Survey: Learning-Augmented Algorithms

Paper review and Proposal

Thang Chu - Nov 2024

Thanks to Shivam Garg and Haruto Tanaka (UofA)

Agenda

1. **Definition**
2. Paper review
3. Research proposal

1. Motivation - Beyond worst-case analysis

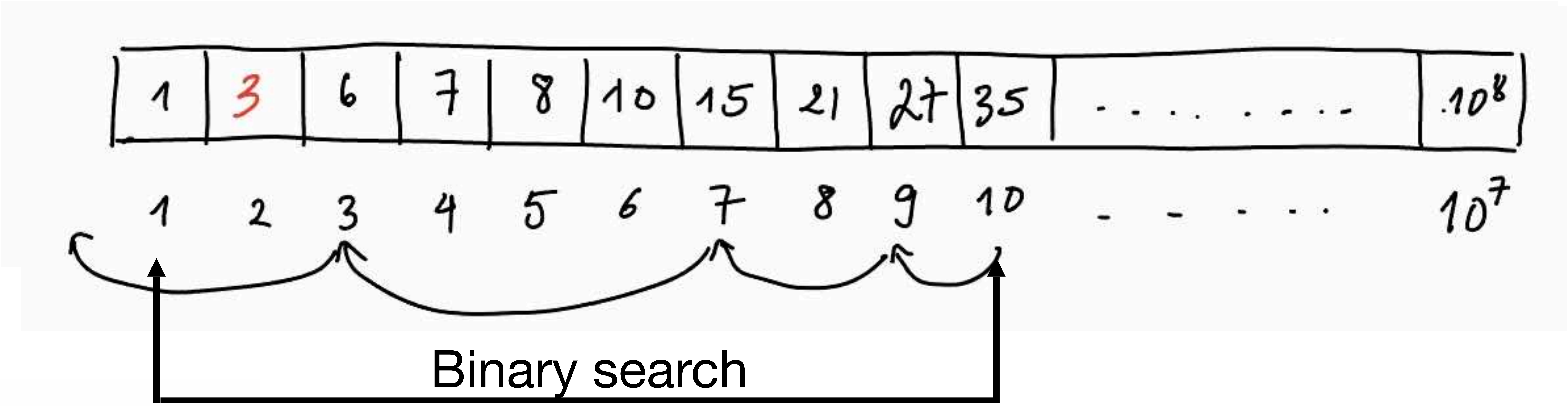
Learning-augmented algorithms

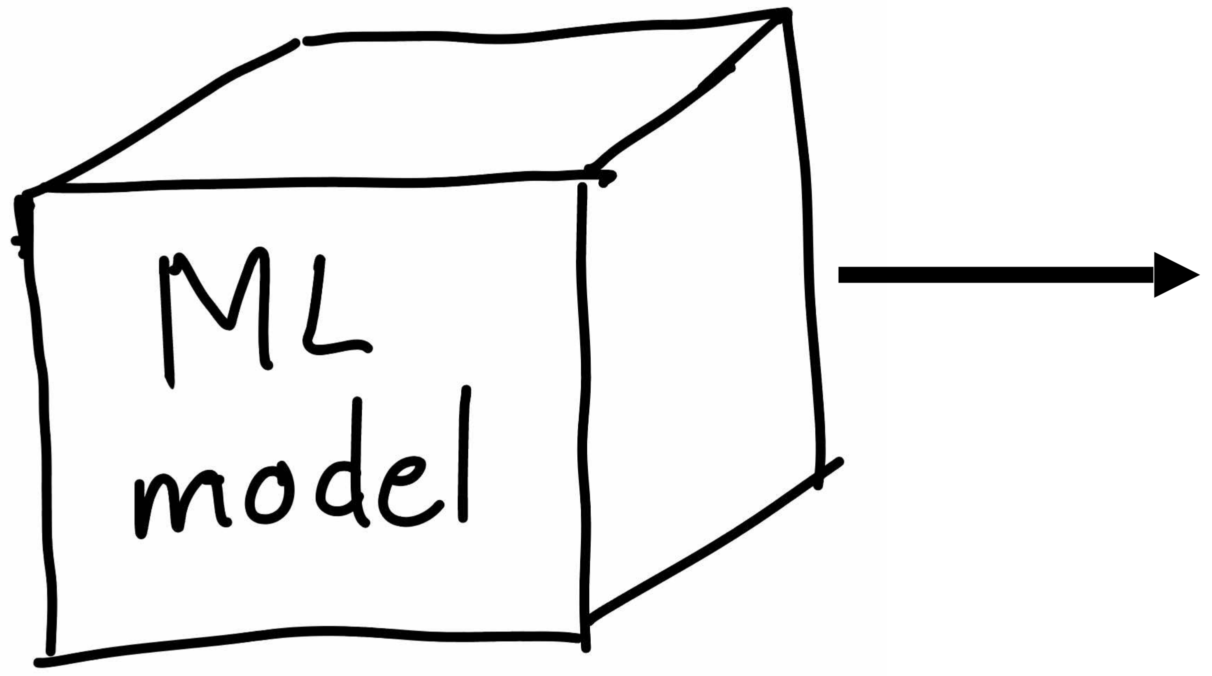
- **Definition (Informal)**: The algorithms use **predictions** obtained from the machine-learned **models** to enhance the performance of **existing algorithms**
    - **Remark**: We make **no assumptions** about the models
- **Predictions:**
    - Generally predicts certain **properties** about the **optimal solutions**
    - Untrusted
- **Robustness & Consistency:**
    - **Consistency:** If the predictions are **accurate**, then the performance is **enhanced**.
    - **Robustness:** If the predictions are **inaccurate**, then worst-case performance can be **bounded** reasonably.

