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

Yongho Shin, **Changyeol Lee**, Gukryeol Lee, and Hyung-Chan An

Yonsei University

연세대학교 컴퓨터과학과
YONSEI UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Learning-Augmented Algorithms

- Take ML prediction as added input
- No assumption on prediction

- Recent success in online optimization

Performance Measure

- An online algorithm is *c-competitive* if SOL $\leq c \cdot$OPT.

- A learning-augmented algorithm is
  - χ *-consistent* if SOL $\leq \chi \cdot$OPT when prediction is accurate.
  - ρ *-robust* if SOL $\leq \rho \cdot$OPT unconditionally.

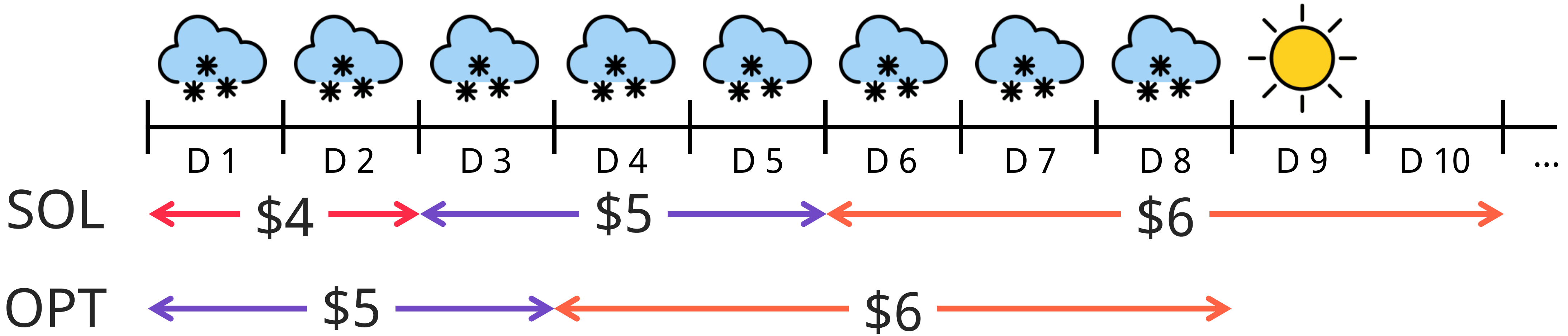
Multi-Option Ski Rental

[Attribute] Photo of a skier: Maarten Duineveld from Unsplash

Multi-Option Ski Rental

Renting options

$4	2 days	
$5	3 days	
$6	5 days	

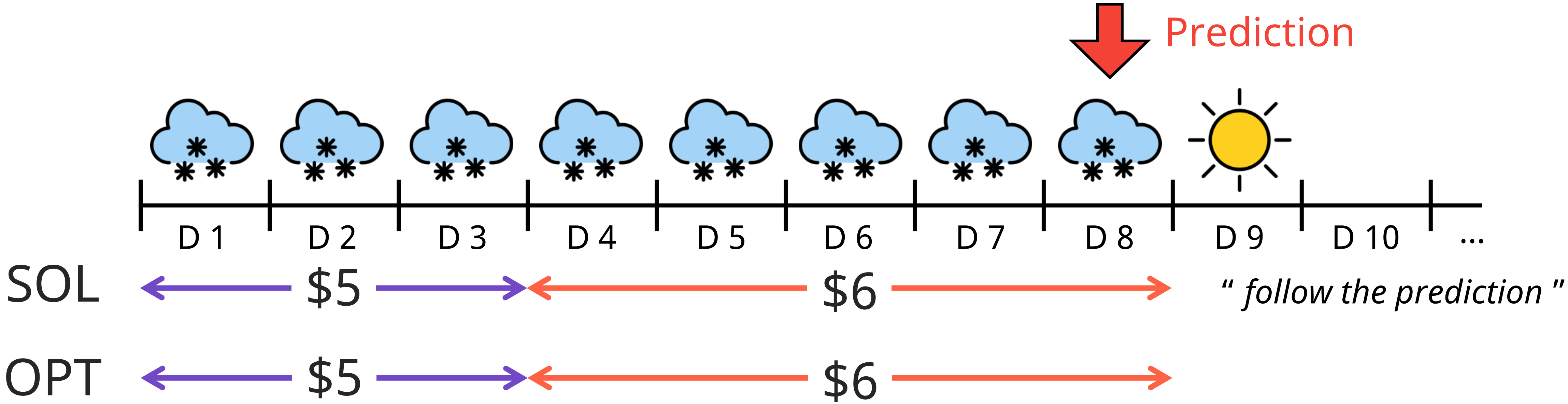


[Attribute] Weather icons: iconixar from Flaticon

Learning-Augmented Multi-Option Ski Rental

Renting options

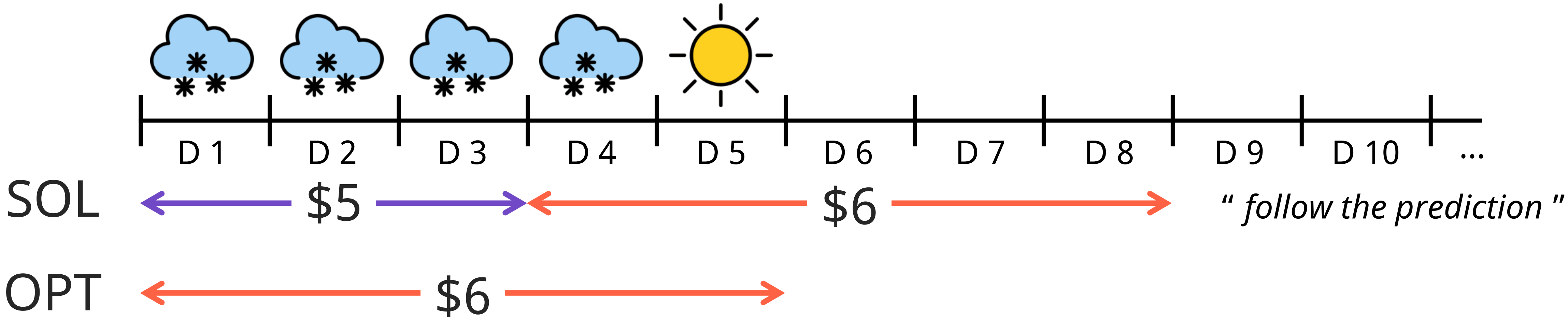
- $4 2 days
- $5 3 days
- $6 5 days



[Attribute] Weather icons: iconixar from Flaticon

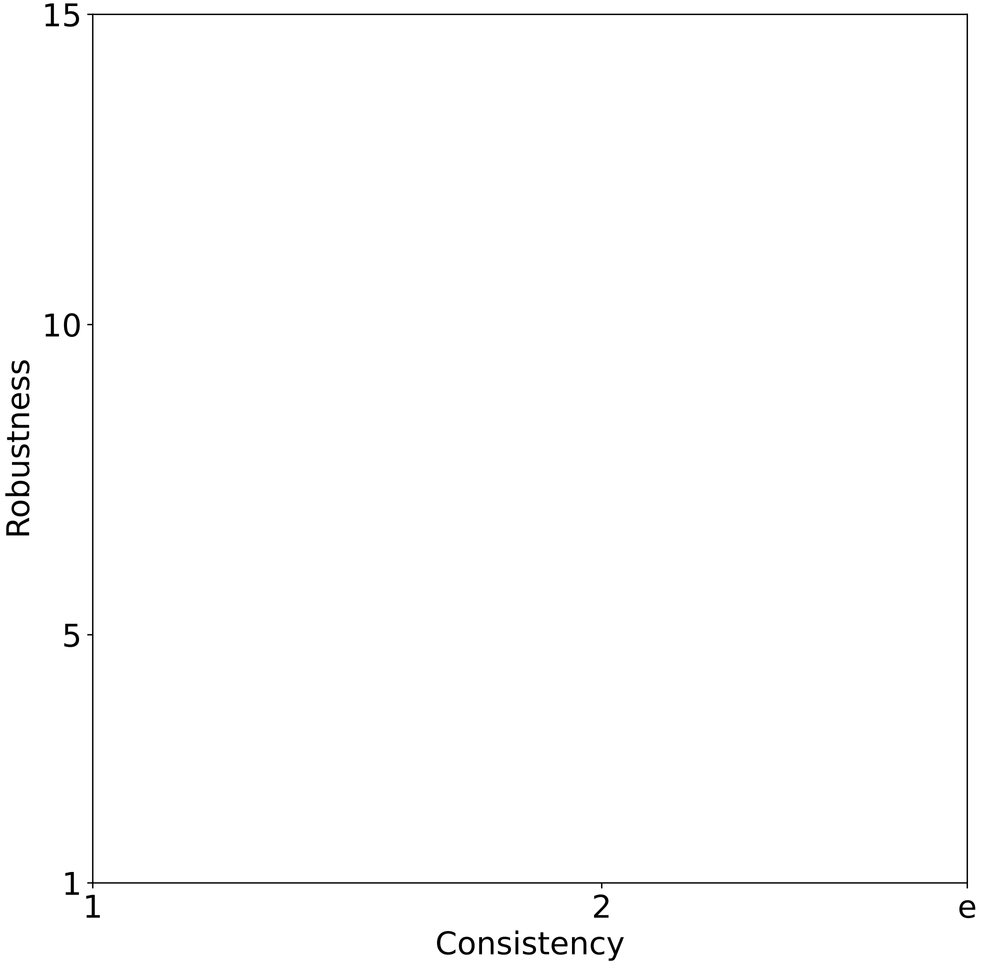
Learning-Augmented Multi-Option Ski Rental

