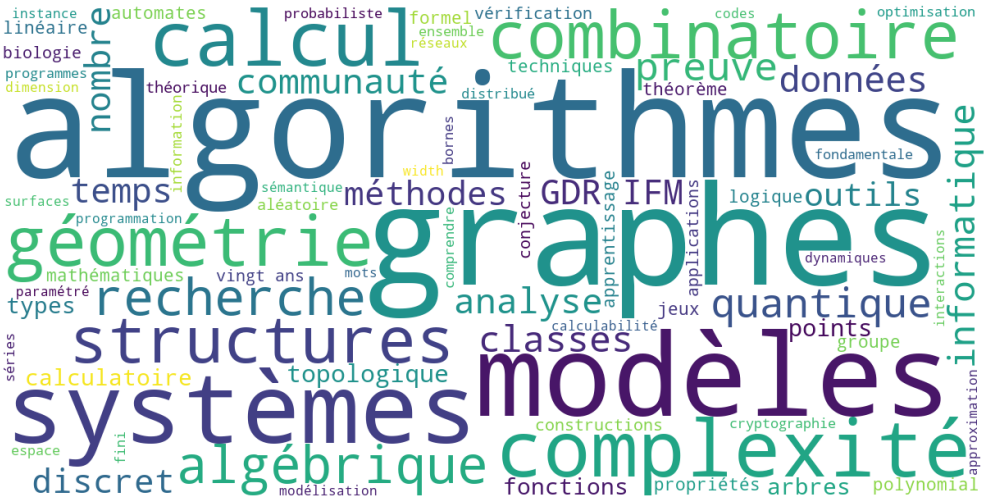
GDR Groupement de recherche
IFM Informatique Fondamentale et ses Mathématiques

Les vingt ans du GdR IFM



Algorithmes, complexité, calculabilité et calcul quantique

Twenty years of GdR IFM, seen from GT CoA

1 Central Topics of GT CoA: Algorithms and Complexity

The aim of the CoA working group of GdR IFM is to bring together all researchers in computer science and mathematics interested in methods and tools for:

- designing and analyzing efficient algorithms,
- establishing lower bounds on properties of algorithms (e.g., computation time, circuit size, approximation factors, quantity of bits exchanged, etc.).

The CoA working group focuses on all forms of algorithms including sequential, parallel or distributed algorithms, online algorithms, streaming algorithms, approximation algorithms, parameterized algorithms, probabilistic algorithms, quantum algorithms, and other new paradigms. We focus on the design and analysis of algorithms, approached from the joint point of view of upper and lower bounds regarding complexity and other bounded resources. The CoA working group is also interested in algorithms motivated by and applied to all types of environments: graphs, networks, biological systems, images, combinatorial objects, etc.

We list below some examples of areas of particular interest to CoA researchers in France and discuss the evolution of these domains over the last twenty years. This list is far from being exhaustive. In fact, algorithmic and complexity-theoretic questions can be found across (almost) all working groups of GDR IFM.

2 Algorithms

Approximation Algorithms

The field of approximation algorithms focuses on the design of provably good polynomial-time solutions typically for NP-hard problems, and has been studied extensively for several decades. The last 20 years have again seen major improvements on longstanding questions thanks to new techniques related for example to linear programming relaxations or randomization over specific distributions. We can cite for illustration the Traveling Salesperson Problem, for which the approximation factor of the metric version has been improved for the first time since the 1970s [51], and the first constant approximation algorithm for the asymmetric case has been designed [73], also following decades of research.

Besides such progress, there is still a significant gap between lower and upper bounds on approximation factors for many problems. A new direction has therefore been focusing on conditional lower bounds, assuming stronger hypotheses than $\mathsf{P} \neq \mathsf{NP}$ to exhibit inapproximability results, thus unifying and making explicit the core open problems at the origin of such gaps. For example, assuming *Unique Games Conjecture* variants has been proven to be sufficient to get tight lower bounds in several scheduling problems [72, 12].

As approximation algorithms focus on guaranteeing the quality of the solution for *all* instances, they may therefore be pessimistic on some real-world instances. This observation has motivated several directions aiming to go "beyond the worst-case", a perspective that has flourished in the last 20 years, leading to new research domains such as smooth analysis, robust optimization, advice complexity and learning-augmented algorithms [69].

Distributed Algorithms

The last 20 years have seen a considerable expansion in the range of applications for distributed computing, originally driven mainly by computer networks (multi-core processors, sensor networks, data centers, peer-to-peer networks, blockchains, etc.), which now extends to biological systems and nanotechnologies, and even sociology (social networks) and physics (complex systems). This evolution has led to a wide diversification of the computational models considered in the context of distributed computing.

In terms of fundamental research, recent algorithmic developments aim to circumvent the numerous obstacles that make it impossible to solve certain tasks in environments prone to failures or attacks (e.g., consensus). Studies focus in particular on the use of sophisticated communication mechanisms and cryptographic primitives. In the context of developing network algorithms for solving graph problems, distributed graph decomposition techniques have seen enormous growth, as well as derandomization techniques. It is also worth mentioning recent advances in distributed quantum computing, and the emergence of a line of research dedicated to limiting the energy consumption of distributed algorithms.

Finally, the identification of lower bounds has also seen enormous progress through the use of tools from algebraic topology, graph theory (e.g., round reduction) and communication complexity, coupled with a better understanding of the power and limitation of using random resources.

Linear Programming Algorithms

Linear programming is one of the landmark achievements of modern mathematics and computing. Developed in the 1940s to formalize questions of resource allocation, it soon became a powerful language for modeling and solving problems across science, industry, and economics. From logistics and scheduling to energy planning and finance, its influence has been enormous, while at the same time it has driven some of the most important advances in algorithms.

The earliest breakthrough was the *simplex method,* designed by Dantzig in 1947. The simplex method is an algorithm that can solve linear programming problems efficiently in practice, and it is a cornerstone of modern optimization software. Unlike its efficiency in practice, in the 1970s the simplex method was proven to require exponential time in the theoretical worst case. Similar to the field of approximation algorithms, this has inspired many researchers to innovate new approaches for algorithm analysis [69]. For the past 25 years, the leading approach has been *smoothed analysis*, introduced by Spielman and Teng. Smoothed analysis shows that if the input is perturbed by tiny random noise, then simplex runs in polynomial time in expectation [71]. This result has launched a wave of refinements, leading to variants with provably much fewer pivot steps [23, 46, 11]. Today, nearly matching upper and lower bounds on the smoothed complexity are known [11]. This has prompted new analytical approaches, basing mathematical assumptions on close observation of algorithm implementations and user manual specifications [10].

A very different approach appeared in the 1980s with *interior point methods,* which follow a continuous path through the interior of the feasible region [52]. These algorithms have a theoretical guarantee of about the square root of the problem dimension in the number of iterations, yet in practice they converge much faster, making them a cornerstone of modern solvers [80]. Their algebraic structure also makes it possible to combine them with fast linear system solvers, which has powered breakthroughs in theory for fundamental algorithmic problems such as maximum flow and minimum cost flow [64, 60, 65, 22, 8, 53, 39, 9, 78, 77,

76, 21, 79].

Recent work has even started to connect the two algorithmic lines. Many analyses of simplex methods are based on a geometric quantity called the shadow size, which measures the number of segments of a two-dimensional projection of the feasible polyhedron. Recent work has shown that the running time of certain interior-point methods is no greater than this number [3]. This led to strongly polynomial algorithms for new classes of problems [24], and opened a promising path towards the solution to Smale's 9th problem, whether all linear programs can be solved in strongly polynomial-time.

Fixed-Parameter Algorithms

Over the past two decades, the parameterized complexity community has developed an extensive toolkit for tackling hard problems, notably the techniques of color coding, iterative compression, bidimensionality, important separators, Cut and Count, and more recently, flow augmentation. The latter four techniques have influenced other areas, especially those concerned with cut and flow problems. In the last 10 years, new width parameters, such as mim-width, twin-width, and merge-width (the successors of treewidth and clique-width), have emerged and yielded logic-based meta-theorems, providing a clearer and more unified landscape of tractability. This is particularly true for the first-order logic model checking problem, where intensive efforts led to the characterization of the FPT tractability barrier for monotone classes (nowhere dense graphs), and significant progress for hereditary classes. Kernelization, a subarea of parameterized complexity, has obtained similar meta-theorems, via a technique called *protrusion replacement.* In parallel, lower-bound techniques have enabled one to match the running time of the fastest known algorithms for most of the parameterized problems.

While historically the focus of parameterized complexity has been on NP-hard graph problems, recent years have seen an expansion to handling problems from computational geometry, computational social choice, bioinformatics, etc., as well as polynomial-time-solvable graph problems, in the so-called FPT in P program. The latter has gradually merged with the field of fine-grained complexity. Another fruitful development of parameterized complexity is its interface with approximation algorithms. For problems that are both hard to approximate and fixed-parameter intractable, can we get improved approximation factors in parameterized time that is not attainable by exact computation? This line of work has brought a wealth of positive and negative results, building upon the toolboxes of (hardness of) approximation and of parameterized complexity.

Online Algorithms

The online algorithm paradigm models situations where the problem instance arrives in form of a request sequence, and the algorithm needs to serve each request immediately, without having knowledge about future requests. Algorithms' performance is then compared with that of an ideal *offline* algorithm having full information about the problem instance in advance. The resulting *competitive ratio* thus measures the price of not knowing the future.

The paging problem, the secretary problem, the prophet inequality problem, the "rent or buy" problem, and the cowpath problem are well-studied examples of an online problem. Many combinatorial optimization problems have their online counterpart.

The online setting differs from that of streaming algorithms, where the algorithms need to store information in small memory to answer a final query. It also differs from robust optimization, which is typically a two stage setting, where a decision has to be made in

the first phase, in such a way that changes in the instance during the second phase can be handled without large cost.

In 2011 [Emek, Fraigniaud, Korman, Rosén] introduced "with advice complexity", asking questions like, how many bits of "additional information" about the whole sequence are needed per request so as to guarantee a particular competitive ratio? In 2018, [Purohit, Svitkina, Kumar] and shortly later [Angelopoulos, Dürr, Jin, Kamali, Renault] studied a model where the algorithm is provided with a prediction on future requests, which could be learned from past instances, and which can have errors. The performance of an ideal online algorithm should degrade smoothly with the error. Such algorithms and matching lower bounds have been developed for many the important online problems.

Randomized Optimization Heuristics

Randomized optimization heuristics (ROHs) such as simple randomized local search algorithms, simulated annealing, evolutionary algorithms, or ant colony optimizers can be traced back to Turing's "learning machine" from 1950. They gained popularity in the 1970s, where in particular genetic algorithms were introduced as a means to reliably optimize settings with very limited problem-specific knowledge. Since then, ROHs thrive as general-purpose optimizers in such black-box settings, making use of more and more complex operators [27, 41, 75].

The majority of the research conducted on ROHs being empirical, their theoretical analysis was pioneered by Ingo Wegener and his students in the 1990s [36, 37, 48]. Initial theoretical results considered very simple algorithms that could be analyzed via elementary tools from probability theory, such as the coupon collector theorem [35]. Soon it was observed that already for the analysis of simple heuristics, the classic toolbox from randomized algorithms does not suffice. A prominent example is the analysis of the runtime of the (1+1) evolutionary algorithms on pseudo-Boolean linear functions, which led to the invention of a set of methods now known as *drift analysis* [29, 30, 44], which allow to translate information on the one-step progress into estimates for the runtime.

Over the years, the mathematical tools for analyzing ROHs became considerably more refined, allowing for the analysis of increasingly complex ROHs and developing complexity-theoretic models. Key results gave very precise estimates for algorithms' expected runtimes [47], they proved guarantees for entire generic classes of operators [61], and they studied dynamically adapting ROHs [59]. This increased body of knowledge also led to the design of new ROHs that were provably better than their predecessors [31, 33, 57]. In particular, in the evolutionary computation community, theory has become a cornerstone of the field [34]. However, theoretical work on ROHs is now also regularly present at the large AI conferences [15, 25, 28, 32, 63, 82].

Quantum Algorithms

In 2006, the 9th edition of the largest international conference devoted to the study of Quantum Information Processing (QIP) was held in Paris. The conference gathered an audience of about 200 participants and featured 40 talks selected from 160 submissions. Over the past decade, attendance has grown to around 1,000 participants annually, with the program now comprising 3 parallel sessions and approximately 130 presentations selected by an international committee. For the 29th edition, in 2026, the program committee consists of 130 members and received about 700 submissions. What has changed?

Since 2006, theoretical progress in quantum algorithms has accelerated, broadening both

research directions and applications. The Harrow–Hassidim–Lloyd (HHL) algorithm [43], introduced in 2009, provided the first quantum method for solving linear systems with exponential improvement in the system dimension, leading to developments in quantum machine learning, including clustering [54], classification [68], and recommendation systems [56]. In addition, generic algorithmic primitives such as quantum walks [74] and quantum Monte Carlo techniques [42] have been refined and applied to problems in optimization [5], sampling, backtracking [67], search [66], and simulation [14]. At the same time, results on oracle separations have clarified the boundaries between classical and quantum computation, providing a rigorous foundation for identifying tasks where quantum algorithms offer provable advantages, for instance in terms of query complexity [2], space complexity [49] and time complexity [1].

These advances have influenced both national research agendas and industrial initiatives. The NIST post-quantum cryptography standardization process has emphasized the practical relevance of quantum algorithms for information security, while national and industrial programs worldwide have supported research in quantum computing. As a result, the current algorithms community includes both researchers focused on fundamental theoretical developments and those working on applied use cases, bridging physics, computer science, and potential industrial applications. Area of applications include simulation of quantum physics for quantum chemistry and materials simulation, post-quantum cryptography and cryptanalysis, and more generally optimization through quantum heuristics such as quantum variational approaches [38]. Some of the proposed solution even include hybrid quantum–hpc approaches [50].

Algorithmic Game Theory

Algorithmic Game Theory lies at the intersection of computer science, economics, and mathematics, focusing on the computational aspects of strategic decision-making and equilibrium analysis. Its foundations were laid in the early 2000s, marked by three seminal papers that were later recognized with the Gödel Prize: “Worst-case equilibria” by Koutsoupias and Papadimitriou, “How bad is selfish routing?” by Roughgarden and Tardos, and “Algorithmic Mechanism Design” by Nisan and Ronen.

Recent advances have deepened our understanding of the algorithmic complexity of finding equilibria, particularly Nash equilibria in large-scale and dynamic games. Researchers have developed efficient approximation algorithms and equilibrium computation techniques for specific game classes, such as congestion, auction, and network games. The rise of online platforms and decentralized systems (Adwords, etc) has further motivated the study of algorithmic mechanisms that ensure efficiency, fairness, and incentive compatibility under computational constraints.

Modern research in Algorithmic Game Theory increasingly integrates machine learning, data-driven modeling, and mechanism design to address real-world challenges in markets, networks, and multi-agent systems. Deep learning-based approaches now allow agents to learn strategies and equilibria in environments too complex for explicit analytical solutions. Meanwhile, algorithmic mechanism design continues to evolve toward robust and adaptive frameworks that account for uncertainty, partial information, and dynamic participation. This convergence of learning, optimization, and strategic reasoning positions Algorithmic Game Theory as a foundational field for understanding and engineering intelligent, self-organizing systems in economics, artificial intelligence, and beyond.

Combinatorial Reconfiguration

Combinatorial reconfiguration is an emerging discipline in theoretical computer science that, since the late 2000s, has led to a systematic study of algorithmic and structural questions about so-called *solution graphs*. For an instance of some combinatorial problem, such as the k-coloring problem, the solution graph has as vertices all solutions to the instance (e.g., k-colorings of a given graph) and two solutions are adjacent whenever they are sufficiently similar. Typically, two solutions are similar, if one can be obtained from the other by a simple modification; for k-colorings, a popular modification is to change the color of a single vertex. Solution graphs are relevant in many applications, such as random generation of combinatorial structures, enumeration, combinatorial games, motion planning, statistical physics, and bioinformatics. However, they are are usually too large to be constructed explicitly. A typical question in combinatorial reconfiguration is whether two combinatorial structures are equivalent in the sense that one can be transformed into the other by applying a sequence of small modifications. Such a sequence of modifications corresponds to a path in a solution graph. We may ask under which conditions the solution graph is connected or what is diameter is, that is, how many modifications suffice in order to transform one k-coloring into any other k-coloring.

The number of combinatorial structures that have been studied from the reconfiguration point of view in the last fifteen years or so is vast, so let us focus the aforementioned k-colorings. The general goal is to classify coloring problems based on the complexity of determining whether two colorings are equivalent. For example, deciding the equivalence of two 3-colorings is polynomial, even though deciding whether a graph admits a 3-coloring is NP-complete [20]. Meanwhile, determining if two 4-colorings of a graph are equivalent is PSPACE-complete [17]. An important insight is that there are certain topological reasons for whether two colorings are equivalent [81]. This insight has lead to a much better understanding of the complexity of testing the equivalence of graph homomorphisms (a generalization of k-colorings), but a complete classification currently seems out of reach [62, 81]. Besides such a classification, a major open problem in the area is the conjecture of Cereceda, which states that for $k \geq d+2$, the diameter of the graph of k-colorings of a d-degenerate graph G is quadratic in the number of vertices of G. The best current bound is polynomial [18].

3 Complexity

Classical Complexity Theory

Classical complexity theory has witnessed several landmark results. One standout achievement is Harvey and van der Hoeven's algorithm for integer multiplication in $O(n \log n)$ time, achieving the theoretically optimal bound for this fundamental operation. Another major breakthrough is Ryan Williams' work on simulating time with square-root space, which established a surprising relationship between time and space complexity, showing that certain time-bounded computations can be simulated using much less space than previously thought.

Fine-grained Complexity

Fine-grained complexity focus on the precise running times of algorithms for fundamental problems. The methodology of fine-grained complexity mimics the approach of NP-hardness: researchers select key problems conjectured to require a certain amount of time (such as k-SAT under the Strong Exponential Time Hypothesis (SETH)) and then use fine-grained reductions to transfer these hardness assumptions to other problems. This has led to a web

of tight conditional lower bounds. At its core the hardness of most of the studied problems in fine-grained complexity can be based on the presumed hardness of three key problems: the 3SUM problem, the All-Pairs-Shortest Paths (APSP) problem and k-SAT. The first problem is assumed to not have subquadratic algorithm, for the second one, the best algorithm is believed to be only cubic.The Strong Exponential Time Hypothesis states that for every $\varepsilon > 0$, there exists a constant k such that k-SAT (the satisfiability problem for conjunctive normal form formulas with k literals per clause) cannot be solved in time $O(2^{(1-\varepsilon)n})$, where n is the number of variables. In other words, SETH asserts that exhaustive search is essentially optimal for SAT, even for large k. Williams showed that this hypothesis implies that the Orthogonal Vectors problem (find in a set of vectors, two which are orthogonal) can not be solved by a subquadratic algorithm. Under these hypotheses, it is shown that the best algorithms for a large class of classical problems (edit distance, graph diameter, shortest cycle, planar motion planning, sequence local alignment, ...) can not be substantially improved.

Circuits Complexity

Circuit complexity has also seen significant progress over the past decade. Rossman, Servedio, and Tan established an average-case depth hierarchy theorem for Boolean circuits, demonstrating that circuits of increasing depth can solve more problems on average, even when size is restricted. Monotone circuits (i.e., without NOT gates) are a fundamental model where exponential lower bounds are known. Pitassi and Robere improved these results by obtaining strongly exponential lower bounds for them, by developing a refined version of the lifting theorems. In the arithmetic realm, Limaye, Srinivasan, and Tavenas showed superpolynomial lower bounds against low-depth algebraic circuits, the similar question has been known for Boolean circuits for 50 years, but this question remained unresolved in the arithmetic case. In fact arithmetic circuits of constant depth seem more powerful. This idea has been strengthened by Andrews and Wigderson's work, arithmetic circuits of constant-depth are able to efficiently compute the gcd of two polynomials, as well as the related problems of the discriminant, resultant, Bézout coefficients, squarefree decomposition.

Meta-complexity

The field of meta-complexity refers to the complexity of problems that are themselves about computational complexity. Central examples are the Minimum Circuit Size Problem (MCSP), which asks for the smallest circuit computing a given Boolean function the problem of determining the Kolmogorov complexity of a string. Research has revealed deep connections between MCSP and other fundamental areas, such as cryptography and learning theory. Notably, quasi-polynomial time algorithms for PAC-learning constant-depth circuits with parity gates were developed by Carmosino, Impagliazzo, Kabanets, and Kolokolova, and new worst-case to average-case reductions for NP problems were established by Hirara.

Query complexity

Query complexity measures the number of queries to an input required to solve a task when access is restricted. For example, sorting a size-n array requires at least $n \log n$ comparisons (queries) to the elements, though it can be faster with direct access. It has also been extensively studied for its deep connections to combinatorial structures. It also has potential impact on data queries in the cloud. Prior to 2006, the relationships among measures of query complexity for total Boolean functions were only partially understood. Deterministic query complexity $D(f)$ could exceed randomized complexity $R(f)$, with the best explicit separation

realized for the recursive NAND tree: $R(f) = \Theta(D(f)^{0.753\cdots})$ [70]. Sensitivity $s(f)$ was known to relate to block sensitivity $bs(f)$, but the *sensitivity conjecture*, asserting a polynomial relationship between $s(f)$ and $bs(f)$ and thus $D(f)$, remained unresolved. Since 2006, Huang's proof established this relationship [45], while pointer functions provide the strongest known explicit separations for total functions: $D(f) = \Omega(n/\log n)$, $R_0(f) = \tilde{O}(\sqrt{n})$, and $Q(f) = \tilde{O}(n^{1/4})$ [4]. These results both clarify the structural role of sensitivity and furnish explicit constructions demonstrating near-optimal gaps among deterministic, randomized, and quantum query complexities, forming a foundational toolkit for lower-bound techniques.

