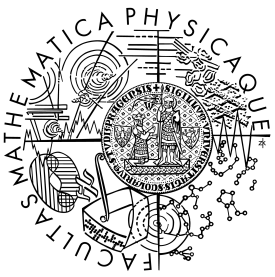

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

HABILITATION THESIS

TOTAL NP SEARCH PROBLEMS: STRUCTURE AND HARDNESS

PAVEL HUBÁČEK

Computer Science Institute of Charles University

December 2025

Pavel Hubáček © December 2025

PREFACE

This habilitation thesis is based on the nine publications listed below. All of them develop an area in computational complexity focusing on *total* NP *search problems*, their relative *structure* and plausible computational *hardness* under various cryptographic assumptions.

[1] Romain Bourneuf, Lukáš Folwarczný, Pavel Hubáček, Alon Rosen, and Nikolaj Ignatieff Schwartzbach. "PPP-Completeness and Extremal Combinatorics." In: *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*. Ed. by Yael Tauman Kalai. Vol. 251. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 22:1–22:20. DOI: 10.4230/LIPICS.ITCS.2023.22. URL: https://doi.org/10.4230/LIPIcs.ITCS.2023.22.

[2] Arka Rai Choudhuri, Pavel Hubáček, Chethan Kamath, Krzysztof Pietrzak, Alon Rosen, and Guy N. Rothblum. "Finding a Nash equilibrium is no easier than breaking Fiat-Shamir." In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*. 2019, pp. 1103–1114. DOI: 10.1145/3313276.3316400. URL: https://doi.org/10.1145/3313276.3316400.

[3] Lukáš Folwarczný, Mika Göös, Pavel Hubáček, Gilbert Maystre, and Weiqiang Yuan. "One-Way Functions vs. TFNP: Simpler and Improved." In: *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*. Ed. by Venkatesan Guruswami. Vol. 287. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, 50:1–50:14. DOI: 10.4230/LIPICS.ITCS.2024.50. URL: https://doi.org/10.4230/LIPIcs.ITCS.2024.50.

[4] Bernd Gärtner, Thomas Dueholm Hansen, Pavel Hubáček, Karel Král, Hagar Mosaad, and Veronika Slívová. "ARRIVAL: Next Stop in CLS." In: *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*. 2018, 60:1–60:13. DOI: 10.4230/LIPICS.ICALP.2018.60. URL: https://doi.org/10.4230/LIPIcs.ICALP.2018.60.

[5] Pavel Hubáček, Chethan Kamath, Karel Král, and Veronika Slívová. "On Average-Case Hardness in TFNP from One-Way Functions." In: *Theory of Cryptography - 18th International Conference, TCC 2020, Durham, NC, USA, November 16-19, 2020, Proceedings, Part III*. Ed. by Rafael Pass and Krzysztof Pietrzak. Vol. 12552. Lecture Notes in Computer Science. Springer, 2020, pp. 614–638. DOI: 10.1007/978-3-030-64381-2_22. URL: https://doi.org/10.1007/978-3-030-64381-2_22.

[6] Pavel Hubáček, Moni Naor, and Eylon Yogev. "The Journey from NP to TFNP Hardness." In: *Proceedings of the Eighth Annual Conference on Innovations in Theoretical Computer Science, ITCS 2017, Berkeley, USA, Simons Institute, January 9-11*. 2017. DOI: 10.4230/LIPICS.ITCS.2017.60. URL: https://doi.org/10.4230/LIPIcs.ITCS.2017.60.

[7] Pavel Hubáček and Jan Václavek. "On Search Complexity of Discrete Logarithm." In: *46th International Symposium on Mathematical Foundations of Computer Science, MFCS 2021, August 23-27, 2021, Tallinn, Estonia*. Ed. by Filippo Bonchi and Simon J. Puglisi. Vol. 202. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 60:1–60:16. DOI: 10.4230/LIPICS.MFCS.2021.60. URL: https://doi.org/10.4230/LIPIcs.MFCS.2021.60.

[8] Pavel Hubáček and Eylon Yogev. "Hardness of Continuous Local Search: Query Complexity and Cryptographic Lower Bounds." In: *SIAM J. Comput.* 49.6 (2020), pp. 1128–1172. DOI: 10.1137/17M1118014. URL: https://doi.org/10.1137/17M1118014.

[9] Pavel Hubáček, Erfan Khaniki, and Neil Thapen. "TFNP Intersections Through the Lens of Feasible Disjunction." In: *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*. Ed. by Venkatesan Guruswami. Vol. 287. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, 63:1–63:24. DOI: 10.4230/LIPICS.ITCS.2024.63. URL: https://doi.org/10.4230/LIPIcs.ITCS.2024.63.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my family and friends, whose unwavering love and support made this thesis possible. I am also immensely thankful for the fruitful collaborations and inspiring discussions with my colleagues—many of whom I am privileged to call my friends. Their insights and encouragement greatly enriched this work.

