \end{aligned}$$

and we give the following lemma:

LEMMA 15. *From the perspective of nature, any strategy $y \in [B, +\infty)$ is dominated. While for the consumer, any strategy in which the buying time $x \in (B, +\infty) \cup \{+\infty\}$ is dominated.*

Similar to MSR, we reduce the consumer’s buying time to the interval $[0, B]$, and nature’s action set to $\{y \in (0, B]\}$.

The following lemma shows that a consumer may switch from shop i to shop j for renting, only when $r_i > r_j$ and $a_i < a_j$.

LEMMA 16. *If a strategy of the consumer:*

$$\psi = \{(0, j_0, x_0), (j_0, j_1, x_1), \dots, (j_{|\psi|-2}, 0, x_{|\psi|-1})\}$$

satisfies any of the following conditions, then it is dominated.

- $\exists 0 < \tau < |\psi| - 1$ such that $x_{\tau-1} = x_\tau$;
- $\exists (i, j, x) \in \psi$ such that $r_i \leq r_j$ and $(j, 0, x) \notin \psi$;
- $\exists (i, j, x) \in \psi$ such that $a_i \geq a_j$ and $(j, 0, x) \notin \psi$.

Here we give some intuitions. In these three cases, we can construct a new action ψ' by deleting one specified operation from ψ , and show that ψ is dominated by ψ' .

This lemma rules out a huge amount of dominated strategies from our action set and allows us to define the operation set:

$$\Sigma \triangleq \left\{ \sigma = (i, j, x) : i, j \in [n], r_i > r_j, a_i < a_j, \right.$$

$$x \in (0, B] \bigcup \left\{ (0, j, 0) : j \in [n] \right\} \\ \bigcup \left\{ (j, 0, x) : j \in [n], x \in [0, B] \right\}$$

Thus, we only need to consider such an action set:

$$\Psi_c \triangleq \left\{ \psi = \{(0, j_0, x_0), (j_0, j_1, x_1), \dots, (j_{|\psi|-2}, 0, x_{|\psi|-1})\} \right. \\ : \quad 0 = x_0 < x_1 < \dots < x_{|\psi|-2} \leq x_{|\psi|-1} \leq B, \\ \quad r_{j_0} > r_{j_1} > \dots > r_{j_{|\psi|-2}}, \\ \quad \left. a_{j_0} < a_{j_1} < \dots < a_{j_{|\psi|-2}} \right\}$$

Since $r_{j_0} > r_{j_1} > \dots > r_{j_k}$, we get $j_0 > j_1 > \dots > j_k$ and $2 \leq |\psi| \leq n + 1$.

4.1.2 Mathematical Expression of the Cost, Ratio and the optimization problem

For nature's action $y \in (0, B]$ and the consumer's action $\psi = \{(j_0, x_0), \dots, (0, x_{|\psi|-1})\} \in \Psi_c$, we define the cost $c(\psi, y)$ as follows:

$$c(\psi, y) \triangleq \begin{cases} \sum_{\tau=0}^{k-1} [a_{j_\tau} + r_{j_\tau} (x_{\tau+1} - x_\tau)] + r_{j_k} (y - x_k), & \text{if } \exists 0 < k < |\psi|, x_{k-1} \leq y < x_k; \\ \sum_{\tau=0}^{|\psi|-2} (a_{j_\tau} + r_{j_\tau} (x_{\tau+1} - x_\tau)) + b_{j_{|\psi|-2}}, & \text{if } y \geq x_{|\psi|-1}. \end{cases}$$

For any action $\psi = \{(j_0, x_0), \dots, (0, x_{|\psi|-1})\}$, we use $\mathbf{s}(\psi)$ to denote the order of the operations:

$$\mathbf{s}(\psi) \triangleq \{(0, j_0), (j_0, j_1), \dots, (j_{|\psi|-2}, 0)\}$$

or the short form:

$$\mathbf{s}(\psi) \triangleq \{j_0, j_1, \dots, j_{|\psi|-2}, 0\}$$

Further, we define \mathcal{S} as the collection $\mathbf{s}(\Psi_c)$ as follows:

$$\mathcal{S} \triangleq \{\mathbf{s} = \{j_0, j_1, \dots, j_k, 0\} : k \geq 0, \\ r_{j_0} > r_{j_1} > \dots > r_{j_k}, a_{j_0} < a_{j_1} < \dots < a_{j_k}\}$$

Note that $j_0, j_1, \dots, j_k \in [n]$ and $\{0\} \notin \mathcal{S}$, so the amount of elements in \mathcal{S} is upper bounded by $|\mathcal{S}| \leq 2^n - 1$.

