SORBONNE UNIVERSITÉ

PROGRAMME INSTITUTS ET INITIATIVES

Appel à projet – campagne 2021

Proposition de projet de recherche doctoral (PRD)

Intitulé du projet de recherche doctoral (PRD): Online algorithms with predictions

Directeur.rice de thèse porteur.euse du projet (titulaire d'une HDR) :

NOM : **ANGELOPOULOS** Prénom : **Spyros**
Titre : CR CNRS
e-mail :spyros.angelopoulos@lip6.fr
Adresse professionnelle : 4 place Jussieu, 26-00, 421.
(site, adresse, bât., bureau)

Unité de Recherche :
Intitulé : Laboratoire d'Informatique de Paris-LIP6
Code *(ex. UMR xxxx)* : UMR 7606

École Doctorale de rattachement de l'équipe (future école doctorale du.de la doctorant.e) : EDITE

Doctorant.e.s actuellement encadré.e.s par la.e directeur.rice de thèse (préciser le nombre de doctorant.e.s, leur année de 1e inscription et la quotité d'encadrement) :

0.

--

Co-encadrant.e :

NOM : Prénom :
Titre : ou HDR ☐
e-mail :

Unité de Recherche :
Intitulé :
Code *(ex. UMR xxxx)* :

École Doctorale de rattachement :

Ou si ED non Alliance SU :

Doctorant.e.s actuellement encadré.e.s par la.e co-directeur.rice de thèse (préciser le nombre de doctorant.e.s, leur année de 1e inscription et la quotité d'encadrement) :

Co-encadrant.e :

NOM : Prénom :
Titre : ou HDR ☐
e-mail :

Unité de Recherche :
Intitulé :
Code *(ex. UMR xxxx)* :

École Doctorale de rattachement : Ou si ED non Alliance SU :

Doctorant.e.s actuellement encadré.e.s par la.e co-directeur.rice de thèse (préciser le nombre de doctorant.e.s, leur année de 1e inscription et la quotité d'encadrement) :

Cotutelle internationale : Non

Selon vous, ce projet est-il susceptible d'intéresser une autre Initiative ou un autre Institut ?
Non Oui, précisez

This proposal is only addressed to EDITE and SCAI.

Description of the project

Background

In online computation we are interested in algorithms that do not know the entire input in advance, but rather process it in the form of a sequence of input items. For each incoming input item, the algorithm must make an irrevocable decision concerning it, in general without any assumptions on the future items. Problems of this type arise very often in real-world applications, such as routing calls within a communications network, managing the cache content of web servers, or deciding how to auction goods among a number of participating agents. For this reason, online computation has developed, over the past few decades, into a prolific field with significant impact in many areas of Computer Science, and in particular in Artificial Intelligence (AI) and Machine Learning (ML). See for example the books [8,14], and a new book in preparation [9], which is a testament to the ever-increasing interest in the field.

How does one analyze and compare the performance of online algorithms? The standard method of evaluation is by means of the *competitive analysis* framework: here, we compare the worst-case performance of the online algorithm to the corresponding performance of an ideal, offline algorithm, namely an algorithm that has advanced knowledge of the entire input. Thus, a good online algorithm guarantees a small gap between the online and the ideal (offline) performance; this measure is called the *competitive ratio* of the algorithm.

General presentation of the topic

The standard model of online computation assumes that the online algorithm has no knowledge whatsoever about the input. This can be problematic since, in the real world, the algorithm may indeed have such (limited) knowledge. For example, in the context of routing calls in a network, we may have information on the participants that are likely to call each other, or the periods of the day during which the majority of the calls are placed. How can then one adapt the main online paradigm to account for situations such as the above? There have been two main approaches:

1. *The advice-complexity framework*. Here, we assume that the online algorithm has access to an oracle that provides the additional information, termed *advice*. More precisely, in the advice setting, the online algorithm receives some bits that encode information concerning the sequence of input items. This additional information can help improve the competitive ratio of the algorithm. It is important to note that this advice is assumed to be i) error-free; and ii) provided by an omnipotent, and potentially unrealistic oracle. This is an area that has been studied for over a decade within theoretical computer science (TCS), see e.g the survey [10] and the book [14].

2. *The ML-prediction framework.* This is a much more recent approach that is rooted in machine learning. Here, the additional information is in the form of a natural prediction concerning the input; for example, the point in time at which certain item will appear. It is important to note that the prediction is assumed to be i) potentially noisy, that is, there is some *error* associated with the prediction; ii) provided by a "reasonable" and "natural" oracle, i.e., an oracle that provides some information on the input that can be easily learnable.