My research presented in this thesis was supported by Czech Science Foundation Grants 17-09142S and 19-27871X, Charles University Research Center UNCE/SCI/004, Charles University grant PRIMUS/17/SCI/9, ERC grant agreement No. 101019547, by the Cariplo CRYPTONOMEX grant, the Neuron Foundation for Support of Science, and the Academy of Sciences of the Czech Republic (RVO 67985840).

CONTENTS

1 Introduction 1
- 1.1 The Taxonomy of Total NP Search Problems 2
- 1.2 Refining the Structure Within TFNP 5
  - 1.2.1 Totality via Pigeonhole Principle 5
  - 1.2.2 Local Search over Continuous Domains 8
  - 1.2.3 The Updated Landscape 14
- 1.3 Cryptographic Lower Bounds 15
  - 1.3.1 Cryptographic Lower Bounds for CLS 16
  - 1.3.2 Average-Case Hardness in TFNP and One-Way Functions 18
- 1.4 Conclusion 23

Bibliography 25

1 INTRODUCTION

A computational search problem is defined by a relation between instances and solutions. Such a problem is called *total* if there exists a solution for every instance. For example, consider the problem of finding a collision in a *shrinking* function h, i.e., a pair of distinct elements in the domain that map to the same output under h. The fact that every shrinking function has a collision is known as the weak pigeonhole principle. Yet the mere existence of collisions does not necessarily simplify the algorithmic task of finding one, even when h is succinctly represented (for instance, by a polynomial-size circuit computing h). Indeed, the security of many widely deployed cryptographic schemes rests on the conjectured hardness of finding such collisions. Interestingly, numerous algorithmic tasks across various domains give rise to such total search problems, many of which have unresolved computational complexity.

Among total search problems, the class TFNP consists of those for which any solution can be efficiently verified, placing TFNP within the search analogue of the famous class of decision problems NP. However, contrary to NP, the complexity class TFNP is unlikely to contain any complete problems [67, 88]. Thus, one cannot simply apply the classic NP-completeness methodology by showing a TFNP problem is at least as hard as every other problem in TFNP. Instead, research on TFNP follows the approach proposed by Papadimitriou [81], characterizing total search problems via the mathematical principles that ensure their totality. In his foundational work, Papadimitriou introduced several subclasses of TFNP defined in terms of such mathematical principles, and a wide range of other subclasses have since been defined, each positioned relative to these original classes.

Over the past two decades, TFNP has become a natural meeting ground for many sub-communities of theoretical computer science beyond computational complexity, such as computational mathematics, algorithm design, logic, and cryptography. Each of these domains has brought its own viewpoints and techniques, pushing our understanding of TFNP further. The goal of this thesis is to present our results in the context of this broad and exciting research endeavor. An actual survey of the recent results on TFNP is out of scope for this type of habilitation thesis, but an interested reader can see, e.g., Goldberg and Papadimitriou [44] and Goldberg [43] and Meunier [80] for some additional interesting open problems.

1.1 THE TAXONOMY OF TOTAL NP SEARCH PROBLEMS

We begin by recalling some standard subclasses of TFNP.

PARITY ON GRAPHS: The classes PPA and PPAD capture search problems whose totality follows from parity arguments on graphs [81]. The class PPA (*Polynomial Parity Argument on graphs*) is defined via the problem LEAF: given an implicitly described undirected graph in which one degree-1 vertex is specified, find another vertex of odd degree. The totality of LEAF follows from the fact that any finite undirected graph has an even number of odd-degree vertices. The subclass of PPA called PPAD (*Polynomial Parity Argument on Directed graphs*) is defined analogously for directed graphs via the canonical problem END-OF-LINE: given succinct predecessor and successor circuits for a directed graph, in which one degree-1 vertex (the "source") is specified, find either another source or a sink.

PPAD quickly grew in importance, as it was shown to capture many important computational problems motivated by famous existential results. For example, the following are PPAD-complete problems: finding Brouwer's fixed points [81], the problem of finding (approximate) Nash equilibria in strategic games [18, 24, 79], or finding a rainbow cell in a properly colored simplicial subdivision guaranteed by Sperner's lemma [17, 81].