We group all the actions in Ψ_c whose $\mathbf{s}(\psi)$ are identical. Thus, we partition Ψ_c into $|\mathcal{S}|$ subsets.

For any action ψ , let $\mathbf{x}(\psi)$ denote the sequence of the operation time, defined as follows:

$$\mathbf{x}(\psi) \triangleq (x_1, \dots, x_{|\psi|-1})$$

For each operation order $\mathbf{s} \in \mathcal{S}$, we define $\mathcal{X}_\mathbf{s}$ as the collection $\{\mathbf{x}(\psi) : \mathbf{s}(\psi) = \mathbf{s}\}$, i.e.,

$$\mathcal{X}_\mathbf{s} \triangleq \{\mathbf{x} = (x_1, x_2, \dots, x_{|\mathbf{s}|-1}) : 0 < x_1 < \dots < x_{|\mathbf{s}|-1} \leq B\}$$

We observe that any $\mathbf{s} \in \mathcal{S}$ and $\mathbf{x} \in \mathcal{X}_\mathbf{s}$ can be combined to a unique action $\psi(\mathbf{s}, \mathbf{x})$. Further, we can use $c_\mathbf{s}(\mathbf{x}, y)$ and $c(\psi(\mathbf{s}, \mathbf{x}), y)$ interchangeably.

For each $\mathbf{s} \in \mathcal{S}$, we define the probability density function $f_\mathbf{s} : \mathcal{X}_\mathbf{s} \rightarrow [0, +\infty) \cup \{+\infty\}$ ⁷, which satisfies

$$\sum_{\mathbf{s} \in \mathcal{S}} \int_{\mathbf{x} \in \mathcal{X}_\mathbf{s}} \dots \int f_\mathbf{s}(\mathbf{x}) d\mathbf{x} = 1$$

⁷ $f_\mathbf{s}(\mathbf{x}) = +\infty$ represents probability mass on \mathbf{x} .

Let $\mathbf{f} = \{f_\mathbf{s} : \mathbf{s} \in \mathcal{S}\}$ denote a mixed strategy for the consumer. Given a mixed strategy \mathbf{f} of the consumer and nature's choice y , the expected competitive ratio is defined as follows:

$$R(\mathbf{f}, y) \triangleq \frac{C(\mathbf{f}, y)}{\text{OPT}(y)} \quad (17)$$

where $\text{OPT}(y) = \min_{j \in [n]} \{a_j + r_j y\}$, and

$$C(\mathbf{f}, y) \triangleq \sum_{\mathbf{s} \in \mathcal{S}} \int_{\mathbf{x} \in \mathcal{X}_\mathbf{s}} \dots \int c_\mathbf{s}(\mathbf{x}, y) f_\mathbf{s}(\mathbf{x}) d\mathbf{x} \quad (18)$$

The objective of the consumer is $\min_{\mathbf{f}} \max_y R(\mathbf{f}, y)$. The following lemma proves that $\forall y \in (0, B]$, $R(\mathbf{f}^*, y)$ is a constant in which \mathbf{f}^* is an optimal mixed strategy.

LEMMA 17. *If \mathbf{f}^* is an optimal solution of the problem $\arg \min_{\mathbf{f}} \max_y R(\mathbf{f}, y)$, then there exists a constant λ such that $\forall y \in (0, B]$, $R(\mathbf{f}^*, y) = \lambda$.*

The formalized optimization problem is as follows:

$$\text{minimize } \lambda \quad (19)$$

$$\text{subject to } \frac{C(\mathbf{f}, y)}{\text{OPT}(y)} = \lambda, \forall y \in (0, B] \quad (19a)$$

$$\sum_{\mathbf{s} \in \mathcal{S}} \int_{\mathbf{x} \in \mathcal{X}_\mathbf{s}} \dots \int f_\mathbf{s}(\mathbf{x}) d\mathbf{x} = 1 \quad (19b)$$

$$f_\mathbf{s}(\mathbf{x}) \geq 0, \forall \mathbf{s} \in \mathcal{S} \quad (19c)$$

