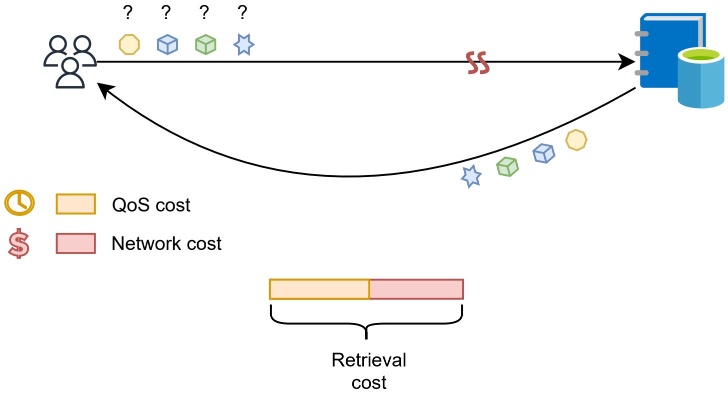
No-Regret Caching via Online Mirror Descent

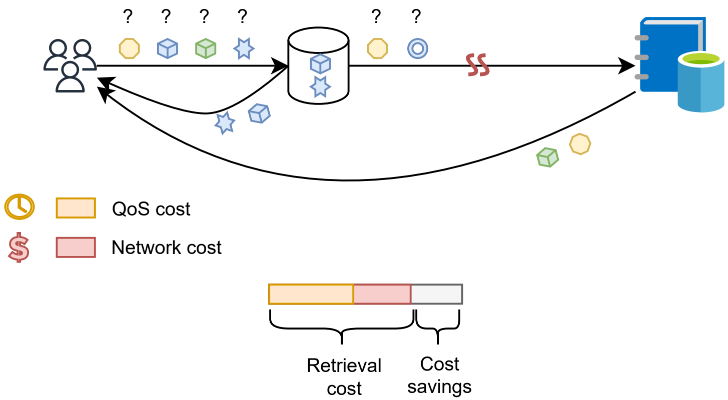
Tareq Si Salem[1] Giovanni Neglia[2] Stratis Ioannidis[1]

[1]Northeastern University, USA [2]Inria, France

Remote Service



Remote Service and Local Cache



Presentation Organization

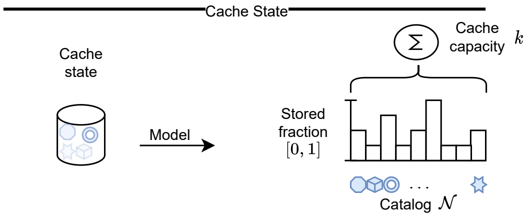
1. Context
2. **Contributions Summary**
3. System Model
4. Classical Caching Policies
5. Gradient-based policies
6. Integral Caching
7. Numerical Experiments

Contributions Summary

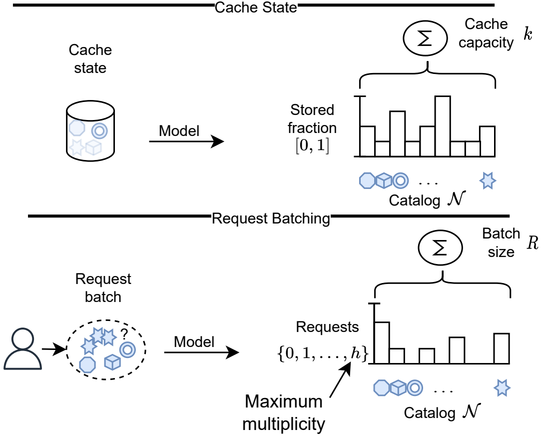
- We initiate the study of mirror descent techniques in the context of caching.
- We demonstrate that the regret of various mirror descent policies depends on the diversity of the request process, and we characterize the optimality of online mirror descent (OMD) caching policies under different diversity regimes.
- We show that gradient-based policies can be extended to the integral setting, where the cache can only store files in their entirety, using opportune randomized rounding techniques while preserving their regret guarantees.