(Quantum) Communication complexity

Communication complexity is a fundamental framework for proving lower bounds across a variety of computational models. Early applications include circuit depth, formula size, VLSI complexity, and time–space tradeoffs. Over the past two decades, it has found new applications in establishing lower bounds and tradeoffs for computational models designed for massive data, while also incorporating techniques from information theory. Some information-theoretic methods were also inspired by quantum information, in particular by non-local games [55]. In a different vein, quantum communication complexity separations developed to distinguish quantum from classical models, such as the *hidden matching problem* [13], have led to further unexpected results. This separation was use to prove space lower bounds for streaming algorithms solving graph problems such as *maximum matching* [6]. More generally, the reduction of communication complexity, and more generally of information complexity [19], to other models, has provided explicit lower bounds for *streaming algorithms* (e.g., statistics estimation [58]), for *distributed computing* tasks(e.g. minimum spanning tree, shortest paths, and minimum cut [26]) and for *property testing* problems (e.g., monotone functions and the class of sparse polynomials [16]). These methods unify lower-bound arguments across deterministic, randomized, and quantum models.

Applications to Learning

Information-theoretic and communication complexity techniques have been instrumental in establishing space-sample tradeoffs for PAC learning. In memory-bounded settings, any one-pass algorithm that learns a class of functions from m labeled examples must use memory proportional to the communication complexity of an associated hard problem. This implies, for instance, that learning parity functions or certain conjunctions with sublinear memory requires an exponential number of samples. These tradeoffs have direct ramifications in algorithm design, limiting what can be learned efficiently in streaming or online settings, and in privacy-preserving learning, since the memory constraints effectively restrict the amount of information retained about individual data points, providing inherent privacy guarantees.

Proof Complexity

Proof complexity studies proofs as computational objects, asking how hard it is to certify that a statement is true. By analyzing the size and structure of proofs in formal systems, it sheds light on the limits of efficient reasoning and their connection to algorithmic complexity. Since 2006, proof complexity has developed into a rich and cross-disciplinary research program linking propositional proofs, circuit complexity, algebraic methods, optimization, and communication-based techniques. Grochow and Pitassi's Ideal Proof System [40] marked an important step toward unifying algebraic proof systems with circuit complexity, connecting algebraic-circuit lower bounds, polynomial identity testing, and the hardness of proving

tautologies. A major milestone established by Atserias and Müller showed that Resolution, a fundamental propositional proof system underlying many SAT solvers, is not automatizable unless $\mathsf{P} = \mathsf{NP}$ [7], resolving a decades-old question about whether short proofs can be found efficiently. This breakthrough spurred a robust line of work extending NP-hardness of automatizability to other systems. In parallel, the Sum-of-Squares/Lasserre proof system emerged as a central bridge between optimization, algorithms, and proof complexity, yielding deep results on both algorithmic power and proof-size lower bounds, in particular for constraint satisfaction and planted problems. Finally, a family of lifting and query-to-communication theorems has provided a versatile framework for transferring query-complexity lower bounds into communication and proof-complexity separations, producing strong lower bounds across multiple proof systems.

Coordination

Carola Doerr (LIP6, Paris) and Alantha Newman (LIP, Lyon).

Contributors

Benjamin Bergougnoux (LIS, Marseille), Marthe Bonamy (LaBRI, Bordeaux), Edouard Bonnet (LIP, Lyon), Monika Csikos (IRIF, Paris), Benjamin Doerr (LIX, Ecole Polytechnique, IP Paris), Christoph Dürr (LIP6, Paris), Laurent Feuilloley (LIRIS, Lyon), Joanna Fijalkow (LaBRI, Bordeaux), Pierre Fraigniaud (IRIF, Paris), George Giakkoupis (IRISA, Rennes), Bruno Grenet (LJK, Grenoble), Sophie Huiberts (LIMOS, Clermont-Ferrand), Martin S. Krejca (LIX, Ecole Polytechnique, IP Paris), Sophie Laplante (IRIF, Paris), William Lochet (LIRMM, Montpellier), Frédéric Magniez (IRIF, Paris), Mathieu Mari (LIRMM, Montpellier), Simon Mauras (INRIA, Paris-Saclay), Moritz Muhlenthaler (G-SCOP, Grenoble), Nicolas Nisse (Inria et I3S, Sophia Antipolis), Adi Rosén (IRIF, Paris), Bertrand Simon (LIG, Grenoble), Tatiana Starikovskaya (DIENS, Paris), Sebastien Tavenas (LAMA, Chambery), Nguyen Kim Thang (LIG, Grenoble), Adrian Vladu (IRIF, Paris).

References

1. Scott Aaronson. BQP and the polynomial hierarchy. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, STOC '10, page 141–150, New York, NY, USA, 2010. Association for Computing Machinery.
2. Scott Aaronson and Andris Ambainis. Forrelation: A problem that optimally separates quantum from classical computing. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '15, page 307–316, New York, NY, USA, 2015. Association for Computing Machinery.
3. Xavier Allamigeon, Daniel Dadush, Georg Loho, Bento Natura, and László A Végh. Interior point methods are not worse than simplex. *SIAM Journal on Computing*, 54(5):FOCS22–178, 2025.
4. Andris Ambainis, Kaspars Balodis, Aleksandrs Belovs, Troy Lee, Miklos Santha, and Juris Smotrovs. Separations in query complexity based on pointer functions. *J. ACM*, 64(5), September 2017.
5. Simon Apers and Ronald de Wolf. Quantum speedup for graph sparsification, cut approximation, and laplacian solving. *SIAM Journal on Computing*, 51(6):1703–1742, 2022.
6. Sepehr Assadi and Janani Sundaresan. Hidden permutations to the rescue: Multi-pass streaming lower bounds for approximate matchings. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 909–932, 2023.

7 Albert Atserias and Moritz Müller. Automating resolution is np-hard. *J. ACM*, 67(5):31:1–31:17, 2020.

8 Kyriakos Axiotis, Aleksander Mądry, and Adrian Vladu. Circulation control for faster minimum cost flow in unit-capacity graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 93–104. IEEE, 2020.

9 Kyriakos Axiotis, Aleksander Mądry, and Adrian Vladu. Faster sparse minimum cost flow by electrical flow localization. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 528–539. IEEE, 2022.

10 Eleon Bach, Alexander E. Black, Sophie Huiberts, and Sean Kafer. Beyond smoothed analysis: Analyzing the simplex method by the book, 2026. To appear in STOC.

11 Eleon Bach and Sophie Huiberts. Optimal smoothed analysis of the simplex method. In *Proceedings of the 66th Annual Symposium on Foundations of Computer Science (FOCS)*, 2025.

12 Nikhil Bansal and Subhash Khot. Optimal long code test with one free bit. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 453–462. IEEE, 2009.

13 Ziv Bar-Yossef, T. S. Jayram, and Iordanis Kerenidis. Exponential separation of quantum and classical one-way communication complexity. *SIAM Journal on Computing*, 38(1):366–384, 2008.

14 Dominic W. Berry and Andrew M. Childs. Black-box hamiltonian simulation and unitary implementation. *Quantum Info. Comput.*, 12(1–2):29–62, January 2012.

15 Chao Bian, Shengjie Ren, Miqing Li, and Chao Qian. An archive can bring provable speed-ups in multi-objective evolutionary algorithms. In *International Joint Conference on Artificial Intelligence, IJCAI 2024*, pages 6905–6913. ijcai.org, 2024.

16 Eric Blais, Joshua Brody, and Kevin Matulef. Property testing lower bounds via communication complexity. *Comput. Complex.*, 21(2):311–358, June 2012.

17 Paul Bonsma and Luis Cereceda. Finding paths between graph colourings: PSPACE-completeness and superpolynomial distances. *Theoretical Computer Science*, 410(50):5215–5226, 2009.

18 Nicolas Bousquet and Marc Heinrich. A polynomial version of cereceda's conjecture. *J. Comb. Theory B*, 155:1–16, 2022.

19 Mark Braverman. Interactive information complexity. *SIAM Journal on Computing*, 44(6):1698–1739, 2015.

20 Luis Cereceda, Jan van den Heuvel, and Matthew Johnson. Finding paths between 3-colorings. *Journal of Graph Theory*, 67(1):69–82, 2011.

21 Li Chen, Rasmus Kyng, Yang Liu, Richard Peng, Maximilian Probst Gutenberg, and Sushant Sachdeva. Maximum flow and minimum-cost flow in almost-linear time. *Journal of the ACM*, 72(3):1–103, 2025.

22 Michael B Cohen, Aleksander Mądry, Piotr Sankowski, and Adrian Vladu. Negative-weight shortest paths and unit capacity minimum cost flow in $\tilde{O}(m^{10/7} \log w)$ time. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 752–771. SIAM, 2017.

23 Daniel Dadush and Sophie Huiberts. A friendly smoothed analysis of the simplex method. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 390–403, 2018.

24 Daniel Dadush, Zhuan Khye Koh, Bento Natura, Neil Olver, and László A Végh. A strongly polynomial algorithm for linear programs with at most two nonzero entries per row or column. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 1561–1572, 2024.

25 Duc-Cuong Dang, Andre Opris, Bahare Salehi, and Dirk Sudholt. A proof that using crossover can guarantee exponential speed-ups in evolutionary multi-objective optimisation. In *Conference on Artificial Intelligence, AAAI 2023*, pages 12390–12398. AAAI Press, 2023.

26 Atish Das Sarma, Stephan Holzer, Liah Kor, Amos Korman, Danupon Nanongkai, Gopal Pandurangan, David Peleg, and Roger Wattenhofer. Distributed verification and hardness of distributed approximation. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, page 363–372, New York, NY, USA, 2011. Association for Computing Machinery.

27 Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6:182–197, 2002.

28 Anh Viet Do, Aneta Neumann, Frank Neumann, and Andrew M. Sutton. Rigorous runtime analysis of MOEA/D for solving multi-objective minimum weight base problems. In *Advances in Neural Information Processing Systems, NeurIPS 2023*, pages 36434–36448, 2023.

29 Benjamin Doerr and Leslie A. Goldberg. Adaptive drift analysis. *Algorithmica*, 65:224–250, 2013.

30 Benjamin Doerr, Daniel Johannsen, and Carola Winzen. Multiplicative drift analysis. *Algorithmica*, 64:673–697, 2012.

31 Benjamin Doerr and Martin S. Krejca. Significance-based estimation-of-distribution algorithms. *IEEE Transactions on Evolutionary Computation*, 24:1025–1034, 2020.

32 Benjamin Doerr, Martin S. Krejca, and Andre Opris. Tight runtime guarantees from understanding the population dynamics of the GSEMO multi-objective evolutionary algorithm. In *International Joint Conference on Artificial Intelligence, IJCAI 2025*, pages 8876–8884. ijcai.org, 2025.

33 Benjamin Doerr, Huu Phuoc Le, Régis Makhmara, and Ta Duy Nguyen. Fast genetic algorithms. In *Genetic and Evolutionary Computation Conference, GECCO 2017*, pages 777–784. ACM, 2017.

34 Benjamin Doerr and Frank Neumann, editors. *Theory of Evolutionary Computation—Recent Developments in Discrete Optimization*. Springer, 2020. Also available at `http://www.lix.polytechnique.fr/Labo/Benjamin.Doerr/doerr_neumann_book.html`.

35 Stefan Droste, Thomas Jansen, and Ingo Wegener. On the optimization of unimodal functions with the $(1 + 1)$ evolutionary algorithm. In *Parallel Problem Solving from Nature, PPSN 1998*, pages 13–22. Springer, 1998.

36 Stefan Droste, Thomas Jansen, and Ingo Wegener. A rigorous complexity analysis of the (1+1) evolutionary algorithm for linear functions with Boolean inputs. In *International Conference on Evolutionary Computation, ICEC 1998*, pages 499–504. IEEE, 1998.

37 Stefan Droste, Thomas Jansen, and Ingo Wegener. A rigorous complexity analysis of the $(1 + 1)$ evolutionary algorithm for separable functions with Boolean inputs. *Evolutionary Computation*, 6:185–196, 1998.

38 Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Leo Zhou. The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size. *Quantum*, 6:759, July 2022.

39 Yu Gao, Yang Liu, and Richard Peng. Fully dynamic electrical flows: Sparse maxflow faster than goldberg–rao. *SIAM Journal on Computing*, (0):FOCS21–85, 2023.

40 Joshua A. Grochow and Toniann Pitassi. Circuit complexity, proof complexity, and polynomial identity testing: The ideal proof system. *J. ACM*, 65(6):37:1–37:59, 2018.

41 Ryoki Hamano, Shota Saito, Masahiro Nomura, and Shinichi Shirakawa. CMA-ES with margin: lower-bounding marginal probability for mixed-integer black-box optimization. In *Genetic and Evolutionary Computation Conference, GECCO 2022*, pages 639—647. ACM, 2022.

42 Yassine Hamoudi and Frédéric Magniez. Quantum Chebyshev's Inequality and Applications. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 69:1–69:16, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

43 Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

44 Jun He and Xin Yao. Drift analysis and average time complexity of evolutionary algorithms. *Artificial Intelligence*, 127:51–81, 2001.

45 Hao Huang. Induced subgraphs of hypercubes and a proof of the sensitivity conjecture. *Annals of Mathematics*, 190(3), November 2019.

46 Sophie Huiberts, Yin Tat Lee, and Xinzhi Zhang. Upper and lower bounds on the smoothed complexity of the simplex method. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 1904–1917, 2023.

47 Hsien-Kuei Hwang, Alois Panholzer, Nicolas Rolin, Tsung-Hsi Tsai, and Wei-Mei Chen. Probabilistic analysis of the (1+1)-evolutionary algorithm. *Evolutionary Computation*, 26:299–345, 2018.

48 Thomas Jansen and Ingo Wegener. On the analysis of evolutionary algorithms – a proof that crossover really can help. In *European Symposium on Algorithms, ESA 1999*, pages 184–193. Springer, 1999.

49 John Kallaugher, Ojas Parekh, and Nadezhda Voronova. Exponential quantum space advantage for approximating maximum directed cut in the streaming model. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, STOC 2024, page 1805–1815, New York, NY, USA, 2024. Association for Computing Machinery.

50 Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, September 2017.

51 Anna R Karlin, Nathan Klein, and Shayan Oveis Gharan. A (slightly) improved approximation algorithm for metric tsp. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 32–45, 2021.

52 Narendra Karmarkar. A new polynomial-time algorithm for linear programming. In *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, pages 302–311, 1984.

53 Tarun Kathuria, Yang P Liu, and Aaron Sidford. Unit capacity maxflow in almost m^4/3 time. *SIAM Journal on Computing*, 53(6):FOCS20–175, 2022.

54 Iordanis Kerenidis, Jonas Landman, Alessandro Luongo, and Anupam Prakash. q-means: A quantum algorithm for unsupervised machine learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

55 Iordanis Kerenidis, Sophie Laplante, Virginie Lerays, Jérémie Roland, and David Xiao. Lower bounds on information complexity via zero-communication protocols and applications. *SIAM Journal on Computing*, 44(5):1550–1572, 2015.

56 Iordanis Kerenidis and Anupam Prakash. Quantum Recommendation Systems. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 49:1–49:21, Dagstuhl, Germany, 2017. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

57 Martin S. Krejca and Carsten Witt. A flexible evolutionary algorithm with dynamic mutation rate archive. *Algorithmica*, 88:1–32, 2025.

58 Ravi Kumar, T. S. Jayram, Ziv Bar-Yossef, and D. Sivakumar. An Information Statistics Approach to Data Stream and Communication Complexity . In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, page 209, Los Alamitos, CA, USA, November 2002. IEEE Computer Society.

59 Jörg Lässig and Dirk Sudholt. Adaptive population models for offspring populations and parallel evolutionary algorithms. In *Foundations of Genetic Algorithms, FOGA 2011*, pages 181–192. ACM, 2011.

60 Yin Tat Lee and Aaron Sidford. Path finding methods for linear programming: Solving linear programs in $O(\sqrt{\text{rank}})$ iterations and faster algorithms for maximum flow. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 424–433. IEEE, 2014.

61 Per Kristian Lehre and Carsten Witt. Black-box search by unbiased variation. *Algorithmica*, 64:623–642, 2012.

62 Benjamin Lévêque, Moritz Mühlenthaler, and Thomas Suzan. Reconfiguration of digraph homomorphisms. *SIAM J. Discret. Math.*, 39(1):327–360, 2025.

63 Mingfeng Li, Qiang Zhang, Weijie Zheng, and Benjamin Doerr. Why popular MOEAs are popular: Proven advantages in approximating the Pareto front. In *Advances in Neural Information Processing Systems, NeurIPS 2025*, 2025. To appear.

64 Aleksander Madry. Navigating central path with electrical flows: From flows to matchings, and back. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 253–262. IEEE, 2013.

65 Aleksander Madry. Computing maximum flow with augmenting electrical flows. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 593–602. IEEE, 2016.

66 Frédéric Magniez, Ashwin Nayak, Jérémie Roland, and Miklos Santha. Search via quantum walk. *SIAM Journal on Computing*, 40(1):142–164, 2011.

67 Ashley Montanaro. Quantum-walk speedup of backtracking algorithms. *Theory of Computing*, 14(15):1–24, 2018.

68 Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113:130503, Sep 2014.

69 Tim Roughgarden, editor. *Beyond the Worst-Case Analysis of Algorithms*. Cambridge University Press, December 2020.

70 Michael Saks and Avi Wigderson. Probabilistic boolean decision trees and the complexity of evaluating game trees. In *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, pages 29–38, 1986.

71 Daniel Spielman and Shang-Hua Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 296–305, 2001.

72 Ola Svensson. Hardness of precedence constrained scheduling on identical machines. *SIAM Journal on Computing*, 40(5):1258–1274, 2011.

73 Ola Svensson, Jakub Tarnawski, and László A Végh. A constant-factor approximation algorithm for the asymmetric traveling salesman problem. *Journal of the ACM (JACM)*, 67(6):1–53, 2020.

74 M. Szegedy. Quantum speed-up of markov chain based algorithms. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 32–41, 2004.

75 Dirk Thierens and Peter A.N. Bosman. Optimal mixing evolutionary algorithms. In *Genetic and Evolutionary Computation Conference, GECCO 2011*, pages 617—-624. ACM, 2011.

76 Jan van den Brand, Yu Gao, Arun Jambulapati, Yin Tat Lee, Yang P Liu, Richard Peng, and Aaron Sidford. Faster maxflow via improved dynamic spectral vertex sparsifiers. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 543–556, 2022.

77 Jan Van Den Brand, Yin Tat Lee, Yang P Liu, Thatchaphol Saranurak, Aaron Sidford, Zhao Song, and Di Wang. Minimum cost flows, mdps, and ℓ_1-regression in nearly linear time for dense instances. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 859–869, 2021.

78 Jan van den Brand, Yin-Tat Lee, Danupon Nanongkai, Richard Peng, Thatchaphol Saranurak, Aaron Sidford, Zhao Song, and Di Wang. Bipartite matching in nearly-linear time on moderately dense graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 919–930. IEEE, 2020.

79 Adrian Vladu. Breaking the barrier of self-concordant barriers: Faster interior point methods for m-matrices. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 2213–2224, 2025.

80 Stephen J. Wright. Optimization in theory and practice, 2025.

81 Marcin Wrochna. Homomorphism reconfiguration via homotopy. *SIAM J. Discret. Math.*, 34(1):328–350, 2020.

82 Weijie Zheng and Benjamin Doerr. Mathematical runtime analysis for the non-dominated sorting genetic algorithm II (NSGA-II). *Artificial Intelligence*, 325:104016, 2023.

Twenty Years of GdR IFM, seen from GT Graphs

At the turn of the 2000s, graph theory experienced major advances. In structural graph theory, the strong perfect graph conjecture had just been proven, and the publication of the Graph Minors series was nearing its end. In graph algorithms, the first applications of Courcelle's Theorem were emerging, and the field of parameterized algorithms was also gaining momentum.

The *Strong Perfect Graph Theorem* [25] characterizes graphs G for which the chromatic number $\chi(G)$ equals the size of their largest clique $\omega(G)$, and for which all induced subgraphs preserve this property. The chromatic number $\chi(G)$ is the smallest number of parts in a partition of the vertices of G such that each edge has its endpoints in distinct parts. It is immediate that $\chi(G) \geq \omega(G)$ for any graph G. This theorem thus characterizes the graphs for which this bound is tight.

The *Graph Minors series* by Robertson and Seymour is an in-depth study of classes of graphs that are closed under the minor relation. This means that in such a class $\mathcal{C}$, for any graph $G \in \mathcal{C}$, deleting a vertex, deleting an edge, or contracting an edge produces a graph that also belongs to $\mathcal{C}$. This is, for example, the case for the class of planar graphs—those that can be drawn in the plane without any edge crossings. One of the highlights of this series is the proof of Wagner's Conjecture: the minor relation is a well-quasi-ordering. This means that every minor-closed class of graphs $\mathcal{C}$ can be characterized by a finite set of excluded minors.

Courcelle's Theorem [31] guarantees that any property Q that can be expressed in a certain logical language — the monadic second-order logic of graphs — and that concerns graphs of tree-width at most t, admits an algorithm running in time $f(Q,t) \cdot |G|$, where G is the input graph and f is some function. Tree-width is a graph invariant that, in a sense, measures how many vertices need to be removed to split a graph—or any of its subgraphs—into smaller pieces (each piece containing at most two-thirds of the original vertices).

Parameterized Complexity is a response to the proven difficulty of many algorithmic problems—the so-called NP-hard problems. For such problems, it is believed that no algorithm can solve them in polynomial time, that is, in time $O(|G|^c)$, where G is the input graph and $c > 0$ is a constant. This approach proposes, for any graph problem Π, to determine the graph parameters $p(G)$ for which there is an algorithm solving Π in time $f(p(G)) \cdot |G|^c$, for some function f and value $c > 0$. This allows characterizing sub-classes of graphs for which Π can be solved efficiently.

Since 2006, graph theory has witnessed an impressive number of works and advances. Given this abundance, it is impossible to provide an exhaustive overview. Likewise, it is difficult to single out the most significant developments. Nevertheless, we will venture to do so: starting from the four major milestones we have just discussed, we will explore in Section 1 how they have nourished our field and fostered the emergence of new theories, powerful tools, and remarkable results. Next, in Section 2, we will look at examples of new tools or new models originating from other fields, which have had a significant impact and that influenced the evolution of our research questions.

