Algorithmic theory of new data models

Théorie algorithmique de nouveaux modèles de données

I. Context, positioning and objectives

Algorithms are everywhere, in every piece of technology, as well as in many decision-making processes. The traditional computation model, with its clean sequence of input-algorithm-output, no longer fits most applications, as we are faced nowadays with new challenges, e.g., when the input is massive, scattered or disorganized, plagued with errors, or constantly evolving. Yet, algorithms are still required "to work", and to produce useful results. Theory has fallen behind practice: our understanding of how to tackle the above mentioned challenges is lagging behind, and it is urgent to lay down sounder theoretical foundations for the types of algorithms that are now "alive" and to better understand how to render them resilient. As theoreticians, we address such challenges with an axiomatic perspective grounded in theoretical models and rigorous theorems. Thus, in order to develop the theory of algorithms for massive, erroneous and dynamic data, we plan on focusing on three settings: (1) Massive data (e.g., the data is too large to fit into memory); (2) Noisy data (e.g., the data cannot be reliably accessed and hence observed with error, or noise); (3) Dynamic data (e.g., the data evolves constantly).

Tackling all of these challenges can benefit from common algorithmic and mathematical tools, such as randomization, communication complexity and information theory, or dynamic data structures. We propose to use these tools to yield novel solutions, either by addressing the above settings separately, or by developing tools that are useful for many of them, or by borrowing ideas and techniques from one setting to the other.

How will we reach our goals? We intend to combine the diverse skills and expertise of the participating researchers and sites: randomization, streaming algorithms, approximation algorithms, testing, online algorithms, communication complexity, query complexity and more. As a consequence, we expect to give important contributions to the algorithmic infrastructure necessary for treating massive, noisy, or dynamic data, as well as to the study of the complexity and the limitations of such settings.

Impact. The expected impact of our project is in substantially improving the theoretical and algorithmic foundations for efficient solutions for today's computing challenges, where data is given and accessed in new ways. We expect to work with, and give results for, more sophisticated models that better reflect the behaviour of the input in today's settings. We expect to both develop new algorithmic techniques to handle these models, as well as to analyze the theoretical limitations that those models impose, and to define new models that better reflect current settings.

Our project thus promises to advance the algorithmic and theoretical infrastructure necessary for the design of better and more efficient solutions in today's scenarios, as well as to improve our understanding of the inherent limitations of our technology today. In the mid and long term, we also believe that the results of this project will contribute to more efficient solutions in practice, e.g., by contributing to a more accurate analysis of modern systems handling noisy, dynamic or massive data.

1. Massive data

To model the limitations on access to the input imposed by massive data, we will use two main data access models: local query and streaming. To capture the diversity and complexity of current technology, we will also build on those models in order to develop novel and more complex ones. We further plan to borrow, from one model to the other, algorithmic methodologies that were designed one specific model, as well as to develop new techniques useful in a wide range of models.

Local query models and local computation algorithms. Instead of computing the entire solution, local computation models, in response to a query, return only a small part of a solution, while querying themselves only a small part of the input. The challenge is to think globally while computing locally: all parts returned must be consistent, i.e., parts of one common global optimal (or approximately optimal) solution. We plan to study graph theoretic problems, as well as the generation of random graphs, in the local computation model, as we did in [ELMR17]. The local computation model is closely related to the research field of query complexity, for which we recently answered a 30-year old open problem by showing a quadratic separation between deterministic and randomized query complexity [ABB+17]. In this field we plan to continue by studying their relation to certificate complexity.

Streaming algorithms. The streaming model of computation handles computation on data that does not fit into a computer memory by means of scanning the input sequentially and then outputting the optimal result (or an approximation thereof). We intend to continue our work on streaming algorithms for graph theoretic and other optimization problems. We also intend to study the setting of distributed streaming algorithms: we plan to continue to work on our novel approach where we analyze simultaneously the communication and the space in such setting [BKM17]. Communication complexity, an area in which we have much experience (e.g., [KLLRX15,KRF16]) is a very important tool for the study of streaming algorithms, and we intend to continue to work in this area both in the two-party and in the multi-party settings. In fact, while many lower bounds for streaming algorithms were given via lower bounds on two-party communication complexity, many times the two-party setting is not sufficient in order to prove meaningful lower bounds for streaming problems (see, e.g., [MMS14,FM13)]. Multi-party communication complexity is also needed for the study of distributed streaming algorithms.

