BSc project proposals

in

Computational Complexity and Game Theory

Ioannis Caragiannis
Kristoffer Arnsfelt Hansen
Srikanth Srinivasan

{iannis, arnsfelt, srikanth}@cs.au.dk
Nygaard 3^{rd} floor

The following pages contain potential topics for BSc projects, but other projects are also possible. The final topic and direction of the project is settled under guidance by the advisor. In particular the weightening between theory and practical implementations is done on an individual basis, and is often adjusted during the BSc project process. Please do not hesitate to pass by our offices on or send us an e-mail for setting up a meeting to discuss potential BSc projects.

Gradient Descent for Convex Optimization

The aim of this project is to study first-order methods in convex optimization. The project should be a mix of theory and practice, with a weighing depending on the preferences of the students. The theoretical part of the project consists presenting the mathematical foundations and of analyzing convergence rates and complexity of several of the main methods used for solving large-scale problems. The practical part of the project consists of implementing and comparing these methods.

References

- To be determined based upon scope of project in terms of theory and practice.

Contact: Kristoffer Arnsfelt Hansen

The Multiplicative Weights Update Method in Game Theory

The idea of the multiplicative weights update method is an important online learning method with applications to many areas, including machine learning and optimization. The goal of this project is to explore the method in the context of game theory. Classic work of Freund and Schapire show that the averages of the strategies produced by the method converge to optimal strategies of zero-sum games. Recent work of Bailey and Pilioras show that while the averages converge to optimal strategies, the sequence may in fact be repelled by optimal strategies. The goal of the project is to present the theoretical background of the multiplicative update method and perform experiments with the method on concrete games.

References

- *Adaptive game playing using multiplicative weights.* Yoav Freund and Robert E Schapire. Games and Economic Behavior, 29(1-2), 79–103, 1999. DOI: 10.1006/game.1999.0738.
- *Multiplicative Weights Update in Zero-Sum Games.* James P. Bailey and Georgios Piliouras. Proceedings 2018 ACM Conference on Economics and Computation (EC), 321–338, 2018. DOI: 10.1145/3219166.3219235.

Contact: Kristoffer Arnsfelt Hansen

Financial Networks and Systemic Risk

The last major financial crisis and its aftermath have revealed the systemic risks and hazards for the society that can arise in financial markets. These are mainly due to the complicated structure of these markets, with many different financial institutions (e.g., banks or firms) that are highly interconnected and interact with each other.

Over the last decade, substantial research has been undertaken to analyse, understand, and manage systemic risks in financial markets. This originates from the seminal work of Eisenberg and Noe, who proposed a graph-theoretic model that is considered the standard today. Interconnections between financial institutions are represented by a directed graph $G = (V, E)$. The node set V contains the financial institutions. A weighted directed edge $e \in E$ expresses a debt relation between two institutions. In addition, each institution has non-negative external assets, which capture the value of property rights (such as real estate, gold, business and mortgage loans, etc.) that the firm has acquired from non-financial institutions. Eisenberg and Noe discuss a clearing mechanism for such a market in which every institution v uses its available assets to pay its debt.

This BSc thesis aims to survey recent work on the model of Eisenberg and Noe and its extensions. Two kinds of questions will be considered. First, we will study computational problems related to identifying risks on the graph representing the interactions between financial institutions. Second, we will study simplified strategic games played by the several institutions. The work will mainly focus on mathematical proofs and analysis of algorithms in the graph-theoretic model above, but it may include implementations and simulations of strategic play by the financial institutions and experiments on these dynamics.