Presentation Organization

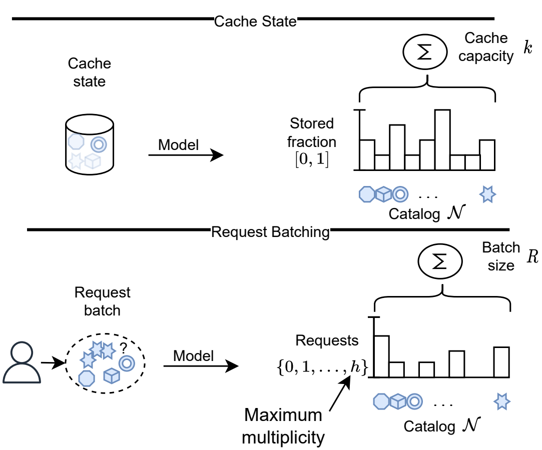
System Model

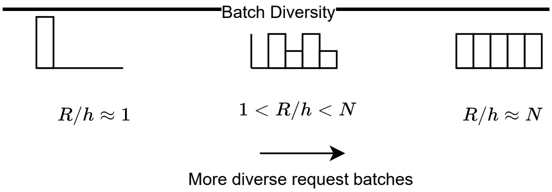


System Model

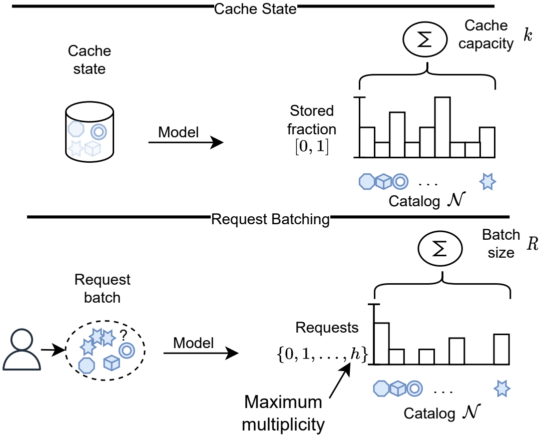


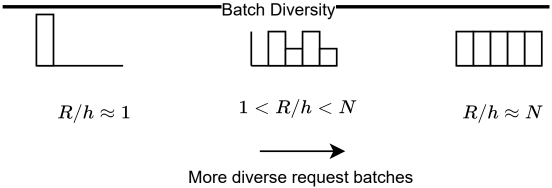
System Model





System Model

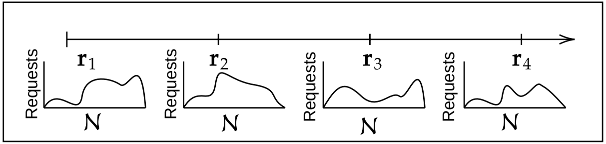




When a request batch $\mathbf{r}_t$ arrives, the cache incurs the following cost:

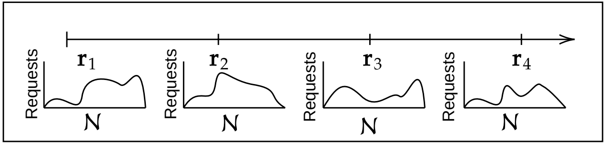
$$f_{\mathbf{r}_t}(\mathbf{x}_t) = \sum_{i=1}^{N} w_i r_{t,i}(1 - x_{t,i}).$$

Setting



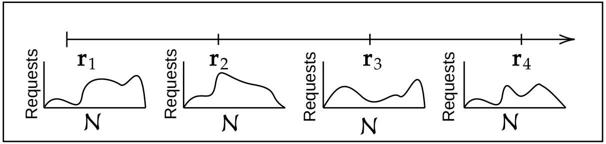
Requests
$\mathbf{r}_1$
$\mathcal{N}$
Requests
$\mathbf{r}_2$
$\mathcal{N}$
Requests
$\mathbf{r}_3$
$\mathcal{N}$
Requests
$\mathbf{r}_4$
$\mathcal{N}$