It should be clear that the two approaches are almost complementary, and one would dare say, in conflict. The advice-complexity framework is almost entirely a theoretical abstraction with very few applications, but allows for very rigorous analysis using information-theoretic approaches. In contrast, the more recent ML-prediction framework is motivated by real applications, but has not yet fully developed the precise theory about the limitations of predictions.

Objectives of this thesis proposal

Our main objective in this project can be summarized as follows:

Can we narrow the big gap between the advice-complexity model and the ML-prediction model? Can we obtain better online algorithms with predictions by combining, and extending techniques from both approaches?

Broadly speaking, our goal is to combine the positive aspects, characteristics, and techniques from these two worlds into a refined model that allows for imperfect predictions. Below we present some specific directions we will pursue with the PhD student.

1. *The price of erroneous advice.* In our work [1] and [2] we studied advice in a setting in which it is either always correct, or always adversarially generated (informally: either provided by a friend or by an enemy). For this reason, some of the performance tradeoffs are only of theoretical significance. A more realistic model would allow for *noisy* advice. This can be interpreted as using a number of

noisy *binary queries* as predictions. One example of an application in AI is our recent work [3], which addresses a well-known resource allocation problem from precisely this perspective. There is a plethora of problems to be studied in this setting. Noisy binary queries have been used in ML problems such as clustering [18], but their application to online problems has been limited so far.

2. *Compact predictions*. The ML-prediction framework so far has not placed any restrictions on the size of the advice, unlike the advice-complexity framework. However, there are situations in which some restriction is necessary in practice (e.g., in caching, we cannot afford too much time in gathering information). Thus, we would like to study *time-sensitive* problems and applications (e.g., caching and its generalizations) in a setting in which there is a prediction of small size.

3. *Combining advice from several experts*. Here, we are interested in using potentially erroneous predictions that may originate from several sources (i.e., experts), instead of a single one. This was only recently explored in [12] for the ski rental problem, and the same question can be asked for many other online problems.

4. Better tools for analysis of algorithms with predictions. The advice complexity field has introduced some powerful theoretical tools (e.g., the string guessing game [11]), which however are tailored to a setting in which the advice is error-free. We would like to extend them to a more general setting in which the advice is erroneous.

Broadly speaking, there is great potential for new results and techniques. This is a rapidly growing area that will continue to attract significant interest for at least the next 5 years (see next section). Any significant online problem that was previously studied in the standard online model is a good candidate for study in this new setting.

The overall approach involves two steps. The first is the theoretical analysis of the online algorithms and their limitations, from the point of view of both positive and negative (i.e., impossibility) results. The second step is the experimental validation of these algorithms. The latter will be a requirement for any submissions to leading AI/ML conferences.

State of the art and global positioning of this project

This line of research was introduced in [16,19] and combines aspects of theoretical computer science and machine learning. These works proved very influential, are already heavily cited, and spearheaded an impressive activity. Further applications include secretary and matching problems [6], page migration [13], ski rental and rent-or-buy problems [7,12], metrical task systems [5], scheduling [15] and many other problems. This list is not inclusive and only represents a sample of recent work. Most of this work is in the context of ML predictions; the theoretical advice framework has relatively lagged.

The scope of this project lies within a broader objective related to the analysis of algorithms beyond the worst-case. Here,the emphasis is on assessing algorithms not only with respect to the pessimistic worst-case, but using refined measures that better reflect the typical algorithmic performance. This is an emerging trend, that will become even more attractive with the publication of a book by Roughgarder on this topic (see, in particular, the chapter by Mitzenmacher and Vassilvitskii on algorithms with predictions in this book [17]).

This line of research is already receiving considerable exposure, with presentations in venues of high visibility (e.g., the talk by Mitzenmacher at the Simons Institute and the talk by Vassilvitskii at Highlights of Algorithms 2019). Related topics have been presented at workshops such as the TTIC Summer Workshop on Learning-Based Algorithms, the STOC 2020 Workshop on Algorithms with Predictions, the 2020 Workshop on Online Algorithms with Advice and Related Models and courses such as the MIT course on Learning-Augmented Algorithms. For recent contributions on the interface between competitive analysis and online learning, refer to this talk by James Lee.

Qualifications and related publications by the supervisor

The supervisor of this project has several publications in leading AI conferences and journals (AAAI, IJCAI, JAIR) and TCS conferences and journals (JACM, SODA, ITCS, ESA, STACS). Below is a list of recent publications directly related to this proposal.

