# PhD projects in the Department of Informatics, AY 25-26 — Foundations of computing (algorithms, computational complexity)

The PhD projects listed below will be considered for 2025/26 studentships available in the Department of Informatics to start on 1 October 2025 or later during the 2025/26 academic year.

Please note that this list is not exhaustive and potential applicants can alternatively identify and contact appropriate supervisors outlining their background and research interests or proposing their own project ideas.

Each project is designated for a single student, meaning it can only be assigned to one successful applicant. Some projects come with allocated studentships, while others are eligible for "unallocated" studentships. Applicants who apply for projects with allocated studentships and are selected will be offered a full studentship. In the project list, these are marked as "studentship allocated." Applicants chosen for other projects will compete for the unallocated studentships.

We welcome applications from students who have secured, or are applying for, or plan to apply for other funding (within other studentships internal to the university or external schemes) and from self-funded students. See also this list of funding opportunities available at King's for post-graduate research in Computer Science.

# PhD projects

- Nominal Specification and Verification Environments (studentship allocated)
- Game-theoretic models in cryptoeconomics: incentives, mechanism design and blockchain dynamics
- Game-theoretic models in multi-agent systems: emergent behaviours, critical phase transactions and learning dynamics
- Sustainable Delivery Logistics with Drones and Cargo Bikes
- Brzozowski Derivatives for Fast Regular Expression Matching
- Validation and Testing of GPUs
- Algorithms for perpetual scheduling problems
- Algorithmic Game Theory, Mechanism Design, or a Related Area
- Verified Complexity Theory: Probabilistic Computation and Verified Post-Quantum Cryptography
- Software Verification and Nominal Dependent Type Theory
- Dealing with imperfect rationality in computational systems
- Network Optimisation Algorithms

# Nominal Specification and Verification Environments

Supervisor: Maribel Fernandez

Areas: Foundations of computing (algorithms, computational complexity), Systems (software engineering, programming)

## Project Description

Software verification techniques have been successfully used to prove correctness of low-level programs, but verification of high-level programming languages is challenging: it requires reasoning about name binding (e.g., visible/hidden channel names, scoping rules defining local and global variable names, parameter passing and substitution of values for variables). A standard approach to deal with name binding in verification tasks is to replace names with numerical codes (de Bruijn indices). While this avoids some of the difficulties of reasoning about names in programs, conducting a formalisation in de Bruijn style is labour-intensive and imposes a significant overhead to comprehending and reusing proofs. Nominal techniques offer an alternative, user-friendly approach, which does not require to replace names with codes. We aim to apply novel nominal techniques to simplify the handling of names and binders in programming languages and verification tasks (e.g., verification of blockchain languages). For this, we aim to develop a core nominal calculus and use it as a basis to enrich with nominal features two successful verification frameworks: Maude and K.

# Game-theoretic models in cryptoeconomics: incentives, mechanism design and blockchain dynamics

Supervisor: Dr. Stefanos Leonardos

Areas: Artificial Intelligence (symbolic AI, logic, etc.), Machine learning / Deep learning, Foundations of computing (algorithms, computational complexity), Game theory

## Project Description

This project is aimed for students who are interested in advancing cutting-edge research at the intersection of game theory and cryptoeconomics. The project will focus on modelling and analyzing blockchain-enabled economies through a game-theoretic lens. Special focus will be placed on transaction fee mechanisms (TFMs), miner extractable value (MEV), proposer-builder separation (PBS) in Ethereum block creation, MEV-boost auctions, dynamics of automated market makers (AMMs), transaction censorship, attacks in decentralized exchanges, and related phenomena. The study will explore cryptoeconomic mechanisms, dissect participants' incentives, and designing mechanisms to optimize blockchain performance. Due to the dynamic nature of these systems, the project will employ elements from algorithmic game theory and dynamical systems, alongside standard tools from economics, computer science, and machine learning. Successful candidates will develop game-theoretic models, conduct rigorous mathematical analyses, and run simulations to validate theoretical predictions in real-world applications, bridging the gap between academia and industry.