Renting options
- $4 2 days
- $5 3 days
- $6 5 days

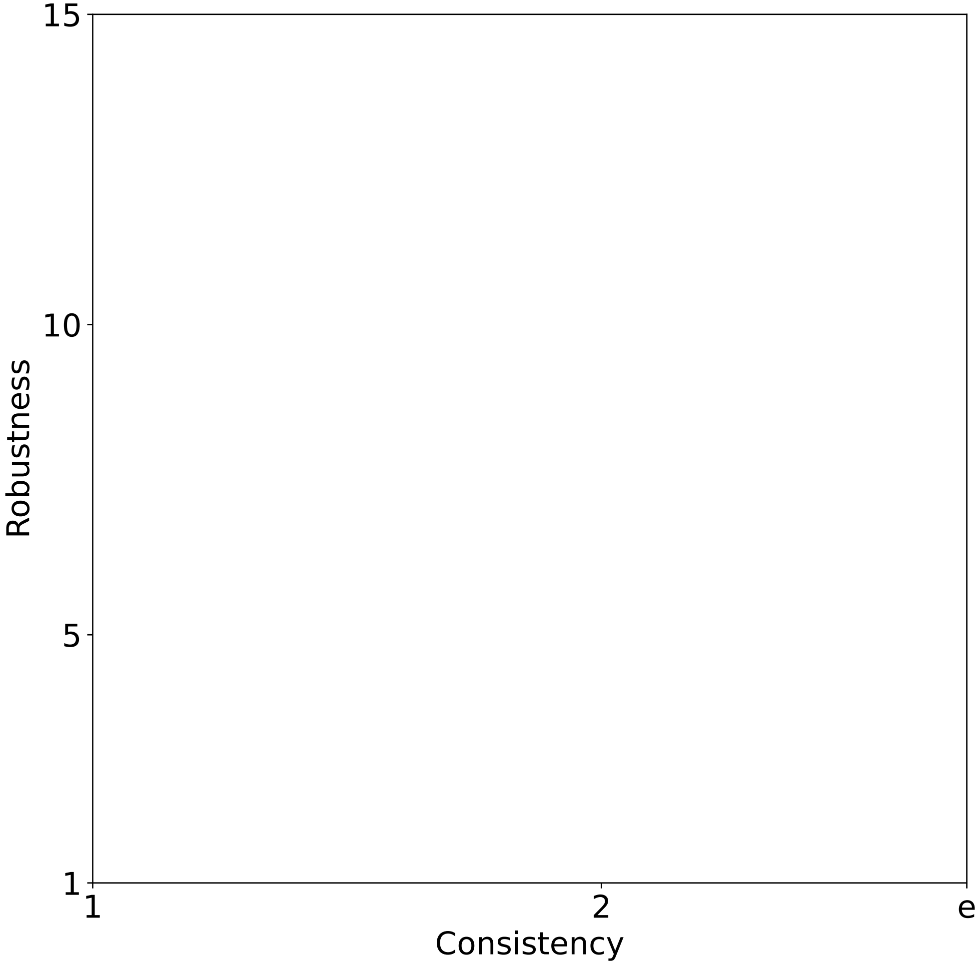


[Attribute] Weather icons: iconixar from Flaticon

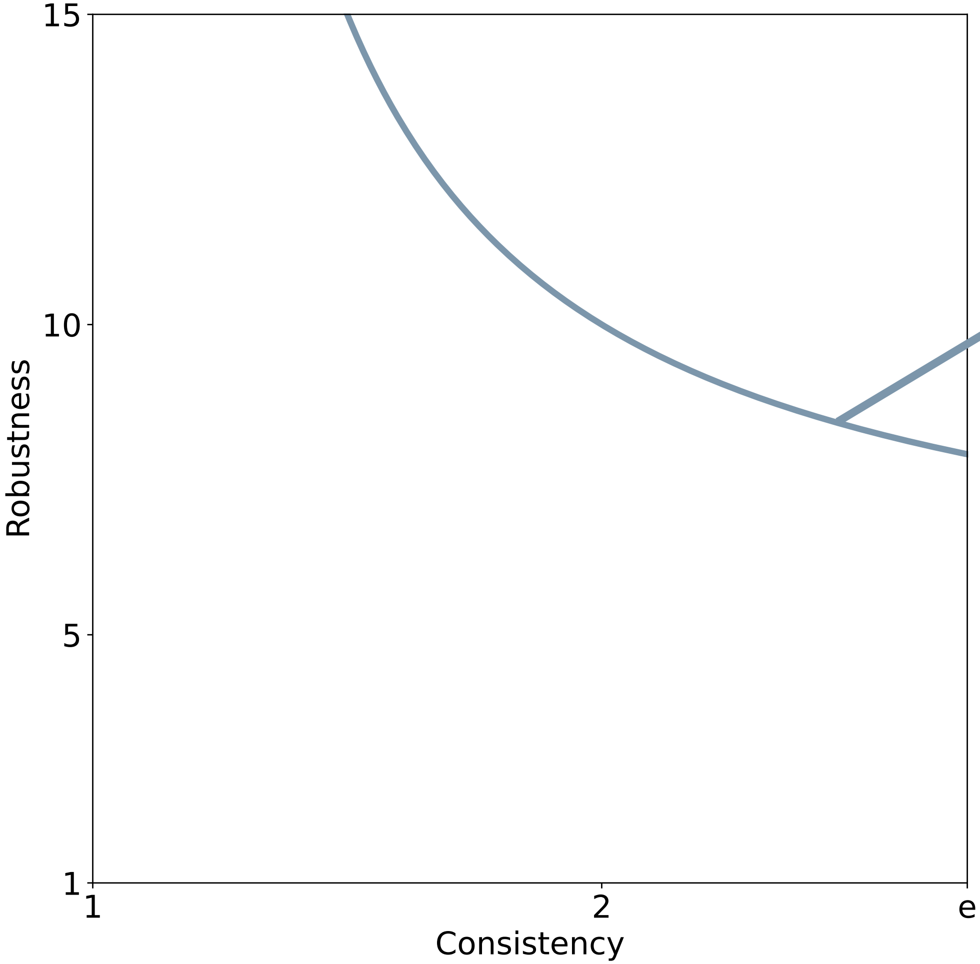
Previous Work



Previous Work (Deterministic)



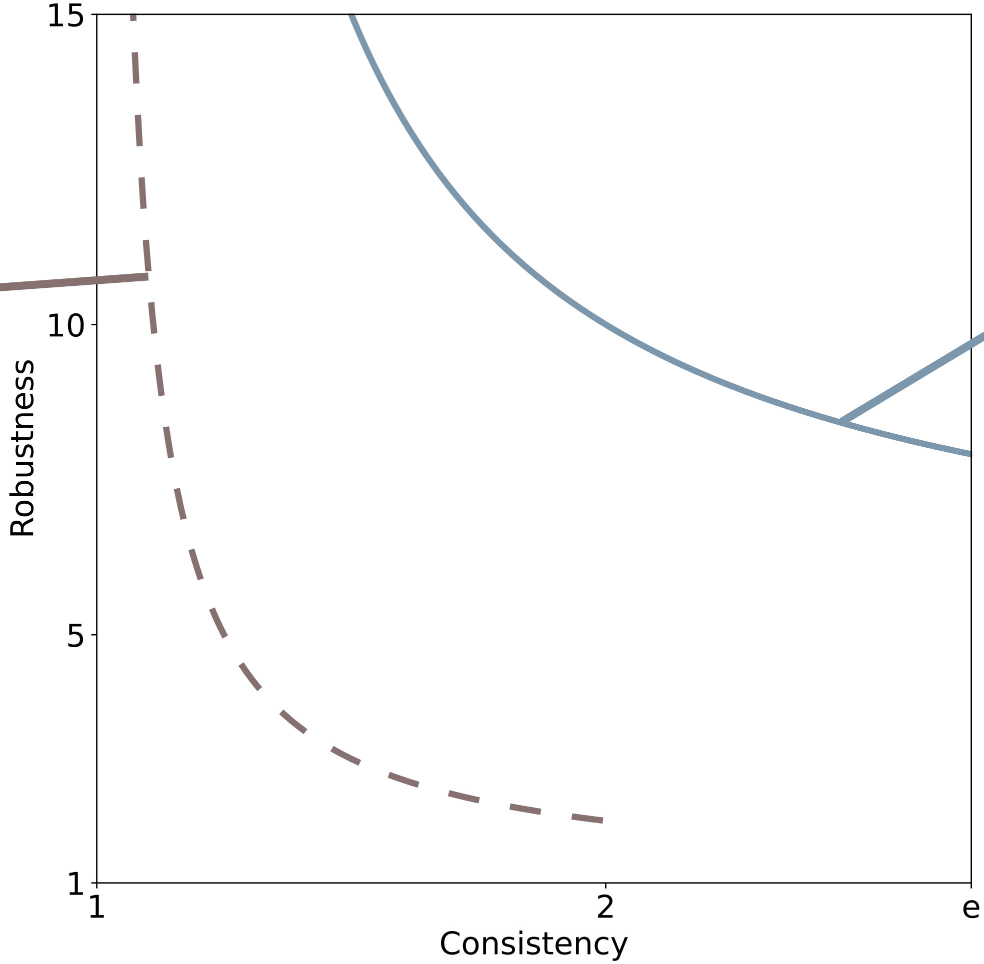
Previous Work (Deterministic)



Deterministic algorithm

[Anand et al., NeurIPS, 2021]

Previous Work (Deterministic)