1 Continuation of major works

1.1 Perfect graphs and χ-boundedness

Understanding why a graph has large chromatic number is a central question in graph theory, and one that most likely does not admit a simple answer. It is clear that a graph containing a large clique (that is a large complete graph) must have a large chromatic number. But the converse is not always true. It has been known since Erdős [48] that there exist triangle-free graphs (and even graphs with no short cycles) with arbitrarily large chromatic number. These examples show that the difficulty of coloring cannot always be explained by local properties.

The strong perfect graph theorem [25] (solved exactly 20 years ago) tells us that a chromatic number strictly larger than the clique number forces the presence of an induced odd cycle in the graph or its complement. In his foundational paper [58], Gyárfás went further and asked which structures must necessarily appear when the gap between $\chi(G)$ and $\omega(G)$ becomes arbitrarily large. He introduced the notion of a χ-bounded class of graphs and formulated several conjectures that have guided subsequent research, including a number concerning the cycle lengths that must occur.

These conjectures remained out of reach for a long time, but the past twenty years have witnessed spectacular progress, starting with the resolution of a special case by Bonamy, Charbit and Thomassé in 2014 [11]. In 2016, Scott and Seymour proved that graphs with no odd hole are indeed χ-bounded [91], that is, the absence of odd holes suffices to bound the chromatic number as a function of the clique number. In 2019, with Chudnovsky and Spirkl, they subsequently established a stronger result: forbidding induced long cycles, or even only induced long odd cycles, also suffices [26, 27].

Among the famous conjectures of Gyárfás, one has resisted attempts for the past forty years: the *Gyárfás–Sumner conjecture* [58, 93], which asserts that for every tree T, the class of graphs with no induced copy of T is χ-bounded. The conjecture is now proved for many specific families of trees (paths, stars, trees of radius two, etc.), and the methods developed to attack it have spread much further, notably in algorithmics (the so-called *Gyárfás path argument*). But the conjecture remains open, and nothing guarantees that a counterexample will not eventually be found. The Gyárfás–Sumner conjecture is still actively studied, in particular by members of the GT Graphs group, both in its original form and in various extensions, such as to signed graphs [2] and oriented graphs [1].

Interestingly, the Gyárfás path argument has also led to new advances on the Erdős–Hajnal conjecture [49], another major open problem in structural graph theory. This conjecture states that forbidding a single graph as an induced subgraph should suffice to move far away from the standard behavior of random graphs, where the largest clique and stable set are only logarithmic in size, and instead guarantee the existence of a clique or a stable set of polynomial size. Perfect graphs, for instance, always contain a clique or a stable set of size $\Omega(\sqrt{n})$. An alternative direction that has proved fruitful is to forbid two (families of) graphs rather than just one, in order to gain additional structure: for example, forbidding a path and its complement [21], or forbidding long cycles and their complements or forbidding just one cycle of fixed length together with its complement [10], or a forest together with its complement [28], already ensures the existence of a very large clique or a very large stable set.

1.2 Sparse Graphs

Sparsity vs denseness is a natural dichotomy when studying the structure of mathematical objects. However, there is not a unique definition of what a sparse graph would look like. While there might be many notions of sparsity, they all agree on some monotonicity conditions. For instance, one can consider as sparse any graph G whose maximum average degree, that is $\max_{H\subseteq G} \frac{1}{|V(H)|}\sum_{v\in V(H)} \deg_H(v)$ where H ranges over the subgraphs of G, is small. However, this definition while can be useful for some partition problems such as coloring problems, is not convenient in other context as simply subdividing every edge of a dense graph gives rise to a graph of small maximum average degree.

In the seminal work of Robertson and Seymour on classifying graph classes with respect to the minor quasi-ordering, the family of non-trivial minor-closed graph classes appear to meet many desired properties of sparsity : (1) the number of edges is linear, (2) they are hereditary w.r.t. minor, and (3) cannot contain all graphs as minors. Nonetheless, graphs of small degree are not minor-closed and still share many properties with non-trivial minor-closed graph classes, indicating better notions of sparsity. In their seminal work [79, 78, 81], rooting from the study of graph homomorphism problems, Nešetřil and Ossona de Mendez proposed to substitute r-shallow minor[1] to subgraph in the definition of maximum average degree. While the definition seems as for maximum average degree to be arbitrary, it allows to define a sparsity parameter for each value $r \geq 0$, ranging from the maximum average degree ($r = 0$) to parameters characterizing minor-closed classes ($r = \infty$). It happens also that this definition is effective enough that an equivalent hierarchy is obtained if we replace minor by other quasi-orders such as topological minor or immersion [82]. Using these parameters, an infinite lattice of sparse graph classes can be defined [82], the most prominent one being the familly of *bounded expansion* graph classes where the parameters are bounded by a function f. For graph classes where these parameters are not bounded by any function $f(r)$, one can distinguish between *nowhere dense* and *somewhere dense* graph classes. Here, the dense graph classes are those that contain all graphs in their r-shallow minors, for some r. Conversely, the *nowhere dense* classes are such that the clique-number of each r-shallow minor is bounded, but they can have super-linear number of edges, a property that cannot be true for non-trivial minor-closed graphs. The links of this class trichotomy with clique number have triggered many questions regarding their links with coloring problems, and several other characterizations of these classes have been proposed, the main structural ones being those based on *generalized coloring* [97], those based on *low tree-width/tree-depth coloring* [80], those based on variants of cops and robber games [56] and those based on *neighbourhood complexity* [86].

Another way of studying a graph class $\mathcal{C}$ is to understand the structure of the set of graphs not belonging to $\mathcal{C}$, and in particular the minimal ones w.r.t. a quasi-order and called obstructions. These obstructions are for instance used as certificates for recognition algorithms. For instance, any minor-closed class of graphs is characterized by a finite list of obstructions, allowing a polynomial recognition algorithm for any minor-closed graph class. Another line of research in studying sparse graph classes is to identify sets of obstructions for important graph properties. One can cite among such results the characterization of NIP and stable graph classes w.r.t. induced subgraph quasi-order [72], and the constructive proof computing the set of obstructions for having the Erdős-Pósa property[2] for any minor-closed

[1] Roughly an r-shallow minor is a minor where contraction are done only on connected subgraphs of diameter at most r.

[2] Having the Erdős-Pósa property w.r.t. a class $\mathcal{H}$, is, for a function f, having either k pairwise

class of graphs [84].

This field of research had important consequence for graph algorithms. It allowed extending the classes of graphs for which FO model checking can be performed efficiently. This is explained in the next subsection.

1.3 Meta-theorems

Algorithmic meta-theorems are general results that identify broad classes of problems which can be solved efficiently when the input graph meets certain structural conditions. Informally, they are expressed as follows:

> *"Every problem expressible in some language $\mathcal{L}$ can be efficiently solved for the graphs of some class $\mathcal{G}$."*

Courcelle's theorem was the first major statement of this form, taking $\mathcal{L}$ as the MSOL-definable properties[3], and $\mathcal{G} = \{G \mid tw(G) \leq t\}$. These last two decades, a *proliferation of algorithmic meta-theorems is reported.* Let us divide them in three groups, the Courcelle-like ones that are using a tree-based measure of graph complexity, the ones doing FO-model checking for wide classes of graphs, and those working in less general classes but for a richer languages $\mathcal{L}$.

Courcelle's theorem and the wide usage of tree-width and variants in several areas motivated the search of more structural parameters that could be used in structural graph theory, and for designing algorithms. This led to several such parameters, that are also defined through a tree-based structure. The most-well known ones are *clique-width* and its equivalent one *rank-width*, because they admit a Courcelle-like theorem, where the tractable problems are those MSO_1-definable, a fragment of MSOL. We should also mention *mim-width* [4], which was very successful with its many algorithmic properties and applications in SAT solvers [89, 22]. However, all these contributions arising from Courcelle's theorem have the disadvantage of being closely connected to tree-based measures. It has been a successful challenge to go beyond these classes of graphs.

Among the graph classes where the above mentioned measures can be arbitrary large, we have planar graphs, of all its generalisations: minor-closed graph classes, bounded expansion classes, and the nowhere dense ones (see Section 1.2). For those graph classes, the expressive power of $\mathcal{L}$ must be reduced, as many MSOL-definable problems are already NP-hard, when restricted to them. Considering sparse graph classes, researchers realized that for such classes, many problems can be solved efficiently, in particular those definable in *first-order logic* (FO for short). This was confirmed, as among graph classes closed under subgraphs, it was shown that FO model-checking is tractable if and only if the graph class is nowhere dense [56].

FO model-checking can also be performed in dense graph classes. The work initiated in [57] for studying parameterized problems in permutation graphs had been surprisingly generalized to all graphs and yielded the so-called *twin-width* parameter, defined through a degree notion based on linear order decomposition [18]. Small twin-width graphs include many studied graph classes, sparse and dense, including those being minor-closed, those with bounded clique-width and many geometric graph classes, being the first of this kind. It has been shown that FO model checking is also tractable for graphs of bounded twin-width, an impressive generalization towards dense graph classes [18].

---

vertex-disjoint subgraphs in $\mathcal{H}$, or having $f(k)$ vertices whose deletions gives rise to a graph in $\mathcal{H}$.

[3] *Monadic second-order logic* (MSOL for short) is a logical language that extends FO logic and allow to express many NP-complete problems such as k-coloring, for any fixed k, or Hamiltonicity.

There are attempts to generalize the graph classes for which FO model-checking is known tractable (nowhere dense classes and graphs with bounded twin-width) through new graph complexity measures, namely *flip-width* and *merge-width*. Another way to present these generalizations is an attempt to revisit the class trichotomy by Nesetril and Ossona de Mendez through the lens of FO model-checking. Here, the three considered families are (1) the structurally sparse ones which are those that are FO-transductions[4] of sparse graph classes (those are conjectured to be exactly the *stable* graph classes studied by Shelah, see for instance [15]), (2) the NIP graph classes which are those that cannot produce all graphs by FO-transductions (for those, the *FPT NIP conjecture* states they correspond to the hereditary graph classes with tractable FO model-checking), and (3) the graph classes that are not NIP. All these characterizations have connected the classification program of graph classes into sparse vs dense with the notions of *VC-dimension* appearing in learning theory [95, 96, 38] and of *neighborhood complexity* appearing in metric graph theory [16, 14].

1.4 Fine grained complexity and parameterized algorithms

Parameterized complexity is a framework that aims at analyzing the running times of algorithms in finer details than classical complexity theory: rather than expressing running times solely as a function of the input size, it also considers the dependence on one or more parameters of the input. These parameters may be related to the solutions (e.g., their size) or to the structure of the input (e.g., its tree-width). Hence, the meta-theorems cited above form a particular type of parameterized algorithms. Among the desired notions of tractability, three play a central role. To define these notions, let us assume we are dealing with a problem whose input size is n and parameter is k. (For example, the problem could be of deciding whether an input graph on n vertices and tree-width k contains a Hamiltonian path: here n is the input size, and k is the parameter.) The first notion is that of *slice-wise polynomial* (XP) algorithms whose running times are of the form $n^{f(k)}$ for some computable function f, which is polynomial for bounded values of k. The second notion is the one of *fixed-parameter tractable* (FPT) algorithms whose running times are of the form $f(k) \cdot n^{O(1)}$, which are expected to be more efficient than XP algorithms when the parameter gets large. For both notions, research has been conducted in obtaining the optimal such functions f, leading to the notion of fine grained complexity. Finally, the third notion is that of kernelization, where the goal is to reduce the instance into one whose size is bounded by a function of the parameter. Problems admitting kernels trivially admit FPT algorithms, and so obtaining kernels of small size is considered as a better degree of tractability.

While not being limited to graph theory, the framework of parameterized complexity has been extensively developed in the context of graphs, as these structures offer numerous parameters, structural properties, and decomposition features that can be exploited by parameterized algorithms. Among the most significant developments in graph theory over the past two decades, one can highlight the emergence of new algorithmic techniques, the development of meta-theorems, the improvement of complexity bounds, and the use of new structural parameters, that we briefly survey here.

Algorithmic techniques. Examples of recent successful algorithmic techniques include the *representative sets technique* [51] that led to efficient algorithms for "cycle-type"

[4] An FO-transduction is roughly a function from relational structures to relational structures, where the domains and relations of the target structures are defined by FO formulas on the input structures, after a non-deterministic k-coloring, for some fixed integer k, of the domains of the inputs.

problems, the *cut-and-count method* [32] that led to improved time bounds for a number of connectivity and acyclic problems, the rank-based approaches allowing deterministic algorithms for many cut-and-count based algorithms [6, 5] and *flow-augmentation techniques* for parametrized graph cut problems [66, 67, 68].

Kernelization. After a growing line of research consisting of characterizing parameterized problems admitting or not polynomial-size kernels, new techniques were found and have expanded the theoretical scope of kernelization. Examples of such successful techniques include the one of *cross-composition* introduced in [9], later applied in various context such as packing problems [36].

Meta-theorems. Algorithmic meta-theorems are general results that identify broad classes of problems which are fixed-parameter tractable (FPT) or admit small kernels when certain structural conditions are met. These last two decades, a proliferation of algorithmic meta-theorems is reported [37, 69], reaching the field of kernelization [8]. We already mentioned the large classes for which FO model checking can be performed efficiently. Further developments allow to cope with larger fragments of logic on fairly large graph classes [54].

Parameterized approximations. The topic of parameterized approximation aims at studying problems that cannot be solved exactly by an FPT algorithm, but that admit an approximation within the same time constraint. It has been growing since its initial stage [75], leading to a systematic study of classical problems such as knapsack problems [61] or domination-type problems [64].

Fine grained complexity. Since the work of Impagliazzo et al. on strong exponential time [60], fine grained complexity has brought a more nuanced view of computational hardness. Influential results include [71] related to tree-width, [24] related to domination-type problems, or [70] related to pathwidth. Fine grained complexity also considers determining for polynomial-time tractable problem such as computing the diameter of a graph, what is the optimal complexity of an algorithm solving this problem. A current trend in this domains aims at finding sub-quadratic time algorithms for the diameter problem [41].

2 New questions and new techniques

The last two decades have seen the emergence of many new questions arising from related domains, and of many new techniques. In the following we sample a few of them.

2.1 New models of algorithms

As we have already seen, graph theory naturally raises many algorithmic questions. Over the past twenty years, new algorithmic models have emerged, which the graph community has naturally embraced. In addition, the tools known in graph theory provide new elements for studying these new models. In this section, we briefly survey a few of them, from distributed computing and temporal graphs to enumeration problems through combinatorial games.

Distributed and local computing. Since a distributed network can be modeled as a graph, the distributed settings offer a new playground for graph algorithms, and the tools of graph theory can be used to improve the understanding of distributed computing. Let us take three concrete examples, among many.

1. Local algorithms have been designed for decades almost only in bounded-degree or general graphs, but thanks to the interaction with graph theorists, this has changed and

there are now distributed analogues of Courcelle's theorem for structured graphs [50] and efficient distributed approximation for minor-closed graphs [12].

2. A new point of view on graph classes has emerged through local certification, a notion stemming from the study of distributed fault-tolerance. It gives a new measure about how well one can check locally the structure of a graph. This has been done for example for minor-closed classes of graphs [20] and for geometric graph classes [35].
3. The blossoming notion of universal graph is at the intersection of graph theory, where it is a tool to understand a graph class, and of distributed computing, where it can serve for routing and for locally encoding adjacency. For example, in [13, 42], the authors uses the product structure theorem (see Section 2.3) to construct a universal graph that leads to good adjacency labelling schemes for planar graphs.

Temporal graphs. A temporal graph is a graph whose edges (and sometimes vertices) are present only at certain points in time. Temporal graphs occur naturally in transportation, communication networks, social networks, robotics, scheduling, and distributed computing. On the theoretical side, these graphs pose important challenges, as many classical concepts and techniques from standard graph theory do not carry on easily. For example, reachability based on temporal paths (paths crossing the edges chronologically) is neither symmetric nor transitive, with important algorithmic consequences. Two seminal papers documenting these effects are [7] and [65], showing respectively that computing a maximum temporal component (set of vertices that can reach each other) is NP-hard, and deciding if there exists two node-disjoint temporal paths between a given source and target is also NP-hard, both problems being polynomial-time solvable in static graphs. Another significant negative result is [3], showing that temporal graphs do not admit sparse spanners in general, i.e. there exist temporal graphs with $\Theta(n^2)$ edges, all of which are critical for connectivity. In recent years, a particular effort is devoted to understanding special cases where the above problems become tractable, for example, through the definition of temporal graph parameters that allows for FPT algorithms (see, e.g. [47]).

Combinatorial game theory. These games involve (generally) two players that take turns on a common board with perfect information. Typically, Alice and Bob alternately select vertices of an hypergraph, Alice wins if eventually she gets all vertices of an hyperedge and Bob wins otherwise [90]. This general definition of combinatorial games is intimately related to graphs and some of the most famous such games (Hex, Tic-tac-toe) are actually games played on grids. Recent progress have been done showing that such games are PSPACE-complete (resp., polynomial) when hyperedges have size 6 [85] (resp., 3 [52]). During the last decades, many combinatorial games defined through graphs have been studied [39, 40], leading to a better understanding of the complexity of these games and, in particular, of their many winning conventions (e.g., Client-Waiter [53]) and their link with reconfiguration [59]. The current research focuses on the limits of the tractability of these games, namely, given a game defined through graphs, in which graph classes it becomes polynomially solvable? Recently, these games have also been considered through the parameterized complexity point of view [17].

Algorithmic enumeration. Algorithmic enumeration is a cross-cutting theme that does not apply only to graphs. Many enumeration problems come from applications in BDD [73], bioinformatics [74], chemoinformatics, data mining, etc. The key issue remains whether it is possible to enumerate the minimal transversals of a hypergraph in output-polynomial time, the best time known to date being output-quasi-polynomial [46]. Nevertheless, there have been significant results in enumeration that have made it possible to solve algorithmic problems in graphs. For example, the Proximity Search technique [29] has

solved the enumeration of maximal subgraphs for a large number of classes admitting orders (chordal graphs, degenerate graphs, etc.), but is not restricted to graphs. Recently, new lines of research have emerged in enumeration: parameterized enumeration and approximate enumeration. We refer to the following survey presenting the computational complexity of enumeration problems and the status of many of them [92].

2.2 Probabilistic method and extremal graph theory

The probabilistic method is a central tool to prove results in (extremal) graph theory and prove the existence of some structures. Although these methods have been known for many years, recent advances have led to new results and proven certain conjectures. Here are a few examples illustrating activity in this area over the past twenty years.

- The constructive proof of the Lovasz's Local Lemma by Moser and Tardos in 2010 [77] – also known as the *entropy compression method* – has been widely used in graph theory, in particular to prove better bounds on colorings and, more generally, on the existence of certain structures in graphs. Recent refinements to this method [87] have yielded even more accurate results.
- In 2014, Bukh [23] presented a random algebraic method to obtain random constructions that are more rigid than with uniform random graphs. The analysis is more difficult and uses profound results from algebraic geometry. Another slightly different approach consists of starting with a rigid algebraic structure (a projective plane, for example), and then applying operations randomly. One of the most spectacular applications of this approach is the new lower bound on the Ramsey number $R(4,t)$ [76].
- The Kahn-Kalai conjecture, proposed in 2006, about the expectation threshold for random graphs has been proved in 2024 by Park and Pham [83]. The result and the technique introduced in the proof have already numerous consequences in probabilistic graph theory.
- Probabilities have also been used to make important progress on big conjectures of graph theory. For example, the Erdős–Faber–Lovász conjecture[5] has been proved in 2023 for large n using *Rödl's niddles* that are constructions of large matchings with an iterative probabilistic procedure [62].

2.3 Product structure theorem and applications

In 2019, Dujmović, Joret, Morin, Micek, Ueckerdt, and Wood proved an unexpected result, the so-called product structure theorem [44]. It asserts that every planar graph is a subgraph of a graph of the form $P \boxtimes H$, the strong product of a path and a graph of bounded tree-width. This structural theorem has been extended to several graph classes, such as graphs embedded on surfaces or graphs admitting an embedding with few crossings per edges.

This result allowed many advances on several longstanding open problems concerning planar graphs. Among others, it allowed proving that :

- Planar graphs have bounded queue-number [44], that is a vertex ordering and an edge partition into boundedly many queues (conjectured since 1992).
- Planar graphs can be colored non-repetitively with a constant number of colors [43]. In this type of coloring, for every even path P, the sequence of colors appearing along the

[5] This conjectures states that a graph made of n cliques of size at most n that intersect two-by-two on at most one vertex is n-colorable.

first half of P is distinct from the one of the second half. This problem was open since 2002.

Quantitatively, the product structure theorem allowed improvement on the number of colors for p-centered colorings [34], on the treedepth fragility [45], or on the size of the adjacency labelling schemes [42]. These results respectively improve the complexity of some parameterized algorithms, speed-up approximation algorithms, and reduce memory requirements in distributed algorithms.

2.4 Computer-assisted graph theory

One of the most famous results in graph theory, the Four Color Theorem, has a proof that contains several computer-assisted parts. With the increase of both memory capacity and processor efficiency (and parallelization), the magnitude of the possibilities for computer assistance has exploded in the last 20 years.

The computer assistance can be found in various contexts within graph theory. First, fast generators have been developed to generate all graphs from a given graph class exhaustively, taking the maximal size of a graph as an input. They are then used to test claims of the form "every graph with property P_1 also has property P_2", or to find the smallest counterexamples to such a claim. Secondly, graph databases (such as the House of Graphs) [30] are available for the researchers so that they can search for a graph with specific properties. Thirdly, various methods have been developed to test specific properties in graphs, which are usually computationally hard, for graphs of reasonable size. Those, are usually based on mixed integer linear program solvers, semi-definite programming solvers and SAT solvers.

Here are a few notable examples where computer assistance was essential to check a simple property many times.

- A famous problem of Hadwiger and Nelson asks for a minimum number of colors needed to color the plane such that no two points at distance 1 receive the same color. Colorings with seven colors are known, and examples of finite unit-distance graphs that need four colors have been known since the 60s. The lower bound was raised to five in 2018, when Aubrey de Grey found a 1581-vertex, non-4-colorable unit-distance graph, heavily relying on computer assistance to check the properties of the gadgets used in his construction [33].
- For the proof of the Barnette-Goodey conjecture [63] (stating that every 3-connected cubic planar graph with faces of size at most 6 is hamiltonian), the computer assistance was succesfully used to rule out a large number of rather small graphs.