## References


1. Buterin, V, Reijsbergen, D, Leonardos, S, Piliouras, G. Incentives in Ethereum's hybrid Casper protocol. Int J Network Mgmt. 2020; 30:e2098. https://doi.org/10.1002/nem.2098
2. Leonardos, S., Reijsbergen, D., Monnot, B., and Piliouras, G., "Optimality Despite Chaos in Fee Markets", arXiv e-prints, 2022. doi:10.48550/arXiv.2212.07175.
3. W. Wu, T. Thiery, S. Leonardos, C. Ventre. Strategic Bidding Wars in On-chain Auctions. IEEE ICBC 2024, https://arxiv.org/abs/2312.14510.
4. Leonardos, S, Reijsbergen, D, Piliouras, G. Weighted voting on the blockchain: Improving consensus in proof of stake protocols. Int J Network Mgmt. 2020; 30:e2093. https://doi.org/10.1002/nem.2093
5. Leonardos, N., Leonardos, S., Piliouras, G. (2020). Oceanic Games: Centralization Risks and Incentives in Blockchain Mining. In: Pardalos, P., Kotsireas, I., Guo, Y., Knottenbelt, W. (eds) Mathematical Research for Blockchain Economy. Springer Proceedings in Business and Economics. Springer, Cham. https://doi.org/10.1007/978-3-030-37110-4_13
6. Leonardos, S., Monnot, B., Reijsbergen, D., Skoulakis, E., and Piliouras, G. (2021). Dynamical analysis of the EIP-1559 Ethereum fee market. In Proceedings of the 3rd ACM Conference on Advances in Financial Technologies (AFT '21). Association for Computing Machinery, New York, NY, USA, 114—126. https://doi.org/10.1145/3479722.3480993
7. D. Reijsbergen, S. Sridhar, B. Monnot, S. Leonardos, S. Skoulakis and G. Piliouras, "Transaction Fees on a Honeymoon: Ethereum's EIP-1559 One Month Later," 2021 IEEE International Conference on Blockchain (Blockchain), Melbourne, Australia, 2021, pp. 196-204, doi: 10.1109/Blockchain53845.2021.00034.
8. Koki, C., Leonardos, S., and Piliouras, G. (2022). Exploring the predictability of cryptocurrencies via Bayesian hidden Markov models, Research in International Business and Finance, Volume 59, 101554, doi: 10.1016/j.ribaf.2021.101554.

# Game-theoretic models in multi-agent systems: emergent behaviours, critical phase transactions and learning dynamics

Supervisor: Dr. Stefanos Leonardos

Areas: Artificial Intelligence (symbolic AI, logic, etc.), Machine learning / Deep learning, Game Theory, Foundations of computing (algorithms, computational complexity)

## Project Description

This project is aimed at students who are interested in cutting-edge research at the intersection of multi-agent systems, game theory and learning dynamics, with applications in economics, machine learning, and artificial intelligence. The project's objective is to explore the intricate patterns of multi-agent systems through a game-theoretic lens, emphasizing on learning dynamics, chaos theory, and their applications. Special focus will be placed on understanding the emergent behaviors in algorithmic decision-making processes that continuously evolve over time. The study will explore phase transitions in strategic interactions, analyze or develop novel algorithms, and quantify their implications on coordination and competition in real-world systems. The analysis will use tools from game theory, mathematics and the theory of dynamical systems, to develop, study and apply learning algorithms in complex multi-agent systems. Successful applicants will have the chance to shape the future of learning systems, bridging theoretical advancements with practical applications with the frameworks of machine learning and artificial intelligence.