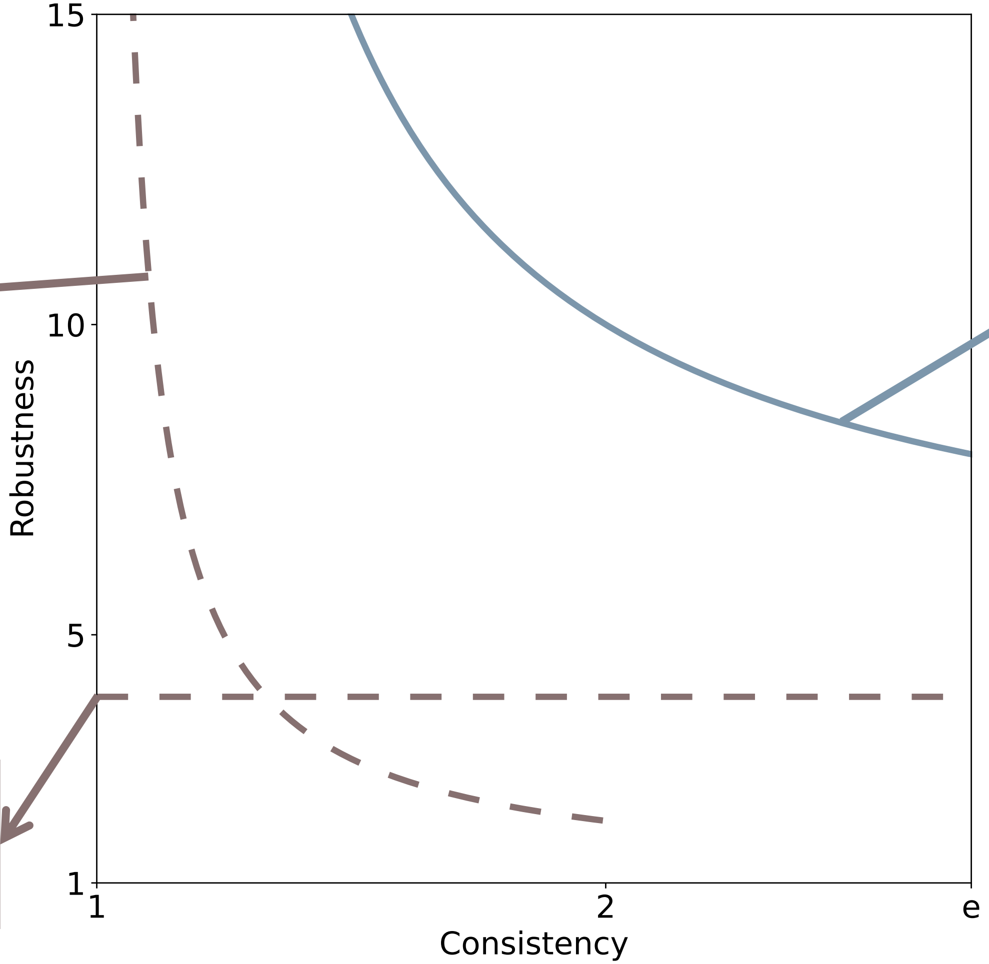
LB on the trade-off

[Anand et al., NeurIPS, 2021]

Deterministic algorithm

Previous Work (Deterministic)

LB on the trade-off

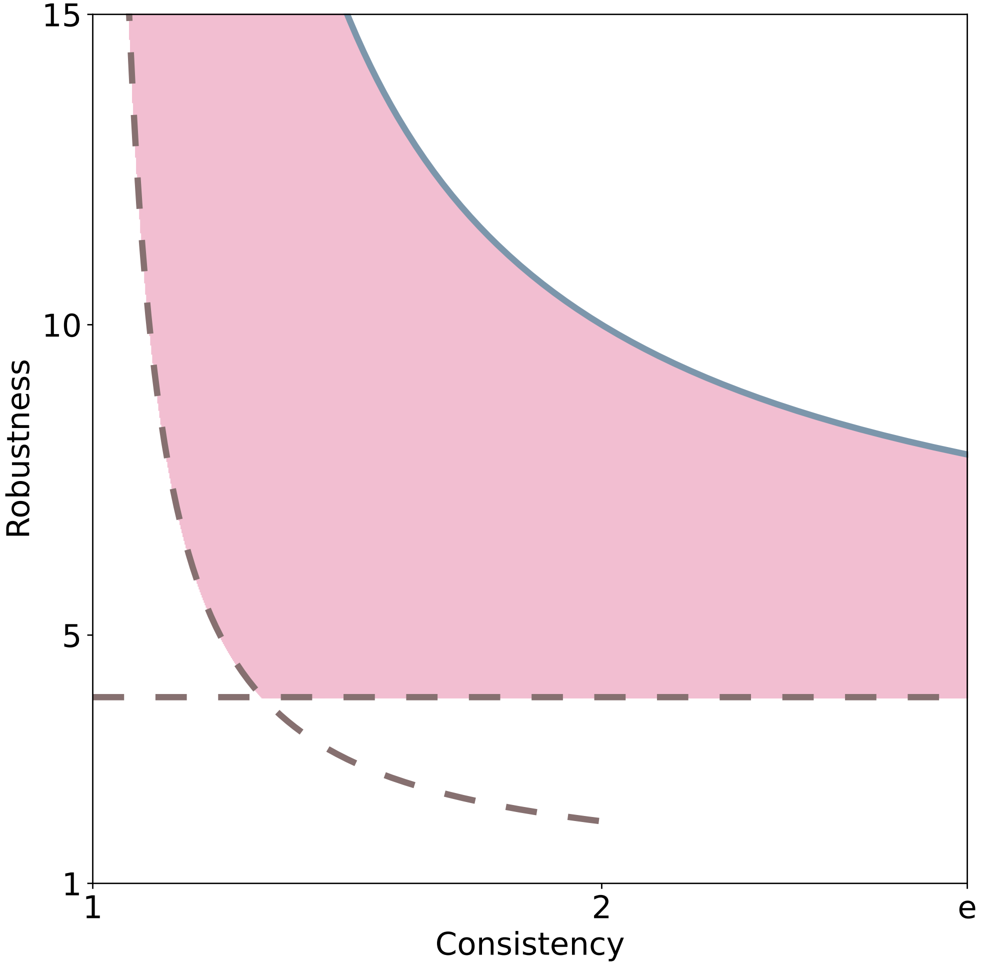


Deterministic algorithm

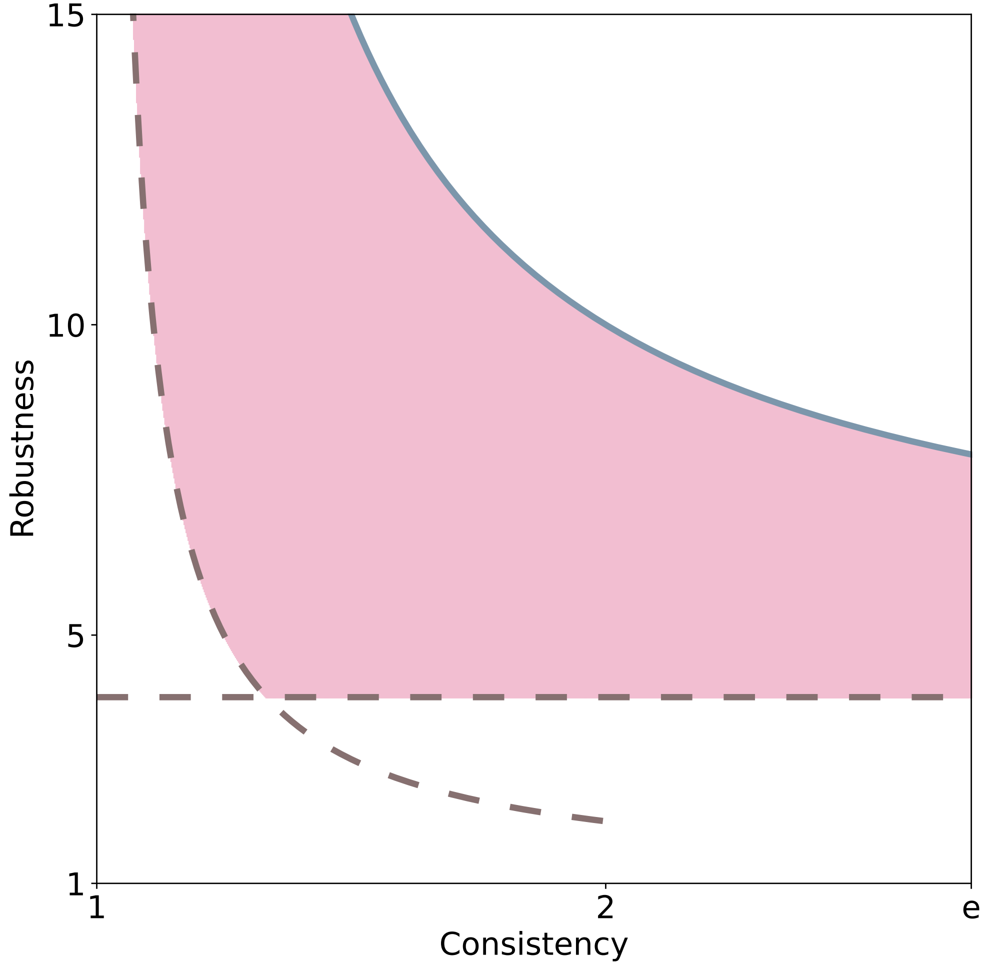
LB on the competitiveness
- 4 is best possible for det

[Zhang et al., Inf. Process Lett, 2011]