There are also cases where the computer assistance lies more in the core of the proofs. For example, in [19] the authors used a routine based on linear programming to construct a set of "discharging rules" leading to progress on Wegner's conjecture about the distance 2-chromatic number of planar graphs with bounded maximum degree. In [88], the author uses the MSOL description of some graph properties, and multilinear algebra, to obtain tight asymptotic bounds on the number of many natural substructures of trees, with respect to their number of vertices n.

Proof assistants also have a history in our field. A little more than twenty years ago, a certified proof of the Four Color Theorem was obtained using Coq (now called Rocq). Today, LLMs enable better navigation in the literature, thereby enhancing researchers' work [55]. There are also a few examples of certified proofs that combine an LLM with a proof assistant [94]. It seems likely that these developments will have a major impact over the next twenty years.

Contributors.

Pierre Aboulker, Caroline Brosse, Arnaud Casteigts, Oscar Defrain, Louis Esperet, Daniel Gonçalves, Laurent Feuilloley, Mamadou Moustapha Kanté, Frantisek Kardos, Aurélie Lagoutte, Vincent Limouzy, Leandro Montero, Nicolas Nisse, Aline Parreau, Christophe Picouleau, Cléophée Robi, Ioan Todinca, Olivier Togni.

References

1 Pierre Aboulker, Pierre Charbit, and Reza Naserasr. Extension of Gyárfás-Sumner conjecture to digraphs. *The Electronic Journal of Combinatorics*, 28(2), May 2021.

2 Guillaume Aubian, Allen Ibiapina, Luis Kuffner, Reza Naserasr, Cyril Pujol, Cléophée Robin, and Huan Zhou. Extension of the gyárfás-sumner conjecture to signed graphs. 2025.

3 Kyriakos Axiotis and Dimitris Fotakis. On the size and the approximability of minimum temporally connected subgraphs. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016*, volume 55 of *LIPIcs*, pages 149:1–149:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. URL: https://doi.org/10.4230/LIPIcs.ICALP.2016.149, doi:10.4230/LIPICS.ICALP.2016.149.

4 Rémy Belmonte and Martin Vatshelle. Graph classes with structured neighborhoods and algorithmic applications. *Theor. Comput. Sci.*, 511:54–65, 2013. URL: https://doi.org/10.1016/j.tcs.2013.01.011, doi:10.1016/J.TCS.2013.01.011.

5 Benjamin Bergougnoux, Jan Dreier, and Lars Jaffke. A logic-based algorithmic meta-theorem for mim-width. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 3282–3304. SIAM, 2023. URL: https://doi.org/10.1137/1.9781611977554.ch125, doi:10.1137/1.9781611977554.CH125.

6 Benjamin Bergougnoux and Mamadou Moustapha Kanté. More applications of the d-neighbor equivalence: Acyclicity and connectivity constraints. *SIAM J. Discret. Math.*, 35(3):1881–1926, 2021. URL: https://doi.org/10.1137/20M1350571, doi:10.1137/20M1350571.

7 Sandeep Bhadra and Afonso Ferreira. Computing multicast trees in dynamic networks and the complexity of connected components in evolving graphs. *J. Internet Serv. Appl.*, 3(3):269–275, 2012. URL: https://doi.org/10.1007/s13174-012-0073-z, doi:10.1007/S13174-012-0073-Z.

8 Hans L Bodlaender, Fedor V Fomin, Daniel Lokshtanov, Eelko Penninkx, Saket Saurabh, and Dimitrios M Thilikos. (meta) kernelization. *Journal of the ACM (JACM)*, 63(5):1–69, 2016.

9 Hans L Bodlaender, Bart MP Jansen, and Stefan Kratsch. Kernelization lower bounds by cross-composition. *SIAM Journal on Discrete Mathematics*, 28(1):277–305, 2014.

10 Marthe Bonamy, Nicolas Bousquet, and Stéphan Thomassé. The Erdős–hajnal conjecture for long holes and antiholes. *SIAM Journal on Discrete Mathematics*, 30(2):1159–1164, 2016.

11 Marthe Bonamy, Pierre Charbit, and Stéphan Thomassé. Graphs with large chromatic number induce $3k$-cycles. *arXiv preprint arXiv:1408.2172*, 2014.

12 Marthe Bonamy, Cyril Gavoille, Timothé Picavet, and Alexandra Wesolek. Local constant approximation for dominating set on graphs excluding large minors. In Alkida Balliu and Fabian Kuhn, editors, *Proceedings of the ACM Symposium on Principles of Distributed Computing, PODC 2025, Hotel Las Brisas Huatulco, Huatulco, Mexico, June 16-20, 2025*, pages 77–87. ACM, 2025. URL: https://doi.org/10.1145/3732772.3733531, doi:10.1145/3732772.3733531.

13 Marthe Bonamy, Cyril Gavoille, and Michał Pilipczuk. *Shorter Labeling Schemes for Planar Graphs*, pages 446–462. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611975994.27, arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611975994.27, doi:10.1137/1.9781611975994.27.

14 Marthe Bonamy and Colin Geniet. χ-boundedness and neighbourhood complexity of bounded merge-width graphs. *CoRR*, abs/2504.08266, 2025. URL: `https://doi.org/10.48550/arXiv.2504.08266`, `arXiv:2504.08266`, `doi:10.48550/ARXIV.2504.08266`.

15 Édouard Bonnet, Samuel Braunfeld, Ioannis Eleftheriadis, Colin Geniet, Nikolas Mählmann, Michal Pilipczuk, Wojciech Przybyszewski, and Szymon Torunczyk. Separability properties of monadically dependent graph classes. In Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis, editors, *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, July 8-11, 2025, Aarhus, Denmark*, volume 334 of *LIPIcs*, pages 147:1–147:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025. URL: `https://doi.org/10.4230/LIPIcs.ICALP.2025.147`, `doi:10.4230/LIPICS.ICALP.2025.147`.

16 Édouard Bonnet, Florent Foucaud, Tuomo Lehtilä, and Aline Parreau. Neighbourhood complexity of graphs of bounded twin-width. *Eur. J. Comb.*, 115:103772, 2024. URL: `https://doi.org/10.1016/j.ejc.2023.103772`, `doi:10.1016/J.EJC.2023.103772`.

17 Édouard Bonnet, Serge Gaspers, Antonin Lambilliotte, Stefan Rümmele, and Abdallah Saffidine. The parameterized complexity of positional games. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80 of *LIPIcs*, pages 90:1–90:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. URL: `https://doi.org/10.4230/LIPIcs.ICALP.2017.90`, `doi:10.4230/LIPICS.ICALP.2017.90`.

18 Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width I: tractable FO model checking. *J. ACM*, 69(1):3:1–3:46, 2022. URL: `https://doi.org/10.1145/3486655`, `doi:10.1145/3486655`.

19 Nicolas Bousquet, Quentin Deschamps, Lucas de Meyer, and Théo Pierron. Square coloring planar graphs with automatic discharging. *SIAM Journal on Discrete Mathematics*, 38(1):504–528, April 2022. URL: `https://hal.science/hal-04960763`, `doi:10.1137/22M1492623`.

20 Nicolas Bousquet, Laurent Feuilloley, and Théo Pierron. Local certification of graph decompositions and applications to minor-free classes. *J. Parallel Distributed Comput.*, 193:104954, 2024. URL: `https://doi.org/10.1016/j.jpdc.2024.104954`, `doi:10.1016/J.JPDC.2024.104954`.

21 Nicolas Bousquet, Aurélie Lagoutte, and Stéphan Thomassé. The Erdős–hajnal conjecture for paths and antipaths. *Journal of Combinatorial Theory, Series B*, 113:261–264, 2015.

22 Simone Bova, Florent Capelli, Stefan Mengel, and Friedrich Slivovsky. On compiling cnfs into structured deterministic dnnfs. In Marijn Heule and Sean A. Weaver, editors, *Theory and Applications of Satisfiability Testing - SAT 2015 - 18th International Conference, Austin, TX, USA, September 24-27, 2015, Proceedings*, volume 9340 of *Lecture Notes in Computer Science*, pages 199–214. Springer, 2015. URL: `https://doi.org/10.1007/978-3-319-24318-4_15`, `doi:10.1007/978-3-319-24318-4\_15`.

23 Boris Bukh. Random algebraic construction of extremal graphs. *Bulletin of the London Mathematical Society*, 47, 2014. URL: `https://api.semanticscholar.org/CorpusID:12387209`.

24 Jianer Chen, Xiuzhen Huang, Iyad A Kanj, and Ge Xia. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences*, 72(8):1346–1367, 2006.

25 Maria Chudnovsky, Neil Robertson, Paul Seymour, and Robin Thomas. The strong perfect graph theorem. *Annals of mathematics*, pages 51–229, 2006.

26 Maria Chudnovsky, Alex Scott, and Paul Seymour. Induced subgraphs of graphs with large chromatic number. iii. long holes. *Combinatorica*, 37(6):1057–1072, 2017.

27 Maria Chudnovsky, Alex Scott, Paul Seymour, and Sophie Spirkl. Induced subgraphs of graphs with large chromatic number. viii. long odd holes. *Journal of Combinatorial Theory, Series B*, 140:84–97, 2020.

28 Maria Chudnovsky, Alex Scott, Paul Seymour, and Sophie Spirkl. Pure pairs. i. trees and linear anticomplete pairs. *Advances in Mathematics*, 375:107396, 2020.

29 Alessio Conte and Takeaki Uno. New polynomial delay bounds for maximal subgraph enumeration by proximity search. In *Proceedings of the 51st Annual ACM SIGACT Symposium*

on Theory of Computing, STOC 2019, pages 1179–1190, New York, NY, USA, 2019. Association for Computing Machinery. URL: https://doi.org/10.1145/3313276.3316402, doi:10.1145/3313276.3316402.

30 Kris Coolsaet, Sven D'hondt, and Jan Goedgebeur. House of graphs 2.0: A database of interesting graphs and more. *Discrete Applied Mathematics*, 325:97–107, 2023. URL: https://www.sciencedirect.com/science/article/pii/S0166218X22004036, doi:https://doi.org/10.1016/j.dam.2022.10.013.

31 Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990. URL: https://doi.org/10.1016/0890-5401(90)90043-H, doi:10.1016/0890-5401(90)90043-H.

32 Marek Cygan, Jesper Nederlof, Marcin Pilipczuk, Michał Pilipczuk, Johan MM Van Rooij, and Jakub Onufry Wojtaszczyk. Solving connectivity problems parameterized by treewidth in single exponential time. *ACM Transactions on Algorithms (TALG)*, 18(2):1–31, 2022.

33 Aubrey D. N. J. de Grey. The chromatic number of the plane is at least 5, 2018. URL: https://arxiv.org/abs/1804.02385, arXiv:1804.02385.

34 Michał Dębski, Stefan Felsner, Piotr Micek, and Felix Schröder. Improved bounds for centered colorings. *Advances in Combinatorics*, 2021.

35 Oscar Defrain, Louis Esperet, Aurélie Lagoutte, Pat Morin, and Jean-Florent Raymond. Local certification of geometric graph classes. In Rastislav Královic and Antonín Kucera, editors, *49th International Symposium on Mathematical Foundations of Computer Science, MFCS 2024, August 26-30, 2024, Bratislava, Slovakia*, volume 306 of *LIPIcs*, pages 48:1–48:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. URL: https://doi.org/10.4230/LIPIcs.MFCS.2024.48, doi:10.4230/LIPICS.MFCS.2024.48.

36 Holger Dell and Dániel Marx. Kernelization of packing problems. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pages 68–81. SIAM, 2012.

37 Erik D Demaine and MohammadTaghi Hajiaghayi. The bidimensionality theory and its algorithmic applications. *The Computer Journal*, 51(3):292–302, 2008.

38 Jan Dreier and Szymon Torunczyk. Merge-width and first-order model checking. In Michal Koucký and Nikhil Bansal, editors, *Proceedings of the 57th Annual ACM Symposium on Theory of Computing, STOC 2025, Prague, Czechia, June 23-27, 2025*, pages 1944–1955. ACM, 2025. URL: https://doi.org/10.1145/3717823.3718259, doi:10.1145/3717823.3718259.

39 Éric Duchêne, Valentin Gledel, Fionn Mc Inerney, Nicolas Nisse, Nacim Oijid, Aline Parreau, and Milos Stojakovic. Complexity of maker-breaker games on edge sets of graphs. *Discret. Appl. Math.*, 361:502–522, 2025. URL: https://doi.org/10.1016/j.dam.2024.11.012, doi:10.1016/J.DAM.2024.11.012.

40 Éric Duchêne, Valentin Gledel, Aline Parreau, and Gabriel Renault. Maker-breaker domination game. *Discret. Math.*, 343(9):111955, 2020. URL: https://doi.org/10.1016/j.disc.2020.111955, doi:10.1016/J.DISC.2020.111955.

41 Guillaume Ducoffe, Michel Habib, and Laurent Viennot. *Diameter computation on <italic>H</italic>-minor free graphs and graphs of bounded (distance) VC-dimension*, pages 1905–1922. 2020. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611975994.117, arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611975994.117, doi:10.1137/1.9781611975994.117.

42 Vida Dujmović, Louis Esperet, Cyril Gavoille, Gwenaël Joret, Piotr Micek, and Pat Morin. Adjacency labelling for planar graphs (and beyond). *Journal of the ACM (JACM)*, 68(6):1–33, 2021.

43 Vida Dujmović, Louis Esperet, Gwenaël Joret, Bartosz Walczak, and David R. Wood. Planar graphs have bounded nonrepetitive chromatic number. *Advances in Combinatorics*, 5:11, March 2020. URL: https://hal.science/hal-02165018, doi:10.19086/aic.12100.

44 Vida Dujmović, Gwenaël Joret, Piotr Micek, Pat Morin, Torsten Ueckerdt, and David R Wood. Planar graphs have bounded queue-number. *Journal of the ACM (JACM)*, 67(4):1–38, 2020.

45 Zdeněk Dvořák and Jean-Sébastien Sereni. On fractional fragility rates of graph classes. *The Electronic Journal of Combinatorics*, pages P4–9, 2020.

46 Thomas Eiter and Georg Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM Journal on Computing*, 24(6):1278–1304, 1995.

47 Jessica A. Enright, Samuel D. Hand, Laura Larios-Jones, and Kitty Meeks. Structural parameters for dense temporal graphs. In Rastislav Královic and Antonín Kucera, editors, *49th International Symposium on Mathematical Foundations of Computer Science, MFCS 2024, August 26-30, 2024, Bratislava, Slovakia*, volume 306 of *LIPIcs*, pages 52:1–52:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. URL: `https://doi.org/10.4230/LIPIcs.MFCS.2024.52`, `doi:10.4230/LIPICS.MFCS.2024.52`.

48 Paul Erdős. Graph theory and probability. *Canadian Journal of Mathematics*, 11:34–38, 1959. `doi:10.4153/CJM-1959-003-9`.

49 Paul Erdős and András Hajnal. Ramsey-type theorems. *Discrete Applied Mathematics*, 25(1-2):37–52, 1989.

50 Fedor V. Fomin, Pierre Fraigniaud, Pedro Montealegre, Ivan Rapaport, and Ioan Todinca. Distributed model checking on graphs of bounded treedepth. In Dan Alistarh, editor, *38th International Symposium on Distributed Computing, DISC 2024, October 28 to November 1, 2024, Madrid, Spain*, volume 319 of *LIPIcs*, pages 25:1–25:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. URL: `https://doi.org/10.4230/LIPIcs.DISC.2024.25`, `doi:10.4230/LIPICS.DISC.2024.25`.

51 Fedor V Fomin, Daniel Lokshtanov, and Saket Saurabh. Efficient computation of representative sets with applications in parameterized and exact algorithms. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 142–151. SIAM, 2014.

52 Florian Galliot. *Hypergraphs and the Maker-Breaker game : a structural approach. (Hypergraphes et jeu Maker-Breaker : une approche structurelle)*. PhD thesis, Grenoble Alpes University, France, 2023. URL: `https://tel.archives-ouvertes.fr/tel-04249805`.

53 Valentin Gledel, Nacim Oijid, Sébastien Tavenas, and Stéphan Thomassé. On the complexity of client-waiter and waiter-client games. In Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis, editors, *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, July 8-11, 2025, Aarhus, Denmark*, volume 334 of *LIPIcs*, pages 89:1–89:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025. URL: `https://doi.org/10.4230/LIPIcs.ICALP.2025.89`, `doi:10.4230/LIPICS.ICALP.2025.89`.

54 Petr A. Golovach, Giannos Stamoulis, and Dimitrios M. Thilikos. Model-checking for first-order logic with disjoint paths predicates in proper minor-closed graph classes. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3684–3699. SIAM, 2023. `doi:10.1137/1.9781611977554.ch141`.

55 Timothy Gowers. Post on x. URL: `https://www.reddit.com/r/math/comments/1olctkm/gowers_on_using_ai_for_math_research/`.

56 Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *J. ACM*, 64(3):17:1–17:32, 2017. URL: `https://doi.org/10.1145/3051095`, `doi:10.1145/3051095`.

57 Sylvain Guillemot and Dániel Marx. Finding small patterns in permutations in linear time. In Chandra Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 82–101. SIAM, 2014. URL: `https://doi.org/10.1137/1.9781611973402.7`, `doi:10.1137/1.9781611973402.7`.

58 A. Gyárfás. Problems from the world surrounding perfect graphs. *Zastowania Matematyki Applicationes Mathematicae*, XIX:413–441, 1987.

59 Marc Heinrich. *Reconfiguration and combinatorial games. (Reconfiguration et jeux combinatoires)*. PhD thesis, University of Lyon, France, 2019. URL: `https://tel.archives-ouvertes.fr/tel-02294749`.

60 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001.

61 Klaus Jansen. Parameterized approximation scheme for the multiple knapsack problem. *SIAM Journal on Computing*, 39(4):1392–1412, 2010.

62 Dong Kang, Tom Kelly, Daniela Kühn, Abhishek Methuku, and Deryk Osthus. A proof of the Erdős–Faber–Lovász conjecture. *Annals of Mathematics*, 198(2):537 – 618, 2023. URL: https://doi.org/10.4007/annals.2023.198.2.2, doi:10.4007/annals.2023.198.2.2.

63 Frantisěk Kardoš. A computer-assisted proof of the barnette–goodey conjecture: Not only fullerene graphs are hamiltonian. *SIAM Journal on Discrete Mathematics*, 34(1):62–100, 2020.

64 CS Karthik, Bundit Laekhanukit, and Pasin Manurangsi. On the parameterized complexity of approximating dominating set. In *50th Annual ACM Symposium on Theory of Computing*, pages 1283–1296, 2018.

65 David Kempe, Jon M. Kleinberg, and Amit Kumar. Connectivity and inference problems for temporal networks. *J. Comput. Syst. Sci.*, 64(4):820–842, 2002. URL: https://doi.org/10.1006/jcss.2002.1829, doi:10.1006/JCSS.2002.1829.

66 Eun Jung Kim, Stefan Kratsch, Marcin Pilipczuk, and Magnus Wahlström. Flow-augmentation II: undirected graphs. *ACM Trans. Algorithms*, 20(2):12, 2024. URL: https://doi.org/10.1145/3641105, doi:10.1145/3641105.

67 Eun Jung Kim, Stefan Kratsch, Marcin Pilipczuk, and Magnus Wahlström. Flow-augmentation I: directed graphs. *J. ACM*, 72(1):5:1–5:38, 2025. URL: https://doi.org/10.1145/3706103, doi:10.1145/3706103.

68 Eun Jung Kim, Stefan Kratsch, Marcin Pilipczuk, and Magnus Wahlström. Flow-augmentation III: complexity dichotomy for boolean csps parameterized by the number of unsatisfied constraints. *SIAM J. Comput.*, 54(4):1065–1137, 2025. URL: https://doi.org/10.1137/23m1553698, doi:10.1137/23M1553698.

69 Stephan Kreutzer. Algorithmic meta-theorems. In *International Workshop on Parameterized and Exact Computation*, pages 10–12. Springer, 2008.

70 Michael Lampis. The primal pathwidth SETH. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1494–1564. SIAM, 2025.

71 Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Known algorithms on graphs of bounded treewidth are probably optimal. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 777–789. SIAM, 2011.

72 Nikolas Mählmann. *Monadically stable and monadically dependent graph classes: characterizations and algorithmic meta-theorems*. PhD thesis, Bremen University, Germany, 2024. URL: https://media.suub.uni-bremen.de/handle/elib/8304, doi:10.26092/ELIB/3338.

73 Heikki Mannila and Kari-Jouko Räihä. *The design of relational databases*. Addison-Wesley Longman Publishing Co., Inc., 1992.

74 Andrea Marino. *Analysis and enumeration: algorithms for biological graphs*, volume 6. Springer, 2015. doi:https://doi.org/10.2991/978-94-6239-097-3.

75 Dániel Marx. Parameterized complexity and approximation algorithms. *The Computer Journal*, 51(1):60–78, 2008.

76 Sam Mattheus and Jacques Verstraete. The asymptotics of $r(4,t)$. *Annals of Mathematics*, 1999(2), 2024. URL: https://arxiv.org/abs/2306.04007, arXiv:2306.04007.

77 Robin A. Moser and Gábor Tardos. A constructive proof of the general lovász local lemma. *J. ACM*, 57(2), February 2010. URL: https://arxiv.org/abs/0903.0544, doi:10.1145/1667053.1667060.

78 Jaroslav Nesetril and Patrice Ossona de Mendez. Linear time low tree-width partitions and algorithmic consequences. In Jon M. Kleinberg, editor, *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 391–400. ACM, 2006. URL: https://doi.org/10.1145/1132516.1132575, doi:10.1145/1132516.1132575.

79 Jaroslav Nesetril and Patrice Ossona de Mendez. Tree-depth, subgraph coloring and homomorphism bounds. *Eur. J. Comb.*, 27(6):1022–1041, 2006. URL: https://doi.org/10.1016/j.ejc.2005.01.010, doi:10.1016/J.EJC.2005.01.010.