## References


1. I. Sakos, S. Leonardos, S. A. Stavroulakis, W. Overman, I. Panageas, G. Piliouras. Beating Price of Anarchy and Gradient Descent without Regret in Potential Games, 12th International Conference on Learning Representations (2024).
2. S. Roesch, S. Leonardos & Y. Du. Selfishness Level Induces Cooperation in Sequential Social Dilemmas, 23rd Conference on Autonomous Agents and Multiagent Systems (2024).
3. Leonardos, S., Sakos, J., Courcoubetis, C. and Piliouras, G. (2023). Catastrophe by Design in Population Games: A Mechanism to Destabilize Inefficient Locked-in Technologies. ACM Trans. Econ. Comput. 11, 1—2, Article 1 (June 2023), 36 pages. doi:10.1145/3583782
4. Leonardos, S., and Piliouras, G. (2022). Exploration-exploitation in multi-agent learning: Catastrophe theory meets game theory, Artificial Intelligence, Volume 304, 103653, doi:10.1016/j.artint.2021.103653.
5. Leonardos, S., Piliouras, G., and Spendlove, K. (2021). Exploration-Exploitation in Multi-Agent Competition: Convergence with Bounded Rationality, in Advances in Neural Information Processing Systems, volume 34, pp. 26318--26331, Curran Associates, Inc., https://proceedings.neurips.cc/paper_files/paper/2021/file/dd1970fb03877a235d530476eb727dabPaper.pdf.
6. Leonardos, S., Overman, W., Panageas I., and Piliouras, G. (2022). Global Convergence of Multi-Agent Policy Gradient in Markov Potential Games, in International Conference on Learning Representations (ICLR 2022), https://openreview.net/forum?id=gfwON7rAm4.

# Sustainable Delivery Logistics with Drones and Cargo Bikes

Supervisor: Dimitrios Letsios

Areas: Robotics, Foundations of computing (algorithms, computational complexity), Artificial Intelligence (symbolic AI, logic, etc.)

## Project Description

Effectively solving inventory optimisation, scheduling, and vehicle routing problems is essential to meet customer demands and manage fleet, human worker, and storage costs for companies in the logistics sector, e.g. courier and delivery services. Drones and cargo bikes offer an alternative to cost-effective and sustainable last-mile deliveries in urban environments. The goal of this project is to develop data science and optimisation methodologies (e.g. metaheuristics and integer programming) for generating solutions to challenging logistics optimisation problems with drones and cargo bikes that attain optimal trade-offs between solution quality and computational effort, while enhancing their performance with simulations and machine/deep learning. Proposed approaches will be evaluated using recent data and benchmark case studies from business partners and in the literature. The project will include a thorough study on effective hyper-parameter tuning (e.g. initialisation components, solution space structure, search operators) of optimisation methodologies for computing the best possible solution within given time frames. Further, it will investigate innovative ways of enhancing performance with dynamic parameter adaptations and machine/deep learning, based on information (e.g. rewards, data properties) collected during the solution process. The project will also assess and strengthen the robustness of proposed approaches via simulations.

# Brzozowski Derivatives for Fast Regular Expression Matching

Supervisor: Christian Urban

Areas: Foundations of computing (algorithms, computational complexity), Systems (software engineering, programming)

## Project Description

If you want to connect a computer directly to the Internet, it must be immediately hardened against outside attacks. The current technology for this is to use regular expressions in order to automatically scan all incoming network traffic for signs when a computer is under attack and if found, to take appropriate counter-actions. Unfortunately, algorithms for regular expression matching are sometimes slow and inefficient. My research is about making breakthroughs in this area. The results are also applicable to DNA searches, security, compilers and algorithms. In addition I am interested in supervising topics in programming languages, formal methods, functional programming, Rust and theorem provers. Skills in these areas are in high demand in both industry and academia: my former PhD students work now at ARM and Imperial College London.