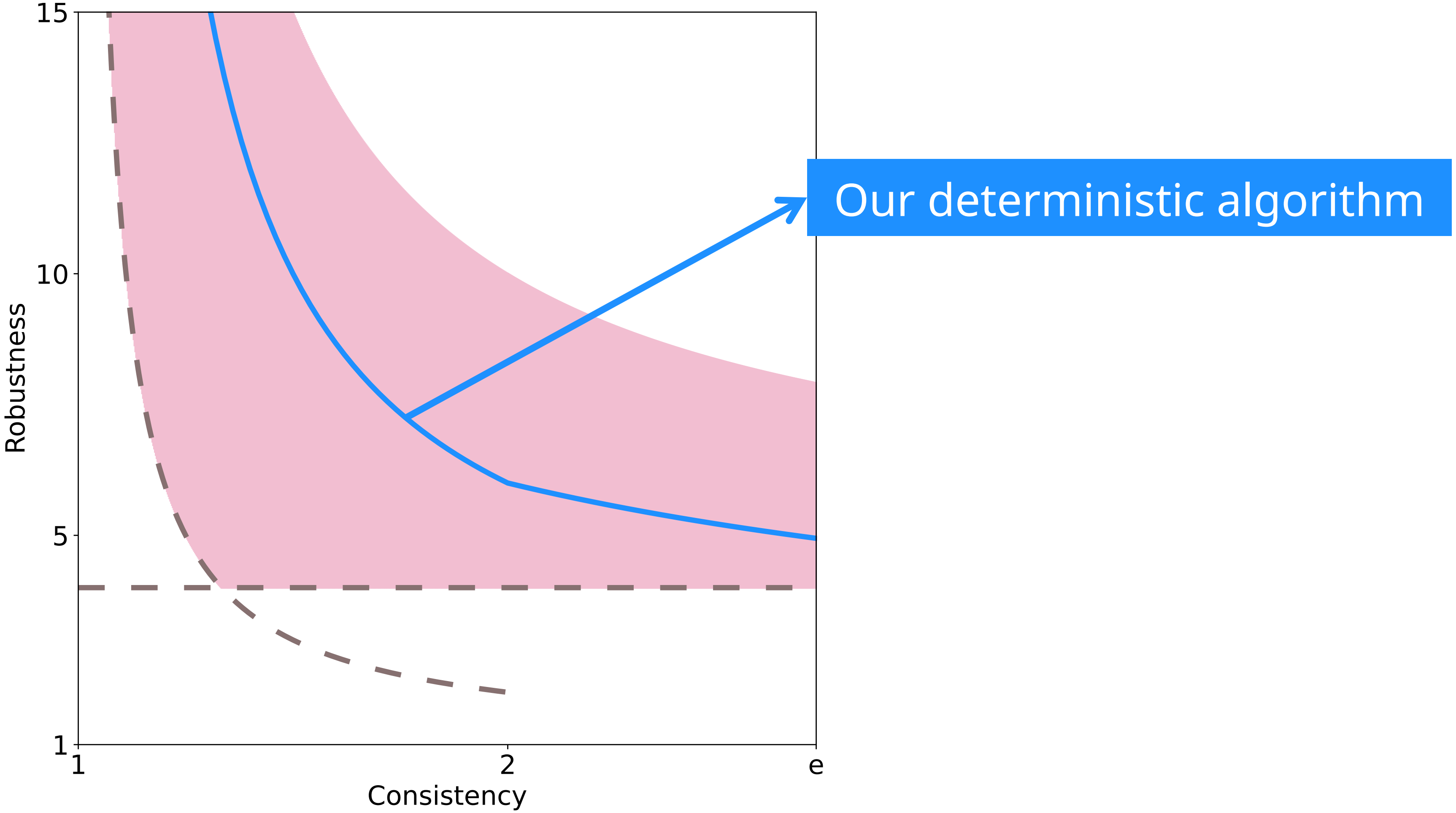
Previous Work (Deterministic)



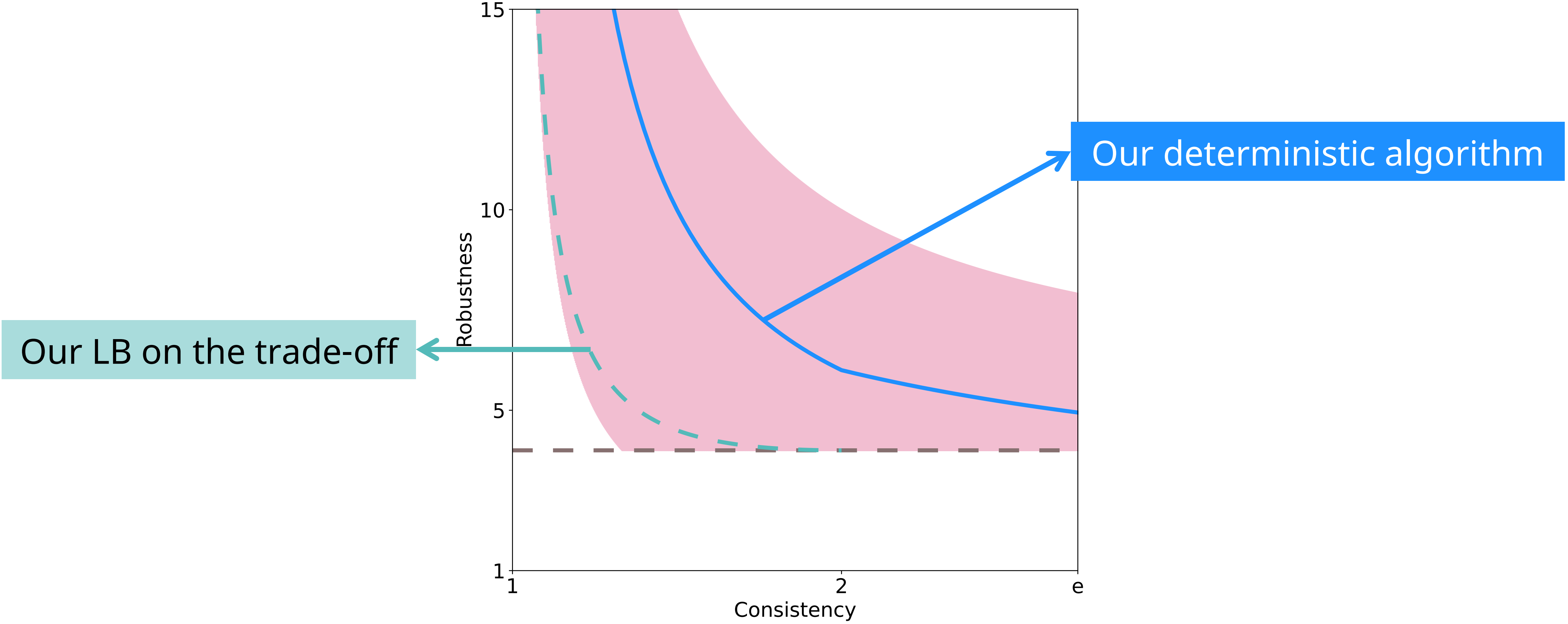
Our Results (Deterministic)



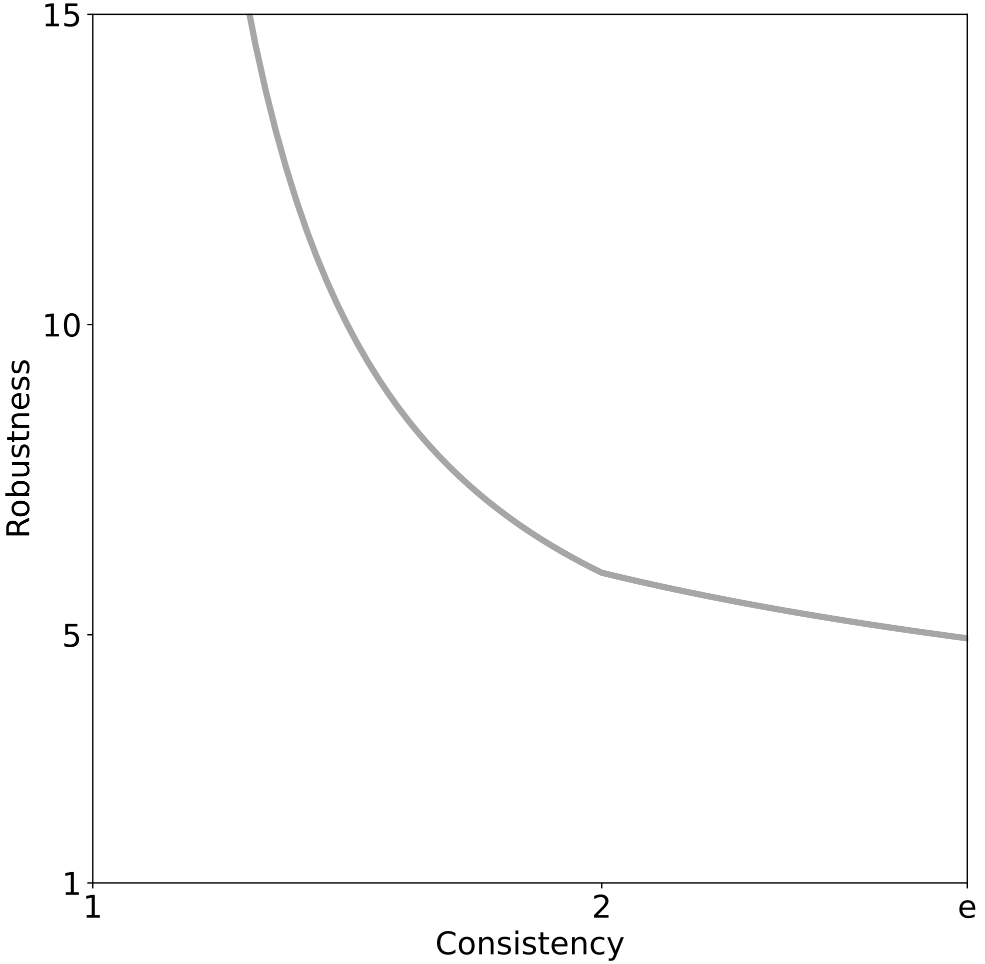
Our Results (Deterministic)