2. Noisy data

Most realistic data is noisy. When an opinion is taken by a poll, there is always a degree of unreliability. When data is obtained from taking a physical measurement, measurement errors creep in. When data is looked up in a database, some of the information may not be updated and may have become obsolete or erroneous. Computations that aim to be reliable must take the existence of errors into account. We intend to study three error models, describing them below from the most restrictive one to the most general one.

Probabilistic noise. When data is gathered by batch processing, consulting experts, or crowdsourcing, requesting the same piece of information several times (from different members of the crowd, say) yields answers that are *independently* noisy with some probability. How can we exploit redundancy wisely so as to efficiently recover the ground truth or at least obtain reliable results in spite of such noisy information? We particularly wish to focus on a fine-grained analysis distinguishing between "easier" and "harder" instances, in the instance-optimal setting. See [MS07,MMV17] for our initial forays into this area.

Intervals of uncertainty. Numerical data such as edge lengths have measurement errors, expressed by giving intervals of uncertainty. Often, one can choose to zoom onto critical data, and use more resources, in order to perform precise (but more costly) measurements and obtain the desired value precisely: this is called *explorable uncertainty*, and is "more general" in the sense that it does not require any assumption of independence. We wish to explore the tradeoff between the accuracy of the solution produced by the algorithm and the number of queries (precise measurements) needed for, e.g., scheduling or network problems. We report initial progress for a scheduling application in [DEMM17].

Property testing. The goal of property testing is to decide whether the input satisfies a certain property up to some margin of uncertainty, e.g., a radar giving a speeding ticket only when the car speed is well above the speed limit. The assumption is that what we see is a noisy version of the data, and there are three possible outputs: "yes, the property holds"; "no, the property definitely does not hold, even if what we are seeing is a corrupted version of the true input"; and "grey area -- not sure". Allowing for this third possibility is an elegant solution for allowing very general errors. We have studied several problems at the intersection of property testing and streaming [FMR10,FMRS16,FMS18]. We now intend to further study property testing, particularly assuming prior information, for instance, in the streaming model, if items arrive as Independent and identically distributed random variables.

3. Dynamic data

Data is many times dynamic, so algorithms and data structures must be robust with respect to changes in their input. Those changes can be seen as another potential source of error, depending how well there are tracked. We intend to study dynamic data for data models that have been recently considered in the literature: change can be pushed onto the algorithm as a stream or pulled explicitly by the algorithm.

Dynamic data stream. In dynamic algorithms, changes to the input (for instance edge insertions and deletions in a graph) are diligently reported to the algorithm, whose task is then to update and maintain its data structure efficiently in order to be able to, e.g., answer queries correctly. This approach has been recently extended to streaming algorithms, where memory is limited [AGM12]. For example, we would like to estimate the size of the Twitter graph of a given request given its stream of edges, with the help and validation of experiments.

Online computing with recourse. In online computing with recourse is allowed more or less freedom (recourse) in reaction to change. Data arrives online as a stream, a solution is maintained online, but the algorithm has some limited flexibility (at some cost) to correct its solution instead of making irrevocable decisions at each step as in "classical"' online algorithms [AML13, GGK16, MSVW16]. This is advisable if the new arrivals show that past decisions were unwise in retrospect. We will revisit classical online problems under this new model.

Evolving data. In the evolving data model, changes to the input are not reported but happen silently in the background (e.g.., medical data), and it is up to the algorithm to send out explicit probes (e.g., requests for medical tests) for updates. The focus is on the tradeoff between the rate of data change between probes, and the quality of the solution maintained by the algorithm. This notion was brought into algorithmic game theory for the stable matching problem [KLM16]. In that model, we will investigate some classical problems such as knapsack, matching, and submodular function optimization. We will also model evolving aspects of the college admissions platform Parcoursup.

II. Partners

1. Scientific coordinator

Claire Mathieu is a renowned scientist in theoretical computer science and has published extensively on approximation algorithms, online algorithms, algorithms for noisy data, streaming algorithms, and more. She has extensive experience in leading research projects and has obtained numerous research grants in France and in the USA. She intends to organize a yearly retreat for the members of the consortium and their students, in a location such as CIRM. In addition, a weekly working group will be organized, its location alternating between the locations of the partners of the consortium.

2. Consortium

Our consortium is constructed of experts in a number of different areas, all having the common interest and experience in efficient algorithms design and complexity lower bounds. Previous collaborations exist and will be reinforced by this project in order to address the emerging algorithmic challenges in today's new technologies and new architectures.