# Validation and Testing of GPUs

Supervisor: Hector Menendez, Maribel Fernandez and Karine Even-Mendoza

Areas: Foundations of computing (algorithms, computational complexity), Systems (software engineering, programming)

## Project Description

Considering the rise of graphics and machine learning libraries, Graphics Processing Units (GPUs) are becoming increasingly relevant, especially for solutions that aim to train large machine learning models. GPUs can significantly reduce the training and inference time of these models by employing multiple graphics kernels. Although these systems present a significant promise for advancing AI and other dependent systems, their reliance on floating point operations limits how they can be validated and tested. The literature shows various contributions to the process of testing GPUs, with the most notable being the work of Alastair Donaldson. He initially developed the GPUVerifier (Betts et al., 2012) and later focused on the problems of concurrency (Alglave et al, 2015) and automatic testing of system rendering (Donaldson et al., 2017). Considering current rendering technologies that focus on universal platforms, such as browsers, through WebGPU technologies (Bernhard et al. 2024), it is also important to evaluate the quality of these specific APIs to ensure that GPU resources are properly utilized and optimized. The main goal of this PhD is to define different verification and testing strategies to validate new GPU technologies. To achieve this, the PhD will start by investigating the work of Alastair Donaldson and others on GPU evaluation, including the GPUVerifier, WebGPU evaluation system, GraphicsFuzz, and other strategies adapted for GPU validation. It may also be beneficial to explore how the testing and validation methods apply to similar accelerators, such as Tensor Processing Units (TPUs) and Neural Processing Units (NPUs). These hardware accelerators are becoming increasingly common in AI. Investigating how these methods translate to TPUs and NPUs could provide additional opportunities for optimizing and ensuring the robustness of AI systems. The thesis will be divided into: Part 1: Literature Review and Identification of Gaps in the Literature: Identify gaps related to the problem of sandboxing current browser APIs and improving the security of systems against potential adversarial attacks. Part 2: Verification: Design different verification strategies based on formal verification methods. These methods will work within a determined context to bound the system's complexity and manage a deeper analysis for validation. Part 3: Testing and Fuzzing: Redesign testing methods to identify specific use cases that expose vulnerabilities in GPU-based systems. These methods focus on the specific technological context, particularly concurrency and floating-point. This PhD will contribute to the development of robust verification and testing strategies, ensuring the efficient and secure use of GPUs in modern computing applications.

## References


Alglave, J., Batty, M., Donaldson, A. F., Gopalakrishnan, G., Ketema, J., Poetzl, D., ... & Wickerson, J. (2015). GPU concurrency: Weak behaviours and programming assumptions. ACM SIGARCH Computer Architecture News, 43(1), 577-591.
Bernhard, L., Schiller, N., Schloegel, M., Bars, N., & Holz, T. (2024). DarthShader: Fuzzing WebGPU Shader Translators & Compilers. arXiv preprint arXiv:2409.01824.
Betts, A., Chong, N., Donaldson, A., Qadeer, S., & Thomson, P. (2012, October). GPUVerify: a verifier for GPU kernels. In Proceedings of the ACM international conference on Object oriented programming systems languages and applications (pp. 113-132).
Donaldson, A. F., Evrard, H., Lascu, A., & Thomson, P. (2017). Automated testing of graphics shader compilers. Proceedings of the ACM on Programming Languages, 1(OOPSLA), 1-29.

# Algorithms for perpetual scheduling problems

Supervisor: Tomasz Radzik

Areas: Foundations of computing (algorithms, computational complexity)

## Project Description

Perpetual scheduling refers to scheduling recurring events where each event must occur at a specified frequency. This task arises in various application contexts, such as determining service schedules for a collection of machines distributed across a geographic area, each with its own service frequency, or scheduling medical checkups for hospital patients, where different patients require checks at different intervals. Our goal is to design efficient algorithms for computing perpetual schedules while analyzing the trade-offs between computational time and the quality of the resulting schedules. There are several variants of the perpetual scheduling problem, depending on the specific constraints and how the quality of schedules is evaluated. For example, in the context of machine servicing, we may aim to minimize either the maximum waiting time for the next service or the average waiting time. This research falls within the fields of optimization, algorithm design, and computational complexity. The reference below provides further information on perpetual scheduling and the relevant computational methodologies

## References

L. Gasieniec, T. Jurdzinski, R. Klasing, C. Levcopoulos, A. Lingas, J. Min, T.Radzik, "Perpetual maintenance of machines with different urgency requirements. Journal of Computer and System Sciences, vol. 139 (2024).