In comparison, our understanding of PPA-completeness was arguably rather limited until very recently (see a broader historical discussion in Deng et al. [26]). However, a recent flurry of PPA-completeness results demonstrated that it is as rich a computational complexity class as PPAD. We now know that the following are PPA-complete: the approximate Consensus-Halving problem [35], Alon's Necklace splitting problem [3] with two thieves [35], the bisection problem underlying the ham sandwich theorem [35], as well as some fixed point problems on non-orientable surfaces [26].

Recent works studying generalizations of PPA motivated by "parity" arguments modulo an arbitrary fixed $p \in \mathbb{N}$ [48, 53] have established that PPA is, in fact, a member of an infinite family of incomparable complexity classes $\{\mathsf{PPA}_p\}_{p\in\mathbb{N}}$, where $\mathsf{PPA}_2 = \mathsf{PPA}$. Interestingly, the class $\mathsf{PPAP} = \bigcap_{p\geq 2} \mathsf{PPA}_p$ is an intermediate class between PPA and PPAD relative to some oracles (see, e.g., [48]).[1] However, PPAP is currently not known to contain any total search problem not contained in PPAD. Moreover, its definition as an intersection of the infinite family of complexity classes $\{\mathsf{PPA}_p\}_{p\in\mathbb{N}}$ is, arguably, rather artificial, and it is also currently not known whether PPAP even has complete problems. On the other hand, there are important total search problems inside PPA that are neither known to be PPA-hard nor contained in PPAD, such as integer factoring [12, 66] and the problem of finding a second Hamiltonian cycle in a cubic graph derived from Smith's theorem [81, 85].

1 We introduce the name $\mathsf{PPAP} = \bigcap_{p\geq 2} \mathsf{PPA}_p$ for conciseness.

THE PIGEONHOLE PRINCIPLE: The class PPP (*Polynomial Pigeonhole Principle*) [81] and its subclass PWPP (*Polynomial Weak Pigeonhole Principle*), introduced by Jeřábek [66], consist of search problems whose totality is guaranteed by some form of the pigeonhole principle. The canonical PPP-complete problem is PIGEON: given a succinct description of a function $f\colon \{0,1\}^n \to \{0,1\}^n$, find either a pre-image of the all-zero string 0^n or a non-trivial collision $x \neq x'$ with $f(x) = f(x')$. The canonical PWPP-complete problem is COLLISION: given a circuit h that shrinks its domain, find a non-trivial collision of h, as in the cryptographic example we used to motivate total search problems above. By definition, we have PWPP $\subseteq$ PPP. The PPAD-complete problem END-OF-LINE can be reduced to PIGEON, thanks to the additional orientation (compared to PPA-complete LEAF), by applying the *strong* pigeonhole principle to the successor function of an END-OF-LINE instance. Until recently, there were far fewer completeness results for PPP and PWPP. We will discuss some of the recent progress in Section 1.2.1.

LOCAL SEARCH: The class PLS (*Polynomial Local Search*) was introduced by Megiddo and Papadimitriou [78] to capture search problems that admit natural formulations as iterated local improvement procedures. A problem is in PLS if instances specify a finite set of candidate solutions, an efficiently computable bounded objective function, and an efficiently computable neighborhood structure. The canonical PLS-complete problem is LOCAL-SEARCH: given circuits describing the neighborhood and objective function over a finite search space, find a solution that is locally optimal with respect to the neighborhood relation. The totality of every PLS problem follows from the finiteness of the search space and the assumption that each improving move strictly increases the objective value. Many natural algorithmic problems, including local search formulations of Max-Cut and other combinatorial optimization tasks [93], finding pure Nash equilibria of congestion games [30], or permutation problems on bit strings [94] have been shown PLS-complete. Daskalakis and Papadimitriou [25] defined a subclass of PLS called CLS (*Continuous Local Search*) that captures optimization problems amenable to local search over a continuous domain. CLS has been extensively studied in the past decade and we discuss it in detail in Section 1.2.2.

THE LANDSCAPE OF SUBCLASSES. In Figure 1.1, we present the known inclusions (higher classes containing the lower ones connected by a line).[2] It is widely conjectured that these classes are pairwise incomparable and strictly contained in TFNP, and that none of their complete problems admit polynomial-time algorithms. There are two main types of argument for separating their powers. A line of work initiated by Beame et al. [7] gave (in modern terms) oracle separations of the decision tree variants of the complexity classes. In that model,

2 So far, we have not mentioned the complexity class PPADS, a natural generalization of PPAD, defined by the variant of END-OF-LINE where we are interested only in sinks.