Setting

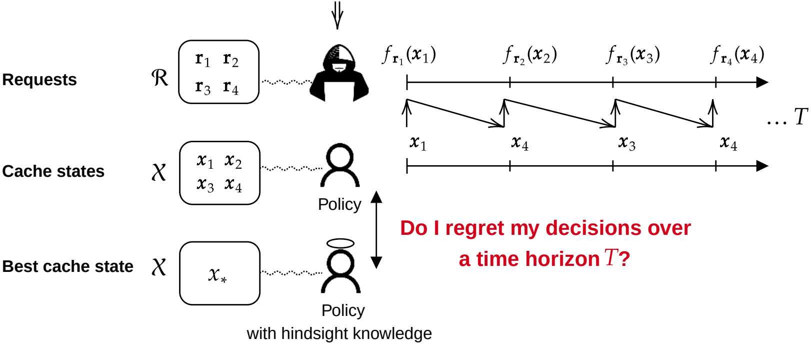


Noisy unpredictable environment can act as an **adversary** in the worst case scenario

Setting



Noisy unpredictable environment can act as an **adversary** in the worst case scenario



Performance Metric

Definition

The regret of a policy $\mathcal{A}$ is defined as

$$\mathrm{Regret}_T(\mathcal{A}) \triangleq \sup_{\{\boldsymbol{r}_t\}_{t=1}^T \in \mathcal{R}^T} \left\{ \sum_{t=1}^T f_{\boldsymbol{r}_t}(\boldsymbol{x}_t) - \min_{x \in \mathcal{X}} \sum_{t=1}^T f_{\boldsymbol{r}_t}(\boldsymbol{x}) \right\}.$$

When $\mathrm{Regret}_T(\mathcal{A})$ is sublinear in T, the policy $\mathcal{A}$ experiences no regret on average as $T \to \infty$.

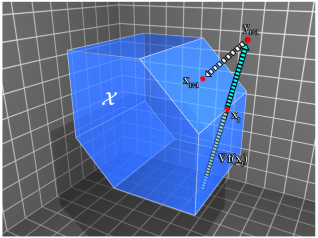
Presentation Organization

Classical Caching Policies

LRU: When a cache hit occurs (item found in the cache), we refresh the content and move it to the head of a queue, otherwise we evict the least recently requested item from the tail of the queue.
LFU: We keep the most requested files in the cache.
W-LFU: We keep the most requested files within a time window W. We obtain a trade-off between adaptability and precision.

Presentation Organization

Online Gradient Descent



At time t, Online Gradient Descent (OGD)'s update rule is

(*additive* gradient update) $$\boldsymbol{y}_{t+1} = \boldsymbol{x}_t - \eta_t \nabla f_t(\boldsymbol{x}_t)$$

(projection step) $$\boldsymbol{x}_{t+1} = \Pi_{\mathcal{X}}(\boldsymbol{y}_{t+1})$$

The operator $\Pi_{\mathcal{X}} : \mathbb{R}^d \to \mathcal{X}$ is the Eulidean projection $\Pi_{\mathcal{X}}(\boldsymbol{y}) \triangleq \arg\min_{\boldsymbol{x} \in \mathcal{X}} \|\boldsymbol{x} - \boldsymbol{y}\|_2$.

Paschos et al. [PDVI19a], show that OGD attains sub-linear ($\mathcal{O}\left(\sqrt{T}\right)$) regret when $R = h = 1$ in the context of caching.

Online Mirror Descent (OMD)

A mirror map $\Phi : \mathcal{D} \subset \mathbb{R}^{\mathcal{N}} \to \mathbb{R}$ defines a unique algorithm.