- IRIF / Algorithms and Complexity group: property testing, query complexity, streaming algorithms, communication complexity, approximation algorithms, online algorithms, algorithmic game theory.
    - Claire Mathieu (main PI, CNRS Senior researcher)
    - Sophie Laplante (Professor at Université Paris Diderot)
    - Frédéric Magniez (CNRS Senior researcher)
    - Adi Rosén (CNRS Senior researcher)
    - Michel de Rougemont (Professor at Université Paris Panthéon-Assas)
    - Miklos Santha (CNRS Senior researcher)
    - Olivier Serre (CNRS Senior researcher, from the Automata and Applications group)
- DI-ENS / TALGO group: analysis of simple approximation algorithms on structured data, algorithms for matching and scheduling, computing with uncertain data, models for generating structured random graphs, randomized computation, structural graph theory.
    - Chien-Chung Huang (PI, CNRS Researcher)
    - Pierre Aboulker (Assistant professor at ENS)
    - Tatiana Starikovskaya (Assistant professor at ENS)
- LIP6 / Operations Research group: online algorithms, approximation algorithm, search heuristics, local search.
    - Spyros Angelopoulos (CNRS Researcher)
    - Evripidis Bampis (Professor at Sorbonne Université)
    - Vincent Cohen-Addad (CNRS Researcher)
    - Christoph Dürr (PI, CNRS Senior researcher)
    - Bruno Escoffier (Professor at Sorbonne Université)

III. References

[ABB+17] A. Ambainis, K. Balodis, A. Belovs, T. Lee, **M. Santha** and J. Smotrovs. *Separations in query complexity based on pointer functions*. Journal of the ACM 64(5):32, 2017.
[AGM12] K. Jin Ahn, S. Guha, A. McGregor. *Analyzing graph structure via linear measurements.* SODA 2012: 459-467
[AML13] T. Avitabile, **C. Mathieu**, L. Parkinson. *Online constrained optimization with recourse.* Information Processing Letter, 113 (2013) 81–86.
[BKM17] L. Boczkowski, I. Kerenidis, **F. Magniez**. *Streaming Communication Protocols*. ICALP 2017: 130:1-130:14
[DEMM17] **C. Dürr**, T. Erlebach, N. Megow, J. Meißner. *Scheduling with Explorable Uncertainty*. arXiv:1709.02592 (2017)
[ELMR17] G. Even, R. Levi, M. Medina, **A. Rosén**. *Sublinear Random Access Generators for Preferential Attachment Graphs.* ICALP 2017: 6:1-6:15.
[FM13] N. François, **F. Magniez**. *Streaming Complexity of Checking Priority Queues.* STACS 2013: 454-465
[FMRS16] N. François, **F. Magniez, M. de Rougemont, O. Serre**. *Streaming Property Testing of Visibly Pushdown Languages*. ESA 2016: 43:1-43:17
[FMR10] E. Fischer, **F. Magniez, M. de Rougemont**. *Approximate Satisfiability and Equivalence*. SIAM J. Comput. 39(6): 2251-2281 (2010)
[FMS18] E. Fischer, **F. Magniez, T. Starikovskaya**. *Improved bounds for testing Dyck languages*. SODA 2018 (arXiv: 1707.06606)
[GGK16] A. Gu, A. Gupta, A. Kumar. *The Power Of Deferral: Maintaining A Constant-competitive Steiner Tree Online.* Siam J. Comput. 45:1, 1–28, 2016.
[GM16] A. Gupta, M. Molinaro. *How the Experts Algorithm Can Help Solve LPs Online.* Mathematics of Operations Research, 41(4), 1404-1431, 2016.
[KLM16] V. Kanade, N. Leonardos, **F. Magniez**. *Stable Matching with Evolving Preferences*. APPROX-RANDOM 2016: 36:1-36:13
[KLLRX15] I. Kerenidis, **S. Laplante**, V. Lerays, J. Roland, D. Xiao. *Lower bounds on information complexity via zero-communication protocols and applications*. SIAM J. Comput. 44(5): 1550-1572 (2015)
[KRF16] I. Kerenidis, **A. Rosén**, F. Urrutia. *Multi-Party Protocols, Information complexity, and privacy*, MFCS 2016, 57:1-57:16.
[MMV17] F. Mallmann-Trenn, C. Mathieu and V. Verdugo. *Skyline Computation with Noisy Comparisons*. arXiv:1710.02058 (2017)
[MS07] **C. Mathieu**, Warren Schudy. *How to rank with few errors*. STOC 2007: 95-103
[MSVW16] N. Megow, M. Skutella, J. Verschae, A. Wiese. *The Power of recourse for online MST and TSP.* SIAM J. Comput. 45:3, 859-880, 2016.