1. Motivation - Beyond worst-case analysis

Learning-augmented algorithms

- **Example** (binary search): $O(\log n)$ where n is the number of elements in array.





Predicted position $= 10$
True position $= 2$

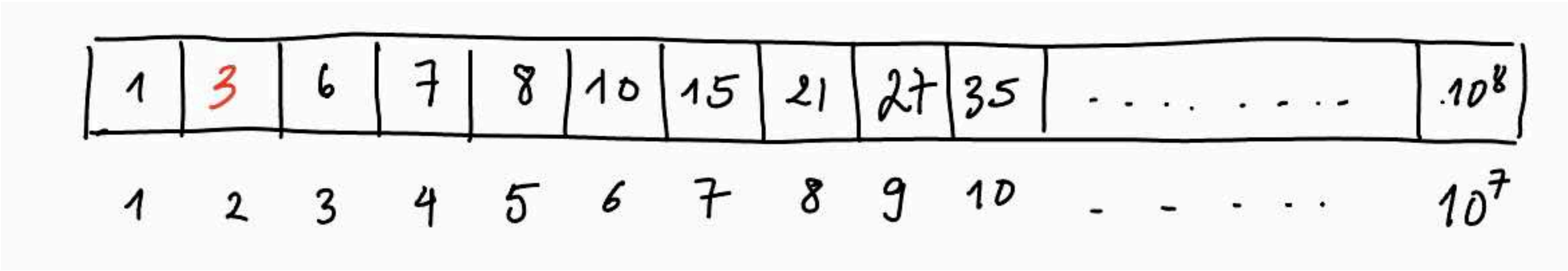
Loss $\eta = 8$

Runtime: $O(\log \eta)$ where η is the loss between prediction and true label

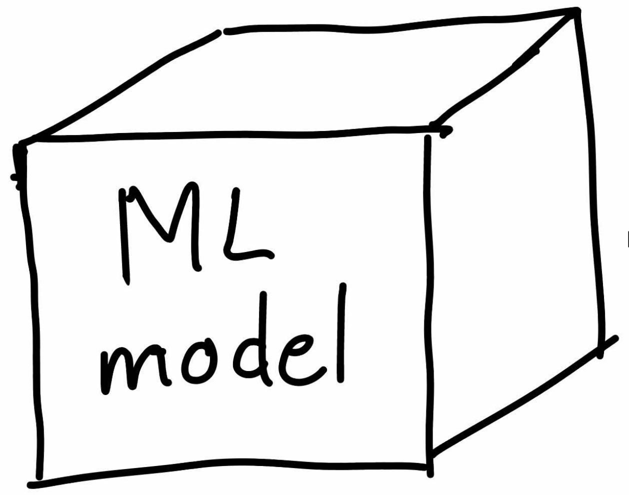
1. Motivation - Beyond worst-case analysis

Learning-augmented algorithms

- **Example** (binary search): Find the index of number 3



Given number $(x) = 3$ →



→ Predicted position $(\hat{y}) = 10$

True position $(y) = 2$

Loss $\eta = \hat{y} - y = 8$

Runtime:

- Previous: $O(\log n)$ where n is the number of elements in array.
- Now: $O(\log \eta)$ where η is the loss between prediction and true label.

Agenda

1. Definition
2. **Paper review**
3. Research proposal

2. Paper Review

The Primal-Dual method for Learning Augmented Algorithms

Etienne Bamas*
EPFL, Lausanne, Switzerland
etienne.bamas@epfl.ch

Andreas Maggiori*
EPFL, Lausanne, Switzerland
andreas.maggiori@epfl.ch

Ola Svensson*
EPFL, Lausanne, Switzerland
ola.svensson@epfl.ch

Abstract

The extension of classical online algorithms when provided with predictions is a new and active research area. In this paper, we extend the primal-dual method for online algorithms in order to incorporate predictions that advise the online algorithm about the next action to take. We use this framework to obtain novel algorithms for a variety of online covering problems. We compare our algorithms to the cost of the true and predicted offline optimal solutions and show that these algorithms outperform any online algorithm when the prediction is accurate while maintaining good guarantees when the prediction is misleading.

Competitive caching with machine learned advice

Thodoris Lykouris* Sergei Vassilvitskii†

First version: February 2018
Current version: August 2020‡

Abstract

Traditional online algorithms encapsulate decision making under uncertainty and give ways to hedge against all possible future events, while guaranteeing a nearly optimal solution, as compared to an offline optimum. On the other hand, machine learning algorithms are in the business of extrapolating patterns found in the data to predict the future, and usually come with strong guarantees on the expected generalization error.

In this work we develop a framework for augmenting online algorithms with a machine learned oracle to achieve competitive ratios that provably improve upon unconditional worst case lower bounds when the oracle has low error. Our approach treats the oracle as a complete black box, and is not dependent on its inner workings, or the exact distribution of its errors.