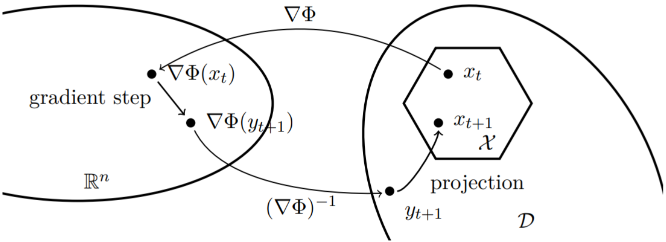


Figure: OMD update rule [Bub15].

Desiderata of a Mirror Map

We are concerned with both (a) the regret attained, and (b) computational complexity issues, particularly pertaining to the associated Bregman projection.

Negative Entropy Online Mirror Descent

A more interesting choice of a mirror map is given by the negative entropy

$$\Phi(\boldsymbol{x}) = \sum_{i=1}^{d} x_i \log(x_i). \quad (1)$$

The policy configured for caching decision sets $\mathcal{X}$ amounts to

(*multiplicative* gradient update) $$y_{t+1,i} = x_{t,i} e^{-\eta l_{t,i}}, i \in \mathcal{N} \quad (2)$$

(projection step) $$\boldsymbol{x}_{t+1} = \arg\min_{\mathcal{X} \in \mathcal{D} \cap \mathcal{X}} D_\Phi(\boldsymbol{x}, \boldsymbol{y}_{t+1}) \quad (3)$$

Mirror Map Selection

- We provide the first results to guide the selection of the best policy.

Theorem

OGD is optimal for $\frac{R}{h} \leq k$ (low diversity and large cache sizes). $\mathrm{OMD}_{\mathrm{NE}}$ is optimal for $\frac{R}{h} > 2\sqrt{Nk}$ (high diversity and small cache sizes).

Mirror Map Selection

- We provide the first results to guide the selection of the best policy.

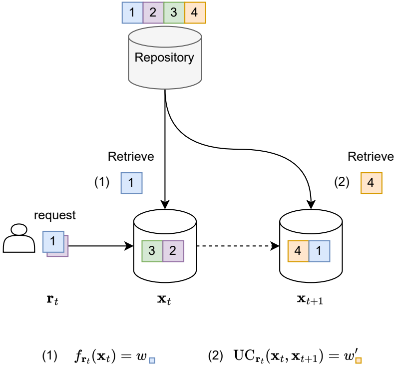
Theorem

OGD is optimal for $\frac{R}{h} \leq k$ (low diversity and large cache sizes). $\mathrm{OMD}_{\mathrm{NE}}$ is optimal for $\frac{R}{h} > 2\sqrt{Nk}$ (high diversity and small cache sizes).

- Highly efficient projection algorithm for $\mathrm{OMD}_{\mathrm{NE}}$ that yields a policy that has the lowest time-complexity per iteration among recent works [PDVI19b, PS21, BBS20, MS21].

Update Costs

We define the update cost at time t as $\mathrm{UC}_{\boldsymbol{r}_t}(\boldsymbol{x}_t, \boldsymbol{x}_{t+1}) \triangleq \sum_{i \notin \mathrm{supp}(\boldsymbol{r}_t)} w'_i \max\{0, x_{t+1,i} - x_{t,i}\}$.

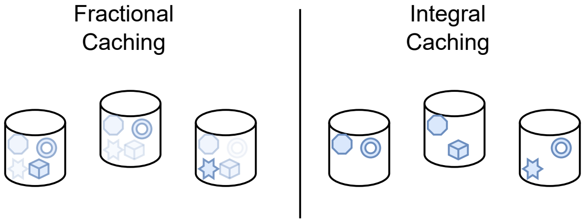


Fractional Caching Incurs no Update Costs

We prove that any request batch $\boldsymbol{r}_t$, for $\mathrm{OMD}_{\mathrm{NE}}$ or OGD, it holds $\mathrm{UC}_{\boldsymbol{r}_t}(\boldsymbol{x}_t, \boldsymbol{x}_{t+1}) = 0$.