References

- *Systemic Risk in Financial Systems.* Larry Eisenberg and Thomas Noe. Management Science, 47(2):236– 249, 2001. Available from: https://www.jstor.org/stable/2661572
- *Strategic Payments in Financial Networks.* Nils Bertschinger, Martin Hoefer, & Daniel Schmand. In *Proceedings of the 11th Innovations in Theoretical Computer Science Conference (ITCS)*, 46:1-16, 2018. Available from: https://drops.dagstuhl.de/opus/volltexte/2020/11731/
- *Forgiving Debt in Financial Network Games.* Panagiotis Kanellopoulos, Maria Kyropoulou, Hao Zhou. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 335-341, 2022. Available from: https://www.ijcai.org/proceedings/2022/48

Contact: Ioannis Caragiannis

Algorithmic Aspects of Participatory Budgeting

Participatory budgeting is a relatively recent trend used by municipalities and regional governments worldwide to decide the distribution of funding to public projects. The main idea is to let the citizens express their opinion through voting over spending priorities and then implement the outcome of the vote as a public decision.

The details of voting are rather complicated (compared to a typical government election) and include a series of steps and features. We briefly present three components that are interesting from an algorithmic viewpoint. The first is the selection of the available alternatives and their nature. For example, alternatives could be discrete such as the construction of a bridge or continuous such as the length of bicycle paths. Second, how are the actual preferences of the citizens will be structured in the ballots? This step may result in a loss of information. Third, how should the votes be aggregated into a collective decision? This part is the most crucial algorithmically but strongly depends and interplays with the previous two.

The interdisciplinary field of computational social choice (lying at the intersection of social choice theory and theoretical computer science), which traditionally studies the computational complexity of voting rules, has extensively considered participatory budgeting recently, together with other modern trends of participatory democracy. The focus has been on axiomatic properties, and on quantifying the lack of efficiency of the voting process, due to the loss of information when expressing preferences.

The aim of the project is to survey the recent approaches in participatory budgeting, focusing on theoretical developments and analysis, case studies from its application to municipalities around the world, and available online tools. The project will involve the implementation of representative aggregation methods and experiments/simulations with synthetic voting scenarios.

References

- Web: www.participatorybudgeting.org, pbstanford.org.
- *Participatory Budgeting: Models and Approaches.* Haris Aziz and Nisarg Shah. In *Pathways Between Social Science and Computational Social Science: Theories, Methods, and Interpretations*, Springer, 2021. Forthcoming (available from Nisarg Shah's homepage).
- *Interactive Democracy.* Markus Brill. In *Proceedings of the 17th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 1183–1187, 2018. dl.acm.org/doi/10.5555/3237383.3237873.

Contact: Ioannis Caragiannis

Pseudorandom Generators from Theory in Practice

Generating random bits is a task of fundamental importance in Computer Science. Many real-world algorithms need access to a large number of high quality random bits in order to perform various computational tasks efficiently and/or securely. Randomness is also useful in computational simulations of complex phenomena such as the climate or the economy.

But where do these algorithms get access to these random bits? It is hard to find sources of 'true randomness'. Many standard practical ways of obtaining 'random' sequences are not random at all: there are standard ways of finding patterns in such sequences (see, e.g. `http://www.pro-technix.com/information/crypto/pages/rfc1750_base.html`).

Given the difficulties involved in obtaining true randomness, it makes sense to think of randomness as a *resource*, much like running time or memory. A standard way to minimize the use of this resource is to use *Pseudorandom Generators* (PRGs).

A PRG is an algorithm that stretches a short, truly random string to a long string that 'looks random' to many standard algorithms.The study of the theory of PRGs began in cryptography in the early 1980s, and since then, researchers have developed PRGs for various kinds of applications.

This project is to understand and possibly develop provably correct kinds of PRGs for various computational tasks.

On the theoretical side, you will need to understand various constructions of PRGs. This will require some elementary notions in probability and algebra, which you will need to pick up during the course of the project.

On the practical side, you will implement these theoretical constructions on standard statistical tests used to test random sequences in practice. Another interesting avenue is to test standard randomized algorithms on these random sequences.

References

- Michael Luby, Avi Wigderson: *Pairwise Independence and Derandomization.* Found. Trends Theor. Comput. Sci. 1(4) (2005)
- Salil Vadhan: *Pseudorandomness*: A monograph on PRG constructions. `https://people.seas.harvard.edu/~salil/pseudorandomness/`
- `https://en.wikipedia.org/wiki/Diehard_tests`: Wikipedia article on standard tests for pseudorandom sequences.