## Algorithmic Game Theory, Mechanism Design, or a Related Area

Supervisor: Bart de Keijzer

Areas: Foundations of computing (algorithms, computational complexity)

Project Description

Algorithmic game theory is a research area that lies in the intersection of computer science, mathematics, and micro-economics. It aims to advance understanding of various computational challenges relating to scenarios of "strategic interaction between self-interested" agents. The project will investigate algorithms, mechanisms, complexity, and computability of various prominent computational challenges for such strategic scenarios, where examples are auctions and two-sided market platforms that are used in the sharing economy, in online ad-auctions, and in e-commerce in general. Desired outcomes of the project are the design of algorithms and mechanisms with provably desirable economic and computational properties. Research in this area has a strong theoretical computer science component and focuses on algorithms and complexity. Please get in touch if you're interested in this project or related topic e.g. in computational social choice, algorithms, computational complexity, or or optimisation.

# Verified Complexity Theory: Probabilistic Computation and Verified Post-Quantum Cryptography

Supervisor: Mohammad Abdulaziz

Areas: Foundations of computing (algorithms, computational complexity), Systems (software engineering, programming)

## Project Description

In 1971, Cook [9] formulated the question P vs. NP for the first time. To this day, this question has not been solved, but could be argued to have been a steady motivator in the field of complexity theory. Many other complexity classes have been defined and studied, like PSPACE, EXP, their nondeterministic variants NPSPACE, NEXP and the polynomial hierarchy PH. The introduction of probabilistic computational models led to further classes like BPP, PP and IP. The relationship between these classes is still being investigated, but there have been some conditional and unconditional results, like PSPACE = NPSPACE [?] and IP = PSPACE [20]. In this project, I propose the use of Interactive Theorem Provers (ITPs) (aka proof assistants) to the area of computational complexity. ITPs are mechanised mathematical systems, i.e. systems which can be used to develop machine-checkable (aka formal) proofs. To prove a theorem in an ITP, the user provides high-level steps of a proof, and the ITP fills in the details, at the level of axioms, culminating in a formal proof. The fact that ITPs can use human expertise is a source of their strength in many applications, e.g. when the properties to prove are undecidable. ITPs have been used to formally prove results from a large number of areas, ranging from the correctness of realistic software systems [16, 14, 13] to formally prove results from more theoretical areas of computer science, especially efficient algorithms and combinatorial optimisation, e.g. algorithms for matching [3, 2], minimum-cost flows [1], maximum flows [15], the simplex algorithm to optimally solve linear programs [17], and geometric algorithms [19]. Furthermore, ITPs are currently attracting the attention of mathematicians and are being used to formally prove many important results in pure mathematics [8, 4, 10]. However, despite all these applications, ITPs have not been used to formally prove anything but the rather elementary results from complexity theory. Most relevant to this proposal, ITPs were recently used to formally prove the Cook-Levin-Theorem, which states that SAT is NP-complete, by Gaher and Kunze [11] in the ITP Coq and Blabach in the ITP Isabelle/HOL [6]. A notable complication when using ITPs to formally prove results from complexity theory is the difficulty of formal reasoning about computational models. Indeed, such reasoning requires the development of a reasoning infrastruc- ture, including formally proving many theorems and developing methods to automate proofs. For instance, Balbach used Turing machines as their underlying computation model, which Gaher and Kunze used Lambda calculus. In this project, you will develop a framework for formal reasoning about probablistic computation models, e.g. Turing machines with a source of random bits. One application area of such a reasoning framework is verifying randomised complexity theoretic reductions. A notable example is the NP-Hardness proof of the problem of Learning with Errors by Atjai [5], which is used to prove the security of CRYSTALS-KYBER [7], one of the few NIST approved post-quantum cryptographic algorithms. An outcome could thus be an implementation of CRYSTAL-KYBER that is formally verified to be secure. Methods that you will need to master include developing automation for the ITP Isabelle/HOL [18] and applying quantitative program logics [12, 21].