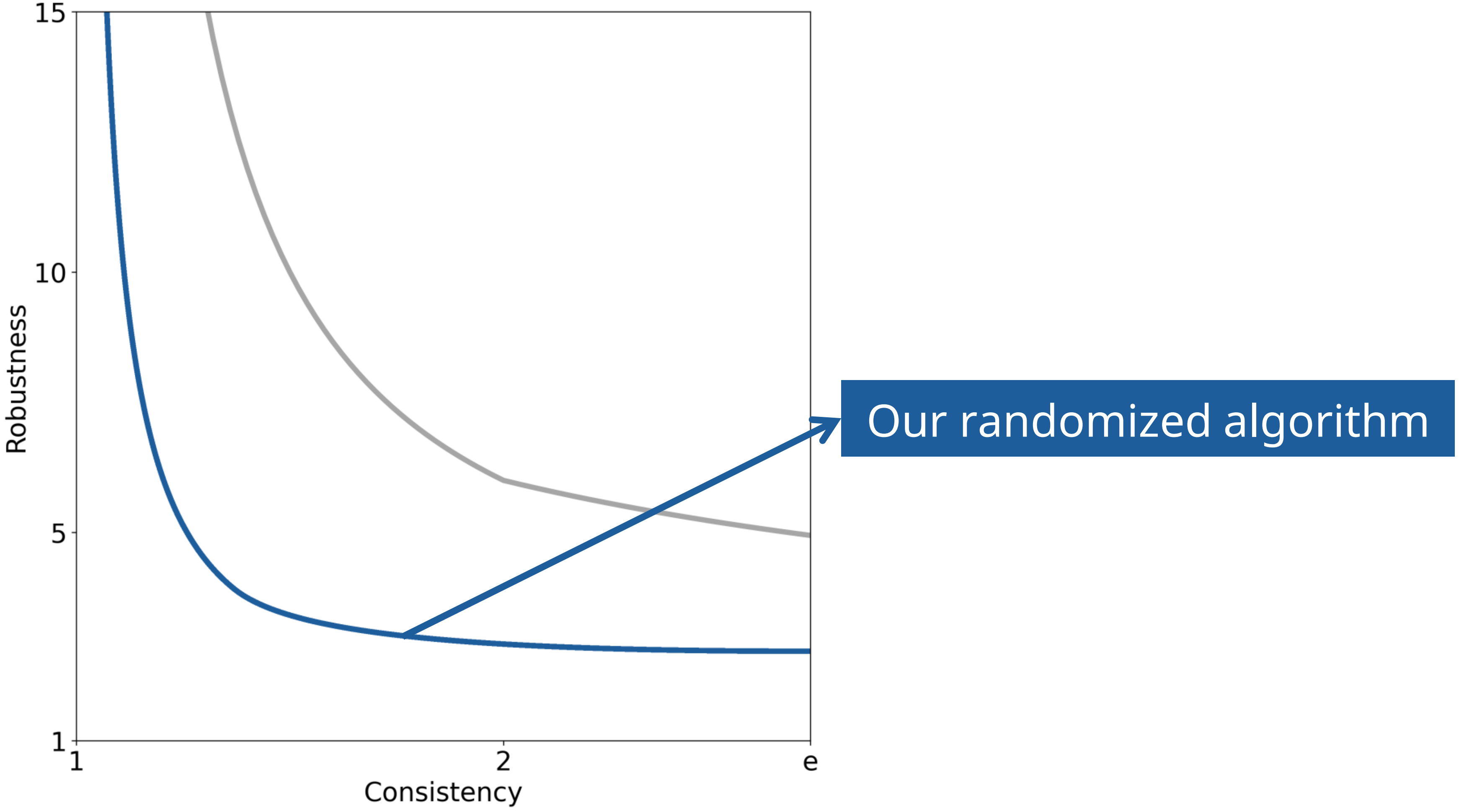
Our Results (Deterministic)



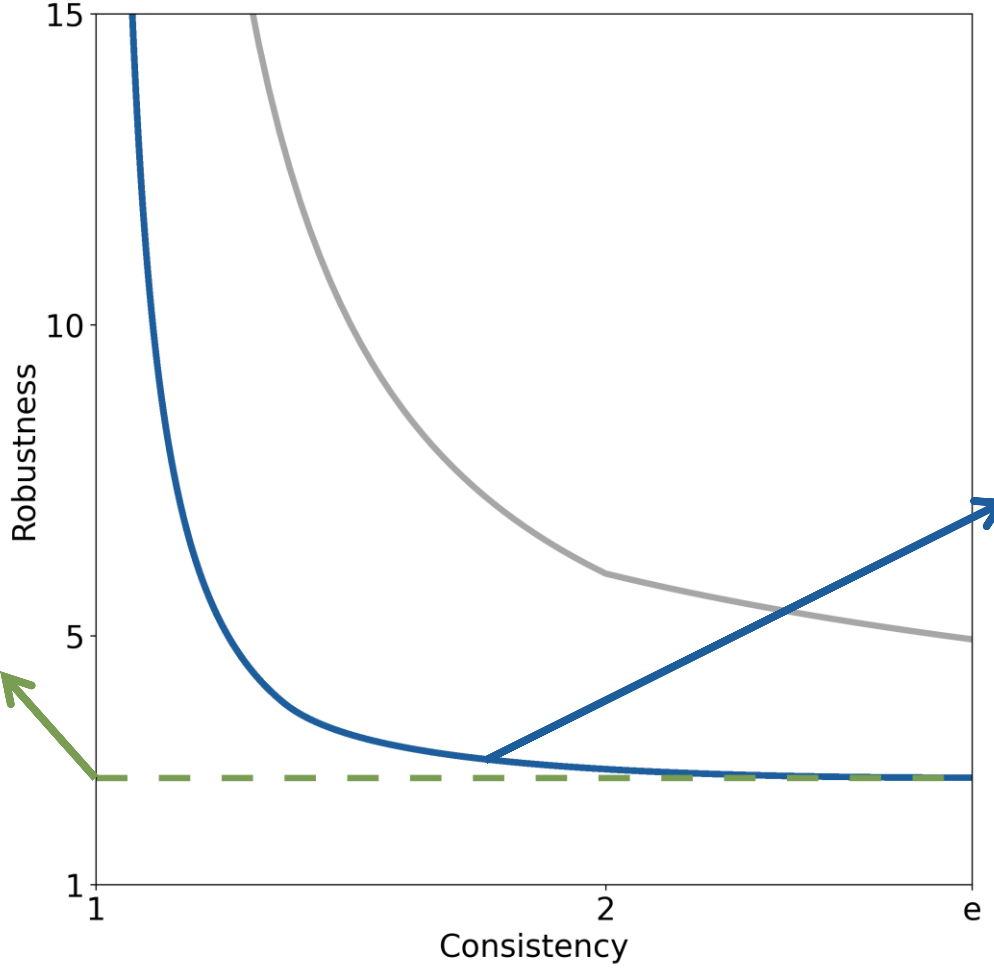
Our Results (Randomized)



Our Results (Randomized)



Our Results (Randomized)



LB on the competitiveness
- *e* is best possible

Our randomized algorithm

Roadmap of Algorithms

Deterministic 4-competitive algorithm → Randomized *e*-competitive algorithm → Randomized learning augmented algorithm

Roadmap of Algorithms

Deterministic 4-competitive algorithm

Randomized e-competitive algorithm

Randomized learning augmented algorithm

Deterministic Competitive Algorithm

for each round i = 0,1,… :

  set *budget* to 2^i

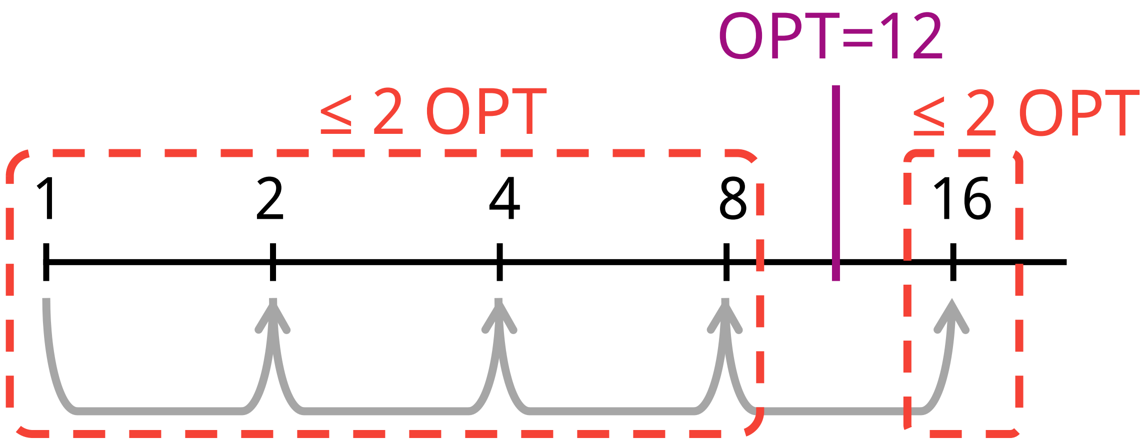
  choose “best” options within the *budget*