80 Jaroslav Nesetril and Patrice Ossona de Mendez. Grad and classes with bounded expansion i. decompositions. *Eur. J. Comb.*, 29(3):760–776, 2008. URL: https://doi.org/10.1016/j.ejc.2006.07.013, doi:10.1016/J.EJC.2006.07.013.

81 Jaroslav Nesetril and Patrice Ossona de Mendez. Grad and classes with bounded expansion III. restricted graph homomorphism dualities. *Eur. J. Comb.*, 29(4):1012–1024, 2008. URL: https://doi.org/10.1016/j.ejc.2007.11.019, doi:10.1016/J.EJC.2007.11.019.

82 Jaroslav Nesetril and Patrice Ossona de Mendez. *Sparsity - Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and combinatorics*. Springer, 2012. URL: https://doi.org/10.1007/978-3-642-27875-4, doi:10.1007/978-3-642-27875-4.

83 Jinyoung Park and Huy Tuan Pham. A proof of the kahn-kalai conjecture. *J. Amer. Math. Soc.*, pages 235–243, 2024. URL: https://arxiv.org/abs/2203.17207, arXiv:2203.17207.

84 Christophe Paul, Evangelos Protopapas, Dimitrios M. Thilikos, and Sebastian Wiederrecht. Obstructions to erdös-pósa dualities for minors. In *65th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2024, Chicago, IL, USA, October 27-30, 2024*, pages 31–52. IEEE, 2024. URL: https://doi.org/10.1109/FOCS61266.2024.00013, doi:10.1109/FOCS61266.2024.00013.

85 Md Lutfar Rahman and Thomas Watson. 6-uniform maker-breaker game is pspace-complete. *Comb.*, 43(3):595–612, 2023. URL: https://doi.org/10.1007/s00493-023-00026-7, doi:10.1007/S00493-023-00026-7.

86 Felix Reidl, Fernando Sánchez Villaamil, and Konstantinos S. Stavropoulos. Characterising bounded expansion by neighbourhood complexity. *Eur. J. Comb.*, 75:152–168, 2019. URL: https://doi.org/10.1016/j.ejc.2018.08.001, doi:10.1016/j.ejc.2018.08.001.

87 Matthieu Rosenfeld. Another Approach to Non-Repetitive Colorings of Graphs of Bounded Degree. *The Electronic Journal of Combinatorics*, 27(3), July 2020. URL: https://hal.science/hal-03583287, doi:10.37236/9667.

88 Matthieu Rosenfeld. The growth rate over trees of any family of sets defined by a monadic second order formula is semi-computable. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 776–795. SIAM, 2021.

89 Sigve Hortemo Sæther, Jan Arne Telle, and Martin Vatshelle. Solving #sat and MAXSAT by dynamic programming. *J. Artif. Intell. Res.*, 54:59–82, 2015. URL: https://doi.org/10.1613/jair.4831, doi:10.1613/JAIR.4831.

90 Thomas J. Schaefer. On the complexity of some two-person perfect-information games. *J. Comput. Syst. Sci.*, 16(2):185–225, 1978. URL: https://doi.org/10.1016/0022-0000(78)90045-4, doi:10.1016/0022-0000(78)90045-4.

91 Alex Scott and Paul Seymour. Induced subgraphs of graphs with large chromatic number. I. Odd holes. *Journal of Combinatorial Theory, Series B*, 121:68–84, 2016.

92 Yann Strozecki. Enumeration complexity. *Bulletin of EATCS*, 1(129), 2019.

93 David P Sumner. Subtrees of a graph and chromatic number. *The theory and applications of graphs*, pages 557–576, 1981.

94 Terence Tao. Post on mathstodon. URL: https://mathstodon.xyz/@tao/115493667607261044.

95 Szymon Torunczyk. Flip-width: Cops and robber on dense graphs. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 663–700. IEEE, 2023. URL: https://doi.org/10.1109/FOCS57990.2023.00045, doi:10.1109/FOCS57990.2023.00045.

96 Szymon Torunczyk. Evaluating first-order formulas in structured graphs (invited talk). In Sudeepa Roy and Ahmet Kara, editors, *28th International Conference on Database Theory, ICDT 2025, March 25-28, 2025, Barcelona, Spain*, volume 328 of *LIPIcs*, pages 3:1–3:2. Schloss

Dagstuhl - Leibniz-Zentrum für Informatik, 2025. URL: https://doi.org/10.4230/LIPIcs.ICDT.2025.3, doi:10.4230/LIPICS.ICDT.2025.3.

97 Xuding Zhu. Colouring graphs with bounded generalized colouring number. *Discret. Math.*, 309(18):5562–5568, 2009. URL: https://doi.org/10.1016/j.disc.2008.03.024, doi:10.1016/J.DISC.2008.03.024.

Twenty years of GdR IFM, seen from GT IQ

1 Context

Quantum computing is a quite young field of research, initiated in the early 80's when Feynman expressed the idea of using well-controlled quantum systems to simulate quantum physics, and which really took off in the mid 90's when Shor and Grover devised quantum algorithms that have a better complexity than their classical counterparts, respectively for prime factorisation and for the unstructured search problem. This aspect constitutes one of the best motivations for the field: the prospect of solving *some* problems more efficiently than what is possible classically, potentially exponentially so, depending on the problem at hand.

Quantum computing is also peculiar in that on the one hand, it covers a vast area of computer science – algorithmics, complexity theory, cryptography, error correcting codes, formal methods, artificial intelligence, ... –, and on the other hand, it is in close contact with physics. Indeed, the existing quantum computers are still in their infancy, and working with them efficiently for practical applications requires a deeper understanding of the underlying physics than what is required nowadays on a classical computer.

Long considered a relatively niche and quirky area of research, quantum computing has gained in traction over the past two decades, both in the academic and the industrial worlds. In the first, this is well illustrated by the surge in research positions in the domain, the integration of quantum computation to master programs, or even the creation of dedicated master programs to the topic, the development of conferences and workshops, etc... On the industrial side, big companies such as Google, Microsoft, IBM, Amazon, or Atos/Bull in France have started a serious quantum-related activity, while several startups have been brought to life, such as, just in France, Quandela, Pasqal, Alice&Bob, C12, Quobly, Welinq, ColibriTD, VeriQloud, etc... On top of these several big actors (EDF, Total, Crédit Agricole, ...) have begun monitoring developments in quantum computing technology in order to better assess the potential benefits of such technology for their businesses. All this now constitutes a rich ecosystem, fostered by recent, massive public and private funding (see e.g. the "Plan Quantique National").

2 Quantum algorithms and complexity

The development of quantum algorithms has been a driving force behind the rapid advancement of quantum computing since its early stages. While the field began with quantum algorithms for toy problems [53, 130], some of the main quantum algorithms, even after all progress in the field, are the foundational quantum algorithm to factor numbers in polynomial time [129] or unstructured database search [78].

In the 2000s, theoretical advancements focused on refining and generalizing quantum algorithms, often with ideas that are inspired both from Computer Science and Physics. Techniques that were fundamental in the first quantum algorithms were refined, generalized and culminated into more general technique that could be applied in different contexts, such as the quantum Fourier transform, which has fundamental to quantum learning theory [13], and the quantum amplitude amplification/estimation [21], that gives a quadratic speedup on "sampling" from a desired event. We also saw the development and consolidation of models of computation that had far-reaching applications such as Adiabatic Quantum

Computing [63] or Quantum Walks [4]. In adiabatic quantum computing, we encode the solution of a (potentially hard) problem into a Hamiltonian, a physical object that describes the evolution of quantum systems, and its groundstate, the state that minimizes the energy of the Hamiltonian. However, finding the groundstate of such a Hamiltonian directly is itself a hard problem. Adiabatic quantum computing offers a solution for it: we start from the groundstate of an easy Hamiltonian, and gradually turn it into the groundstate of the Hamiltonian of interest. Initially proposed for solving specific optimization problems, it was later shown that adiabatic quantum computing is an universal model for quantum computing [63], and it is a building block of modern frameworks such as VQE and QAOA that we describe below. Quantum walks emerged as a powerful framework within quantum algorithm design, offering a quantum analogue to classical random walks. Unlike their classical counterparts, quantum walks leverage superposition and interference to explore complex spaces more efficiently, enabling exponential speedups for certain problems [38] and they have found applications in a wide range of areas. We also have seen the proposal of new algorithmic techniques, such as HHL [82], an algorithm for solving linear systems of equations exponentially faster than classical methods, paving the way for more recent applications in machine learning and data analysis [92, 20]. Moreover, quantum algorithms also played a key role in the development of post-quantum cryptography, both from a cryptanalysis perspective, and as a tool to provide security proofs, such as celebrated the worst-case to average-case quantum reduction by Regev to support the hardness of LWE [121].

The 2010s and 2020s marked the beginning of the Noisy Intermediate-Scale Quantum (NISQ) era, characterized by the availability of quantum computers with limited qubits and high error rates. This era shifted the focus to hybrid quantum-classical algorithms, which combined quantum and classical computing to mitigate hardware limitations. Algorithms like the Variational Quantum Eigensolver (VQE) [119] and the Quantum Approximate Optimization Algorithm (QAOA) [62] were developed to tackle optimization and quantum chemistry problems, making quantum computing more accessible for near-term applications. The rise of quantum machine learning also gained momentum, with algorithms such as Quantum Support Vector Machines (QSVM) [120] and Quantum Neural Networks (QNNs) [125] exploring the intersection of quantum computing and artificial intelligence. Interestingly, some of these quantum machine learning algorithms have been later *dequantized* [134], leading to new efficient classical algorithms for a variety of problems. Moreover, new tools and frameworks, such as Quantum Singular-Value Transformation (QSVT) and block-encoding of matrices [72], provide a new conceptual way of developing quantum algorithms, and provide a significant improvement for important physical tasks such as Hamiltonian simulation.

From a complexity perspective, we have seen many fundamental results in the early 2000s. With Kitaev's seminal result on defining QMA, the quantum version of NP, and proving that the local Hamiltonian problem, a natural problem in condensed matter physics, is QMA-complete [96], initialized the field which is now called Hamiltonian complexity [70]. This field has gained a lot of attention since it not only allows us to study problems related to complexity theory and hardness of approximation, but also enables the study of the structure in low-energy states of interest from a Physics perspective. In parallel, we have also seen the development of techniques for proving lower-bounds in the quantum setting in different setups such as query complexity [9, 17] and communication complexity [32]. Such techniques have also influenced other fields in classical TCS, e.g., it has been shown, via lower bounds on quantum communication complexity, that TSP requires superpolynomial linear programs [65]. In the 2010s, we have seen the development of Hamiltonian complexity, and the study of the quantum PCP conjecture [5]. The classical PCP theorem connects

many key topics of classical complexity theory: hardness of approximation, efficient proof verification, hardness of finding the best game strategies. One of the most important open questions in quantum complexity theory today states whether a similar conjecture also holds in the quantum setting. We have also seen the development of quantum interactive proof systems, which extend NP and adds interaction between polynomially bounded verifiers and unbounded provers. Using techniques from classical optimization, it has been shown that in the single prover case, quantum and classical interactive proof systems have the same computational power [87]. However, in the multi-prover case, we have seen the breakthrough result that multiple untrusted entangled provers can be used to prove even undecidable problems such as the Halting problem [89]. This has not only profound implications for complexity theory, but one of its main consequences is to disprove the long-standing Connes embedding conjecture in Operator Algebra that had been open for decades [137].

3 Quantum cryptography and communication

Quantum cryptography emerged as an area of interest following the invention of quantum key distribution (QKD) [18] in the 1980s, marking a pivotal moment in the history of cryptography. The development of QKD represented a significant leap forward, introducing a method by which two parties could share a secure key over an insecure channel without any risk of interception by eavesdroppers. Since its inception, the field of quantum cryptography has expanded rapidly, addressing many privacy related problems that classical cryptography was previously unable to solve. A prominent example is the concept of quantum money [140], which presents a groundbreaking approach to using the "no-cloning" principle of quantum physics to create digital currency that cannot be counterfeited. As quantum cryptography has continued to evolve, it has been realized that information-theoretic security (i.e. security against even unbounded adversaries) also has its limitations in the quantum setting [106, 109].

In the 2000s, the field of quantum cryptography was mainly focused on improving the security and the efficiency of QKD protocols. First of all, there was a lot of effort in defining the desired security definitions [123], pushing protocols closer to practical setups [77] and proving their security – which sometimes come many years later [102]. Many of these improvements, which are still currently being pushed forward, come from redefining notions of information theory to the quantum setting [141], leading to a plethora of quantities such as quantum entropies and divergence, that enable tighter security proofs leading to improvements on QKD rate. Secondly, there has been a lot of effort in finding models under which we can achieve cryptographic protocols with provable security. Two examples of these directions are the bounded storage model [47], where the adversary has only limited quantum memory, or verifiable delegation of quantum computation: a classical client is able to delegate some quantum computation to a quantum server, who can solve the computation and prove to the client that the output is correct. The first protocols managed to prove this type of result but with clients with limited quantum resources [27, 66, 26, 7].

In the 2010s and 2020s, there was an impressive growth of the field, which branched in many directions. Firstly, new security notions such as device-independence, allowed us to find protocols where we have minimal trust in the quantum devices used in cryptographic protocols, e.g. in QKD [136] or to achieve protocols for verifiable delegation with quantum computation by classical clients [122, 44]. Secondly, inspired by classical cryptography, there was a big development on quantum cryptography under computational assumptions. Here, the goals are two-fold: to provide quantum protocols for classical functionalities while relying on a weaker computational assumption [76, 16], but also to provide protocols for quantum

functionalities [58, 55]. Within this research direction, it has been realized that considering assumptions that are inherently quantum (such as the existence of pseudo-random states), we can have computational cryptography even if P = NP [88, 100], which cannot be done classically. There is an intense line of research in quantum cryptography to understand the difference of variants of quantum cryptographic assumptions, and in particular to find the minimal one. Finally, there was also the "reborn of uncloneability". It has been shown that the fact that quantum states cannot be generically cloned can be used to achieve many more applications than just quantum money schemes, and we have today many protocols with one-time properties, i.e. we can prove that their functionalities can be used a single time [80], with uncloneable properties [29, 146], where the information is exclusively revealed to a single party, or even with certified deletion, guaranteeing that an untrusted party has destroyed information [28].

4 Quantum error-correction and fault-tolerant quantum computation

As quantum hardware is expected to be quite noisy, hence prone to errors, it is paramount to try and mitigate them. The first quantum error correcting code is due to Shor in the 90s, who showed in particular that through measurements, continuous errors could actually be brought to only two kinds of discrete errors: bit-flips and phase-flips. Over the years, many new error correction schemes, that heavily rely on the theory of classical codes, have been proposed [103, 14, 60, 105]. Notice that to have a good, practical error correction scheme, it is not enough to have a good code, as one also needs to efficiently decode it (which in all generality is an NP-hard problem for linear codes).

CSS codes are special cases of quantum error correcting codes, that can be built out of two classical linear codes, making their study easier. This constitutes one of the most studied classes of codes. To date, one of the codes that seems the most promising – and which turns out to be a CSS code –, due to its conceptual simplicity, its good theoretical performance, and its compliance with most reasonable qubit layouts, is the surface code [67]. It was introduced in the early 2000's, building on top of Kitaev's toric code [97], and it encodes a single logical qubit into an array of $n \times n$ physical qubits – the size of the array allowing us to control the error tolerance of the scheme: the code distance scales as $\sqrt{n}$.

This scaling capability is however not on par with the best classical codes, whose distance and number of logical qubits scale linearly with n. A lot of effort has hence been devoted to the study of *low-density parity check codes* [23], which benefit from efficient decoding algorithms and can approach channel capacities in the classical case. A breakthrough was [135] that shows that quantum LDPC codes may have both a non-vanishing rate[1] and a better-than-logarithmic distance. The question of the existence of an LDPC code with linear rate and linear distance was open for a long time and only positively answered in 2021 [113, 104].

This is particularly important if one wants to overcome the *threshold* problem [6, 99]: applying an error correcting scheme implies applying to the quantum memory some additional quantum gates, which themselves can bring new errors, hence, for an error correcting code to be useful on a given hardware, the error rates of the gates have to be below a certain threshold. The threshold of course depends on the chosen scheme and its properties. The threshold theorem has been extended in [64] for constant-overhead fault-tolerance using expander codes.

[1] the ratio of the number of logical qubits over the number of physical qubits

A recent avenue for obtaining better such threshold is dynamical codes, such as Floquet codes [83, 54] i.e. stabilizer codes (themselves a generalisation of CSS codes) that evolve over time. They may circumvent certain no-go theorems that apply to static stabilizer codes, exhibit reduced complexity in error detection, can often be implemented on lattices with low qubit connectivity and employ simple, two-qubit measurements.

Quantum error correction obviously benefits from results in its classical counterpart, but it so happens that the converse is also true. For instance, a result on locally decodable codes was provided in [91], following a quantum argument. Since then, several results in quantum error correction have been provided hand-in-hand with their classical counterparts (e.g. [25, 113]).

An important aspect of error correction is the ability to translate an algorithm or protocol defined on ideal logical qubits, into the encoded realm (e.g. a logical 1-qubit gate will translate into a series of logical gates on the array of qubits used for the surface code). Most error correction schemes natively handle a non-universal subset of gates (called Clifford gates), and need an extra push to reach universal computation. This additional computational resource is usually provided as a state (called *magic state*) [35, 71], which is then incorporated into the circuit. Providing this resource as a state allows us to better control the quality of that resource. Since magic states are usually hard to create, they may go through a round of distillation, whereby we take several low-quality magic states and create a better-quality one. Interestingly, this process itself uses error correcting codes.

With the advent of actual quantum computers, an obvious goal for manufacturers was to reach the threshold for a given error correction scheme using their hardware. This achievement has only recently been announced for the first time [60].

Physicists have also approached the problem of qubit robustness, and have proposed new physical implementations of qubits that allow to correct errors at a more fundamental level. The cat qubit [61], for instance, is a quantum harmonic oscillator that makes use of its coherent states as its logical 0 and 1 states, and which at a more abstract level can be considered as a qubit that is free from phase-flip errors. In such systems, one can go back to classical codes to correct for the remaining type of errors (bit-flips).

With such a setup that is easier to handle, a manufacturer with fault-tolerant quantum computation as its goal can more easily try to *co-design* its chip together with an error-correcting scheme. Since the qubit layout will most probably remain 2-dimensional (array-like), one may use the 2nd dimension in the case of the cat-qubit as a way to implement 2D classical codes [124].

5 Programming languages and compilation

Twenty years ago, several programming languages were already introduced [126], trying to identify and promote concepts deemed useful for quantum computation, that would allow the programmer to abstract away from the low-level inner workings of a putative quantum computer, and reason at a more conceptual level. A few imperative approaches existed at the time, that introduced run-time checks to ensure the validity of the execution of the program. This approach arguably raises several problems, first due to the probabilistic nature of quantum computation (is there an error because of a bug in the code, or because the quantum data unexpectedly changed?), and second because of the cost of quantum resources: one may prefer to classically and statically check the program before it is run on a quantum machine. This is allowed by functional programming languages.

All of these prototypical programming languages were either theoretical constructions

meant to study the properties of the concepts used as commands for the language, or simply lacked the ability to program large-scale algorithms. One of the first practical programming languages that overcomes this scalability issue is Quipper [75]. It is a higher-order functional programming language embedded in Haskell, and it comes with a categorical semantics.

Several scalable programming languages have since then been developed, often linked to a specific platform (Qiskit, developed and used by IBM, Q# by Microsoft, MyQLM by Atos/Eviden, Cirq by Google, Perceval by Quandela...). These usually have a preferred gate set (driven by the quantum hardware they interface with), and have to compile, in their own way (e.g. [132, 84]), the high-level code provided by the user to this set of gates. On top of complying with the imposed gate set, one has to account for the topological constraints of the quantum chip: indeed, physically, qubits have a given geographical location, and multi-qubit gates usually cannot be applied on qubits that are far apart. Solutions to this problem often use heuristics to classical optimisation problems, e.g. from graph theory in [131].

Since the lifetime of quantum memory is short and quantum gates may incur new errors, it is paramount to make the quantum circuit we want to execute as short as possible (without changing the results of the computation). Different metrics exist, depending on the setting and the kind of errors we want to avoid the most: circuit depth, circuit size (or gate count), number of ancillas (the additional qubits that are used for storage during the computation), ... Quantum volume is a more recent notion that takes into account both the full memory size, and either the depth or the gate count, leading to a more practical metric that one wants to minimise when optimising a circuit, but which is also used by manufacturers to gauge the quality of their device [46] (they look at the maximum volume they can get before the error rate becomes prohibitive). On most fault-tolerant architectures, the Clifford subset of gates is much simpler to implement than the others, it is hence customary to try and optimise the non-Clifford gate count. In other settings, two-qubit gates incur more errors than single-qubit ones, hence we are interested in those cases in the two-qubit gate count. In certain fragments of quantum computation, optimisation is fairly simple and can be done in polynomial time [143, 108], in other cases, heuristics are being sought to reach a better but non-optimal result [10, 110, 31].

Many quantum algorithms require very fine rotation gates (e.g. the QFT in Shor's algorithm that requires the $\frac{\pi}{2^n}$ on $n-1$ qubits). Such gates are believed to be very hard to implement, especially fault-tolerantly. Thankfully, the Solovay-Kitaev theorem [95] provides a way to turn any such ideal circuit into a circuit that uses a fixed (universal) gate-set with a very good gate-count w.r.t. the target accuracy of the transformation. This complexity has even been improved for specific gate sets, such as Clifford+T [127].

Compilation is also concerned with integrating classical data (part of the problem's parameters) into the quantum circuit. A main historical such component is the so-called quantum oracle, which, given a boolean function f, implements either $|x\rangle \mapsto (-1)^{f(x)} |x\rangle$ or $|x, y\rangle \mapsto |x, y \oplus f(x)\rangle$, for which there now exist several synthesis methods (e.g. [68, 144]).

Other such primitives include state preparation (encoding a given vector into a quantum state) [74, 148], or block-encoding (encoding an arbitrary given matrix as part of a quantum circuit) [133]. These last routines have recently gained interest because of their use in the powerful QSP and QSVT algorithms [73].