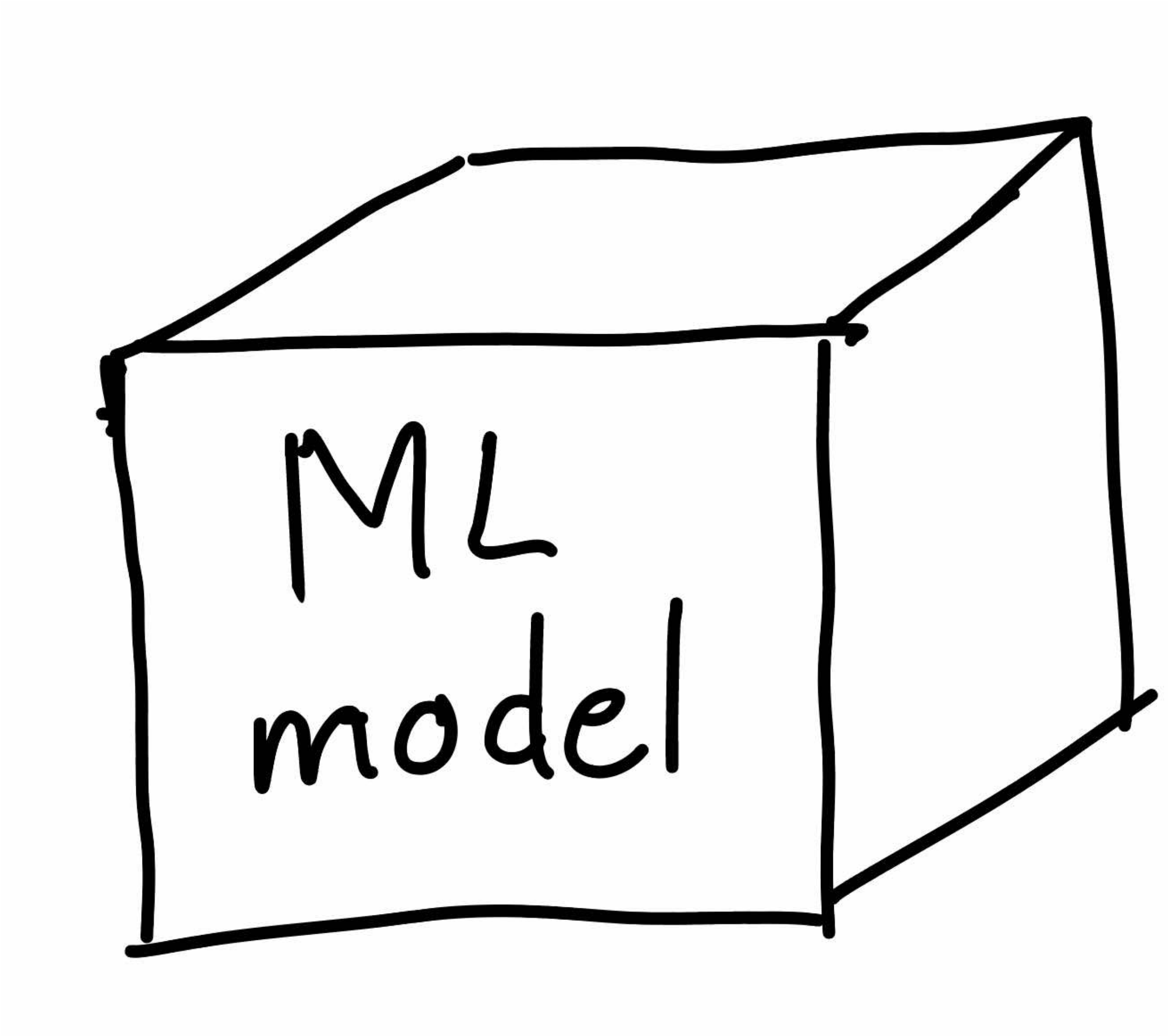
We apply this framework to the traditional caching problem—creating an eviction strategy for a cache of size k. We demonstrate that naively following the oracle's recommendations may lead to very poor performance, even when the average error is quite low. Instead we show how to modify the Marker algorithm to take into account the oracle's predictions, and prove that this combined approach achieves a competitive ratio that both decreases as the oracle's error decreases, and is always capped by $O(\log k)$, which can be achieved without any oracle input. We complement our results with an empirical evaluation of our algorithm on real world datasets, and show that it performs well empirically even when using simple off-the-shelf predictions.

2. Paper review

2.1 The primal-dual method for learning augmented algorithms - Overview

- Previously, online primal-dual algorithms were designed to solve a wide range of covering and packing problems (**ski-rental, set cover, TCP acknowledge**, etc.)

+



2. Paper review

2.1 The primal-dual method for learning augmented algorithms - Setting

Primal-Dual formulation

Primal
minimize $\sum_{S\in\mathcal{F}} w_S x_S$ subject to: $\sum_{S\in\mathcal{F}(e)} x_S \geq 1 \quad \forall e \in \mathcal{U}$ $x_S \geq 0 \quad \forall S \in \mathcal{F}$
Dual
maximize $\sum_{e\in\mathcal{U}} y_e$ subject to: $\sum_{e\in S} y_e \leq w_S \quad \forall S \in \mathcal{F}$ $y_e \geq 0 \quad \forall e \in \mathcal{U}$

U: set of elements

$\mathcal{F}$: set of subsets

$\mathcal{F}(e)$: subsets cover element e

x_s: (fractional) primal solution of subset

w_s: cost of subsets

Primal-Dual algorithm

Algorithm 1 PRIMAL DUAL METHOD FOR ONLINE WEIGHTED SET COVER [1].

Initialize: $x_S \leftarrow 0, y_e \leftarrow 0 \,\forall S, e$
for all element e that just arrived **do**
  while $\sum_{S\in\mathcal{F}(e)} x_S < 1$ **do**
    / Primal Update*
    for all $S \in \mathcal{F}(e)$ **do**
      $x_S \leftarrow x_S\left(1 + \frac{1}{w_S}\right) + \frac{1}{w_S|\mathcal{F}(e)|}$
    end for
    / Dual Update*
    $y_e \leftarrow y_e + 1$
  end while
end for

y_e: (fractional) dual solution of element

$$d = \max_{f \in \mathcal{F}} |f|$$

Competitive ratio is $O(\log d)$
Can you do better?