## References


[1] Mohammad Abdulaziz and Thomas Ammer. A Formal Analysis of Capacity Scaling Algorithms for Minimum Cost Flows. In The 15th International Conference on Interactive Theorem Proving (ITP 2024).
[2] Mohammad Abdulaziz and Christoph Madlener. A Formal Analysis of RANKING. In The 14th Conference on Interactive Theorem Proving (ITP).
[3] Mohammad Abdulaziz, Kurt Mehlhorn, and Tobias Nipkow. Trustworthy graph algorithms (invited paper). In The 44th International Symposium on Mathematical Foundations of Computer Science (MFCS).
[4] Mohammad Abdulaziz and Lawrence C. Paulson. An Isabelle/HOL Formalisation of Green's Theorem.
[5] Miklos Ajtai. The shortest vector problem in L 2 is NP -hard for randomized reductions (extended abstract). In Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing - STOC '98.
[6] Frank J. Balbach. The Cook-Levin theorem.
[7] Joppe W. Bos, Leo Ducas, Eike Kiltz, Tancrede Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, and Damien Stehle. CRYSTALS - Kyber: A CCA-secure module-lattice-based KEM.
[8] Davide Castelvecchi. Mathematicians welcome computer-assisted proof in 'grand unification' theory.
[9] Stephen A. Cook. The Complexity of Theorem-Proving Procedures. In 3rd Annual ACM Symposium on Theory of Computing (STOC).
[10] prefix=van useprefix=false family=Doorn, given=Floris, Patrick Massot, and Oliver Nash. Formalising the h- Principle and Sphere

Eversion. In Proceedings of the 12th ACM SIGPLAN International Conference on Certified Programs and Proofs, CPP 2023, Boston, MA, USA, January 16-17, 2023.
[11] Lennard Gaher and Fabian Kunze. Mechanising Complexity Theory: The Cook-Levin Theorem in Coq. In 12th International Conference on Interactive Theorem Proving (ITP).
[12] Maximilian Paul Louis Haslbeck. Verified Quantitative Analysis of Imperative Algorithms (Verifizierte Quanti- tative Analyse von Imperativen Algorithmen).
[13] Sudeep Kanav, Peter Lammich, and Andrei Popescu. A Conference Management System with Verified Document Confidentiality. In The 26th International Conference on Computer Aided Verification (CAV).
[14] Gerwin Klein, Kevin Elphinstone, Gernot Heiser, June Andronick, David Cock, Philip Derrin, Dhammika Elkaduwe, Kai Engelhardt, Rafal Kolanski, Michael Norrish, Thomas Sewell, Harvey Tuch, and Simon Winwood. seL4: Formal verification of an OS kernel. In 22nd ACM Symposium on Operating Systems Principles 2009 (SOSP).
[15] Peter Lammich and S. Reza Sefidgar. Formalizing Network Flow Algorithms: A Refinement Approach in Isabelle/HOL.
[16] Xavier Leroy. Formal verification of a realistic compiler.
[17] Filip Maric, Mirko Spasic, and Rene Thiemann. An Incremental Simplex Algorithm with Unsatisfiable Core Generation.
[18] Tobias Nipkow, Lawrence C. Paulson, and Markus Wenzel. Isabelle/HOL - A Proof Assistant for Higher-Order Logic.
[19] Martin Rau and Tobias Nipkow. Verification of Closest Pair of Points Algorithms. In Automated Reasoning - 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part II.
[20] Adi Shamir. IP = PSPACE.
[21] Joseph Tassarotti and Robert Harper. A separation logic for concurrent randomized programs.