6 Formal methods and models of computation

Formal methods for quantum computation arose when the first quantum programming languages were defined, and needed to be given a semantic. As discussed above, some of

their most important motivations is the analysis and providing of guaranties to quantum programs, in a classical manner.

Theoretical quantum programming languages, such as the quantum λ-calculus [128], can be defined specifically with a given set of paradigmatic features in mind (linearity, higher-order, ...). It is then possible to study the properties of such sets of features, and ensure some properties on the language and its semantic, such as type safety, soundness (the fact the semantic cannot identify observationally different programs), full abstraction (the fact that the semantic identifies two programs if and only if they are observationally equivalent, see e.g. [40] for the result for the quantum λ-calculus). Although not meant to be practical, these may serve as foundations for programming languages such as those mentioned in the previous section.

One may for instance make use of deductive verification (using Hoare triple) together with an intermediate language to give and check specifications, as in the QBricks system [37], where algorithms such as Shor's were verified using the Why3 proof assistant. Other verification frameworks make for instance use of Rocq [115]. Other methods to improve the trust in the fact that an implementation is bug-free include type systems [48] and abstract interpretations [118, 145]. One may also make sure that the cost of the quantum circuit that is being built is under control, for instance by using implicit complexity (being able to write the program in a dedicated language ensures the polynomial time of the execution [81]), or by assigning a semantic to the program that takes this cost into account (e.g. [15]).

Formal methods are also being used to verify or certify the classical processing of the program. For instance, certain optimisers for quantum circuits has been checked using the Rocq proof assistant [85].

Low-level formal methods are interested in the gate-based behaviour and verification of quantum circuits and similar graphical representations. Quantum decision diagrams [142] for instance make use of a weighted version of decision diagrams – usually used for boolean function analysis – to perform equivalence checks of quantum circuits. They have been used to check the compilation flow of the Qiskit software [33]. More complex automata-based structures (e.g. [3]) are also being investigated and provide very good results.

The equations that govern the behaviour of quantum logic gates can be used to perform verification or to simplify the circuits we consider, by simple local manipulations of the circuit. It is natural to wonder whether a given set of such equations entirely captures the semantical equivalence of circuits. This central property is called the completeness of the equational theory. The first complete presentation for quantum circuits was only given recently, in 2023 [42]. Interestingly, it derives from an analogous result in linear optics. The result has since been improved [41], to reach minimality: the fact that none of the equations in the presentation is a consequence of the others.

Category theory has been used to provide alternative representations of quantum processes, in a way that in particular eliminates the rather rigid unitarity constraint, leading to the family of ZX-like calculi [43], whose main feature is the ability to deform the diagrams (representations of quantum operators) at will, without changing their semantics. In practice, such diagrams get rid of a lot of bureaucracy and become easier to manipulate than circuits. Completeness has also been one of the focuses there [138], and on top of verification [116], these can be used for instance in circuit optimisation [56, 93] and emulation of quantum circuits [94], or as it was first intended, to bridge between different models of quantum computation [57, 50, 36].

Indeed, the prominent way to devise and represent computations, the circuit model, is not the only one at our disposal. In this model, the focus is put on the unitary part (i.e. where

there is no interaction with the environment), as measurements can be deferred to the very end [111]. Another model of computation, the measurement-based model, instead pushes the core of the computation to the measurements, after a round of entangling gates [24]. Due to the probabilistic nature of measurements, the question of performing deterministic computations by applying corrections later in the execution has become central. Solutions to this question interestingly serve as a way to turn from a measurement-based scheme to a full-fledged quantum circuit [49, 30]. We may also mention the topological quantum computer which performs computations by manipulating anyons and swapping them [98] (although the existence of the kind of anyons required here is still unsure, but now benefits from good evidence [86]). Since this operation is non-involutive, computations can be modeled by braids from knot theory.

We have already mentioned adiabatic quantum computing in the context of algorithms and complexity, with results like [8] showing its polynomial equivalence with quantum circuits, hence providing a bridge between the continuous and the digital worlds. Another such bridge can be made by discretising the continuous space-time and simulating the evolution of a quantum system using circuits. This results in quantum cellular automata, whose local functioning is very simple, and from which emerge the physical phenomena one wants to simulate. The main concern of quantum cellular automata is quite naturally to ensure that its evolution converges to the simulated one when the time steps and space steps tend to 0 [12].

We close this section by mentioning a way to generalise quantum circuits that came to light in recent years: the capacity, allowed by quantum theory, to have superpositions of orders of execution of operators [39, 101], not only of data. The study of indefinite causal order is an ongoing research topic that may redefine the way we think about quantum computation, and with potential far reaching consequences, for instance due to its link with quantum gravity [52] (to quickly explain that link, relativity tells us that masses bend space-time around them, but quantum theory tells us that a particle may be in a superposition of two different places, resulting in a superposition of two space-time curvatures, where the causal order of two distinct events may be different). The most prototypical example of this new resource is the quantum switch, which, given two circuits U and V and an input state $|\psi\rangle$, applies a superposition of the two possible orders of U and V to $|\psi\rangle$, i.e. $|\psi\rangle \mapsto (\alpha UV + \beta VU)\,|\psi\rangle$. Physical realisations of this have been performed, although it is still debated whether it really is the quantum switch that was implemented or merely a simulation of it. A generalisation of the quantum switch may be used to show that indefinite causal orders yield a computational advantage over fixed-ordered quantum computations [11].

7 Near-term algorithms and quantum advantage

With the development of quantum devices, we have achieved what is called the NISQ area, where we have **N**oisy **I**ntermediate-**S**cale **Q**uantum devices. While these devices are not sufficient for implementing quantum algorithms with provable advantage such as Shor, HHL, and others, they are a landmark on the progress on quantum technologies and they open an era where we can start to investigate the behaviour of quantum devices in practice.

More concretely, with access to quantum devices, we can now start to study how quantum heuristics behave to solve problems of interest in practice. For example, the variational algorithms (see above) have been developed to accommodate these limitations. They use short, parameterised quantum circuits whose parameters are to be optimised classically. These can in theory be applied to many problems, but they face many challenges (in particular the so-called barren plateau, where the parameters' optimisation seems to be stuck in a local

but not global optimum). Lately, they have drawn attention for their potential for machine learning [45], they seem to be well designed for quantum chemistry (i.e. simulating chemical systems) [79].

The promise of solving some problems more efficiently quantumly than classically, has fueled the development of quantum computers, together with the definition of relevant problems for such demonstration, and provides a good way to assess the validity of a quantum device. The goal is to perform a computation on a quantum computer, that would take much longer classically – the precise scales vary from an author to the other, but think a few minutes quantumly vs at least thousands of years classically.

Theoretically, finding a suitable problem relies heavily on complexity theory. One such problem is Boson Sampling [1], which can be physically setup by, say, scattering photons through linear interferometers. This problem is related to computing permanents of matrices, a known hard problem, and it has been shown that an efficient simulation of the Boson Sampling problem would imply the collapse of the polynomial hierarchy to its third level.

Since 2019, several platforms have claimed to have reached quantum supremacy with the Random Circuit Sampling setup, starting with Google [59]. There is still some discussion as to whether this milestone really has been crossed, for several of these claims have been hindered by the existence of good classical simulation algorithms that brought the computation time to reasonable scales. These usually use tensor networks, and exploit the fact that current quantum computers are still very noisy [147].

8 Quantum HPC

The advent of the NISQ era has underlined the need for powerful classical resources to run hand in hand with the quantum algorithms. Some of them (like the variational ones) require a heavy classical treatment to optimise quantum circuit parameters [139]; others use the quantum primitives to get a rough estimate of the solution of a problem, that can then be refined classically, or with several backs-and-forth with the quantum processing unit [112]. Many numerical stability issues arising in quantum algorithms such as QSVT, HHL and variational solvers could be mitigated by the use of HPC-inspired methods, or by cleverly interfacing the classical and quantum parts. HPC may also be used in the preprocessing step, where for instance circuit synthesis may prove rather resource intensive [114, 51].

Since the NISQ-era computers are very limited in memory size, cutting the circuit we want to execute into chunks and stitching together the results obtained on each smaller circuit has been proposed [117, 107]. Finding the best spots to cut the circuit, and dealing with the data, requires large and fast classical capabilities.

In the LSQ era (the large-scale, fault tolerant era), it is expected that performing error correction will require fast classical methods to not hinder the quantum part. Some schemes have already been proposed with HPC methods in mind [105].

More pragmatically, HPC has already been used to simulate quantum circuits (as pointed out above). Different methods have been explored for that purpose, the most naive (although amenable to many optimisation possibilities) is the state-vector, where an n-qubit quantum state is represented by a 2^n-sized complex vector, and the application of quantum gates results in modifications in the vector [90]. Another method is that of tensor networks, where the state is represented by a set of multidimensional matrices connected together following a predefined topology. This structure may be more amenable to simulate the effect of quantum operators, and benefits from efficient data compression [19]. Other methods include stabiliser-based simulation, whereby we use the fact that the Clifford fragment is

efficiently simulable [2], so we try to decompose the non-Clifford parts as linear combinations of Clifford parts. This method has the benefit of scaling exponentially with the number of non-Clifford gates rather than the number of qubits [22].

Finally, developments have already started in the integration of quantum computing in the workflow of HPC developers. For instance, the Q-pragma [69] is a C++ framework developed to offload specific tasks to the quantum processing unit. Workflow abstractions and workflow management are being investigated [34] to propose the best ways to interact with the QPU.

9 Conclusion

The field of quantum computing has boomed over the past 20 years, following the advent of actual quantum computers. Once an almost completely theoretical field, it now evolves hand-in-hand with experimenters, who can now test some of the theoretical results, and pose new problems. The putative large-scale quantum computing era, where virtually any quantum algorithm or protocol can be implemented using fault-tolerant schemes, promises powerful results, as well as interesting new challenges, in the way we program, verify, or implement such algorithms.

Contributors

Alex Bredariol Grilo (CNRS, LIP6) and Renaud Vilmart (Inria, LMF).

References

1 Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. *Theory of Computing*, 9(4):143–252, 2013. URL: https://theoryofcomputing.org/articles/v009a004, doi:10.4086/toc.2013.v009a004.

2 Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70(5), November 2004. URL: http://dx.doi.org/10.1103/PhysRevA.70.052328, doi:10.1103/physreva.70.052328.

3 Parosh Aziz Abdulla, Yo-Ga Chen, Yu-Fang Chen, Lukáš Holík, Ondřej Lengál, Jyun-Ao Lin, Fang-Yi Lo, and Wei-Lun Tsai. Verifying quantum circuits with level-synchronized tree automata. *Proc. ACM Program. Lang.*, 9(POPL), January 2025. URL: https://doi.org/10.1145/3704868, doi:10.1145/3704868.

4 Dorit Aharonov, Andris Ambainis, Julia Kempe, and Umesh V. Vazirani. Quantum walks on graphs. In Jeffrey Scott Vitter, Paul G. Spirakis, and Mihalis Yannakakis, editors, *Proceedings on 33rd Annual ACM Symposium on Theory of Computing, July 6-8, 2001, Heraklion, Crete, Greece*, pages 50–59. ACM, 2001. URL: https://doi.org/10.1145/380752.380758, doi:10.1145/380752.380758.

5 Dorit Aharonov, Itai Arad, and Thomas Vidick. Guest column: the quantum pcp conjecture. *SIGACT News*, 44(2):47–79, June 2013. URL: https://doi.org/10.1145/2491533.2491549, doi:10.1145/2491533.2491549.

6 Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error rate. *SIAM Journal on Computing*, 38(4):1207–1282, 2008. URL: https://doi.org/10.1137/S0097539799359385, arXiv:https://doi.org/10.1137/S0097539799359385, doi:10.1137/S0097539799359385.

7 Dorit Aharonov, Michael Ben-Or, and Elad Eban. Interactive proofs for quantum computations. In Andrew Chi-Chih Yao, editor, *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 5-7, 2010. Proceedings*, pages 453–469. Tsinghua University

Press, 2010. URL: http://conference.iiis.tsinghua.edu.cn/ICS2010/content/papers/35.html.

8 Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev. Adiabatic quantum computation is equivalent to standard quantum computation. *SIAM Review*, 50(4):755–787, 2008. URL: https://doi.org/10.1137/080734479, arXiv:https://doi.org/10.1137/080734479, doi:10.1137/080734479.

9 Andris Ambainis. Quantum lower bounds by quantum arguments. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, pages 636–643. ACM, 2000.

10 Matthew Amy, Dmitri Maslov, and Michele Mosca. Polynomial-time t-depth optimization of clifford+t circuits via matroid partitioning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 33(10):1476–1489, October 2014. URL: http://dx.doi.org/10.1109/TCAD.2014.2341953, doi:10.1109/tcad.2014.2341953.

11 Mateus Araújo, Fabio Costa, and Časlav Brukner. Computational advantage from quantum-controlled ordering of gates. *Physical Review Letters*, 113(25), December 2014. URL: http://dx.doi.org/10.1103/PhysRevLett.113.250402, doi:10.1103/physrevlett.113.250402.

12 P. Arrighi. An overview of quantum cellular automata. *Natural Computing*, 18(4):885–899, September 2019. URL: http://dx.doi.org/10.1007/s11047-019-09762-6, doi:10.1007/s11047-019-09762-6.

13 Srinivasan Arunachalam and Ronald de Wolf. Guest column: A survey of quantum learning theory. *SIGACT News*, 48(2):41–67, June 2017. URL: https://doi.org/10.1145/3106700.3106710, doi:10.1145/3106700.3106710.

14 Benjamin Audoux and Alain Couvreur. On tensor products of css codes. *Annales de l'Institut Henri Poincaré D, Combinatorics, Physics and their Interactions*, 6(2):239–287, March 2019. URL: http://dx.doi.org/10.4171/aihpd/71, doi:10.4171/aihpd/71.

15 Martin Avanzini, Georg Moser, Romain Péchoux, Simon Perdrix, and Vladimir Zamdzhiev. Quantum Expectation Transformers for Cost Analysis. In *Symposium on Logic In Computer Science LICS '22*, Haifa, Israel, August 2022. URL: https://inria.hal.science/hal-03540366.

16 James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma. One-way functions imply secure computation in a quantum world. In *Advances in Cryptology–CRYPTO 2021: 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16–20, 2021, Proceedings, Part I 41*, pages 467–496. Springer, 2021.

17 Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. *J. ACM*, 48(4):778–797, 2001.

18 Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. In *EEE International Conference on Computers, Systems and Signal Processing*, 1984.

19 Aleksandr Berezutskii, Minzhao Liu, Atithi Acharya, Roman Ellerbrock, Johnnie Gray, Reza Haghshenas, Zichang He, Abid Khan, Viacheslav Kuzmin, Dmitry Lyakh, Danylo Lykov, Salvatore Mandrà, Christopher Mansell, Alexey Melnikov, Artem Melnikov, Vladimir Mironov, Dmitry Morozov, Florian Neukart, Alberto Nocera, Michael A. Perlin, Michael Perelshtein, Matthew Steinberg, Ruslan Shaydulin, Benjamin Villalonga, Markus Pflitsch, Marco Pistoia, Valerii Vinokur, and Yuri Alexeev. Tensor networks for quantum computing. *Nature Reviews Physics*, 7(10):581–593, July 2025. URL: http://dx.doi.org/10.1038/s42254-025-00853-1, doi:10.1038/s42254-025-00853-1.

20 Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549, 2017.

21 Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation, 2002. URL: http://dx.doi.org/10.1090/conm/305/05215, doi:10.1090/conm/305/05215.

22 Sergey Bravyi, Dan Browne, Padraic Calpin, Earl Campbell, David Gosset, and Mark Howard. Simulation of quantum circuits by low-rank stabilizer decompositions. *Quantum*, 3:181,

September 2019. URL: http://dx.doi.org/10.22331/q-2019-09-02-181, doi:10.22331/q-2019-09-02-181.

23 Nikolas P. Breuckmann and Jens Niklas Eberhardt. Quantum low-density parity-check codes. *PRX Quantum*, 2(4), October 2021. URL: http://dx.doi.org/10.1103/PRXQuantum.2.040101, doi:10.1103/prxquantum.2.040101.

24 H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. Van den Nest. Measurement-based quantum computation. *Nature Physics*, 5(1):19–26, January 2009. URL: http://dx.doi.org/10.1038/nphys1157, doi:10.1038/nphys1157.

25 Jop Briet and Ronald de Wolf. Locally Decodable Quantum Codes. In Susanne Albers and Jean-Yves Marion, editors, *26th International Symposium on Theoretical Aspects of Computer Science*, volume 3 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 219–230, Dagstuhl, Germany, 2009. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.STACS.2009.1813, doi:10.4230/LIPIcs.STACS.2009.1813.

26 Anne Broadbent. How to verify a quantum computation. *Theory Comput.*, 14(1):1–37, 2018. URL: https://doi.org/10.4086/toc.2018.v014a011, doi:10.4086/TOC.2018.V014A011.

27 Anne Broadbent, Joseph F. Fitzsimons, and Elham Kashefi. Universal blind quantum computation. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, Atlanta, Georgia, USA, October 25-27, 2009*, pages 517–526. IEEE Computer Society, 2009. URL: https://doi.org/10.1109/FOCS.2009.36, doi:10.1109/FOCS.2009.36.

28 Anne Broadbent and Rabib Islam. Quantum encryption with certified deletion. In Rafael Pass and Krzysztof Pietrzak, editors, *Theory of Cryptography*, pages 92–122, 2020.

29 Anne Broadbent and Sébastien Lord. Uncloneable Quantum Encryption via Oracles. In *15th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020)*, Leibniz International Proceedings in Informatics (LIPIcs), pages 4:1–4:22, 2020.

30 Daniel E Browne, Elham Kashefi, Mehdi Mhalla, and Simon Perdrix. Generalized flow and determinism in measurement-based quantum computation. *New Journal of Physics*, 9(8):250–250, August 2007. URL: http://dx.doi.org/10.1088/1367-2630/9/8/250, doi:10.1088/1367-2630/9/8/250.

31 Timothée Goubault de Brugière, Marc Baboulin, Benoît Valiron, Simon Martiel, and Cyril Allouche. Reducing the depth of linear reversible quantum circuits. *IEEE Transactions on Quantum Engineering*, 2:1–22, 2021. doi:10.1109/TQE.2021.3091648.

32 Harry Buhrman and Ronald de Wolf. Communication complexity lower bounds by polynomials. In *Proceedings of the 16th Annual IEEE Conference on Computational Complexity*, pages 120–130. IEEE Computer Society, 2001.

33 Lukas Burgholzer, Rudy Raymond, and Robert Wille. Verifying results of the ibm qiskit quantum circuit compilation flow. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 356–365, 2020. doi:10.1109/QCE49297.2020.00051.

34 Silvina Caino-Lores, Daniel Claudino, Eugene Dumitrescu, Travis S. Humble, Sonia Lopez Alarcon, and Elaine Wong. *Rethinking Programming Paradigms in the QC-HPC Context*, page 84–91. Springer Nature Switzerland, 2024. URL: http://dx.doi.org/10.1007/978-3-031-61763-8_8, doi:10.1007/978-3-031-61763-8_8.

35 Earl T. Campbell, Barbara M. Terhal, and Christophe Vuillot. Roads towards fault-tolerant universal quantum computation. *Nature*, 549(7671):172–179, September 2017. URL: http://dx.doi.org/10.1038/nature23460, doi:10.1038/nature23460.

36 Kostia Chardonnet, Marc de Visme, Benoît Valiron, and Renaud Vilmart. The many-worlds calculus. *Logical Methods in Computer Science*, Volume 21, Issue 2, May 2025. URL: http://dx.doi.org/10.46298/lmcs-21(2:13)2025, doi:10.46298/lmcs-21(2:13)2025.

37 Christophe Chareton, Sébastien Bardin, François Bobot, Valentin Perrelle, and Benoît Valiron. *An Automated Deductive Verification Framework for Circuit-building Quantum Programs*,

page 148–177. Springer International Publishing, 2021. URL: http://dx.doi.org/10.1007/978-3-030-72019-3_6, doi:10.1007/978-3-030-72019-3_6.

38 Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman. Exponential algorithmic speedup by a quantum walk. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '03, page 59–68. Association for Computing Machinery, 2003.

39 Giulio Chiribella, Giacomo Mauro D'Ariano, Paolo Perinotti, and Benoit Valiron. Quantum computations without definite causal structure. *Physical Review A*, 88(2), August 2013. URL: http://dx.doi.org/10.1103/PhysRevA.88.022318, doi:10.1103/physreva.88.022318.

40 Pierre Clairambault and Marc de Visme. Full abstraction for the quantum lambda-calculus. *Proc. ACM Program. Lang.*, 4(POPL), December 2019. URL: https://doi.org/10.1145/3371131, doi:10.1145/3371131.

41 Alexandre Clément, Noé Delorme, and Simon Perdrix. Minimal equational theories for quantum circuits. In *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '24, New York, NY, USA, 2024. Association for Computing Machinery. URL: https://doi.org/10.1145/3661814.3662088, doi:10.1145/3661814.3662088.

42 Alexandre Clément, Nicolas Heurtel, Shane Mansfield, Simon Perdrix, and Benoît Valiron. A complete equational theory for quantum circuits. In *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13, 2023. doi:10.1109/LICS56636.2023.10175801.

43 Bob Coecke and Ross Duncan. Interacting quantum observables: categorical algebra and diagrammatics. *New Journal of Physics*, 13(4):043016, April 2011. URL: http://dx.doi.org/10.1088/1367-2630/13/4/043016, doi:10.1088/1367-2630/13/4/043016.

44 Andrea Coladangelo, Alex B. Grilo, Stacey Jeffery, and Thomas Vidick. Verifier-on-a-leash: New schemes for verifiable delegated quantum computation, with quasilinear resources. *Theory Comput.*, 20:1–87, 2024.

45 Brian Coyle, Snehal Raj, Natansh Mathur, El Amine Cherrat, Nishant Jain, Skander Kazdaghli, and Iordanis Kerenidis. Training-efficient density quantum machine learning. *npj Quantum Information*, 11(1), November 2025. URL: http://dx.doi.org/10.1038/s41534-025-01099-6, doi:10.1038/s41534-025-01099-6.

