Learning-Augmented Dynamic Power Management with Multiple States via New Ski Rental Bounds

Antonios Antoniadis, Christian Coester, Marek Eliáš, Adam Polak, Bertrand Simon

DYNAMIC POWER MANAGEMENT WITH MULTIPLE STATES (DPM)

Machine with multiple sleep states:

Sx	Description
S0	Active
S0ix	Low Power S0 Idle
S1	*Power on Suspend (POS)*
S2	CPU powered off.
S3	*Standby*
S4	Hibernation
S5	Soft Off

Deeper sleep → higher wake-up cost

Input: sequence of idle periods

Our task: choose sleep states during each idle period

- duration not known in advance!

RAISON D'ÊTRE OF LEARNING-AUGMENTED ALGORITHMS

	worst-case	typical
classical algorithms	☺	☹
machine-learned predictions	☹	☺
learning-augmented algorithms	☺	☺

DESIDERATA OF LEARNING-AUGMENTED ALGORITHMS

= ALGORITHMS WITH PREDICTIONS

Consistency

close-to-optimal performance when predictions accurate

Robustness

worst-case guarantees, even when predictions adversarial

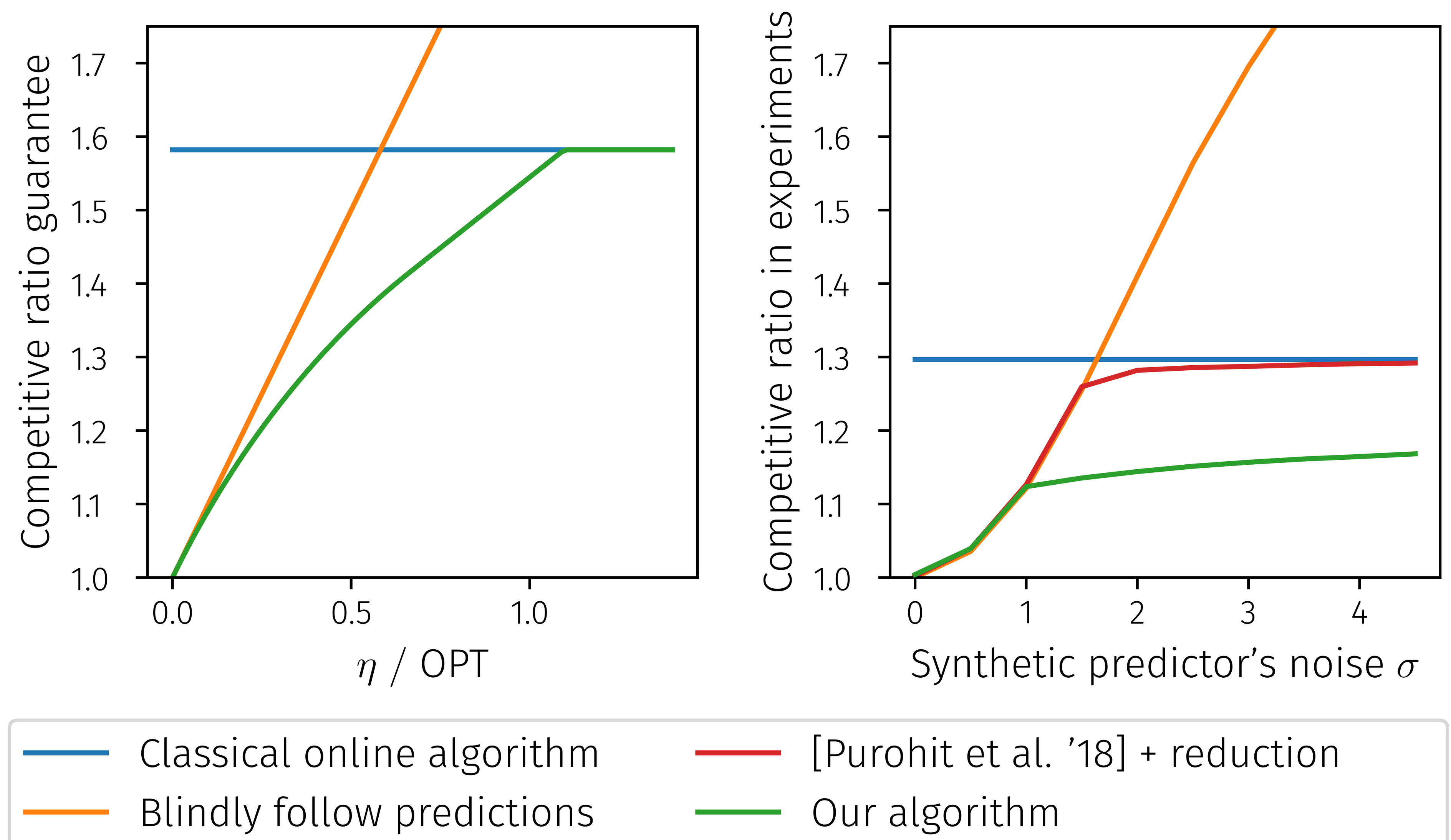
Smoothness

performance degrades slowly in the prediction error

OUR LEARNING-AUGMENTED DPM ALGORITHM

When i-th idle period starts, get prediction τ_i of its length ℓ_i