YES!

2. Paper review

$\lambda \in [0,1]$: trust on prediction quality

A: prediction

2.1 The primal-dual method for learning augmented algorithms - Algorithm

Algorithm 1 PRIMAL DUAL METHOD FOR ONLINE WEIGHTED SET COVER [1].

Initialize: $x_S \leftarrow 0,\ y_e \leftarrow 0\ \forall S, e$
for all element e that just arrived **do**
  while $\sum_{S\in\mathcal{F}(e)} x_S < 1$ **do**
    /* *Primal Update*
    for all $S \in \mathcal{F}(e)$ **do**
      $x_S \leftarrow x_S\left(1+\frac{1}{w_S}\right)+\frac{1}{w_S|\mathcal{F}(e)|}$
    end for
    /* *Dual Update*
    $y_e \leftarrow y_e + 1$
  end while
end for

$\Rightarrow$

Algorithm 7 PDLA FOR ONLINE WEIGHTED SET COVER.

Input: $\boxed{\lambda}, \boxed{\mathcal{A}}$
Initialize: $x_S \leftarrow 0,\ y_e \leftarrow 0\ \forall S, e$
for all element e that just arrived **do**
  while $\sum_{S\in\mathcal{F}(e)} x_S < 1$ **do**
    for all $S \in \mathcal{F}(e)$ **do**
      if $|\mathcal{F}(e)\cap\mathcal{A}| \geqslant 1$ **then**
        /* *Primal Update (more aggressive if* $\mathbb{1}\{S\in\mathcal{A}\}=1$ *)*
        $x_S \leftarrow x_S\left(1+\frac{1}{w_S}\right)+\boxed{\frac{\lambda}{w_S\cdot|\mathcal{F}(e)|}}+\boxed{\frac{(1-\lambda)\cdot\mathbb{1}\{S\in\mathcal{A}\}}{w_S\cdot|\mathcal{F}(e)\cap\mathcal{A}|}}$
      else
        /* *e is not covered by the prediction*
        $x_S \leftarrow x_S\cdot\left(1+\frac{1}{w_S}\right)+\boxed{\frac{1}{w_S\cdot|\mathcal{F}(e)|}}$
      end if
    end for
    /* *Dual Update*
    $y_e \leftarrow y_e + 1$
  end while
end for

$e = 1$ arrives $F(e) = \{\boxed{\{1,2\}}, \boxed{\{1\}}\}$

$A = \{\boxed{\{1,2\}}, \{3\}\}$

$A = \{\{4\}, \{3\}\}$

2. Paper review

2.1 The primal-dual method for learning augmented algorithms - Competitive ratio

$$c_{ALG}(I, A, \lambda) = min\{O(\frac{1}{1-\lambda})S(A, I), O(\log\frac{d}{\lambda})OPT(I)\}$$

	Trust (small λ)	Don't trust (high λ)
Correct A $S(A, I) \approx OPT(I)$	$OPT(I)$	$O(\log d)OPT(I)$
Incorrect A	$O(\log\frac{d}{\lambda})OPT(I)$	

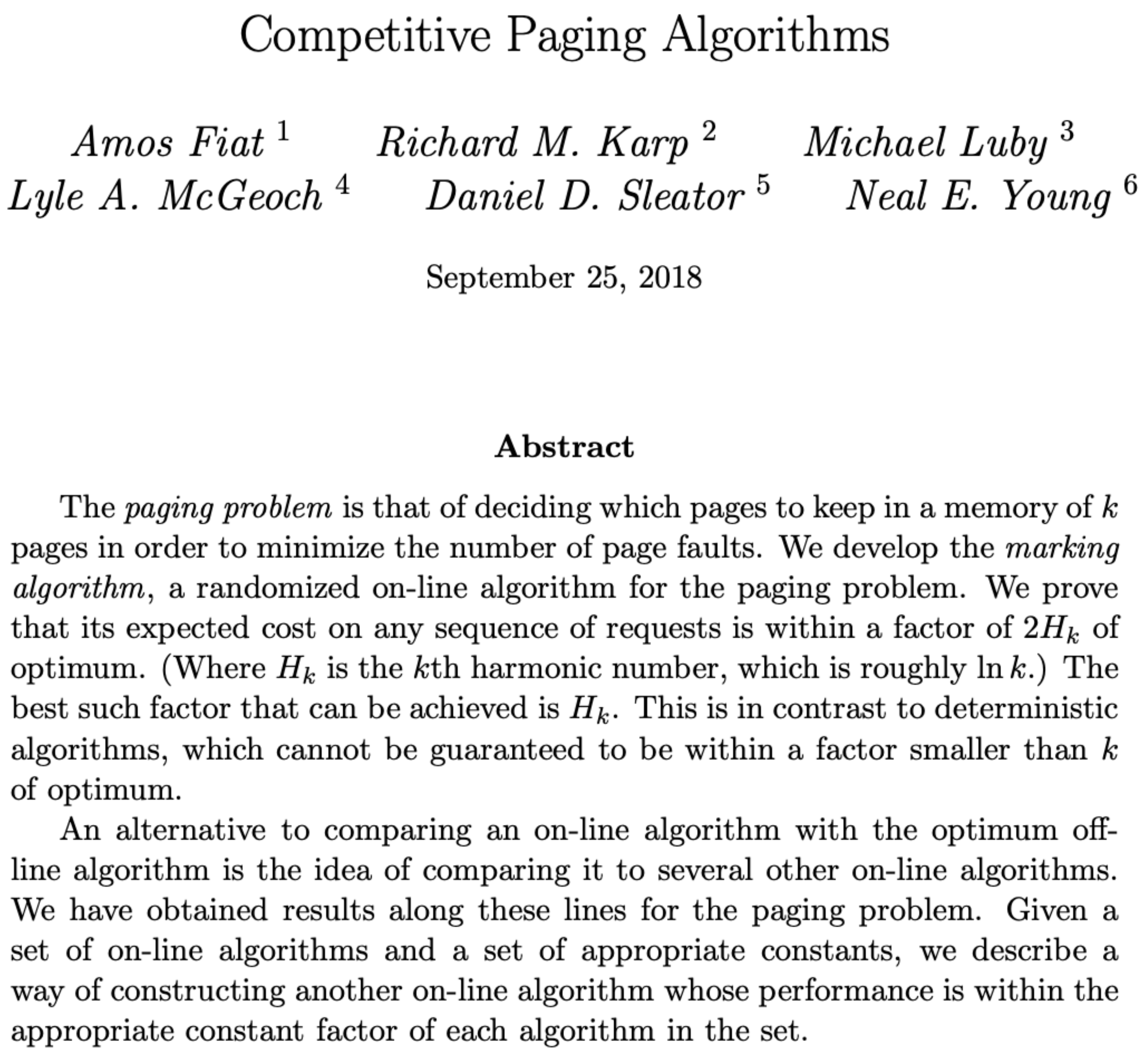
- c_{ALG}: cost of the algorithm
- $S(A, I)$: cost of following the prediction A blindly on instance I
- $OPT(I)$: offline solution on instance I