Deterministic Competitive Algorithm

for each round i = 0,1,... :

set *budget* to 2^i

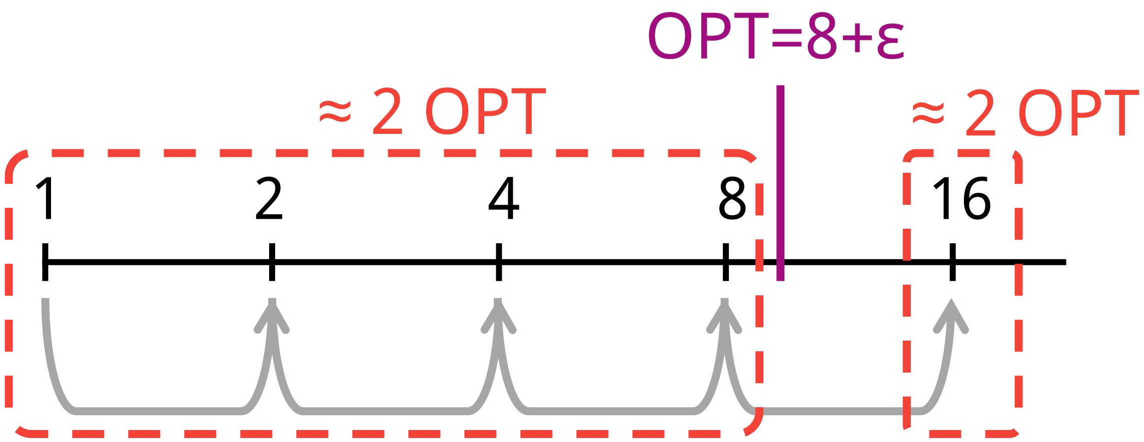
choose "best" options within the budget



Theorem. (Anand et al.) This algorithm is 4-competitive

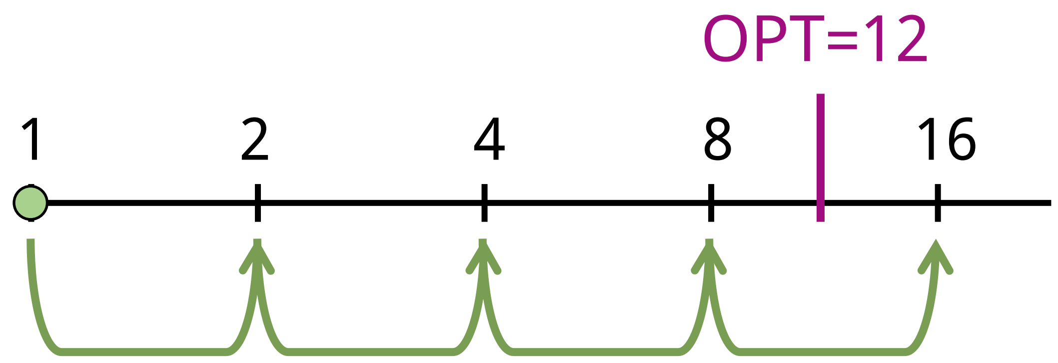
Worst Case of Deterministic Algorithm

for each round i = 0,1,… :
  set *budget* to 2^i
  choose "best" options within the *budget*



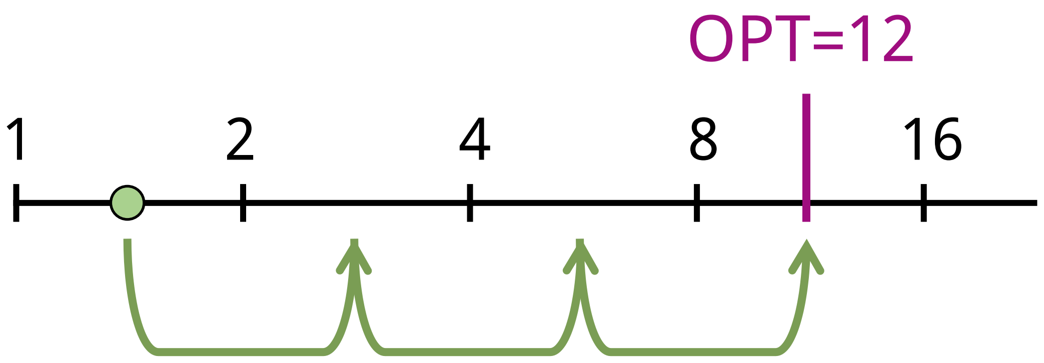
Randomized Initial Budget

sample initial *budget* $\alpha \in [1, 2)$
for each round $i = 0,1,\ldots$:
  set *budget* to $\alpha \cdot 2^i$
  choose "best" options within the *budget*



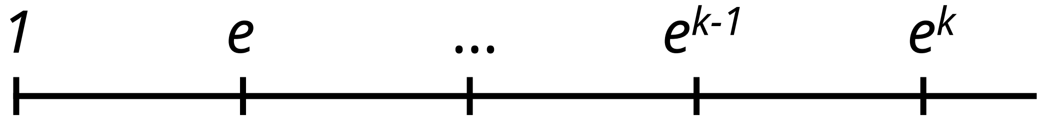
Randomized Initial Budget

sample initial *budget* $\alpha \in [1, 2)$
for each round $i = 0,1,\ldots$:
set *budget* to $\alpha \cdot 2^i$
choose "best" options within the *budget*



Randomized Competitive Algorithm

sample initial *budget* $\boldsymbol{\alpha} \in [1, e)$ from pdf $f(\alpha) = 1/\alpha$
for each round $i = 0,1,\ldots$:
  set *budget* to $\boldsymbol{\alpha} \cdot e^i$
  choose "best" options within the *budget*



Theorem. (SLLA, 2023) This algorithm is e-competitive

Theorem. (SLLA, 2023) This algorithm is best possible

Roadmap of Algorithms

Deterministic 4-competitive algorithm

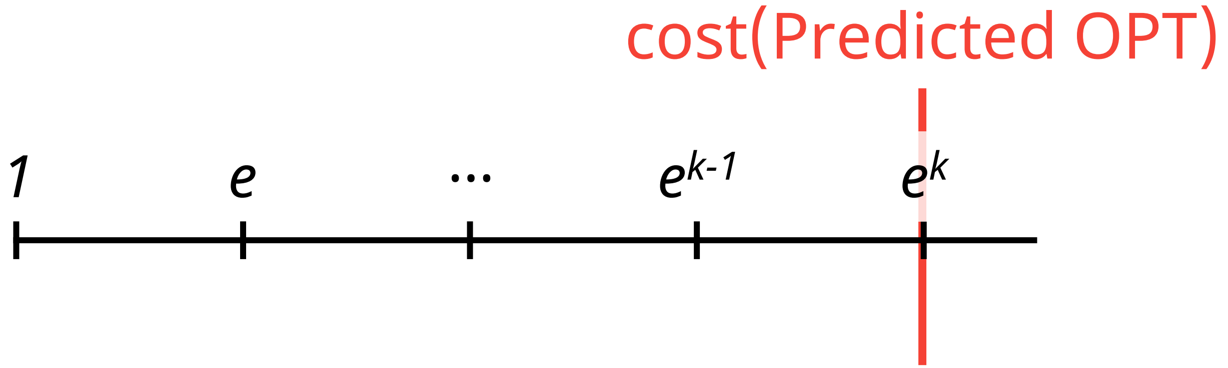
Randomized *e*-competitive algorithm

Randomized learning augmented algorithm

Randomized Learning Augmented Algorithm

Given: prediction on the last day

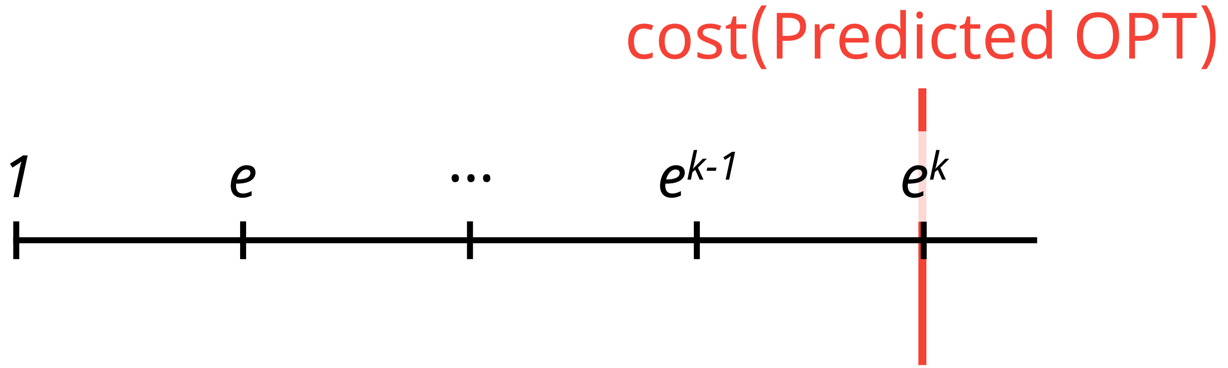
assume *cost* (“*predicted opt soln*”) = e^k for some int k



Randomized Learning Augmented Algorithm

Given: prediction on the last day

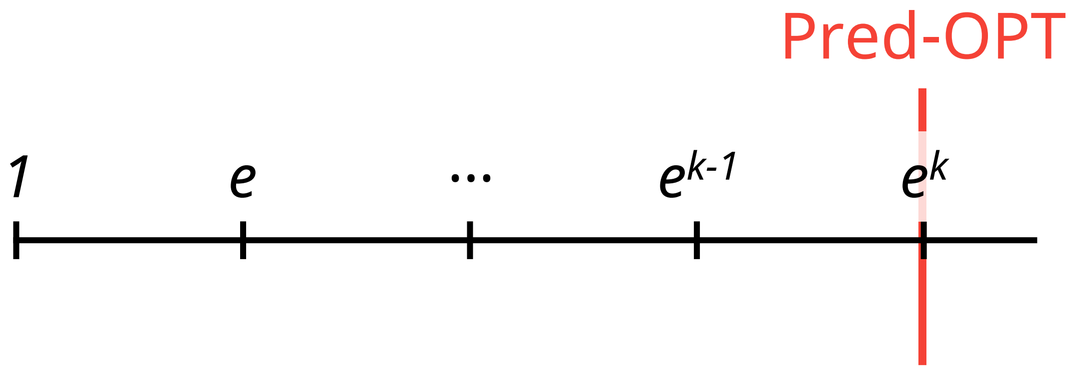
assume *cost* (“*predicted opt soln*”) = e^k for some int k, WLOG