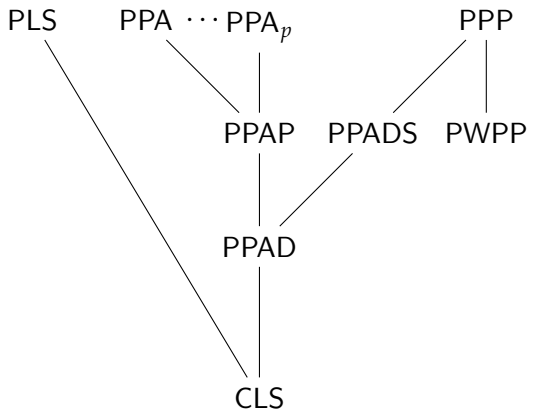

Figure 1.1: The partially ordered set defined by the known inclusions among important subclasses of TFNP.

all the inclusions depicted in Figure 1.1 are proper. More recent work uses cryptographic assumptions to argue that complete problems for these classes are infeasible for polynomial-time algorithms in the "white-box" setting, i.e., when given an explicit succinct representation of the corresponding oracles.

In the remainder of this habilitation thesis, we discuss structural results related to the subclasses CLS, PPP, and PWPP (Section 1.2). We then move on to cryptographic arguments for the infeasibility of problems in subclasses of TFNP such as, in particular, CLS, and to the limits of this methodology (Section 1.3). We conclude with some open problems motivated by recent progress (Section 1.4). The subsequent chapters contain published versions of the relevant papers.

1.2 REFINING THE STRUCTURE WITHIN TFNP

In this section, we discuss our results refining the structure of TFNP, for example by defining new total search problems and, when possible, proving their completeness for some subclass of TFNP, or by separating TFNP subclasses relative to suitable oracles. In Section 1.2.1, we discuss our new characterizations of PPP and PWPP. In Section 1.2.2, we discuss our results related to CLS.

1.2.1 *Totality via Pigeonhole Principle*

Unlike subclasses of TFNP characterized by many diverse complete problems such as PPAD, the subclasses PPP and PWPP, whose totality is guaranteed by the (weak) pigeonhole principle, had not been extensively studied until quite recently. Ban et al. [6] and Sotiraki, Zampetakis, and Zirdelis [97] studied PPP as a potential home for computational problems in number theory related to integral lattices, but beyond these works no non-trivial completeness results were known.

1.2.1.1 *Search Complexity of Discrete Logarithm and Collision Resistance*

In Hubáček and Václavek [57], we revisited PPP and PWPP from the perspective of known results on the relationship between collision-resistant hashing and the discrete logarithm problem in cryptography. Sotiraki, Zampetakis, and Zirdelis [97] proposed a generic formulation of the discrete logarithm problem, for which they asked whether it could be PPP-complete. With Václavek, we refined their formulation of the discrete logarithm problem and showed that it is indeed PPP-complete under this generic formulation.

First, we recall a generic TFNP formulation of the discrete logarithm problem, which we call Index. An instance of Index consists of:

- a finite set $\mathbb{G}$ of size s, together with a binary operation $\star\colon \mathbb{G} \times \mathbb{G} \to \mathbb{G}$ represented by a Boolean circuit f,
- a "generator" element $g \in \mathbb{G}$, an "identity" element $i \in \mathbb{G}$, and a "target" element $t \in \mathbb{G}$.

When f computes a binary operation on $\mathbb{G} = \langle g \rangle$ that satisfies the group axioms, it is possible to efficiently access the group elements via an *indexing function* $\mathcal{I}_{\mathbb{G}}\colon [s] \to \mathbb{G}$ computed as the corresponding powers of the generator g (e.g., via the classical square-and-multiply algorithm). Using this notation, the goal is to find either a discrete logarithm of the target t, i.e., $x \in [s]$ such that $\mathcal{I}_{\mathbb{G}}(x) = t$, or a collision in the indexing function, i.e., distinct $x, y \in [s]$ such that $\mathcal{I}_{\mathbb{G}}(x) = \mathcal{I}_{\mathbb{G}}(y)$ (ensuring the totality of the problem in case of discrepancies in f).

Since we cannot promise that the binary operation $\star$ on $\mathbb{G}$ satisfies the group axioms, the structure of instances of Index is quite rich. To restrict the structure and bring the formulation closer in spirit to the discrete logarithm problem, we defined DLog, where we are given an

instance of INDEX and define additional types of solutions induced by local violations of the group axioms (specifically, the associativity of $\star$). It turns out that the two problems are qualitatively different, as established by the following theorem.