Presentation Organization

Integral Caching – Necessity of Randomization



Proposition

Any deterministic policy restricted to select integral cache states in $\mathcal{Z} \triangleq \mathcal{X} \cap \{0,1\}^{\mathcal{N}}$ has linear regret, i.e.,

$$\text{Regret}_T(\mathcal{A}) \geq k\,(1 - k/N)\,T.$$

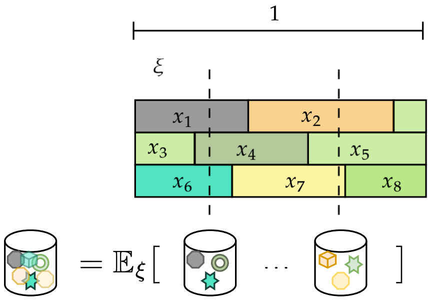
Randomized Integral Caching

We restrict ourselves to randomized rounding schemes that output $\boldsymbol{z}_t \in \mathcal{Z}$ such that $\mathbb{E}[\boldsymbol{z}_t] = \boldsymbol{x}_t$ through some rounding Ξ.

Remark

The expected regret $\mathrm{Regret}_T(\mathcal{A}, \Xi)$ is the same as the regret of $\mathcal{A}$.

A scheme that has this property is Madow's sampling [MM44]:



Randomized Integral Caching

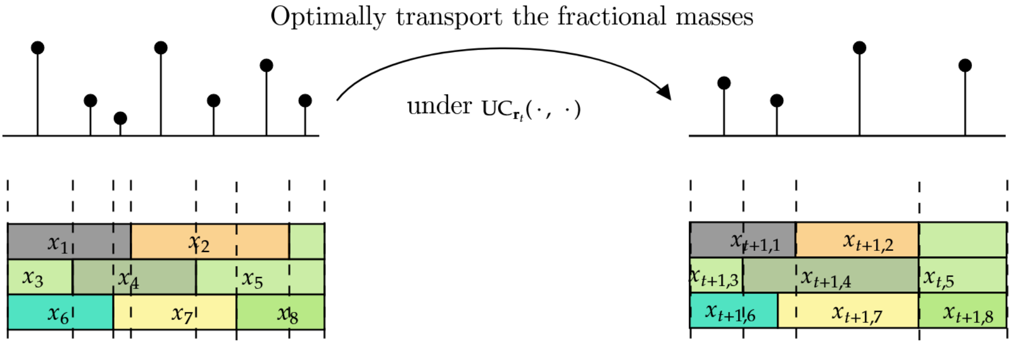
When considering the extended regret ($\text{E-Regret}_T(\mathcal{A}, \Xi)$) we lose immediately the regret guarantee:

Theorem

Any randomized caching policy constructed by an online policy $\mathcal{A}$ combined with online independent rounding as Ξ leads to $\Omega(T)$ $\text{E-Regret}(\mathcal{A}, \Xi)$.

Imposing dependence (coupling) between the two consecutive random states may significantly reduce the expected update cost.

Randomized Integral Caching – Optimal Transport

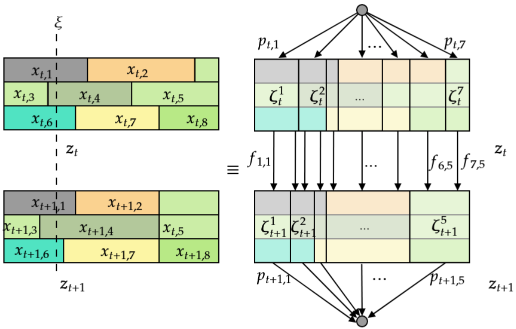


Remark

We prove that this scheme selected as Ξ, coupled with a no-regret policy $\mathcal{A}$, has sublinear extended regret guarantee. However, it has a time-complexity $\mathcal{O}\left(N^3\right)$.