4.2 Reduction to MSR-E

For a mixed strategy \mathbf{f} , we define the probability density function of an operation $\sigma = (i, j, x)$ as follows:

$$\mathbf{p}^{(\mathbf{f})}(\sigma) \triangleq \sum_{\mathbf{s} \in \mathcal{S}: (i, j) \in \mathbf{s}} \int_{\mathbf{x}_{-\{x\}}: \sigma \in \psi(\mathbf{s}, \mathbf{x})} \dots \int f_\mathbf{s}(\mathbf{x}) d(\mathbf{x}_{-\{x\}}) \quad (20)$$

where $\mathbf{x}_{-\{x\}}$ is the vector \mathbf{x} in which the element corresponding to x is eliminated. Here $\mathbf{p}_{(i, j)}^{(\mathbf{f})}(x)$ can also be viewed as a marginal probability density function. Also the p.d.f of an operation can be expressed in another form:

$$\mathbf{p}_{(i, j)}^{(\mathbf{f})}(x) \triangleq \mathbf{p}^{(\mathbf{f})}((i, j, x))$$

Then we give the following lemma:

LEMMA 18. *For any 2 mixed strategies $\mathbf{f}_1, \mathbf{f}_2$ for the consumer, we have $C(\mathbf{f}_1, y) = C(\mathbf{f}_2, y)$ for all $y \in (0, B]$ if $\mathbf{p}^{(\mathbf{f}_1)}(\sigma) = \mathbf{p}^{(\mathbf{f}_2)}(\sigma)$ for all $\sigma \in \Sigma$, i.e., for a mixed strategy \mathbf{f} , we only care about its marginal $\mathbf{p}^{(\mathbf{f})}(\sigma)$.*

Therefore, the target of the problem converts from the optimal \mathbf{f} to the optimal \mathbf{p} . Note that for the switching operation from shop i to shop j , we do not care about which action ψ it belongs to. Instead, the only thing that matters is when this switching operation happens. This is similar to the one shop case, in which we only care about when the consumer decides to buy. Thus, for each switching pair (i, j) , we consider it as a virtual shop. Among these $O(n^2)$ virtual shops, no switching will appear. Thus, we show that the MSR-ES problem is almost the same as MSR-E.

Now we show our settings for virtual shops. For all $i, j \in [n]$ such that $a_i < a_j, r_i > r_j$, we define the virtual shop (i, j) with entry fee $a_{(i, j)} = a_i - a_j$, renting price $r_{(i, j)} = r_i - r_j$ and buying price $b_{(i, j)} = a_j$. We regard the switching time from

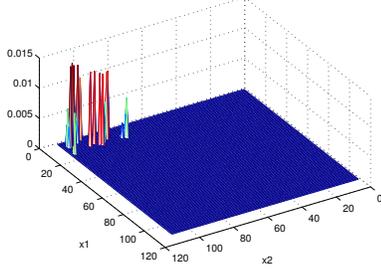


Figure 2: PDF of the strategies with switching actions, i.e., function $f_s^*(x)$ when $s = \{2, 1, 0\}$. x_1 is the switching time and x_2 is the buying time.

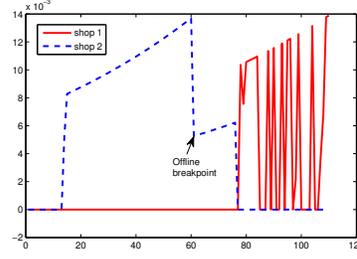


Figure 3: PDF of the strategies without switching actions, i.e., function $f_s^*(x)$ when $|s| = 2$. Blue: $f_s^*(x)$ when $s = \{2, 0\}$; Red: $f_s^*(x)$ when $s = \{1, 0\}$.