Theorem 1.1 (Hubáček and Václavek [57])**.** INDEX *is* PPP*-complete and* DLOG *is* PWPP*-complete.*

Our reduction establishing the containment of DLOG in PWPP is very much inspired by constructions of collision-resistant hash functions under the hardness of the discrete logarithm problem via *claw-free permutations* [23, 91].[3] We define CLAW as a TFNP formulation of the problem of breaking claw-free permutations and, while establishing the PWPP-completeness of DLOG, we show that it is also PWPP-complete. Even though somewhat expected, this is, to the best of our knowledge, the first example of an alternative characterization of one of the standard TFNP classes via a generic cryptographic problem.

Our PPP- and PWPP-hardness results establishing the other direction of Theorem 1.1 are mostly technical and rely on the expressive power of the two problems. Ultimately, they exemplify a downside of INDEX and DLOG, as our reductions do not necessarily construct a valid group structure and instead leverage collision in the indexing function or the solutions certifying violations of the group axioms.

We then turned to a less generic formulation of the discrete logarithm problem in $\mathbb{Z}_p^*$, i.e., the multiplicative group modulo a prime p, which we call DLOG_p. In this specific context, our formulation ensures that there is a proper group and, moreover, that the solution is unique for any instance of DLOG_p.

Theorem 1.2 (Hubáček and Václavek [57])**.** $\text{DLOG}_p \in \text{PWPP} \cap \text{TFUP}$.

Here, TFUP is the subset of TFNP consisting of total search problems with *syntactically* guaranteed uniqueness of the solution for every instance, i.e., the search analogue of $\text{UP} \cap \text{coUP}$. It is unlikely that PWPP is contained in TFUP, which led us to conjecture that DLOG_p is strictly easier than the generic DLOG (see Section 1.4 for further discussion on TFUP).

RELATED WORK: Fleming et al. [36] recently studied closure properties of PPP. They showed that two non-adaptive queries to a PPP oracle are strictly more powerful than a single query, demonstrating that PPP is not closed under Turing reductions. For PWPP, Jeřábek [66] already observed that many non-adaptive queries to a PWPP oracle can be simulated by a single query. Nevertheless, it is not known whether PWPP is closed under *adaptive* Turing reductions, and the techniques of Fleming et al. [36] do not seem to apply to PWPP. It would be interesting to see whether the connection between PWPP and collision-resistant hashing might be helpful in clarifying the relationship between PWPP and its adaptive Turing closure.

3 In cryptography, a family of claw-free permutations is an efficiently sampleable family of pairs of permutations such that, given a "random" pair h_0 and h_1 of permutations from the family, it is computationally infeasible to find a *claw* for the two permutations, i.e., inputs u and v such that $h_0(u) = h_1(v)$.

1.2.1.2 *Pigeonhole Principle and Extremal Combinatorics*

In Bourneuf et al. [11], we explored a different source of TFNP problems, motivated by famous results in extremal combinatorics, and used it to give new characterizations of the classes PWPP and PPP. Recall the following classical results in extremal combinatorics:

ERDŐS–KO–RADO [29]. Any family of distinct pairwise-intersecting k-sets on a universe of size m has size at most $\binom{m-1}{k-1}$.

SPERNER [98]. The largest antichain, i.e., a family of subsets such that no member is contained in any other, on a universe of size $2n$ is unique and consists of all subsets of size n.

CAYLEY [13]. There are exactly n^{n-2} spanning trees of the complete graph on n vertices.

MANTEL [77]. Any triangle-free graph on n vertices has at most $n^2/4$ edges.

TURÁN [100]. If $G = (V, E)$ is a graph on $n = |V|$ vertices that does not contain any $r+1$-clique, then $|E| \leq (1 - \frac{1}{r})\frac{n^2}{2}$.

WARD–SZABÓ [101]. Any edge-coloring of the complete graph on n vertices with $2 \leq r \leq \sqrt{n}$ colors must contain a bichromatic triangle.

Distinguishing between *weak* and *strong* (i.e., tight) variants of the above theorems, we associated corresponding "weak" and "strong" total search problems to the Erdős–Ko–Rado, Sperner, and Cayley theorems, and we proved that the weak variants are PWPP-complete, while the corresponding strong variants are PPP-complete. We further defined TFNP problems motivated by Mantel's theorem, Turán's theorem, and the Ward–Szabó theorem on swell colorings, and showed that suitable variants of these problems are PWPP-hard and PPP-hard, while their precise placement inside PWPP or PPP remained open.

Our containment results relied on a general framework of *property-preserving encodings*, that is, efficiently computable shrinking maps whose collisions correspond exactly to the combinatorial witnesses required by the underlying theorem. Concretely, we employed Cover encodings for k-subsets [22], Prüfer codes for spanning trees [86], and Catalan factorization for chains in the Boolean lattice [28] to obtain optimal encodings that translate combinatorial structure into collisions.