Contact: Srikanth Srinivasan

The Minimum Circuit Size Problem

Consider the following two strings:

$x = 01010101010101010101010101010101010101 \quad y = 11101000101000011010110001111011110 1.$

Which of these is more random? The reasonable answer is that y is more random than x, but what does this mean? And more importantly, what does this have to do with Computer Science?

The question of defining and quantifying notions of randomness has been taken up by many mathematicians. Some of the most interesting such notions are *computational*, defined by means of computers and algorithms. The notion of *Kolomogorov complexity* deals with how compactly a given string x can be compressed by a general algorithm. A string is defined to be random if it cannot be meaningfully compressed by any algorithm.

Unfortunately, the Kolomogorov complexity of a string is itself a difficult quantity to understand, as it is uncomputable. However, to overcome this, we have a number of resource-bounded notions of Kolomogorov complexity, all of which quantify various notions of compressibility. The *Minimum Circuit Size Problem* is an umbrella term for the corresponding computational questions: it is the algorithmic problem of computing the most 'compressed' version of a given string x.

These notions are useful in both theory and practice. In practice, these problems are related to fundamental problems in logic design. In theory, the computational complexity of this problem has been linked to the existence of fundamental objects in cryptography.

The aim of the project is to understand variants of the Minimum Circuit Size Problem, the algorithms we know for these variants, and where we suspect (or can show) that the problem is computationally hard. Understanding these results will require some elementary notions related to probability, algebra, approximation algorithms and basic polynomial-time reductions, which you will need to pick up as we go along.

References

- Eric Allender: *The Complexity of Complexity.* Computability and Complexity 2017: 79-94.
- Eric Allender, Lisa Hellerstein, Paul McCabe, Toniann Pitassi, Michael E. Saks: *Minimizing Disjunctive Normal Form Formulas and AC0 Circuits Given a Truth Table.* SIAM J. Comput. 38(1): 63-84 (2008).
- Shuichi Hirahara, Igor Carboni Oliveira, Rahul Santhanam: *NP-hardness of Minimum Circuit Size Problem for OR-AND-MOD Circuits.* Computational Complexity Conference 2018: 5:1-5:31.

Contact: Srikanth Srinivasan

Secret sharing via Monotone Boolean Formulae

(Jointly offered by CCGT and Cryptography groups.)

Secret sharing is a way of splitting a secret up into several shares, such that individually, the shares completely hide all information about the secret, but the secret can be recovered if a sufficient number of shares are combined.

Secret sharing can be used to distribute sensitive information across several servers, providing both security and redundancy. It is also an essential tool in building secure multi-party computation protocols, which allow performing computations on private data that is held by several participants.

This project will look at a classic method of building secret sharing schemes using monotone Boolean formulae, which are a special type of Boolean circuit. This method is very general, and has some nice properties that are useful for applications. Unfortunately, the best known construction of a suitable monotone Boolean formula is quite expensive, with a size of $O(n^{5.3})$ for n participants, using a probabilistic construction by Valiant.

The goal of the project is to understand these constructions and get a better idea of the hidden constants in the big-oh notation, as well as try to optimize and perhaps implement the construction for small values of n.

Depending on interest, it's also possible to look into other aspects such as different types of secret sharing schemes and applications.

References

- Secret sharing from monotone Boolean formulae: `https://viterbi-web.usc.edu/~shanghua/teaching/Fall2014-476/BenalohLeichter.pdf`
- Valiant's construction of monotone Boolean formulae for threshold functions: `https://www.sciencedirect.com/science/article/pii/0196677484900166`
- A variant of Valiant's construction: `https://www.wisdom.weizmann.ac.il/~oded/COL2/mono-maj.pdf`

Contact: Peter Scholl (`peter.scholl@cs.au.dk`), Srikanth Srinivasan