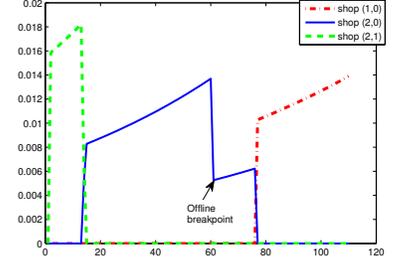


Figure 4: PDF of the virtual shops, i.e., function $p_{(i,j)}^{(f)}(x)$ when (i, j) is a virtual shop. Green: $p_{(2,1)}^*(x)$; Blue: $p_{(2,0)}^*(x)$; Red: $p_{(1,0)}^*(x)$.

i to j as the buying time in virtual shop (i, j) . For special case, the prices of virtual shop $(j, 0)$ is the same as the real shop j . Through this setting, it is not hard to verify that for any action ψ and any $0 \leq y \leq B$, the cost function $c(\psi, y)$ is exactly the summation of the cost in the corresponding virtual shops. Similar to a real shop, we define the cost for each virtual shop (i, j) :

$$C_{(i,j)}(\mathbf{p}^{(f)}, y) \triangleq \int_0^y (a_{(i,j)} + r_{(i,j)}x + b_{(i,j)})p_{(i,j)}^{(f)}(x)dx + \int_y^B (a_{(i,j)} + r_{(i,j)}y)p_{(i,j)}^{(f)}(x)dx$$

Now we are ready to formalize MSR-ES by the following theorem:

THEOREM 4. *The optimization problem for the consumer can be formalized as follows:*

$$\text{minimize } \lambda \quad (21)$$

$$\text{subject to } \frac{C(\mathbf{f}, y)}{\text{OPT}(y)} = \lambda \quad (21a)$$

$$C(\mathbf{f}, y) = \sum_{(i,j) \in [n]^2: a_i < a_j, r_i > r_j} C_{(i,j)}(\mathbf{p}^{(f)}, y) + \sum_{j \in [n]} C_{(j,0)}(\mathbf{p}^{(f)}, y), \quad \forall y \in [0, B] \quad (21b)$$

$$\sum_{j \in [n]} \int_0^B p_{(j,0)}^{(f)}(x)dx = 1 \quad (21c)$$

$$\sum_{j \in [n]: a_j < a_i, r_j > r_i} \int_y^B p_{(j,i)}^{(f)}(x)dx \leq \int_y^B p_{(i,0)}^{(f)}(x)dx + \sum_{j \in [n]: a_i > a_j, r_i < r_j} \int_y^B p_{(i,j)}^{(f)}(x)dx, \quad \forall i \in [n], y \in (0, B] \quad (21d)$$

Thus, MSR-ES can be regarded as MSR-E with $O(n^2)$ shops. The difference is that the summation of the buying probabilities in each virtual shop may be larger than 1. Fortunately, those nice properties of MSR-E still hold for MSR-ES.

LEMMA 19. *Lemma 13 and 14 still hold for the virtual shops in the MSR-ES problem.*

As in other extensions before, the probability density function of the virtual shops is segmented and each segment is an exponential function. The consumer only assigns positive buying probability in exactly one virtual shop at any time. As the buying time increases, she follows the virtual shop order in which the ratio between buying price and renting price is increasing.

In the 3 figures above, we give a simple example when $n = 2$ in order to make Lemma 19 easier to understand. We approach the optimal strategy through the discrete model and the figures show the p.d.f. functions in the optimal strategy. It can be seen that Lemma 19 is verified. Since there is a proof for Lemma 19, we do not give more complicated examples. The parameters for the 2 shops are as follows: $a_1 = 80, r_1 = 1, b_1 = 110, a_2 = 20, r_2 = 2, b_2 = 180$.

5. CONCLUSIONS

In this paper, we consider the multi-shop ski rental problem (MSR) and its extensions (MSR-S, MSR-E, and MSR-ES), in which there are multiple shops and the consumer wants to minimize the competitive ratio.

For each problem, we prove that in the optimal mixed strategy of the consumer, she only assigns positive buying probability to *exactly one* shop at any time. The shop order strongly relates to the ratio between buying price and renting price, even in which entry fee is involved. Further, in the basic problem (MSR), we derive a linear time algorithm for computing the optimal strategy of the consumer. For MSR-S, we prove that under the optimal mixed strategy, the consumer only switches to another shop at the buying time.