For hardness, we systematically leveraged a refined *graph–hash product* construction: we combine near-extremal combinatorial objects with shrinking circuits so that any required substructure in the induced instance must arise from a collision in the hash. This approach was originally used by Komargodski, Naor, and Yogev [72] and Krajíček [74] to establish the PWPP-hardness of the total search problem associated with the famous theorem of Ramsey [89] on monochromatic subgraphs.

RELATED WORK: Ishizuka [63] recently resolved the complexity of the TFNP formulation of the Ward–Szabó theorem by proving its PPP-completeness.

Independently of our work, Pasarkar, Papadimitriou, and Yannakakis [82] also investigated total search problems arising from extremal combinatorics and their relationship to subclasses of TFNP. Their crucial observation was that many classical extremal theorems are established via an *iterated* application of the pigeonhole principle, which motivated them to introduce a new subclass PLC (*Polynomial Long Choice*). They showed that PPP $\subseteq$ PLC and that a range of extremal principles, including Ramsey's theorem, the sunflower lemma, Erdős–Ko–Rado, and Kőnig's lemma give rise to total search problems in PLC, with specialized formulations of Kőnig's lemma and the Erdős–Ko–Rado lemma being PPP-complete. Further, they introduced hierarchies based on Turán-type and coloring problems that are PPP-hard but plausibly strictly harder than PPP itself. While both works attach total search problems to Erdős–Ko–Rado and to Turán/Mantel-type statements, our results focus on characterizations of PPP and PWPP via extremal theorems, whereas Pasarkar, Papadimitriou, and Yannakakis [82] used such theorems as test cases for the broader landscape of iterated pigeonhole principles and towards separating PPP from stronger TFNP subclasses.

1.2.2 *Local Search over Continuous Domains*

Daskalakis and Papadimitriou [25] introduced the complexity class CLS to isolate, as they put it, a "benign" form of total search that simultaneously has a local-search flavour (via PLS) and a fixpoint flavour (via PPAD). On the one hand, they observed that PLS and PPAD can both be recast geometrically in terms of continuous functions with Lipschitz guarantees, and that the known complete problem for PLS $\cap$ PPAD, EITHER-FIXPOINT, is conceptually unsatisfying, since it asks for a fixpoint of one of two unrelated functions. On the other hand, they identified several natural problems previously known to lie in PLS $\cap$ PPAD but otherwise unclassified, including finding approximate fixpoints of contraction maps, solving the linear complementarity problem for P-matrices, finding approximate stationary points of low-degree polynomial objectives, computing values in simple stochastic games, and finding mixed Nash equilibria in congestion, implicit congestion, and network coordination games (see [25] for works establishing the respective inclusion). Motivated by the search for a single, structurally coherent class capturing exactly this kind of continuous local optimization with guaranteed totality, they defined CLS via the search problem CLOPT: given a continuous map f and a continuous potential p (both specified by arithmetic circuits and promised to be Lipschitz), one must either find an approximate local optimum for f with respect to p or a certificate that one of the Lipschitz conditions for f or p is violated. Finally, they showed that all of the above problems lie in CLS.

1.2.2.1 *Towards a Canonical Combinatorial Problem in* CLS

In Hubáček and Yogev [58], we initiated the study of the combinatorial structure of CLS-complete problems.

LOCAL-SEARCH AND PLS. As we discussed in Section 1.1, the class PLS was introduced to capture the complexity of "hill-climbing" style algorithms [67]. An instance of the canonical LOCAL-SEARCH problem is specified by two circuits

$$\mathsf{S} : \{0,1\}^n \to \{0,1\}^n \qquad \text{and} \qquad \mathsf{V} : \{0,1\}^n \to [2^n] \cup \{0\}.$$

We think of each vertex $u \in \{0,1\}^n$ as a feasible solution, $\mathsf{S}(u)$ as a designated "neighbor" of u, and $\mathsf{V}(u)$ as its objective value. The goal in LOCAL-SEARCH is to find a vertex u such that moving to its designated neighbor does not strictly improve the value, i.e., $\mathsf{V}(\mathsf{S}(u)) \leq \mathsf{V}(u)$. Such a vertex is a local optimum with respect to the neighborhood structure implicitly defined by S. The totality of the problem follows from the simple observation that any strictly improving path under S must terminate, because the valuation V is bounded from above and the domain is finite.

