Improved Learning-Augmented Algorithms for the Multi-Option Ski Rental Problem via Best-Possible Competitive Analysis

Yongho Shin Changyeol Lee Gukryeol Lee Hyung-Chan An

Yonsei University

Paper URL

Problem Definition

Multi-Option Ski Rental Problem

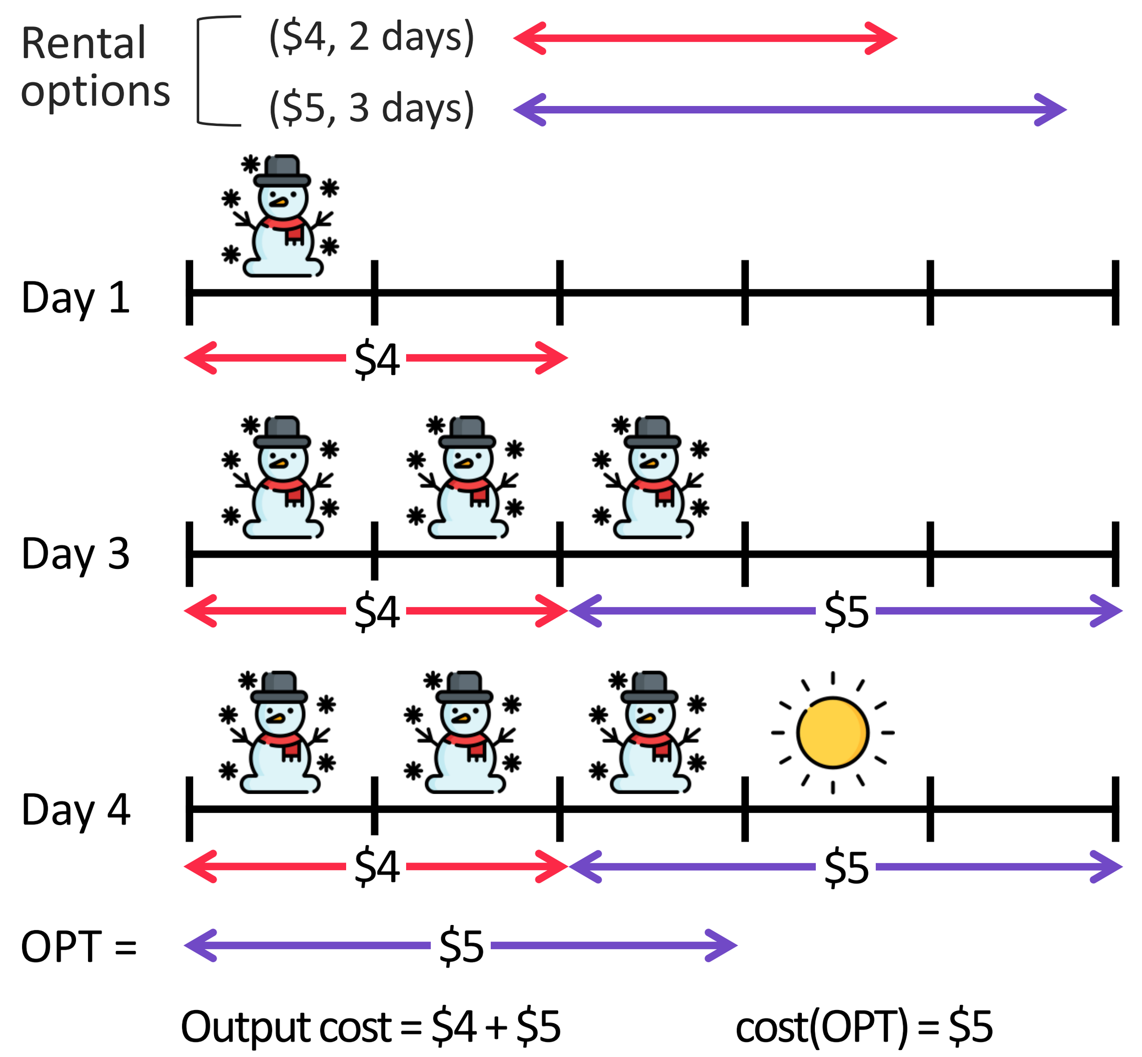
- **Input**: ski rental options (cost, #days)
- Must have skis every day
- Last day T not known in advance
- **Goal**: minimize expected total cost
- Rent-or-buy type applications

Without Learning Augmentation (LA)

- **γ-competitive** if (output cost) ≤ γ·cost(OPT)