2. Paper review

2.2 Competitive caching with machine learned advice - Overview

- Previously, **Marker** algorithm is designed to solve the **online caching problem** and achieved $\Omega(\log k)$ competitive ratio, where k is the cache size.

- **Predictive Marker** extends the previous algorithm by **incorporating the prediction**



Competitive Paging Algorithms

Amos Fiat [1] *Richard M. Karp* [2] *Michael Luby* [3]
Lyle A. McGeoch [4] *Daniel D. Sleator* [5] *Neal E. Young* [6]

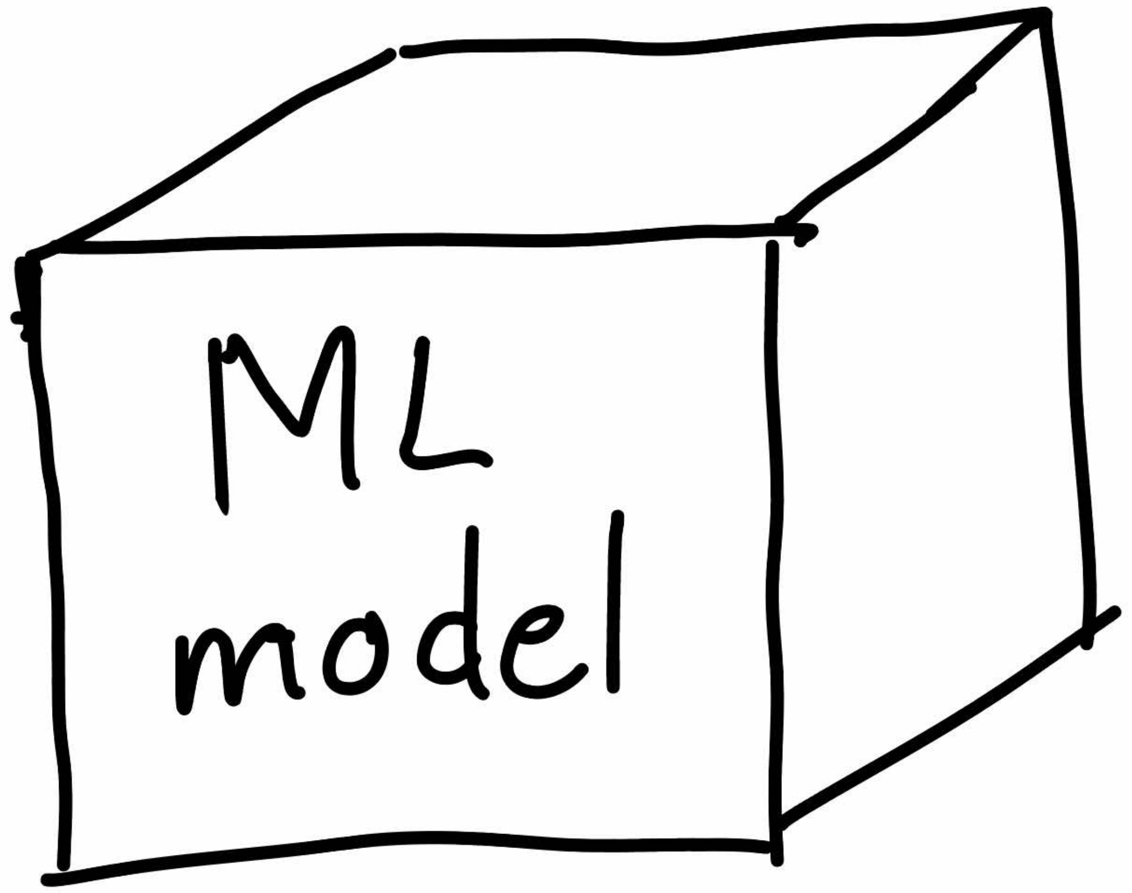
September 25, 2018

Abstract

The *paging problem* is that of deciding which pages to keep in a memory of k pages in order to minimize the number of page faults. We develop the *marking algorithm*, a randomized on-line algorithm for the paging problem. We prove that its expected cost on any sequence of requests is within a factor of $2H_k$ of optimum. (Where H_k is the kth harmonic number, which is roughly $\ln k$.) The best such factor that can be achieved is H_k. This is in contrast to deterministic algorithms, which cannot be guaranteed to be within a factor smaller than k of optimum.