END-OF-LINE AND PPAD. The class PPAD is defined via the END-OF-LINE problem, which captures parity arguments on implicitly described directed graphs [81]. An END-OF-LINE instance consists of two circuits

$$\mathsf{S}, \mathsf{P} : \{0,1\}^n \to \{0,1\}^n$$

that define a directed graph on $\{0,1\}^n$ in which each vertex has in-degree and out-degree at most one: S plays the role of a successor function and P the predecessor function, and we put a directed edge from u to v iff it is witnessed by S and P, i.e., $\mathsf{P}(\mathsf{S}(u)) = u$. The vertex 0^n is guaranteed to be a source by the local condition $\mathsf{P}(0^n) = 0^n \neq \mathsf{S}(0^n)$. Thus, any instance of END-OF-LINE defines a collection of vertex-disjoint directed paths and cycles. The special source 0^n guarantees, by a simple parity argument, that there must exist another "unbalanced" vertex: a sink or another source. The END-OF-LINE problem asks us to find any such unbalanced vertex. In this sense, END-OF-LINE expresses a combinatorial parity-based argument for totality, in contrast to the potential-based argument in LOCAL-SEARCH.

END-OF-METERED-LINE AS A GATEWAY TO CLS. In Hubáček and Yogev [58], we defined the END-OF-METERED-LINE problem, which refines END-OF-LINE by enriching the structure with an additional valuation circuit that "meters" distance along the distinguished directed path starting at the canonical source 0^n.

Formally, an END-OF-METERED-LINE instance consists of three circuits

$$\mathsf{S}, \mathsf{P} : \{0,1\}^n \to \{0,1\}^n, \qquad \text{and} \qquad \mathsf{V} : \{0,1\}^n \to [2^n] \cup \{0\}.$$

Here, S and P play exactly the same role as in End-of-Line: they define a directed graph of maximum in- and out-degree one, where the vertex 0^n is guaranteed to be a source. The new ingredient is the valuation circuit V, which is intended to act as an *odometer*: for vertices on the directed path starting from 0^n, the value $\mathsf{V}(u)$ should equal the position of u on the line starting at 0^n, i.e., it takes $\mathsf{V}(u) - 1$ steps of S to reach u from 0^n. Concretely, the "ideal" behavior is:

- 0^n is labeled by value 1, and
- whenever u has nonzero value $\mathsf{V}(u) = i$, its successor and predecessor (with respect to S, P) on the same path should have values $i + 1$ and $i - 1$, respectively.

Of course, since End-of-Metered-Line is not a promise problem, we cannot assume that V behaves correctly everywhere. Instead, the *solutions* to the End-of-Metered-Line instance are defined to be precisely those vertices that either:

1. witness the usual End-of-Line condition, i.e., a sink or a nontrivial source, or
2. witness a deviation of V from its prescribed "metering" behavior, i.e., a vertex with a nonzero valuation whose successor or predecessor labels do not increment or decrement by one, or a vertex other than 0^n that is incorrectly assigned value 1.

Thus, the valuation circuit is guaranteed to be useful from the algorithmic perspective: either it correctly describes distance along the special path (which might be exploited), or it immediately introduces a solution by its own inconsistency. From a structural point of view, one can focus on the subgraph consisting only of vertices with nonzero valuation such that edges respect the increment condition $\mathsf{V}(\mathsf{S}(u)) = \mathsf{V}(u) + 1$. This subgraph is a collection of directed paths with strictly increasing labels and no cycles.

It is easy to see that End-of-Metered-Line is contained in PLS $\cap$ PPAD by straightforward reductions to either of the classes. Our main technical result was establishing a reduction to CLOpt:

Theorem 1.3 (Hubáček and Yogev [58])**.** End-of-Metered-Line *is many-one reducible to* CLOpt*, and, thus,* End-of-Metered-Line $\in$ CLS.

Our reduction builds on previous embeddings of discrete combinatorial problems, such as End-of-Line, into the continuous domain [16, 17, 52]. The main technical hurdle that we had to overcome was dealing with the additional solutions introduced by the valuation. Specifically, our embedding simultaneously handles the discrete graph and the valuation of vertices, which did not follow easily using the known embeddings that were concerned only with the discrete graph.

RELATED WORK: Our definition of End-of-Metered-Line motivated further refinements of the combinatorial structure within CLS, leading to the formulation of the End-of-Potential-Line problem

and the corresponding class EOPL by Fearnley et al. [33], which captures problems in PLS ∩ PPAD via potential-monotone directed paths with efficient access to the predecessor and successor of each vertex. Crucially, in End-of-Potential-Line the potential is only required to be monotone along the path and does not need to serve as an exact odometer as in our definition of End-of-Metered-Line. Nevertheless, Fearnley et al. [33] established a polynomial-time equivalence between End-of-Metered-Line and End-of-Potential-Line, confirming the robustness of our odometer-based definition. Additionally, they defined a natural refinement of EOPL, denoted UEOPL, that captures problems in which the "natural" potential line is guaranteed to be unique for well-formed instances, e.g., finding the fixed point of a contraction. The search problem Sink-of-Potential-Line defined by Göös et al. [47] is another variant of End-of-Potential-Line, analogous to the PPADS-complete problem Sink-of-Line obtained from the PPAD-complete End-of-Line by focusing only on sinks as solutions.