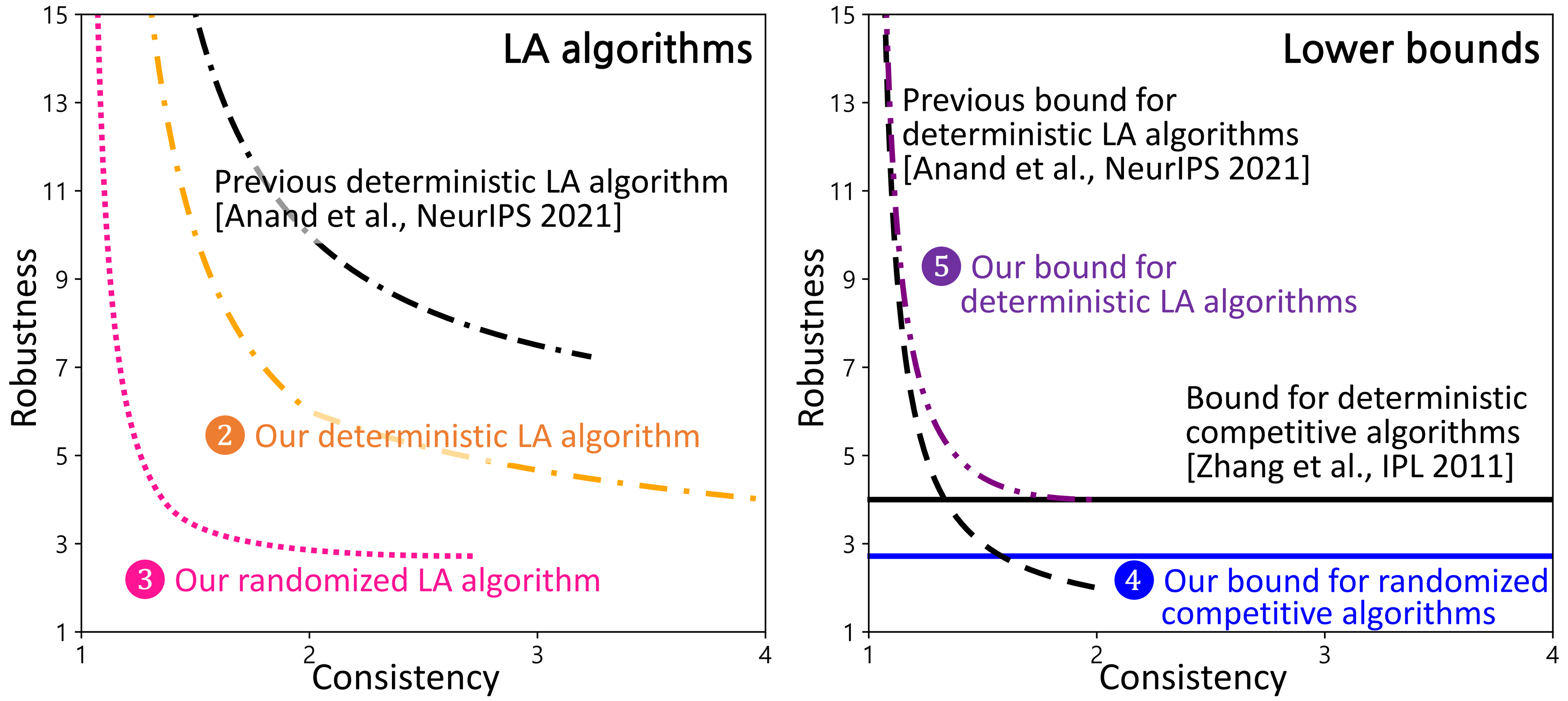
With Learning Augmentation (LA)

- ML prediction $\hat{T}$ on T additionally given
- **χ-consistent** if (output cost) ≤ χ·cost(OPT) when $\hat{T}$ is accurate
- **ρ-robust** if (output cost) ≤ ρ·cost(OPT) regardless of $\hat{T}$



Our Contributions

1 Randomized e-competitive algorithm
(Previous result: Deterministic 4-competitive algorithm [Anand et al., NeurIPS 2021])



Algorithms

Notation

- OPT(t): optimal options covering t days
- Best(B): "best" (i.e., covering most days) options within budget B

DET: deterministic 4-competitive algorithm

- Key idea: doubling the budget

```
B ← cost(OPT(1))      // initial budget
for each day (until the last day)
  if no skis available then
    choose Best(B)
    B ← B + cost(Best(B))
```

Theorem. DET is 4-competitive.

RAND: randomized e-competitive algorithm 1

- Key idea: randomizing the budget

```
sample α ∈ [0,e) from pdf f(α)=1/α
B ← α                  // initial budget
for each day (until the last day)
  if no skis available then
    choose Best(B)
    B ← e·B
```

Theorem. RAND is e-competitive.

Notation

- $\lambda \in [0,1]$: "level of confidence" on the prediction $\hat{T}$

DET-LA: improved deterministic LA algorithm 2

- Key idea: ignoring the prediction

```
run DET until B > λ·cost(OPT(T̂))
if B ≤ cost(OPT(T̂)) then
  choose OPT(T̂)
  B ← B + cost(OPT(T̂))
run DET with initial budget B
```

Theorem. DET-LA is max(1+2λ,4λ)-consistent and (2+2/λ)-robust.