46 Andrew W. Cross, Lev S. Bishop, Sarah Sheldon, Paul D. Nation, and Jay M. Gambetta. Validating quantum computers using randomized model circuits. *Physical Review A*, 100(3), September 2019. URL: http://dx.doi.org/10.1103/PhysRevA.100.032328, doi:10.1103/physreva.100.032328.

47 Ivan B. Damgard, Serge Fehr, Louis Salvail, and Christian Schaffner. Cryptography in the bounded quantum-storage model. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '05, page 449–458, USA, 2005. IEEE Computer Society. URL: https://doi.org/10.1109/SFCS.2005.30, doi:10.1109/SFCS.2005.30.

48 Liliane-Joy Dandy, Emmanuel Jeandel, and Vladimir Zamdzhiev. Type-safe quantum programming in idris. In Thomas Wies, editor, *Programming Languages and Systems*, pages 507–534, Cham, 2023. Springer Nature Switzerland.

49 Vincent Danos and Elham Kashefi. Determinism in the one-way model. *Phys. Rev. A*, 74:052310, Nov 2006. URL: https://link.aps.org/doi/10.1103/PhysRevA.74.052310, doi:10.1103/PhysRevA.74.052310.

50 Niel de Beaudrap and Dominic Horsman. The zx calculus is a language for surface code lattice surgery. *Quantum*, 4:218, 2020.

51 T. Goubault de Brugière, M. Baboulin, B. Valiron, and C. Allouche. Quantum circuits synthesis using householder transformations. *Comput. Phys. Commun.*, 248:107001, 2020.

52 Anne-Catherine de la Hamette, Viktoria Kabel, Marios Christodoulou, and Časlav Brukner. Indefinite causal order and quantum coordinates. *Phys. Rev. Lett.*, 135:141402, Oct 2025. URL: https://link.aps.org/doi/10.1103/bnkn-4p3f, doi:10.1103/bnkn-4p3f.

53 David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558, 12 1992. URL: https://doi.org/10.1098/rspa.1992.0167, arXiv:https://royalsocietypublishing.org/rspa/article-pdf/439/1907/553/68698/rspa.1992.0167.pdf, doi:10.1098/rspa.1992.0167.

54 Arpit Dua, Nathanan Tantivasadakarn, Joseph Sullivan, and Tyler D Ellison. Engineering 3d floquet codes by rewinding. *PRX Quantum*, 5(2):020305, 2024.

55 Yfke Dulek, Alex B. Grilo, Stacey Jeffery, Christian Majenz, and Christian Schaffner. Secure multi-party quantum computation with a dishonest majority. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, volume 12107 of *Lecture Notes in Computer Science*, pages 729–758. Springer, 2020.

56 Ross Duncan, Aleks Kissinger, Simon Perdrix, and John Van De Wetering. Graph-theoretic simplification of quantum circuits with the zx-calculus. *Quantum*, 4:279, 2020.

57 Ross Duncan and Simon Perdrix. Rewriting measurement-based quantum computations with generalised flow. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide, and Paul G. Spirakis, editors, *Automata, Languages and Programming*, pages 285–296, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

58 Frédéric Dupuis, Jesper Buus Nielsen, and Louis Salvail. Secure two-party quantum evaluation of unitaries against specious adversaries. In Tal Rabin, editor, *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference*, volume 6223 of *Lecture Notes in Computer Science*, pages 685–706, 2010.

59 Frank Arute et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, October 2019. URL: http://dx.doi.org/10.1038/s41586-019-1666-5, doi:10.1038/s41586-019-1666-5.

60 Rajeev Acharya et al. Quantum error correction below the surface code threshold. *Nature*, 638(8052):920–926, December 2024. URL: http://dx.doi.org/10.1038/s41586-024-08449-y, doi:10.1038/s41586-024-08449-y.

61 Ulysse Réglade et al. Quantum control of a cat qubit with bit-flip times exceeding ten seconds. *Nature*, 629(8013):778–783, May 2024. URL: http://dx.doi.org/10.1038/s41586-024-07294-3, doi:10.1038/s41586-024-07294-3.

62 Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm, 2014. URL: https://arxiv.org/abs/1411.4028, arXiv:1411.4028.

63 Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution, 2000. URL: https://arxiv.org/abs/quant-ph/0001106, arXiv:quant-ph/0001106.

64 Omar Fawzi, Antoine Grospellier, and Anthony Leverrier. Constant overhead quantum fault tolerance with quantum expander codes. *Commun. ACM*, 64(1):106–114, December 2020. URL: https://doi.org/10.1145/3434163, doi:10.1145/3434163.

65 Samuel Fiorini, Serge Massar, Sebastian Pokutta, Hans Raj Tiwary, and Ronald de Wolf. Exponential lower bounds for polytopes in combinatorial optimization. *J. ACM*, 62(2), May 2015. URL: https://doi.org/10.1145/2716307, doi:10.1145/2716307.

66 Joseph F. Fitzsimons and Elham Kashefi. Unconditionally verifiable blind quantum computation. *Phys. Rev. A*, 96:012303, Jul 2017. URL: https://link.aps.org/doi/10.1103/PhysRevA.96.012303, doi:10.1103/PhysRevA.96.012303.

67 Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A*, 86(3), September 2012. URL: http://dx.doi.org/10.1103/PhysRevA.86.032324, doi:10.1103/physreva.86.032324.

68 Peng Gao, Yiwei Li, Marek Perkowski, and Xiaoyu Song. Realization of quantum oracles using symmetries of boolean functions. *Quantum Information and Computation*, 20(5 & 6):418–448, May 2020. URL: http://dx.doi.org/10.26421/QIC20.5-6-4, doi:10.26421/qic20.5-6-4.

69 Arnaud Gazda and Océane Koska. A pragma based c++ framework for hybrid quantum/classical computation. *Science of Computer Programming*, 236:103119, September 2024. URL: http://dx.doi.org/10.1016/j.scico.2024.103119, doi:10.1016/j.scico.2024.103119.

70 Sevag Gharibian, Yichen Huang, Zeph Landau, and Seung Woo Shin. Quantum hamiltonian complexity. *Found. Trends Theor. Comput. Sci.*, 10(3):159–282, 2015. URL: https://doi.org/10.1561/0400000066, doi:10.1561/0400000066.

71 Craig Gidney and Austin G. Fowler. Efficient magic state factories with a catalyzed $|CCZ\rangle$ to $2|T\rangle$ transformation. *Quantum*, 3:135, April 2019. URL: https://doi.org/10.22331/q-2019-04-30-135, doi:10.22331/q-2019-04-30-135.

72 András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 193–204, 2019.

73 András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC '19, page 193–204. ACM, June 2019. URL: http://dx.doi.org/10.1145/3313276.3316366, doi:10.1145/3313276.3316366.

74 Niels Gleinig and Torsten Hoefler. An efficient algorithm for sparse quantum state preparation. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*, pages 433–438, 2021. doi:10.1109/DAC18074.2021.9586240.

75 Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron. Quipper: a scalable quantum programming language. In *Proceedings of the 34th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '13, page 333–342, New York, NY, USA, 2013. Association for Computing Machinery. URL: https://doi.org/10.1145/2491956.2462177, doi:10.1145/2491956.2462177.

76 Alex B Grilo, Huijia Lin, Fang Song, and Vinod Vaikuntanathan. Oblivious transfer is in miniqcrypt. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 531–561. Springer, 2021.

77 Frédéric Grosshans and Philippe Grangier. Continuous variable quantum cryptography using coherent states. *Phys. Rev. Lett.*, 88:057902, Jan 2002. URL: https://link.aps.org/doi/10.1103/PhysRevLett.88.057902, doi:10.1103/PhysRevLett.88.057902.

78 Lov K. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Phys. Rev. Lett.*, 79:325–328, Jul 1997. URL: https://link.aps.org/doi/10.1103/PhysRevLett.79.325, doi:10.1103/PhysRevLett.79.325.

79 Shaojun Guo, Jinzhao Sun, Haoran Qian, Ming Gong, Yukun Zhang, Fusheng Chen, Yangsen Ye, Yulin Wu, Sirui Cao, Kun Liu, Chen Zha, Chong Ying, Qingling Zhu, He-Liang Huang, Youwei Zhao, Shaowei Li, Shiyu Wang, Jiale Yu, Daojin Fan, Dachao Wu, Hong Su, Hui Deng, Hao Rong, Yuan Li, Kaili Zhang, Tung-Hsun Chung, Futian Liang, Jin Lin, Yu Xu, Lihua Sun, Cheng Guo, Na Li, Yong-Heng Huo, Cheng-Zhi Peng, Chao-Yang Lu, Xiao Yuan, Xiaobo Zhu, and Jian-Wei Pan. Experimental quantum computational chemistry with optimized unitary coupled cluster ansatz. *Nature Physics*, 20(8):1240–1246, June 2024. URL: http://dx.doi.org/10.1038/s41567-024-02530-z, doi:10.1038/s41567-024-02530-z.

80 Aparna Gupte, Jiahui Liu, Justin Raizes, Bhaskar Roberts, and Vinod Vaikuntanathan. Quantum one-time programs, revisited. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, 2025.

81 Emmanuel Hainry, Romain Péchoux, and Mário Silva. A polytime quantum programming language. *ACM Transactions on Quantum Computing*, 7(1), November 2025. URL: https://doi.org/10.1145/3769851, doi:10.1145/3769851.

82 Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15), October 2009. URL: http://dx.doi.org/10.1103/PhysRevLett.103.150502, doi:10.1103/physrevlett.103.150502.

83 Matthew B. Hastings and Jeongwan Haah. Dynamically Generated Logical Qubits. *Quantum*, 5:564, October 2021. URL: https://doi.org/10.22331/q-2021-10-19-564, doi:10.22331/q-2021-10-19-564.

84 Nicolas Heurtel, Andreas Fyrillas, Grégoire De Gliniasty, Raphaël Le Bihan, Sébastien Malherbe, Marceau Pailhas, Eric Bertasi, Boris Bourdoncle, Pierre-Emmanuel Emeriau, Rawad Mezher, Luka Music, Nadia Belabas, Benoît Valiron, Pascale Senellart, Shane Mansfield, and Jean Senellart. Perceval: A Software Platform for Discrete Variable Photonic Quantum Computing. *Quantum*, 7:931, February 2023. URL: https://hal.science/hal-03874624, doi:10.22331/q-2023-02-21-931.

85 Kesha Hietala, Robert Rand, Shih-Han Hung, Xiaodi Wu, and Michael Hicks. A verified optimizer for quantum circuits. *Proc. ACM Program. Lang.*, 5(POPL), January 2021. URL: https://doi.org/10.1145/3434318, doi:10.1145/3434318.

86 Mohsin Iqbal, Nathanan Tantivasadakarn, Ruben Verresen, Sara L. Campbell, Joan M. Dreiling, Caroline Figgatt, John P. Gaebler, Jacob Johansen, Michael Mills, Steven A. Moses, Juan M. Pino, Anthony Ransford, Mary Rowe, Peter Siegfried, Russell P. Stutz, Michael Foss-Feig, Ashvin Vishwanath, and Henrik Dreyer. Non-abelian topological order and anyons on a trapped-ion processor. *Nature*, 626(7999):505–511, February 2024. URL: http://dx.doi.org/10.1038/s41586-023-06934-4, doi:10.1038/s41586-023-06934-4.

87 Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. QIP = PSPACE. *J. ACM*, 58(6):30:1–30:27, 2011. URL: https://doi.org/10.1145/2049697.2049704, doi:10.1145/2049697.2049704.

88 Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III*, volume 10993 of *Lecture Notes in Computer Science*, pages 126–152. Springer, 2018. URL: https://doi.org/10.1007/978-3-319-96878-0_5, doi:10.1007/978-3-319-96878-0_5.

89 Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. Mip*=re, 2022. URL: https://arxiv.org/abs/2001.04383, arXiv:2001.04383.

90 Tyson Jones, Anna Brown, Ian Bush, and Simon C. Benjamin. Quest and high performance simulation of quantum computers. *Scientific Reports*, 9(1), July 2019. URL: http://dx.doi.org/10.1038/s41598-019-47174-9, doi:10.1038/s41598-019-47174-9.

91 Iordanis Kerenidis and Ronald de Wolf. Exponential lower bound for 2-query locally decodable codes via a quantum argument. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '03, page 106–115, New York, NY, USA, 2003. Association for Computing Machinery. URL: https://doi.org/10.1145/780542.780560, doi:10.1145/780542.780560.

92 Iordanis Kerenidis and Anupam Prakash. Quantum recommendation systems. In *8th Innovations in Theoretical Computer Science Conference, ITCS 2017, Berkeley, CA, USA, January 9-11, 2017*, volume 67 of *LIPIcs*, pages 49:1–49:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

93 Aleks Kissinger and John van de Wetering. Reducing the number of non-clifford gates in quantum circuits. *Phys. Rev. A*, 102:022406, Aug 2020. URL: https://link.aps.org/doi/10.1103/PhysRevA.102.022406, doi:10.1103/PhysRevA.102.022406.

94 Aleks Kissinger, John van de Wetering, and Renaud Vilmart. Classical Simulation of Quantum Circuits with Partial and Graphical Stabiliser Decompositions. In François Le Gall and Tomoyuki Morimae, editors, *17th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2022)*, volume 232 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 5:1–5:13, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.TQC.2022.5, doi:10.4230/LIPIcs.TQC.2022.5.

95 A Yu Kitaev. Quantum computations: algorithms and error correction. *Russian Mathematical Surveys*, 52(6):1191–1249, December 1997. URL: http://dx.doi.org/10.1070/RM1997v052n06ABEH002155, doi:10.1070/rm1997v052n06abeh002155.

96 Alexei Y. Kitaev, A. H. Shen, and Mikhail N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate studies in mathematics*. American Mathematical Society, 2002. URL: https://bookstore.ams.org/gsm-47/.

97 A.Yu. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, January 2003. URL: http://dx.doi.org/10.1016/S0003-4916(02)00018-0, doi:10.1016/s0003-4916(02)00018-0.

98 A.Yu. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, January 2003. URL: http://dx.doi.org/10.1016/S0003-4916(02)00018-0, doi:10.1016/s0003-4916(02)00018-0.

99 Emanuel Knill, Raymond Laflamme, and Wojciech H. Zurek. Resilient quantum computation. *Science*, 279(5349):342–345, 1998. URL: https://www.science.org/doi/abs/10.1126/science.279.5349.342, arXiv:https://www.science.org/doi/pdf/10.1126/science.279.5349.342, doi:10.1126/science.279.5349.342.

100 William Kretschmer, Luowen Qian, Makrand Sinha, and Avishay Tal. Quantum cryptography in algorithmica. In Barna Saha and Rocco A. Servedio, editors, *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 1589–1602. ACM, 2023. URL: https://doi.org/10.1145/3564246.3585225, doi:10.1145/3564246.3585225.

101 Hlér Kristjánsson, Giulio Chiribella, Sina Salek, Daniel Ebler, and Matthew Wilson. Resource theories of communication. *New Journal of Physics*, 22(7):073014, July 2020. URL: http://dx.doi.org/10.1088/1367-2630/ab8ef7, doi:10.1088/1367-2630/ab8ef7.

102 Anthony Leverrier. Security of continuous-variable quantum key distribution via a gaussian de finetti reduction. *Phys. Rev. Lett.*, 118:200501, May 2017. URL: https://link.aps.org/doi/10.1103/PhysRevLett.118.200501, doi:10.1103/PhysRevLett.118.200501.

103 Anthony Leverrier, Simon Apers, and Christophe Vuillot. Quantum xyz product codes. *Quantum*, 6:766, July 2022. URL: http://dx.doi.org/10.22331/q-2022-07-14-766, doi:10.22331/q-2022-07-14-766.

104 Anthony Leverrier and Gilles Zémor. Quantum tanner codes. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 872–883, 2022. doi:10.1109/FOCS54457.2022.00117.

105 Anthony Leverrier and Gilles Zémor. Decoding quantum tanner codes. *IEEE Transactions on Information Theory*, 69(8):5100–5115, 2023. doi:10.1109/TIT.2023.3267945.

106 Hoi-Kwong Lo and Hoi Fung Chau. Is quantum bit commitment really possible? *Physical Review Letters*, 1997.

107 Angus Lowe, Matija Medvidović, Anthony Hayes, Lee J. O'Riordan, Thomas R. Bromley, Juan Miguel Arrazola, and Nathan Killoran. Fast quantum circuit cutting with randomized measurements. *Quantum*, 7:934, March 2023. URL: https://doi.org/10.22331/q-2023-03-02-934, doi:10.22331/q-2023-03-02-934.

108 Dmitri Maslov and Ben Zindorf. Depth optimization of cz, cnot, and clifford circuits. *IEEE Transactions on Quantum Engineering*, 3:1–8, 2022.

109 Dominic Mayers. Unconditionally secure quantum bit commitment is impossible. *Physical review letters*, 1997.

110 Nikolaj Moll, Panagiotis Barkoutsos, Lev S Bishop, Jerry M Chow, Andrew Cross, Daniel J Egger, Stefan Filipp, Andreas Fuhrer, Jay M Gambetta, Marc Ganzhorn, Abhinav Kandala, Antonio Mezzacapo, Peter Müller, Walter Riess, Gian Salis, John Smolin, Ivano Tavernelli, and Kristan Temme. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Science and Technology*, 3(3):030503, June 2018. URL: http://dx.doi.org/10.1088/2058-9565/aab822, doi:10.1088/2058-9565/aab822.

111 Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

112 Arnaud Gazda Océane Koska, Marc Baboulin. A mixed-precision quantum-classical algorithm for solving linear systems. *IPDPS 2025 - 39th IEEE International Parallel and Distributed Processing Symposium Workshops*, pages 501–508, 2025.

113 Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical ldpc codes. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2022, page 375–388, New York, NY, USA, 2022. Association for Computing Machinery. URL: https://doi.org/10.1145/3519935.3520017, doi:10.1145/3519935.3520017.

114 Anouk Paradis, Jasper Dekoninck, Benjamin Bichsel, and Martin Vechev. Synthetiq: Fast and versatile quantum circuit synthesis. *Proc. ACM Program. Lang.*, 8(OOPSLA1), April 2024. URL: https://doi.org/10.1145/3649813, doi:10.1145/3649813.

115 Jennifer Paykin, Robert Rand, and Steve Zdancewic. Qwire: a core language for quantum circuits. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages*, POPL '17, page 846–858, New York, NY, USA, 2017. Association for Computing Machinery. URL: https://doi.org/10.1145/3009837.3009894, doi:10.1145/3009837.3009894.

116 Tom Peham, Lukas Burgholzer, and Robert Wille. Equivalence checking of quantum circuits with the zx-calculus. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 12(3):662–675, 2022. doi:10.1109/JETCAS.2022.3202204.

117 Tianyi Peng, Aram W. Harrow, Maris Ozols, and Xiaodi Wu. Simulating large quantum circuits on a small quantum computer. *Physical Review Letters*, 125(15), October 2020. URL: http://dx.doi.org/10.1103/PhysRevLett.125.150504, doi:10.1103/physrevlett.125.150504.

118 Simon Perdrix. *Quantum Entanglement Analysis Based on Abstract Interpretation*, page 270–282. Springer Berlin Heidelberg. URL: http://dx.doi.org/10.1007/978-3-540-69166-2_18, doi:10.1007/978-3-540-69166-2_18.

119 Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5, 2014.

120 Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113:130503, Sep 2014. URL: https://link.aps.org/doi/10.1103/PhysRevLett.113.130503, doi:10.1103/PhysRevLett.113.130503.

121 Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), September 2009. URL: https://doi.org/10.1145/1568318.1568324, doi:10.1145/1568318.1568324.

122 Ben W. Reichardt, Falk Unger, and Umesh V. Vazirani. Classical command of quantum systems. *Nat.*, 496(7446):456–460, 2013. URL: https://doi.org/10.1038/nature12035, doi:10.1038/NATURE12035.

123 Renato Renner. Security of quantum key distribution, 2006. URL: https://arxiv.org/abs/quant-ph/0512258, arXiv:quant-ph/0512258.

124 Diego Ruiz, Jérémie Guillaud, Anthony Leverrier, Mazyar Mirrahimi, and Christophe Vuillot. Ldpc-cat codes for low-overhead quantum computing in 2d. *Nature Communications*, 16(1), January 2025. URL: http://dx.doi.org/10.1038/s41467-025-56298-8, doi:10.1038/s41467-025-56298-8.

125 Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. The quest for a quantum neural network. *Quantum Information Processing*, 13(11), 2014.

126 Peter Selinger. *A Brief Survey of Quantum Programming Languages*, page 1–6. Springer Berlin Heidelberg, 2004. URL: http://dx.doi.org/10.1007/978-3-540-24754-8_1, doi:10.1007/978-3-540-24754-8_1.

127 Peter Selinger. Efficient clifford+t approximation of single-qubit operators. *Quantum Info. Comput.*, 15(1–2):159–180, January 2015.

128 Peter Selinger and Benoît Valiron. Quantum Lambda Calculus. In Simon J. Gay and Ian Mackie, editors, *Semantic Techniques in Quantum Computation*, pages 135–172. Cambridge University Press, Cambridge, November 2009. URL: `https://www.mscs.dal.ca/~selinger/papers/qlambdabook.pdf`, `doi:10.1017/CBO9781139193313.005`.

129 P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, SFCS '94, page 124–134, 1994. `doi:10.1109/SFCS.1994.365700`.

130 D.R. Simon. On the power of quantum computation. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 116–123, 1994. `doi:10.1109/SFCS.1994.365701`.

131 Marcos Yukio Siraichi, Vinicius Fernandes Dos Santos, Caroline Collange, and Fernando Magno Quintão Pereira. Qubit allocation as a combination of subgraph isomorphism and token swapping. In *OOPSLA*, volume 3, pages 1–29, Athens, Greece, October 2019. URL: `https://inria.hal.science/hal-02316820`, `doi:10.1145/3360546`.

132 Seyon Sivarajah, Silas Dilkes, Alexander Cowtan, Will Simmons, Alec Edgington, and Ross Duncan. t|ket): a retargetable compiler for NISQ devices. *Quantum Science and Technology*, 6(1):014003, November 2020. URL: `http://dx.doi.org/10.1088/2058-9565/ab8e92`, `doi:10.1088/2058-9565/ab8e92`.