In problems MSR-E and MSR-ES, we show that the optimal strategy can be solved if the breakpoints are known. Similar to the basic problem (MSR), we conjecture that the quasi-concave property also holds for these two variants. Further, we conjecture that there exists an iteration algorithm using gradient decent technique, which might converge to the optimal solution.

6. ACKNOWLEDGEMENT

This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61361136003, 041302007, the China youth 1000-talent grant 610502003, and the MSRA-Tsinghua joint lab grant 041902004.

7. REFERENCES

- [1] Amazon EC2, <http://aws.amazon.com/ec2/>.
- [2] Astrium, <http://www.astrium-geo.com/>.
- [3] Digitalglobe, <http://www.digitalglobe.com/>.
- [4] Elastichosts, <http://www.elastichosts.com/>.
- [5] The multi-shop ski rental problem, <http://sites.google.com/site/multishopskirental/MSRtech.pdf>.
- [6] Microsoft Windows Azure, <http://www.gogrid.com/>.
- [7] C. Bodenstein, M. Hedwig, and D. Neumann. Strategic decision support for smart-leasing infrastructure-as-a-service. 2011.
- [8] M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, and M. Overmars. Computational geometry: algorithms and applications. 2000. *New York, New York*.
- [9] R. Fleischer. On the bahncard problem. *Theoretical Computer Science*, 268(1):161–174, 2001.
- [10] B. Guenter, N. Jain, and C. Williams. Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning. In *INFOCOM, 2011 Proceedings IEEE*, pages 1332–1340. IEEE, 2011.
- [11] Y.-J. Hong, J. Xue, and M. Thottethodi. Dynamic server provisioning to minimize cost in an iaas cloud. In *Proceedings of the ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems*, pages 147–148. ACM, 2011.
- [12] A. R. Karlin, C. Kenyon, and D. Randall. Dynamic tcp acknowledgement and other stories about $e/(e-1)$. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 502–509. ACM, 2001.
- [13] A. R. Karlin, M. S. Manasse, L. A. McGeoch, and S. Owicki. Competitive randomized algorithms for nonuniform problems. *Algorithmica*, 11(6):542–571, 1994.
- [14] A. R. Karlin, M. S. Manasse, L. Rudolph, and D. D. Sleator. Competitive snoopy caching. *Algorithmica*, 3(1-4):79–119, 1988.
- [15] A. Khanafer, M. Kodialam, and K. P. Puttaswamy. The constrained ski-rental problem and its application to online cloud cost optimization. In *Proc. IEEE INFOCOM*, 2013.
- [16] M. Lin, A. Wierman, L. L. Andrew, and E. Thereska. Dynamic right-sizing for power-proportional data centers. In *INFOCOM, 2011 Proceedings IEEE*, pages 1098–1106. IEEE, 2011.
- [17] Z. Lotker, B. Patt-Shamir, D. Rawitz, S. Albers, and P. Weil. Rent, lease or buy: Randomized algorithms for multislope ski rental. In *25th International Symposium on Theoretical Aspects of Computer Science (STACS 2008)*, volume 1, pages 503–514. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2008.
- [18] T. Lu and M. Chen. Simple and effective dynamic provisioning for power-proportional data centers. In *Information Sciences and Systems (CISS), 2012 46th Annual Conference on*, pages 1–6. IEEE, 2012.
- [19] S. Shankland. Google to buy GeoEye satellite imagery, http://news.cnet.com/8301-1023_3-10028842-93.html.
- [20] R. Singh, U. Sharma, E. Cecchet, and P. Shenoy. Autonomic mix-aware provisioning for non-stationary data center workloads. In *Proceedings of the 7th international conference on Autonomic computing*, pages 21–30. ACM, 2010.
- [21] C. Wang, B. Urgaonkar, Q. Wang, G. Kesidis, and A. Sivasubramaniam. Data center cost optimization via workload modulation under real-world electricity pricing. *arXiv preprint arXiv:1308.0585*, 2013.
- [22] W. Wang, B. Li, and B. Liang. To reserve or not to reserve: Optimal online multi-instance acquisition in iaas clouds. *arXiv preprint arXiv:1305.5608*, 2013.