Randomized Learning Augmented Algorithm

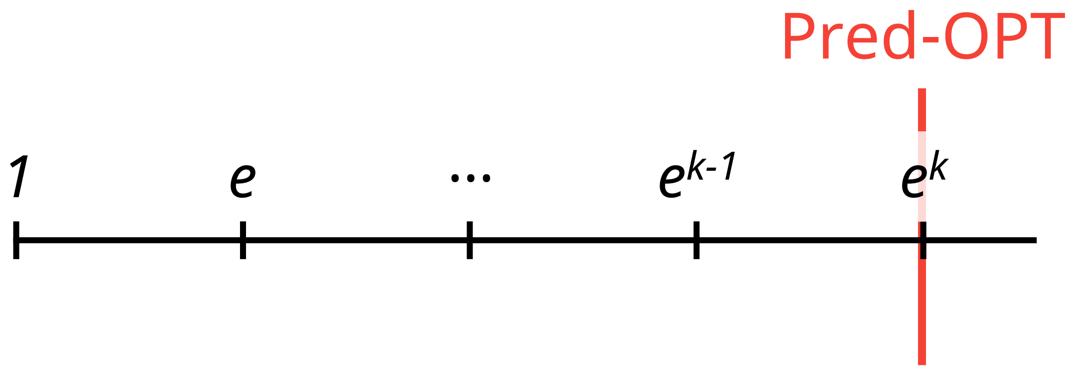
Given: prediction on the last day

Pred-OPT = e^k for some int k, WLOG



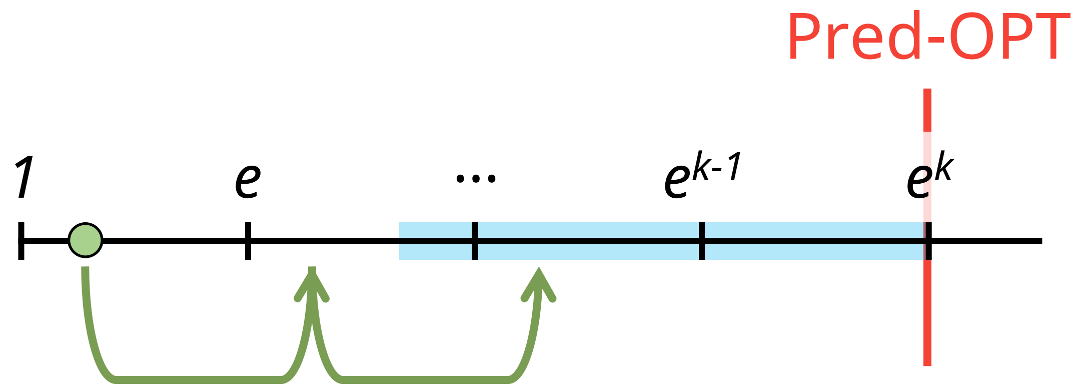
Randomized Learning Augmented Algorithm

Given: prediction on the last day, "level of confidence" $\boldsymbol{\lambda} \in [0,1]$
Pred-OPT = e^k for some int k, WLOG
sample initial *budget* $\alpha \in [1, e)$ from pdf $f(\alpha) = 1/\alpha$
for each round $i = 0,1,\ldots$:
  set *budget* to $\alpha \cdot e^i$
  if the *budget* $\in [\boldsymbol{\lambda} \cdot$ Pred-OPT, Pred-OPT), " *follow the prediction* " (at most once)
  otherwise, choose "best" options within the *budget*



Randomized Learning Augmented Algorithm

Given: prediction on the last day, ”level of confidence” $\boldsymbol{\lambda} \in [0,1]$
Pred-OPT = e^k for some int k, WLOG
sample initial *budget* $\alpha \in [1, e)$ from pdf $f(\alpha) = 1/\alpha$
for each round $i = 0,1,\ldots$:
  set *budget* to $\alpha \cdot e^i$
  if the *budget* ∈ [$\boldsymbol{\lambda}$ · Pred-OPT, Pred-OPT), “ *follow the prediction* ” (at most once)
  otherwise, choose “best” options within the *budget*



Randomized Learning Augmented Algorithm

Given: prediction on the last day, "level of confidence" $\boldsymbol{\lambda} \in [0,1]$

Pred-OPT = e^k for some int k, WLOG

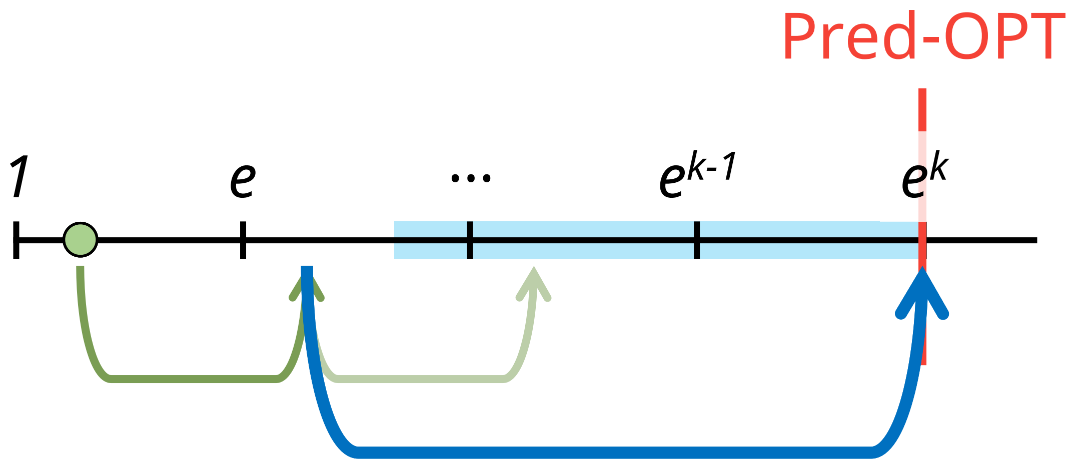
sample initial *budget* $\alpha \in [1, e)$ from pdf $f(\alpha) = 1/\alpha$

for each round $i = 0,1,\ldots$:

  set *budget* to $\alpha \cdot e^i$

  if the *budget* $\in [\boldsymbol{\lambda} \cdot \text{Pred-OPT}, \text{Pred-OPT})$, " *follow the prediction* " (at most once)

  otherwise, choose "best" options within the *budget*



Randomized Learning Augmented Algorithm

Given: prediction on the last day, "<u>level of confidence</u>" $\boldsymbol{\lambda} \in [0,1]$

Pred-OPT = e^k for some int k, WLOG

sample initial *budget* $\alpha \in [1, e)$ from pdf $f(\alpha) = 1/\alpha$

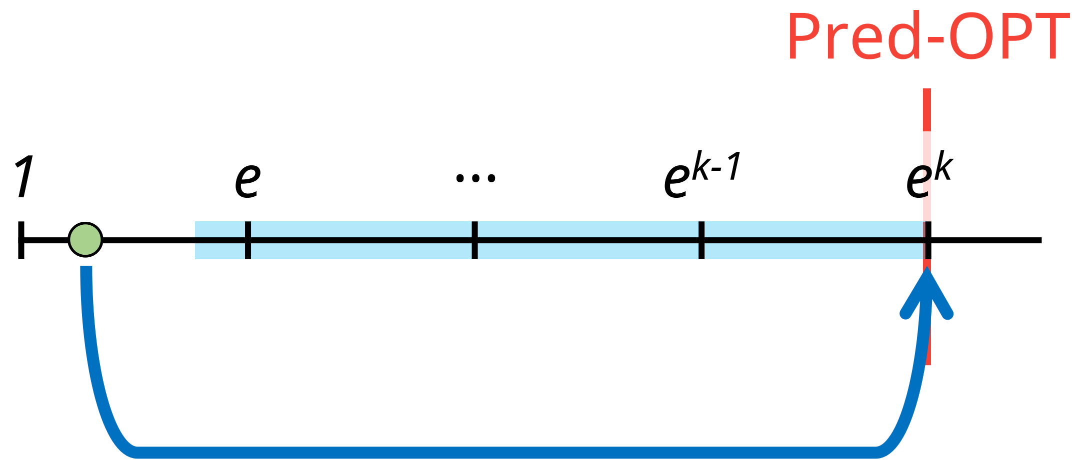
for each round $i = 0,1,\ldots$:

  set *budget* to $\alpha \cdot e^i$

  if the *budget* $\in [\boldsymbol{\lambda} \cdot$ Pred-OPT, Pred-OPT), " *follow the prediction* " (at most once)

  otherwise, choose "best" options within the *budget*

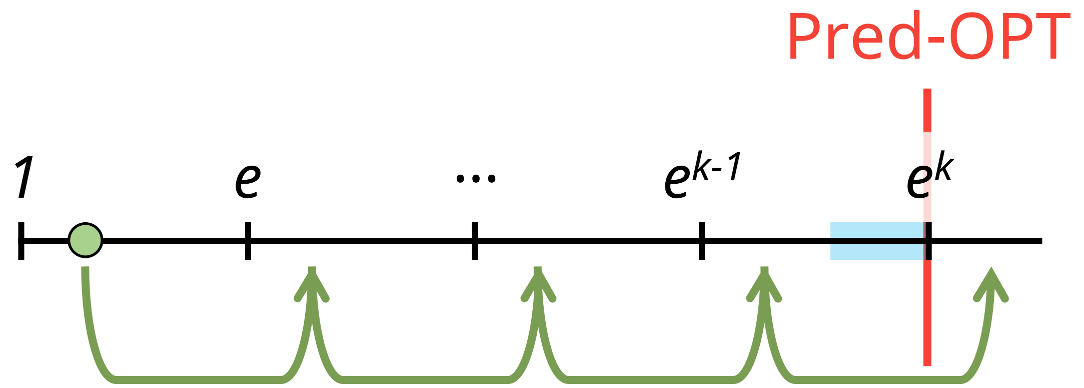
$\boldsymbol{\lambda}$ close to 0



Randomized Learning Augmented Algorithm

Given: prediction on the last day, "level of confidence" $\boldsymbol{\lambda} \in [0,1]$
Pred-OPT = e^k for some int k, WLOG
sample initial *budget* $\alpha \in [1, e)$ from pdf $f(\alpha) = 1/\alpha$
for each round $i = 0,1,\ldots$:
    set *budget* to $\alpha \cdot e^i$
    if the *budget* $\in$ [$\boldsymbol{\lambda}$ · Pred-OPT, Pred-OPT), " *follow the prediction* " (at most once)
    otherwise, choose "best" options within the *budget*