133 Christoph Sünderhauf, Earl Campbell, and Joan Camps. Block-encoding structured matrices for data input in quantum computing. *Quantum*, 8:1226, January 2024. URL: `https://doi.org/10.22331/q-2024-01-11-1226`, `doi:10.22331/q-2024-01-11-1226`.

134 Ewin Tang. Dequantizing algorithms to understand quantum advantage in machine learning. *Nature Reviews Physics*, 4, 2022.

135 Jean-Pierre Tillich and Gilles Zémor. Quantum ldpc codes with positive rate and minimum distance proportional to the square root of the blocklength. *IEEE Transactions on Information Theory*, 60(2):1193–1202, 2014. `doi:10.1109/TIT.2013.2292061`.

136 Umesh Vazirani and Thomas Vidick. Fully device independent quantum key distribution. *Commun. ACM*, 62(4):133, March 2019.

137 Thomas Vidick. MIP*=RE: A negative resolution to Connes' embedding problem and Tsirelson's problem. *ICM International Congress of Mathematicians 2022 July 6-14. Sections 12-14*, 2022.

138 Renaud Vilmart. A near-minimal axiomatisation of zx-calculus for pure qubit quantum mechanics. In *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–10, 2019. `doi:10.1109/LICS.2019.8785765`.

139 Aleksander Wennersteen, Kemal Bidzhiev, Mauro D'Arcangelo, Matthieu Moreau, Anton Quelle, Alexandre Dauphin, and Mourad Beji. *Hybrid Quantum Classical Algorithms: A Cloud On-Demand Viewpoint*, page 117–125. Springer Nature Switzerland, 2026. URL: `http://dx.doi.org/10.1007/978-3-032-13855-2_11`, `doi:10.1007/978-3-032-13855-2_11`.

140 Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1), 1983.

141 Mark M. Wilde. *Quantum Information Theory*. Cambridge University Press, 2 edition, 2017.

142 Robert Wille, Stefan Hillmich, and Lukas Burgholzer. *Decision Diagrams for Quantum Computing*, page 1–23. Springer International Publishing, August 2022. URL: `http://dx.doi.org/10.1007/978-3-031-15699-1_1`, `doi:10.1007/978-3-031-15699-1_1`.

143 Bujiao Wu, Xiaoyu He, Shuai Yang, Lifu Shou, Guojing Tian, Jialin Zhang, and Xiaoming Sun. Optimization of cnot circuits on limited-connectivity architecture. *Physical Review Research*, 5(1):013065, 2023.

144 Shuai Yang, Wei Zi, Bujiao Wu, Cheng Guo, Jialin Zhang, and Xiaoming Sun. Efficient quantum circuit synthesis for sat-oracle with limited ancillary qubit. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(3):868–877, 2024. `doi:10.1109/TCAD.2023.3325974`.

145 Nengkun Yu and Jens Palsberg. Quantum abstract interpretation. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*, PLDI 2021, page 542–558, New York, NY, USA, 2021. Association for Comput-

ing Machinery. URL: https://doi.org/10.1145/3453483.3454061, doi:10.1145/3453483.3454061.

146 Mark Zhandry. Quantum lightning never strikes the same state twice. In *Advances in Cryptology – EUROCRYPT 2019*, pages 408–438, Cham, 2019. Springer International Publishing.

147 Yiqing Zhou, E. Miles Stoudenmire, and Xavier Waintal. What limits the simulation of quantum computers? *Phys. Rev. X*, 10:041038, Nov 2020. URL: https://link.aps.org/doi/10.1103/PhysRevX.10.041038, doi:10.1103/PhysRevX.10.041038.

148 Julien Zylberman and Fabrice Debbasch. Efficient quantum state preparation with walsh series. *Phys. Rev. A*, 109:042401, Apr 2024. URL: https://link.aps.org/doi/10.1103/PhysRevA.109.042401, doi:10.1103/PhysRevA.109.042401.

Les vingt ans du GdR IFM, vus du GT « Calculabilités »

Les thématiques fédérées par le GT Calculabilités du GdR IFM s'inscrivent dans une tradition centrale de la recherche en informatique théorique et en logique mathématique : comprendre les limites fondamentales du calcul, leurs variations selon les modèles, et leurs interactions avec d'autres domaines des mathématiques.

Si le GT en tant que structure n'a qu'une dizaine d'années [1], les axes scientifiques qu'il rassemble constituent un fil directeur majeur des activités du GdR IFM sur cette période, avec un approfondissement constant des résultats et une diversification des modèles étudiés.

1 Calculabilité classique, logique et degrés

Un premier socle scientifique concerne la théorie classique de la calculabilité et ses liens profonds avec la logique. Les travaux menés sur les hiérarchies arithmétiques et analytiques, les degrés de calculabilité et les relations de réduction ont permis d'affiner considérablement la compréhension des frontières entre décidabilité, indécidabilité et définissabilité. Au cours des vingt dernières années, plusieurs contributions ont mis en évidence des séparations fines entre notions de calculabilité proches, ainsi que des phénomènes de non-robustesse sous relativisation ou sous contraintes de ressources, montrant que des variations apparemment mineures dans les modèles peuvent entraîner des changements profonds de pouvoir de calcul. Par exemple, les méthodes utilisées pour comprendre le problème de Post, à savoir l'existence de problèmes qui ne sont ni calculables ni calculatoirement énumérables, ont été approfondies et bien mieux comprises. On est passé d'un système à priorité à un système calquant les mécanismes de forcing utilisés en théorie des ensembles, permettant de ne plus avoir à décrire les mécanismes de calculs pour se concentrer sur ce qui est calculé.

Dans cette dynamique, les interactions avec les mathématiques à rebours et la combinatoire effective ont joué un rôle structurant. Partant d'un énoncé de combinatoire (e.g. si on colorie en deux couleurs les arêtes d'un graphe infini, il existe un sous-graphe induit infini mono-chromatique), on cherche d'une part à trouver les axiomes minimaux de l'arithmétique du second ordre permettant de prouver ce résultat, et d'autre part à comprendre sa nature calculatoire (e.g. l'ensemble mono-chromatique est-il calculable si le graphe est calculable ?). Ceci permet de relier des principes combinatoires précis à des niveaux bien identifiés de calculabilité, offrant une lecture computationnelle de résultats logiques fondamentaux, et renforçant le dialogue entre logique mathématique et théorie du calcul. Cette perspective est particulièrement visible dans l'étude systématique de théorèmes de type Ramsey – l'exemple précédent correspondant au théorème de Ramsey pour les paires [19] – et montre comment la calculabilité se lit à travers des hiérarchies fines et des séparations délicates.

Un second fil directeur a porté sur l'articulation entre *hasard effectif* et *degrés / généricité*. Mathématiquement, il existe plusieurs façons de dire qu'une propriété est vraie presque partout : d'un point de vue mesurable (l'ensemble des points où elle est vraie est de mesure pleine) ou topologiquement (l'ensemble des points où elle est vraie est un ouvert dense), qui ont toutes les deux leur pendant calculatoire. Sur ce sujet, des résultats identifient précisément ce que des oracles aléatoires permettent (ou ne permettent pas) de calculer en matière de généricité, et réciproquement [3].

1. La première édition des journées du GT ont eu lieu à Fontainebleau en 2015.

2 Calculabilité sur les réels et analyse effective

Un axe majeur des vingt dernières années concerne la calculabilité sur les structures continues, c'est-à-dire des structures mettant en jeu des éléments ayant la cardinalité des réels. Les travaux sur l'analyse effective et la calculabilité sur les espaces métriques et topologiques ont permis de mieux comprendre la nature algorithmique des objets analytiques classiques, au-delà du cadre discret, le point de départ étant le fait que toute fonction de $\mathbb{R}$ dans $\mathbb{R}$ (par exemple) est continue si et seulement si elle est calculable relativement à un oracle. Ce fait se généralise aux ensembles fermés et compacts, et à n'importe quel espace topologique, une fois qu'on a une *représentation* de cet espace.

L'étude des espaces représentables a montré que des propriétés analytiques classiques telles que la continuité ou la convergence ne garantissent en rien une bonne effectivité, et que des phénomènes systématiques de non-calculabilité peuvent apparaître même dans des cadres très réguliers [10]. Les liens établis entre topologie descriptive et calculabilité ont ainsi permis de préciser la structure effective d'espaces mathématiques usuels.

Un résultat représentatif de cette approche est l'effectivisation d'objets probabilistes continus et pas seulement des suites infinies de bits. Par exemple, dans [17], sont étudiés des espaces métriques généraux sur des objets mathématiques plus complexes (mesures, fonctions, treillis), donnant une compréhension robuste des notions de calculabilité et de hasard dans des cadres non purement discrets. Dans un autre registre, les journées du GT ont régulièrement mis en avant des contributions portant sur la calculabilité de structures topologiques et analytiques, où les questions de représentation effective sont centrales et où l'on cherche à comprendre ce que signifie concrètement « calculer » dans un espace topologique [9].

3 Calcul analogique, systèmes continus et complexité

Les modèles de calcul continus et analogiques constituent un autre axe structurant du GT. Des travaux consacrés aux équations différentielles comme modèles de calcul ont montré que des classes de calculabilité sur les réels peuvent être capturées de manière intrinsèque par des systèmes dynamiques continus, et en particulier par des équations différentielles polynomiales effectives [6].

Ces résultats contribuent à mettre sur un pied d'égalité conceptuel calcul discret et calcul continu : le calcul analogique devient un cadre où l'on peut formuler des questions de pouvoir de calcul et de complexité avec une précision comparable au cadre classique. Il a été montré que l'on peut également définir des notions naturelles de complexité dans ces modèles : le temps correspond à la longueur des trajectoires [8], et l'espace à la précision requise pour représenter les configurations [4].

Un autre apport marquant est l'étude de la *robustesse* : il ne s'agit pas seulement de savoir ce qui est calculable en théorie, mais ce qui demeure calculable sous perturbations réalistes (approximation, bruit, erreurs). Des résultats mettent en évidence des phénomènes de stabilité et de non-stabilité qui structurent l'espace des modèles analogiques [5]. Dans le même esprit, la clarification du lien entre le paradigme GPAC (Shannon) et l'analyse calculable, ainsi que la description des fonctions générées, ont joué un rôle important dans l'unification du paysage des modèles analogiques.

4 Systèmes dynamiques discrets, pavages et universalité

Les systèmes dynamiques discrets — automates cellulaires, dynamique symbolique, pavages — ont constitué un terrain privilégié pour étudier l'émergence du calcul universel. Une

ligne forte des vingt dernières années est que des systèmes localement simples peuvent simuler des calculs universels, et donc engendrer de l'indécidabilité pour des propriétés dynamiques naturelles.

Un résultat emblématique est le développement de constructions *pavages apériodiques par point fixe*. Il existe des liens forts entre pavages et calculabilité, qui s'illustrent par exemple par le fait qu'un diagramme espace-temps d'une machine de Turing peut être vu comme un pavage. Cependant toutes les constructions de problèmes indécidables, à différents niveaux de la hiérarchie arithmétique, s'appuyaient sur l'existence de pavages apériodiques, dont les constructions historiques sont en général combinatoires ou géométriques (et donc pas de nature calculatoire). Dans [13], les auteurs montrent qu'on peut obtenir un pavage apériodique par des arguments de calculabilité et plus exactement par le point fixe de Kleene : une machine qui écrit son propre code se transforme en un pavage qui se désubstitue en lui-même.

Cette approche présente à la fois un intérêt conceptuel — en fournissant une explication computationnelle unifiée de l'apériodicité — et un intérêt technique, la flexibilité de la construction permettant d'imposer des propriétés supplémentaires, afin de rendre par exemple l'ensemble de pavages robuste aux erreurs locales [13], ou minimal.

Par ailleurs, l'étude de la dynamique de réseaux d'automates a mis en évidence des phénomènes de type « théorème de Rice » en complexité : tester des propriétés non triviales du comportement dynamique devient rapidement intractable, fournissant une explication structurelle aux difficultés de vérification de dynamiques finies [15]. Dans le même esprit, plusieurs travaux sur les automates cellulaires et les systèmes discrets ont analysé finement la complexité et la décidabilité de propriétés dynamiques telles que la prédiction, l'atteignabilité ou la stabilité, contribuant à clarifier les liens entre dynamique symbolique, calculabilité et complexité [14].

5 Information algorithmique et hasard

Un autre axe important concerne l'aléa algorithmique et la théorie de l'information algorithmique. Les notions classiques (complexité de Kolmogorov, hasard, normalité) ont été revisitées sous l'angle de contraintes de ressources, en particulier dans des modèles à mémoire finie.

Une contribution marquante, explicitement identifiée dans le cadre du GT, est la structuration d'une *théorie de l'information algorithmique à états finis*. Il était connu de longue date que la normalité de Borel est étroitement liée aux machines à états finis et aux mécanismes de compression à mémoire bornée, mais ces résultats restaient dispersés.

Des travaux récents ont proposé un cadre unifié fournissant des versions à états finis de notions fondamentales de l'information algorithmique, telles que la complexité de Kolmogorov, la probabilité a priori et la complexité conditionnelle. Une contribution technique centrale est l'introduction de la notion de *mesure de complexité superadditive*, qui permet de donner des preuves plus conceptuelles et uniformes de nombreux résultats auparavant techniques [18].

6 Calculabilité transfinie et modèles de calcul proches

Le GT s'est aussi intéressé aux modèles de calcul à temps transfini, qui, contrairement aux automates cellulaires et machines de Turing classiques, ne s'arrêtent pas en un temps fini, mais en un temps ordinal. L'idée est d'ajouter, en plus du mécanisme classique permettant de passer d'un temps t au temps $t+1$ (ordinal successeur), un mécanisme expliquant comment

passer au temps t si on dispose de l'état du système à tous les temps $t' < t$ (ordinal limite). Ces modèles servent d'interface naturelle entre calculabilité et logique transfinie.

Dans ce cadre, il a été montré que les *ordinaux admissibles*, qui jouent en logique un rôle central en tant qu'ordinaux bien clos pour les fonctions élémentaires Σ_1, sont profondément liés à des propriétés algorithmiques précises des machines. Plusieurs travaux ont permis de préciser ces correspondances, en reliant la structure des calculs des machines de Turing à temps infini à la hiérarchie des ensembles constructibles de Gödel. Ceci résume la direction générale de plusieurs articles [2, 1, 11, 12] qui couvrent un spectre assez large de résultats, de la caractérisation de tout admissible comme premier ordinal non récursif avec un oracle réel (qui est explicitement proposé), jusqu'au temps minimal nécessaire pour construire les ensembles de Gödel.

7 Ouvertures contemporaines

Enfin, les activités récentes du GT montrent une ouverture vers des modèles contemporains issus de l'informatique moderne, en particulier l'apprentissage automatique et les réseaux de neurones. Abordés avec les outils de la calculabilité, de la dynamique et de la complexité, ces modèles sont analysés du point de vue de leur pouvoir de calcul, de la décidabilité de problèmes de vérification et des limites intrinsèques de leur expressivité [7]. Ces travaux prolongent naturellement les questions classiques du GT, en montrant que des modèles largement utilisés en pratique soulèvent des questions fondationnelles profondes.

Dans un autre registre, des travaux récents ont montré comment des approches issues de l'analyse de programmes et de la sécurité de l'information peuvent fournir des outils conceptuels pour raisonner sur la complexité, illustrant une autre facette des interactions entre calculabilité, analyse de programmes et contrôle de l'information [16].

8 Conclusion

La rétrospective des activités du GT Calculabilités sur les vingt dernières années met en évidence une communauté scientifique cohérente, structurée autour de questions de fond sur les limites du calcul. En combinant logique, analyse, dynamique et complexité, les travaux menés ont contribué à une compréhension plus unifiée et plus profonde de la calculabilité dans une grande diversité de modèles. Ce positionnement transversal constitue un socle solide pour les développements futurs.

Contributeurs.

Olivier Bournez, Julien Cervelle, Bruno Durand, Mathieu Hoyrup, Ludovic Patey, Romain Péchoux, Emmanuel Rauzy, Alexander Shen.

Références

1 Kenza Benjelloun and Bruno Durand. Infinite Time Turing Machines for elementary proofs on recursive reals. In *JAF 2024 - Journées sur les Arithmétiques Faibles*, Passau, Germany, September 2024. URL : `https://hal-lirmm.ccsd.cnrs.fr/lirmm-04509148`.

2 Kenza Benjelloun, Bruno Durand, and Grégory Lafitte. Writability power of ITTMs : ordinals and constructible sets. working paper or preprint, February 2023. URL : `https://hal-lirmm.ccsd.cnrs.fr/lirmm-04505369`.

3 Laurent Bienvenu and Christopher P Porter. On the interplay between effective notions of randomness and genericity. *The Journal of Symbolic Logic*, 84(1) :393–407, 2019.

4 Manon Blanc and Olivier Bournez. The complexity of computing in continuous time : space complexity is precision. In Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, editors, *ICALP : Annual International Colloquium on Automata, Languages and Programming 2024 (ICALP'2024)*. LIPICS, July 2024. doi:10.4230/LIPIcs.ICALP.2024.129.

5 Manon Blanc and Olivier Bournez. Quantifiying the robustness of dynamical systems. Relating time and space to length and precision. In Aniello Murano and Alexandra Silva, editors, *32nd EACSL Annual Conference on Computer Science Logic, CSL 2024, February 19-23, 2024, Naples, Italy*, volume 288 of *LIPIcs*, pages 17 :1–17 :20, Dagstuhl, Germany, 2024. Schloss Dagstuhl - Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.CSL.2024.17.

6 Olivier Bournez, Manuel L. Campagnolo, Daniel S. Graça, and Emmanuel Hainry. Polynomial differential equations compute all real computable functions on computable compact intervals. *Journal of Complexity*, 23(3) :317–335, June 2007. URL : http://dx.doi.org/10.1016/j.jco.2006.12.005, doi:10.1016/j.jco.2006.12.005.

7 Olivier Bournez, Johanne Cohen, and Adrian Wurm. A Universal Uniform Approximation Theorem for Neural Networks. In Paweł Gawrychowski, Filip Mazowiecki, and Michał Skrzypczak, editors, *50th International Symposium on Mathematical Foundations of Computer Science (MFCS 2025)*, volume 345 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 29 :1–29 :20, Dagstuhl, Germany, 2025. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL : https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.MFCS.2025.29, doi:10.4230/LIPIcs.MFCS.2025.Kol29.

8 Olivier Bournez, Daniel Silva Graça, and Amaury Pouly. Polynomial time corresponds to solutions of polynomial ordinary differential equations of polynomial length. *Journal of the ACM*, 64(6) :38 :1–38 :76, 2017. doi:10.1145/3127496.

9 Vasco Brattka and Emmanuel Rauzy. Effective second countability in computable analysis. In Arnold Beckmann, Isabel Oitavem, and Florin Manea, editors, *Crossroads of Computability and Logic : Insights, Inspirations, and Innovations*, volume 15764 of *LNCS*, pages 19–33, Cham, 2025. Springer. 21st Conference on Computability in Europe. doi:10.1007/978-3-031-95908-0_2.

10 Antonin Callard and Mathieu Hoyrup. Descriptive complexity on non-polish spaces. In *International Symposium on Teoretical Aspects of Computer Science (STACS)*, 2020.

11 Merlin Carl, Bruno Durand, Grégory Lafitte, and Sabrina Ouazzani. Admissibles in gaps. In Jarkko Kari, Florin Manea, and Ion Petre, editors, *Unveiling Dynamics and Complexity - 13th Conference on Computability in Europe, CiE 2017, Turku, Finland, June 12-16, 2017, Proceedings*, volume 10307 of *Lecture Notes in Computer Science*, pages 175–186. Springer, 2017. URL : https://doi.org/10.1007/978-3-319-58741-7_18, doi:10.1007/978-3-319-58741-7_18.

12 Bruno Durand and Grégory Lafitte. An algorithmic approach to characterizations of admissibles. In Florin Manea, Barnaby Martin, Daniël Paulusma, and Giuseppe Primiero, editors, *Computing with Foresight and Industry - 15th Conference on Computability in Europe, CiE 2019, Durham, UK, July 15-19, 2019, Proceedings*, volume 11558 of *Lecture Notes in Computer Science*, pages 181–192. Springer, 2019. URL : https://doi.org/10.1007/978-3-030-22996-2_16, doi:10.1007/978-3-030-22996-2_16.

13 Bruno Durand, Andrei Romashchenko, and Alexander Shen. Fixed-point tile sets and their applications. *Journal of Computer and System Sciences*, 78(3) :731–764, 2012.

14 Enrico Formenti. Complexity of local, global and universality properties in finite dynamical systems. In Jérôme Durand-Lose and György Vaszil, editors, *Machines, Computations, and Universality - 9th International Conference, MCU 2022, Debrecen, Hungary, August 31 - September 2, 2022, Proceedings*, volume 13419 of *Lecture Notes in Computer Science*, pages 1–11. Springer, 2022. URL : https://doi.org/10.1007/978-3-031-13502-6_1, doi:10.1007/978-3-031-13502-6_1.

15 Guilhem Gamard, Pierre Guillon, Kevin Perrot, and Guillaume Theyssier. Rice-like theorems for automata networks. In *38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*, 2021.

16 Emmanuel Hainry, Bruce M. Kapron, Jean-Yves Marion, and Romain Péchoux. Declassification policy for program complexity analysis. In Pawel Sobocinski, Ugo Dal Lago, and Javier Esparza, editors, *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2024, Tallinn, Estonia, July 8-11, 2024*, pages 41 :1–41 :14. ACM, 2024. URL : `https://doi.org/10.1145/3661814.3662100`, `doi:10.1145/3661814.3662100`.

17 Mathieu Hoyrup and Cristóbal Rojas. Computability of probability measures and martin-löf randomness over metric spaces. *Information and Computation*, 207(7) :830–847, 2009.

18 Alexander Kozachinskiy and Alexander Shen. Automatic kolmogorov complexity, normality, and finite-state dimension revisited. *Journal of Computer and System Sciences*, 118 :75–107, 2021.

19 Ludovic Patey and Keita Yokoyama. The proof-theoretic strength of ramsey's theorem for pairs and two colors. *Advances in Mathematics*, 330 :1034–1070, 2018.