Randomized Integral Caching – Simpler Approach

Online Coupled Rounding



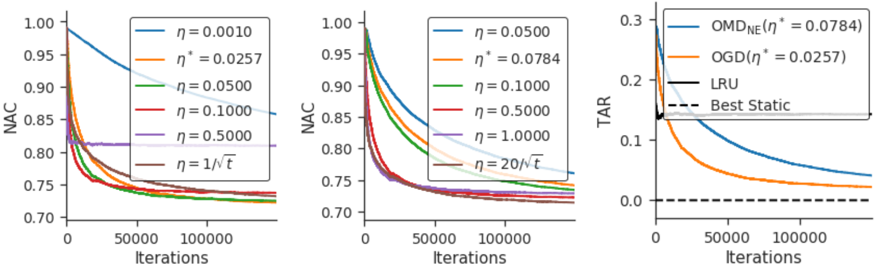
Theorem

A no-regret policy $\mathcal{A}$ combined with online coupled rounding Ξ has $\mathcal{O}\left(\sqrt{T}\right)$ $\text{E-Regret}_T(\mathcal{A}, \Xi)$.

Online Coupled Rounding has linear time complexity.

Presentation Organization

Fixed Popularity

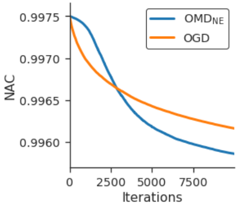


(a) NAC of OGD (b) NAC of $\mathrm{OMD}_{\mathrm{NE}}$ (c) Time-Averaged Regret

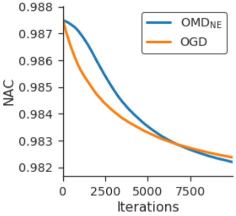
Remark

The learning rate denoted by η^* is the learning rate that gives the tightest worst-case regrets for OGD and $\mathrm{OMD}_{\mathrm{NE}}$. While this learning rate is selected to protect against any (adversarial) request sequence, it is not too pessimistic.

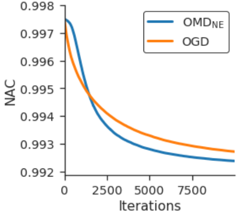
Effect of Diversity



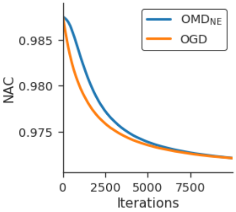
(a) $k = 25, \alpha = 0.1$



(b) $k = 125, \alpha = 0.1$



(c) $k = 25, \alpha = 0.2$

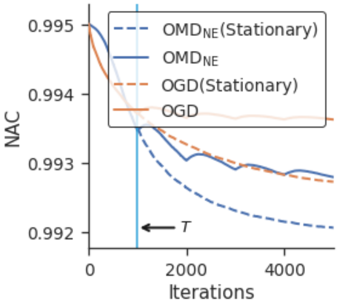


(d) $k = 125, \alpha = 0.2$

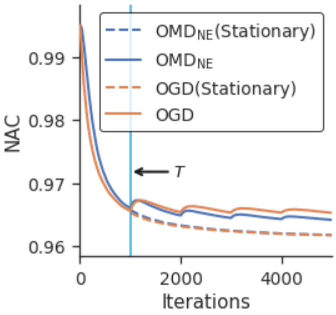
Remark

$\mathrm{OMD}_{\mathrm{NE}}$ outperforms OGD in diverse regimes and small cache sizes, while OGD outperforms for large cache sizes and concentrated requests.