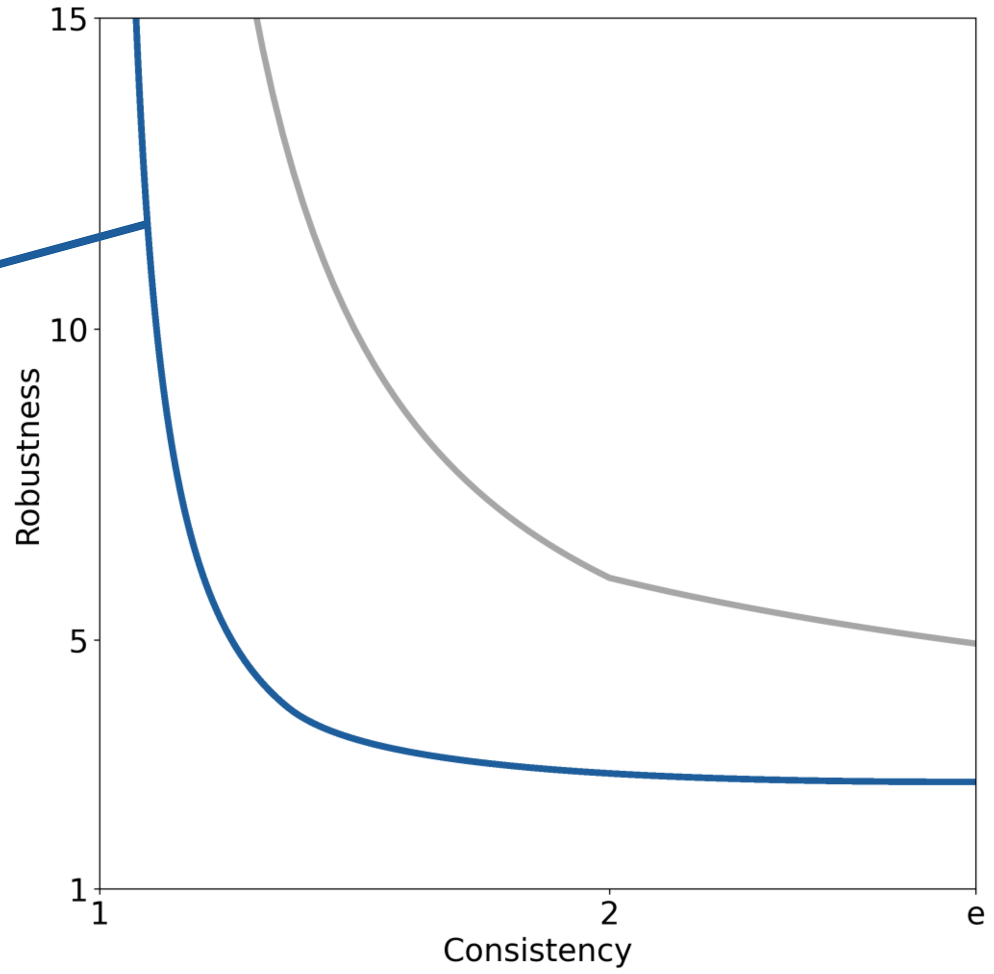
$\boldsymbol{\lambda}$ close to 1



Randomized Learning-Augmented Algorithm

Theorem. (SLLA, 2023) For $\lambda \in [0,1]$, this algorithm is $\chi(\lambda)$-consistent (e^{λ}/λ)-robust, where

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



Lower Bounds

- Button problem

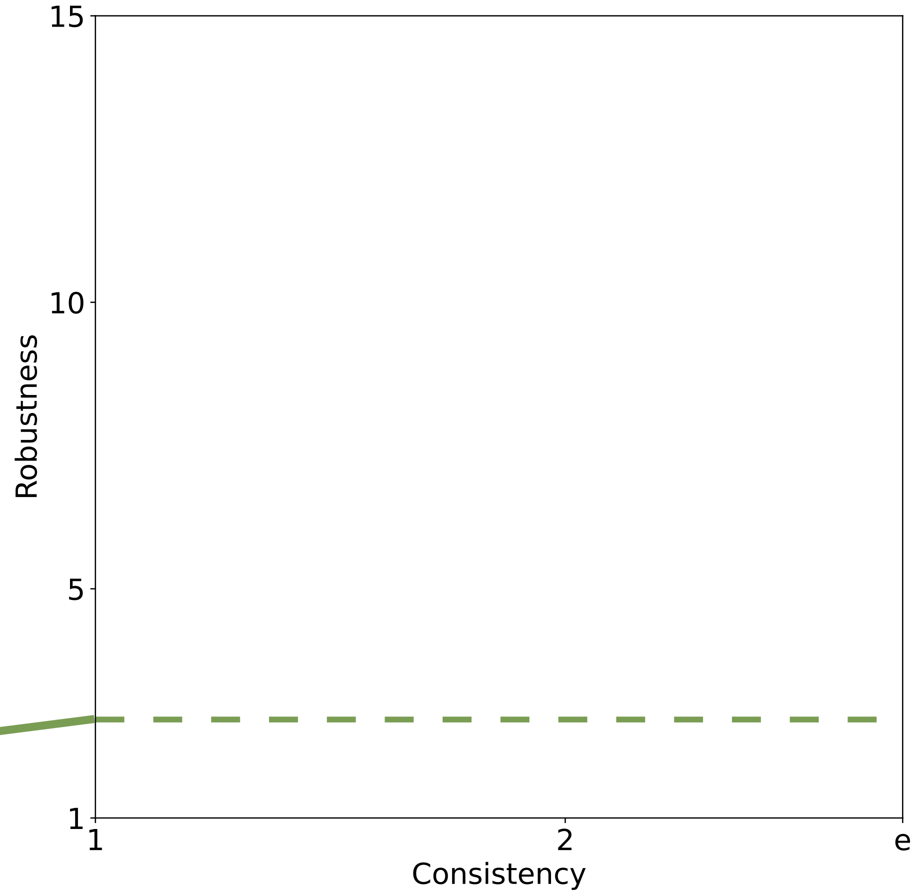
  - is (almost) as hard as multi-option ski rental

- Competitive ratio of any algorithm for the button problem ≥

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

- Construct a family of instances and consider the dual solutions

Lower Bounds for Button Problem

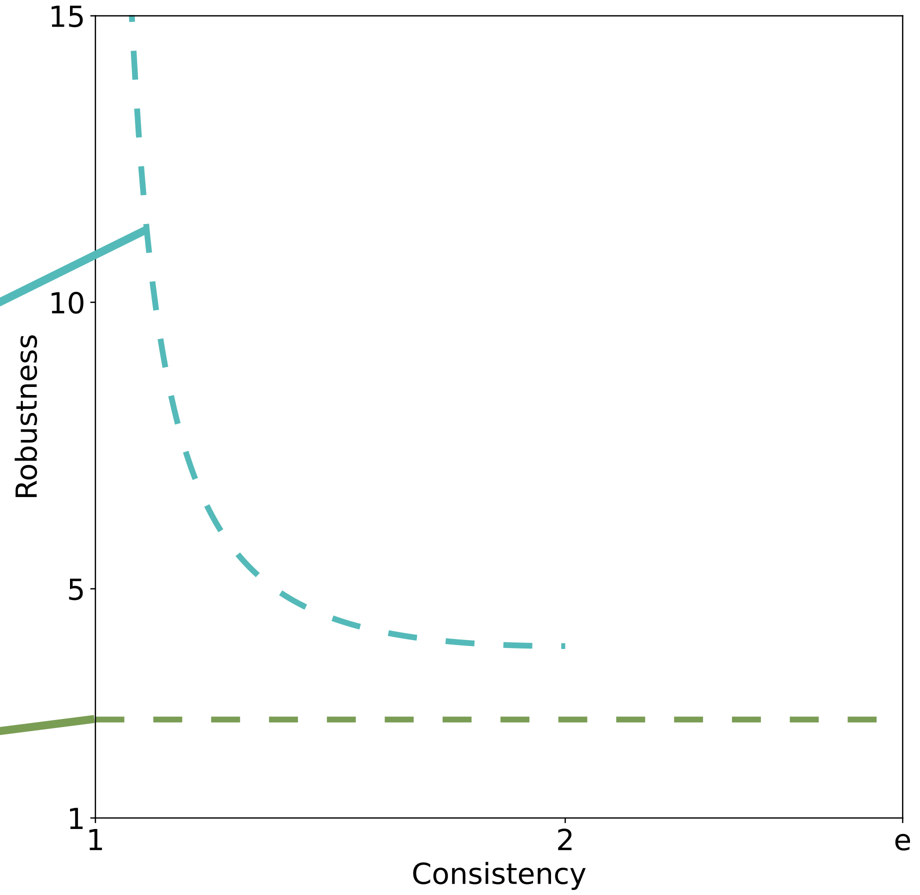


Theorem. (SLLA, 2023)
LB on the competitiveness

Lower Bounds for Button Problem

Theorem. (SLLA, 2023)
LB on the trade-off
for the deterministic algorithm

Theorem. (SLLA, 2023)
LB on the competitiveness



Thank you for listening!