An alternative to comparing an on-line algorithm with the optimum off-line algorithm is the idea of comparing it to several other on-line algorithms. We have obtained results along these lines for the paging problem. Given a set of on-line algorithms and a set of appropriate constants, we describe a way of constructing another on-line algorithm whose performance is within the appropriate constant factor of each algorithm in the set.

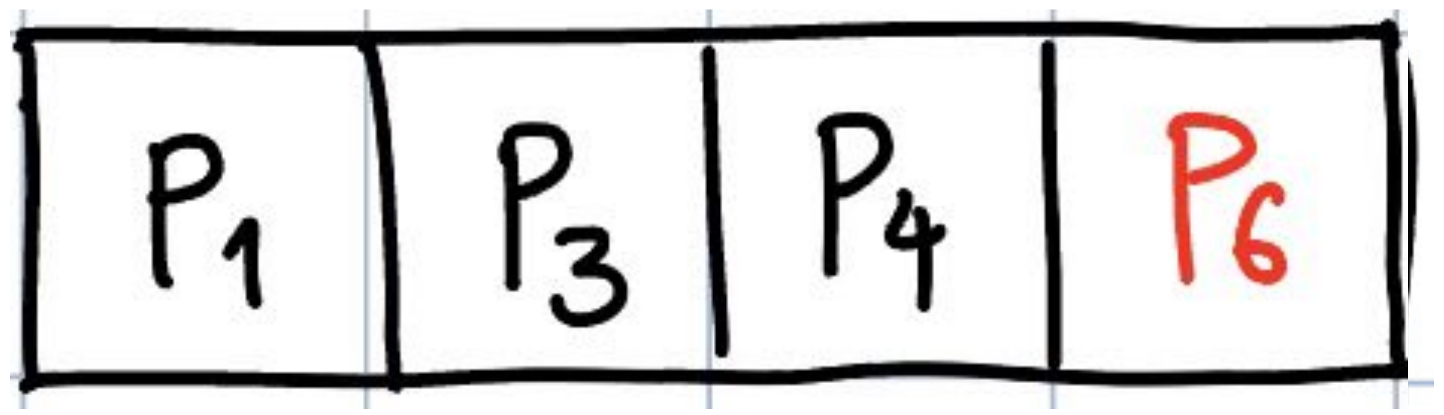
+

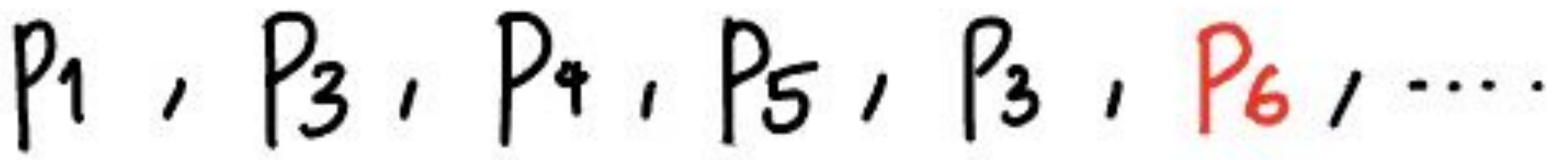


2. Paper review

2.2 Competitive caching with machine learned advice - Setting

- **Notation**:
  - **Set of pages**: $U = \{p_1, p_2, \ldots, p_n\}$
  - **Cache size**: k





- **Setting**: Unweighted, online
- **Cache hit**: The arrived page has already been in the cache
- **Cache miss**: The arrived page has not been in the cache.
- **Eviction strategy**: For each cache miss, we need to evict one page to make room for a new page.
- **Goal**: Minimize the number of **cache misses** by creating a effective eviction strategy.
- **Performance metric**: competitive ratio $c_{ALG} \leq \alpha \cdot OPT$

2. Paper review

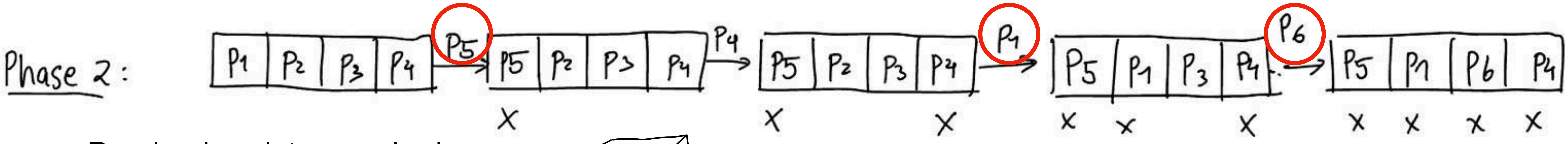
2.2 Competitive caching with machine learned advice - Algorithm