$$\underbrace{\eta}_{\text{total prediction error}} = \sum_{i=1}^{n} \underbrace{\alpha}_{\text{power consumption of the active state}} \cdot |\underbrace{\tau_i}_{\text{predicted idle period length}} - \underbrace{\ell_i}_{\text{actual idle period length}}|$$



Main ingredient:

- New **smooth** learning-augmented algorithm for **ski rental**
  - optimal trade-off between consistency and error-dependence

Other ingredients:

- Reduction from DPM to ski-rental
  - extension of [Lotker et al. '12] to the learning-augmented setting
- Online learning for hyperparameter optimization

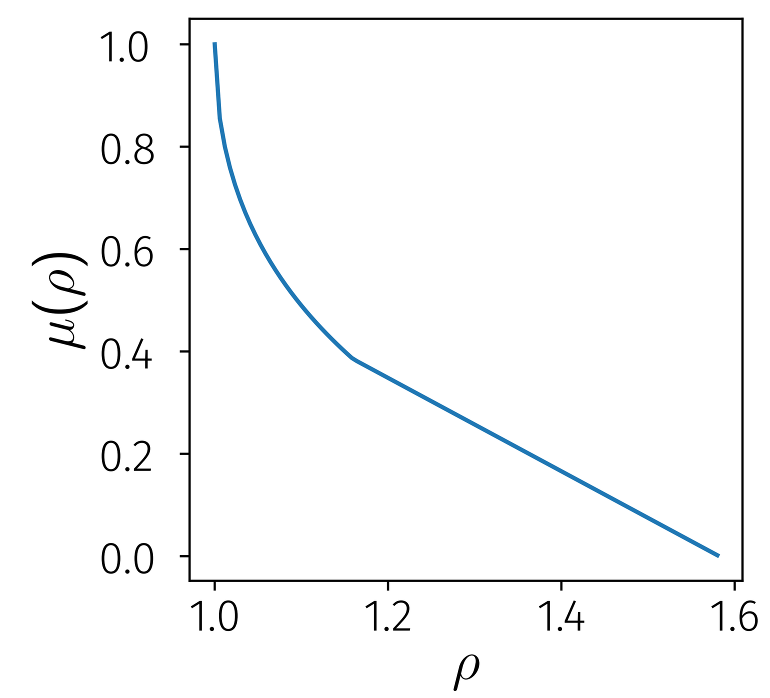
SKI RENTAL

During a ski season of unknown length, each day either

per day cost α ⟶ **rent** or **buy** ⟵ one-time cost β

Previous algorithms:

- deterministic 2-competitive [folklore]
- randomized $e/(e-1)$-competitive [Karlin et al. '90]
- learning-augmented [Purohit et al. '18], [Angelopoulos et al. '20]
  - optimal consistency/robustness trade-off, no smoothness



Our new ski rental algorithm is $(\rho, \mu(\rho))$-competitive, i.e.

$$\text{cost}(ALG) \leqslant \rho \cdot \text{cost}(OPT) + \mu(\rho) \cdot \eta$$

where $\rho \in [1, \frac{e}{e-1}]$, $T^2 e^{-T} = 1 - \frac{1}{\rho}$

$$\mu(\rho) = \max\left\{\frac{1-\rho\frac{e-1}{e}}{\ln 2}, \rho(1-T)e^{-T}\right\}$$