* Contract scheduling with predictions (with S. Kamali). In Proceedings of the 35th AAAI Conference, 2021. Link to full paper.
* Online search with a hint. In Proceedings of the 12th International Conference on Innovations in Theoretical Computer Science (ITCS) 2021. Link to paper.
* Online computation with untrusted advice (with C. Dürr, S. Jin, S. Kamali, and M. Renault). In Proceedings of the 11th International Conference on Innovations in Theoretical Computer Science (ITCS) 2020. Link to paper.
* Online bin packing with predictions (with S. Kamali and K. Shadkami). Submitted to a ML conference, 2021. Link to full paper.

Relevance

This proposal is on a rapidly-growing area within ML and more broadly within AI, as supported by the numerous publications over the last two years (see sections above). As such, it should be clearly of interest to SCAI.

Student profile

The successful applicant must have a strong background in the design and analysis of algorithms, and the desire to apply mathematical approaches in computation. Solid programming skills, and willingness to work with real data are also essential.

References

[1] Spyros Angelopoulos. Online search with a hint. Proceedings of the 12th Innovations in Theoretical Computer Science Conference (ITCS): 51:1-51:16, 2021

[2] Spyros Angelopoulos, Christoph Dürr, Shendan Jin, Shahin Kamali, Marc P. Renault. Online Computation with Untrusted Advice. Proceedings of the 11th Innovations in Theoretical Computer Science Conference (ITCS): 52:1-52:15, 2020

[3] Spyros Angelopoulos and Shahin Kamali. Contract scheduling with predictions. Proceedings of the 35th AAAI Conference on Artificial Intelligence, 2021. Full paper at https://arxiv.org/pdf/2102.03311.pdf

[4] Spyros Angelopoulos, Shahin Kamali and Kimia Shadkami. Online bin packing with predictions. Available at https://arxiv.org/pdf/2102.03311.pdf

[5] Antonios Antoniadis, Christian Coester, Marek Eliás, Adam Polak, and Bertrand Simon. Online metric algorithms with untrusted predictions. In Proceedings of the 37th International Conference on Machine Learning (ICML), pages 345–355, 2020.

[6] Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. Secretary and online matching problems with machine learned advice. In Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS), 2020.

[7] Soumya Banerjee. Improving online rent-or-buy algorithms with sequential decision making and ML predictions. In Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS), 2020.

[8] Allan Borodin and Ran El-Yaniv. Online Computation and Competitive Analysis. Cambridge University Press, 1998

[9] Allan Borodin and Denis Pankratov. Online computation. In preparation. Available at http://www.cs.toronto.edu/~bor/2420s19/papers/draft-ch1-8.pdf

[10] Joan Boyar, Lene M. Favrholdt, Christian Kudahl, Kim S. Larsen, and Jesper W.Mikkelsen. Online algorithms with advice: A survey. SIGACT News,47(3):93–129, 2016

[11] Joan Boyar, Lene M. Favrholdt, Christian Kudahl, Jesper W. Mikkelsen: Advice Complexity for a Class of Online Problems. Proceedings of the 32nd International Symposium on Theoretical Aspects of Computer Science (STACS) 116-129, 2015

[12] Sreenivas Gollapudi and Debmalya Panigrahi. Online algorithms for rent-or-buy with expert advice. In Proceedings of the 36th International Conference on Machine Learning (ICML), 2319–2327, 2019

[13] Piotr Indyk,, Frederik Mallmann-Trenn, Slobodan Mitrović, and Ronitt Rubinfeld. Online Page Migration with ML Advice. *arXiv preprint arXiv:2006.05028*, 2020.

[14] Dennis Komm. Introduction to Online Computation. Springer, 2016

[15] Silvio Lattanzi, Thomas Lavastida, Benjamin Moseley, and Sergei Vassilvitskii. Online scheduling via learned weights. In Proceedings of the 14th ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 1859–1877, 2020.

[16] Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. In Proceedings of the 35th International Conference on Machine Learning (ICML) 3302–3311, 2018

[17] Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. In Tim Roughgarden, editor, Beyond the Worst-Case Analysis of Algorithms, pages 646–662. Cambridge University Press, 2020.

[18] Arya Mazumdar, Barna Saha. Clustering with Noisy Queries. Advances in Neural Information Processing Systems (NeurIPS) 2017: 5788-5799

[19] Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In Advances in Neural Information Processing Systems (NeurIPS) ,volume 31, 9661–9670, 2018.

Merci d'enregistrer votre fichier au <u>format PDF</u> et de le nommer :
«ACRONYME de l'initiative/institut – AAP 2021 – NOM Porteur.euse Projet »

Fichier envoyer simultanément par e-mail à l'ED de rattachement et au programme :
cd_instituts_et_initiatives@listes.upmc.fr avant le 20 février.