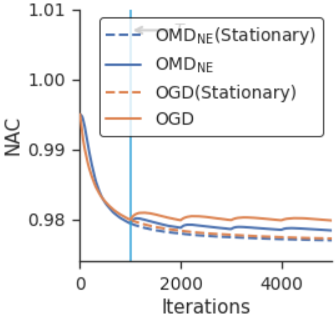
Robustness to Transient Requests



(a) $\alpha = 0.1$



(b) $\alpha = 0.3$

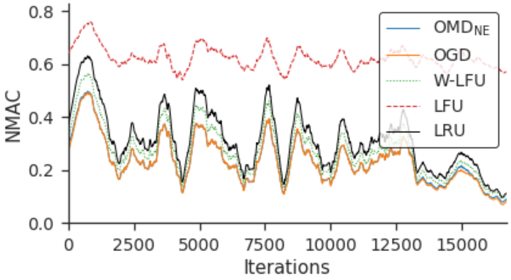


(c) $\alpha = 0.4$

Remark

We observe that $\mathrm{OMD}_{\mathrm{NE}}$ is consistently more robust to partial popularity changes than OGD.

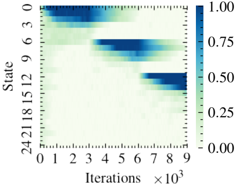
Akamai Trace



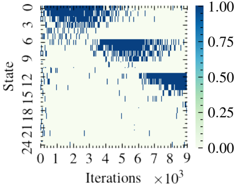
Remark

NMAC of the different caching policies evaluated on the *Akamai Trace*. $\mathrm{OMD}_{\mathrm{NE}}$ and OGD provide consistently the best performance compared to W-LFU, LRU and LFU. OGD performs slightly better than OMD in some parts of the trace.

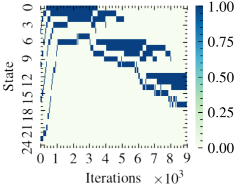
Online Randomized Rounding



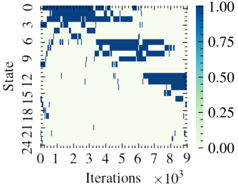
(a) Fractional cache states



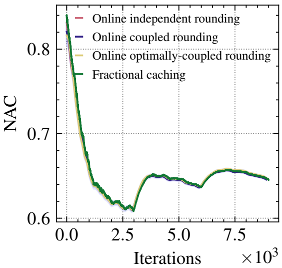
(b) Online independent rounding



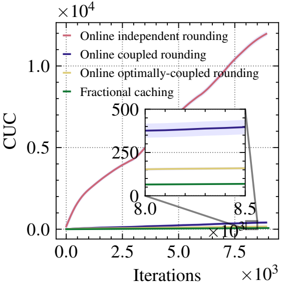
(c) Online coupled rounding



(d) Online optimally-coupled rounding



(a) Normalized average cost



(b) Cumulative update cost

Thank you for your attention.

References I

- Rajarshi Bhattacharjee, Subhankar Banerjee, and Abhishek Sinha, *Fundamental Limits on the Regret of Online Network-Caching*, Proceedings of the ACM on Measurement and Analysis of Computing Systems **4** (2020), no. 2.
- Sébastien Bubeck, *Convex Optimization: Algorithms and Complexity*, Foundations and Trends in Machine Learning **8** (2015), no. 3–4, 231–357.
- William G Madow and Lillian H Madow, *On the Theory of Systematic Sampling*, Ann. Math. Statist. **15** (1944), no. 4, 1–24.
- Samrat Mukhopadhyay and Abhishek Sinha, *Online Caching with Optimal Switching Regret*, 2021 IEEE International Symposium on Information Theory (ISIT), 2021, pp. 1546–1551.

References II

G. S. Paschos, A. Destounis, L. Vigneri, and G. Iosifidis, *Learning to Cache With No Regrets*, IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, 2019, pp. 235–243.

Georgios S Paschos, Apostolos Destounis, Luigi Vigneri, and George Iosifidis, *Learning to Cache With No Regrets*, IEEE INFOCOM 2019-IEEE Conference on Computer Communications, 2019, pp. 235–243.

Debjit Paria and Abhishek Sinha, *LeadCache: Regret-Optimal Caching in Networks*, Advances in Neural Information Processing Systems **34** (2021), 4435–4447.