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 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 shop[1].

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.

[1] For 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.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGMETRICS'14, June 16–20, 2014, Austin, Texas, USA.
Copyright 2014 ACM 978-1-4503-2789-3/14/06 ...$15.00.
http://dx.doi.org/10.1145/2591971.2591984

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 le is replicated and stored in di erent machines in the cluster. Some node undertaking some computing job needs data in the le during the execution. The node can either request the corresponding data block of the le from some selected machine whenever it needs to, which incurs some delay, or it can simply ask that machine to transmit the whole le beforehand, at the sacri ce 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 le beforehand.

2. Cost management in IaaS cloud: Multiple IaaS cloud vendors, such as Amazon EC2 [1], ElasticHosts [4] and Microsoft Windows Azure [6], o er di erent price options, which can be classi ed 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 con gurations o ered 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 o ering 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 di erent telecommunication companies.

1.1 Related Work

The ski rental problem is rst 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 rst 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 su ciently 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 o 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 o 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