RAND-LA: first randomized LA algorithm 3

- Key idea: randomizing budget & ignoring the prediction

```
sample α∈[0,e) from pdf f(α)=1/α
B ← α                  // initial budget
for each day (until the last day)
  if no skis available then
    if λcost(OPT(T̂)) ≤ B < cost(OPT(T̂)) then
      choose OPT(T̂) (at most once)
    else
      choose Best(B)
    B ← e·B
```

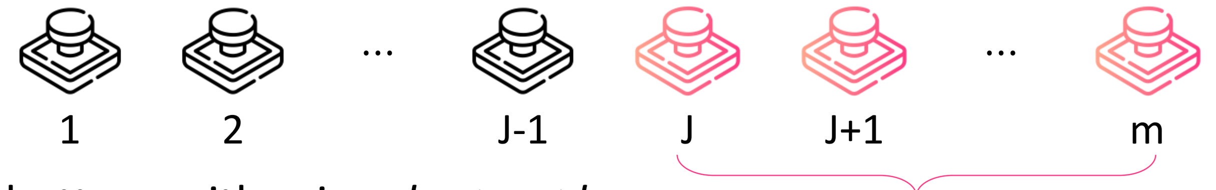
Theorem. RAND-LA is $\chi(\lambda)$-consistent and (e^λ/λ)-robust, where

$$\chi(\lambda) := \begin{cases} 1+\lambda, & \lambda < 1/e, \\ (e+1)\lambda - \ln\lambda - 1, & \lambda \ge 1/e. \end{cases}$$

Proof sketches and computational evaluation →

Lower Bounds

Button problem



- **Input**: m buttons with prices $b_1 \le \dots \le b_m$
- Button J and later are targets, where J is not known in advance
- Click buttons until a target is clicked
- **Goal:** minimize (expected) total price

Theorem. (informal)
Lower bounds on this problem (almost) immediately extend to multi-option ski rental problem.

Lower bound on competitive ratio of randomized algorithms 4

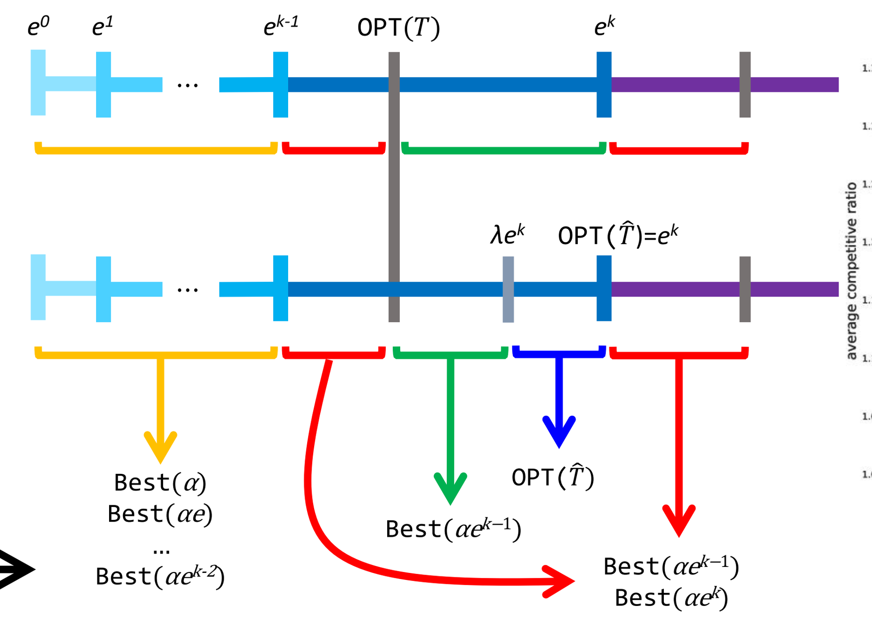
Theorem. No algorithm can achieve a constant competitive ratio better than e.

- x_j: marginal probability that the algorithm clicks button j as the first button
- $y_{t,j}$: marginal probability that the algorithm clicks buttons t and j in a row

$$\begin{aligned} \text{minimize} \quad & \gamma \\ \text{subject to} \quad & \textstyle\sum_{j=1}^m x_j = 1, \\ & \textstyle\sum_{j=t+1}^m y_{t,j} = x_t + \sum_{j=1}^{t-1} y_{j,t}, && \forall t = 1,\cdots,m-1, \\ & \textstyle\sum_{j=1}^m b_j \cdot \left(x_j + \sum_{t=1}^{\min(J,j)-1} y_{t,j}\right) \le \gamma \cdot b_J, && \forall J = 1,\cdots,m, \\ & x, y \ge 0. \end{aligned}$$

The rest of the proof involves constructing a family of instances and considering the dual LP.

Proof Sketches



Computational Evaluation