$\mathcal{U} = \{p_1, \dots, p_7\}, \; k = 4$

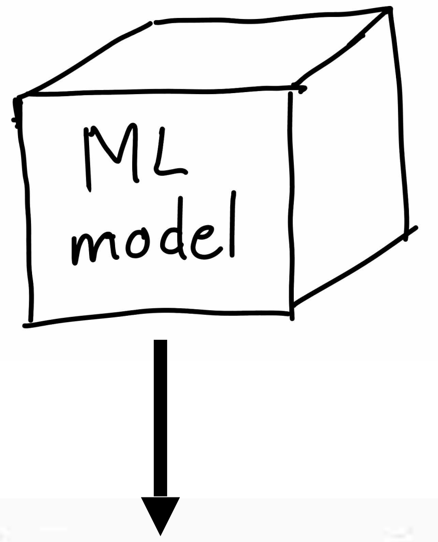
- **Marker**

Time step 1 2 3 4 5 6 7 8 9 10 11 12

$P_1, P_2, P_3, P_4, P_5, P_4, P_1, P_6, P_7, P_2, P_6, P_5$

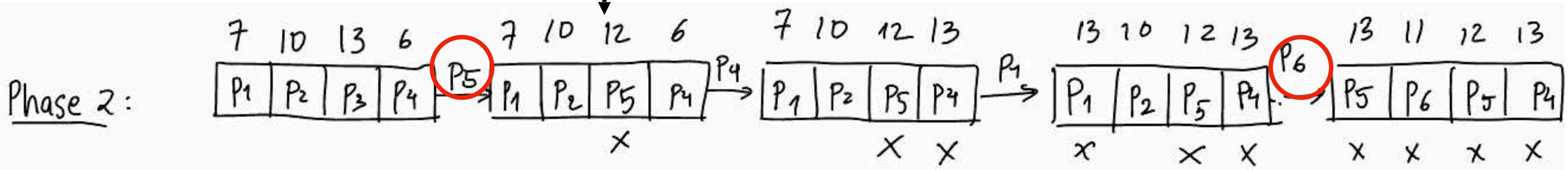


Randomly evict unmarked pages



(Perfectly) predict next time arrival

- **Predictive Marker**



Evict highest-prediction-time item

2. Paper review

2.2 Competitive caching with machine learned advice - Competitive ratio

- **Marker** - $O(\log k)$
- **Predictive Marker** - $2 + O(min\{\sqrt{\frac{\eta}{OPT}}, \log k\})$ where η is the loss function (L1, L2)
  - **Prefect predictor**: 2
  - **Nearly perfect predictor**: $2 + O(\sqrt{\frac{\eta}{OPT}})$
  - **Worst predictor**: $2 + O(\log k)$

2. Paper review

Summary

<table>
<tr><th></th><th>PDLA</th><th>Predictive Marker</th></tr>
<tr><td>Strengths</td><td colspan="2">• The competitive ratio is improved if the predictions are accurate
• The competitive ratio is slightly worse if the predictions are completely wrong, but the bound is reasonable.
• The extension is simple and the algorithm is implementable
• The experiments are provided as a proof of concept.</td></tr>
<tr><td>Drawbacks</td><td>• Continuous version of PDLA
• Optimal λ trade-off
• Restricted choice of prediction</td><td>• Complex setting (weighted)
• Optimal η loss func</td></tr>
</table>

* There are also feedback for minor issues in the written proposal

Both papers established the foundation for new research area.

However, the setting is quite simple.

Survey: Learning-augmented algorithms

3. Proposal

Survey - Previous work

1.1 Warm-up: Binary search

1.2 Online algorithms: Ski Rental

2 Counting Sketches

3 Learned Bloom Filters

4 Caching with Predictions

5 Scheduling with Predictions

Algorithms with Predictions*

Michael Mitzenmacher† Sergei Vassilvitskii‡

June 17, 2020

Abstract

We introduce algorithms that use predictions from machine learning applied to the input to circumvent worst-case analysis. We aim for algorithms that have near optimal performance when these predictions are good, but recover the prediction-less worst case behavior when the predictions have large errors.

- The survey **groups** different **algorithms** by **problems**.
  - Usually an online algorithm is designed to solve for **a specific problem**!

3. Proposal

Survey - Previous work

- However, this choice of categorization shows **several drawbacks**
  - The previous paper only **covers small** number of problems. Even if it covers **all problems as well as their variations**, it is neither possible nor interesting.
  - Even if it can cover all of the problems, this approach typically compares **within the same category (problem)**. However, it does not compares **across different categories** (difference/similarity/strength/drawback).
  - Some papers do **not aim to solve for a particular problem**, rather solve for more **generalized setting** (predictors, optimal robustness-consistency, etc.)

3. Proposal

Survey - New approach

- Inspired from paper reviews, we plan to categorize the papers based on 5 dimensions

1. Prediction: limited versus full information

Reduce the number of queries from one black-box model (query once every x rounds)

2. Predictor: one versus multiple

Advice from multiple models instead of one model

3. Predictor: black-box versus grey-box

More information on how the model is trained, the model can also be more accurate

4. Performance measure: λ**-** versus η-competitive ratio

Relationship between them, optimal value of λ and choice of loss function

5. Algorithm design: systematic versus ad-hoc

Primal-dual formulation versus the rest

3. Proposal

Survey - Summary

- My idea is based on
  - General setting: Bridging the gap between **theory and practice**.
  - General framework: More **unified and principled** (primal-dual approach).
- It is **difficult** to have a **perfect separation** strategy*