# Software Verification and Nominal Dependent Type Theory

Supervisor: Maribel Fernandez

Areas: Systems (software engineering, programming), Foundations of computing (algorithms, computational complexity)

## Project Description

Dependent Type Theory is a mathematical tool to write formal specifications and prove the correctness of software implementations. The proof assistants used to certify the correctness of programs (such as Coq), are based on dependently typed higher-order abstract syntax. The goal of this project is to explore alternative foundations for proof assistants using nominal techniques. The nominal approach has roots in set theory and has been successfully used to specify programming languages. This project will focus on the combination of dependent types and nominal syntax and explore the connections between the nominal approach and the higher-order syntax approach used in current proof assistants.

## References

Typed Nominal Rewriting, Elliot Fairweather and Maribel Fernandez, ACM Transactions on Computational Logic, vol.19, number 1, 2018.

# Dealing with imperfect rationality in computational systems

Supervisor: Carmine Ventre

Areas: Foundations of computing (algorithms, computational complexity)

## Project Description

Dealing with imperfect rationality in computational systems. Project Description Computational/AI systems often collect their input from humans. For example, parents are asked to input their preferences over primary schools before a centralised algorithm allocates children to schools. Should the AI trust the input provided by parents who may try to game the system? Should the parents trust that the AI system has optimised their interests? Would it be safe to run the algorithm with a potentially misleading input? Algorithmic Game Theory (AGT) is a research field that attempts to add safety and trustworthiness to AI systems vis-a-vis strategic reasoning. With its set of symbolic tools, one aims to align the goals of the AI system (e.g., the allocation algorithm above) with those of the agents (e.g., the parents above) involved. The AI will then be safe, in that we can analytically predict end states of the system, and trustworthy, since no rational agent will attempt to misguide the system and the system will work on truthful inputs. One assumption underlying much of the work in AGT is, however, pretty limiting: agents need to be fully rational. This is unrealistic in many real-life scenarios; we, in fact, have empirical evidence that people often misunderstand the incentives and try to game the system even when it is against their own interest. Moreover, modern software agents, often built on top of AI tools, are seldom able to perfectly optimise their rewards. This project will look at novel approaches to deal with imperfect rationality, including analysis of known AI systems and the design of novel ones. This will involve theoretical work that builds on the recent advances on mechanism design for imperfectly rational agents (namely obvious strategyproofness and not obvious manipulability) to include more complex domains and the modelling of further behavioural biases in mechanism design. Prospective applicants are encouraged to consult the publications of Prof Ventre at https://kclpure.kcl.ac.uk/portal/en/persons/carmine.ventre/publications/.

# Network Optimisation Algorithms

Supervisor: Tomasz Radzik, Kathleen Steinhofel

Areas: Foundations of computing (algorithms, computational complexity)

## Project Description

Network Optimisation problems are computational problems where the input data has a network structure. Such problems occur in computer science, operations research, logistics, engineering, and applied mathematics. From the computer science point of view, the general objective of studying network optimisation problems is to analyse their computational complexity with the aim of developing efficient algorithms which provide strict performance guarantees. This project will focus on algorithms for network optimisation problems with the dynamic underlying network structure, which changes over time. One of the applications is to provide efficient routing in networks where individual node-to-node links are not always available. The reference below is a recent survey of algorithms for dynamic network optimisation problems.

## References

K. Hanauer, M. Henzinger, C. Schulz, "Recent Advances in Fully Dynamic Graph Algorithms." 1st Symposium on Algorithmic Foundations of Dynamic Networks (SAND 2022). DOI: 10.4230/LIPIcs.SAND.2022.1.