* I will only cover more papers for each problem in the final paper

Reference

1. Keerti Anand, Rong Ge, Amit Kumar, and Debmalya Panigrahi. 2022a. Online algorithms with multiple predictions.
2. Keerti Anand, Rong Ge, and Debmalya Panigrahi. 2022b. Customizing ml predictions for online algorithms.
3. Antonios Antoniadis, Joan Boyar, Marek Eli´aˇs, Lene M. Favrholdt, Ruben Hoeksma, Kim S. Larsen, Adam Polak, and Bertrand Simon. 2022. Paging with succinct predictions.
4. Niv Buchbinder, Joseph Seffi Naor, et al. 2009. The design of competitive online algorithms via a primal–dual approach. Foundations and Trends® in Theoretical Computer Science, 3(2–3):93–263.
5. Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, Ali Vakilian, and Nikos Zarifis. 2021. Learning online algorithms with distributional advice. In Marina Meila and Tong Zhang, editors, Proceedings of the 38th International Conference on Machine Learning, volume 139 of Proceedings of Machine Learning Research, pages 2687–2696. PMLR, 18–24 Jul.
6. Marina Drygala, Sai Ganesh Nagarajan, and Ola Svensson. 2023. Online algorithms with costly predictions. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, Proceedings of The 26th International Conference on Artificial Intelligence and Statistics, volume 206 of Proceedings of Machine Learning Research, pages 8078–8101. PMLR, 25–27 Apr.
7. Paul D¨utting, Silvio Lattanzi, Renato Paes Leme, and Sergei Vassilvitskii. 2020. Secretaries with advice.
8. Amos Fiat, Richard M Karp, Michael Luby, Lyle A McGeoch, Daniel D Sleator, and Neal E Young. 1991. Competitive paging algorithms. Journal of Algorithms, 12(4):685–699.
9. Sreenivas Gollapudi and Debmalya Panigrahi. 2019. Online algorithms for rent-or-buy with expert advice. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research, pages 2319–2327. PMLR, 09–15 Jun.
10. Elena Grigorescu, Young-San Lin, Sandeep Silwal, Maoyuan Song, and Samson Zhou. 2022. Learning-augmented algorithms for online linear and semidefinite programming.
11. Elena Grigorescu, Young-San Lin, and Maoyuan Song. 2024. A simple learning-augmented algorithm for online packing with concave objectives.
12. Thodoris Lykouris and Sergei Vassilvitskii. 2020. Competitive caching with machine learned advice.
13. Michael Mitzenmacher and Sergei Vassilvitskii. 2020. Algorithms with predictions

Thank you

Appendix

3. Proposal
Survey - New approach

- **Prediction:** limited versus full

Paging with Succinct Predictions

Antonios Antoniadis [*1] Joan Boyar [*2] Marek Eliáš [*3] Lene M. Favrholdt [*2] Ruben Hoeksma [*1]
Kim S. Larsen [*2] Adam Polak [*4] Bertrand Simon [*5]

Online Algorithms with Costly Predictions

Marina Drygala, EPFL
Sai Ganesh Nagarajan, EPFL
Ola Svensson, EPFL

Limit the information of prediction

Limit the number of queries

Open problems:

- Extending the ideas to solve for a specific problem!
- Given extra statistics (mean, variance) of the problem, can we further reduce the number of predictions?

* There are more papers will be reviewed in the final paper

3. Proposal
Survey - New approach

- **Predictor:** single versus multiple

Online Algorithms with Multiple Predictions

Keerti Anand[*] Rong Ge[†] Amit Kumar[‡] Debmalya Panigrahi[§]

July 14, 2022

Online Algorithms for Rent-or-Buy with Expert Advice

Sreenivas Gollapudi[*1] Debmalya Panigrahi[*2]

Open problems:

- Multiple predictions for packing problems, convex-objective problems, etc.
- Instead of choosing among predictors, can we combine multiple predictions (Boosting algorithm)?

* There are more papers will be reviewed in the final paper

3. Proposal
Survey - New approach

- **Predictor:** black-box versus grey-box

Customizing ML Predictions for Online Algorithms

Keerti Anand* Rong Ge* Debmalya Panigrahi*

Learning Online Algorithms with Distributional Advice

Ilias Diakonikolas [*1] Vasilis Kontonis [*1] Christos Tzamos [*1] Ali Vakilian [*2] Nikos Zarifis [*1]

Open problems:

Are there any instances such that it is not learnable? If yes, how to overcome the issue?

* There are more papers will be reviewed in the final paper

3. Proposal

Survey - New approach

- **Performance measure:** λ- versus η-competitive ratio

Overcoming Brittleness in Pareto-Optimal Learning-Augmented Algorithms

Spyros Angelopoulos
Sorbonne Université, CNRS, LIP6, Paris, France

Christoph Dürr
Sorbonne Université, CNRS, LIP6, Paris, France

Alex Elenter
Sorbonne Université, CNRS, LIP6, Paris, France

Yanni Lefki
Ecole Polytechnique, Palaiseau, France

Optimal robustness-consistency tradeoffs for learning-augmented metrical task systems

Nicolas Christianson Junxuan Shen Adam Wierman
California Institute of Technology

Open problems:

- Are there relationships between these two types of competitive ratio? In other words, can we convert from one type to the other?
- Currently, there are results for showing the optimal trade-off for ski-rental and metrical task system. What about other problems?

* There are more papers will be reviewed in the final paper

3. Proposal

Survey - New approach

- **Algorithm design:** systematic versus ad-hoc

Learning-Augmented Algorithms for Online Linear and Semidefinite Programming

Elena Grigorescu[*]
Purdue University, USA
elena-g@purdue.edu

Young-San Lin[*†]
University of Melbourne, Australia
nilnamuh@gmail.com

Sandeep Silwal[‡]
MIT, USA
silwal@mit.edu

Maoyuan Song[*§]
Purdue University, USA
song683@purdue.edu

Samson Zhou[¶]
UC Berkeley and Rice University, USA
samsonzhou@gmail.com

A Simple Learning-Augmented Algorithm for Online Packing with Concave Objectives

Elena Grigorescu[*]
Purdue University
elena-g@purdue.edu

Young-San Lin
Melbourne Business School
y.lin@mbs.edu

Maoyuan Song[†]
Purdue University
maoyuanrs@gmail.com

Open problems:

- How the framework will handle multiple predictors, limited predictions, incorporate advice from grey-box models are unknown.
- Efficient implementation for continuous version of PD to compare with existing ad-hoc solutions.

* There are more papers will be reviewed in the final paper