This line of work culminated in the collapse of the intersection class PLS ∩ PPAD to EOPL by Fearnley et al. [32], confirming that CLS coincides with the whole of PLS ∩ PPAD. On the other hand, Göös et al. [46] gave an oracle separation of EOPL and UEOPL, suggesting that the uniqueness of the solution for well-formed instances may actually make UEOPL strictly weaker. Finally, Göös et al. [45] completed the picture of inclusions and oracle separations among these new line-based subclasses and the classical subclasses of TFNP. To the best of our knowledge, all of the above results rely on our reduction from End-of-Metered-Line to CLOpt in the chain of reductions establishing the characterization CLS = PLS ∩ PPAD.

Additionally, in Section 1.3.1, we discuss implications of our reduction for lower bounds on CLS-complete problems.

1.2.2.2 Arrival

Dohrau et al. [27] presented the following simple-to-state combinatorial decision problem Arrival, which seems to captivate many researchers with a taste for combinatorial optimization:

"Suppose that a train is running along a railway network, starting from a designated origin, with the goal of reaching a designated destination. The network, however, is of a special nature: every time the train traverses a switch, the switch will change its position immediately afterwards. Hence, the next time the train traverses the same switch, the other direction will be taken, so that directions alternate with each traversal of the switch.
Given a network with origin and destination, what is the complexity of deciding whether the train, starting at the origin, will eventually reach the destination?" [27]

Correspondingly, an instance of Arrival is given by an *explicit* directed graph such that each vertex has exactly two outgoing edges (for the alternating exits) and designated origin and destination vertices. The *decision* variant of Arrival asks whether it is possible to reach the destination vertex. The *search* variant, which we consider throughout the rest of this section, asks us to find a "transcript" of the train's path

from the origin to the destination. As stated above, it is not obvious whether ARRIVAL is a *total* search problem, but it is easy to modify any instance so that, in the negative case, one can also efficiently witness the fact that the train cannot reach its destination. Interestingly, it is one of the few TFNP problems with an explicit representation that does not easily fall into FP.

In the context of TFNP, Karthik C. S. [71] showed that the search variant of ARRIVAL lies in PLS. In Gärtner et al. [40], we gave a reduction from ARRIVAL to END-OF-METERED-LINE, showing that it lies among other computational problems in CLS with unresolved computational complexity, i.e., at the lowest non-trivial end of TFNP containing plausibly the easiest such search problems. As an additional refinement, we established that ARRIVAL is contained in TFUP; in other words, that there is a unique solution for any instance of ARRIVAL. Finally, due to the simplicity of our reduction to CLS, we were able to leverage a generic randomized algorithm for solving local search instances by Aldous [2] and give an (exponential-time) algorithm for ARRIVAL, improving over the naive strategy that simulates the train's run.

Theorem 1.4 (Gärtner et al. [40])**.** ARRIVAL *is contained in* CLS ∩ TFUP. *Moreover, it can be solved in expected time* $\mathcal{O}(2^{n/2}\,\mathrm{poly}(n))$ *on n-vertex road networks.*

RELATED WORK: Due to its explicit representation, we believe proving completeness of ARRIVAL for some subclass of TFNP, such as UEOPL, would likely be challenging. On the other hand, the algorithmic progress on ARRIVAL is, arguably, slow. For some restricted classes of graphs, Auger, Coucheney, and Duhaze [4] presented a polynomial-time algorithm. However, the state-of-the-art algorithm for solving ARRIVAL on general graphs by Gärtner, Haslebacher, and Hoang [41] runs in subexponential time. A recent result of Haslebacher [51] established upper bounds on the complexity of ARRIVAL quite similar to those for the classical problem of solving simple stochastic games [20, 96] (see Section 1.1 in Haslebacher [51] for further discussion). ARRIVAL was positioned within a broader framework of Reachability Switching Games of Fearnley et al. [31], whose zero/one/two-player variants are NL-hard, NP-complete, and PSPACE-hard respectively. The best known lower bound for ARRIVAL is currently its NL-hardness.

A variant of a spanning tree partition problem motivated by the György-Lovász theorem [49, 76] is known to be contained in PLS as shown by Chandran, Cheung, and Issac [14]. Similarly to ARRIVAL, it is an explicit combinatorial problem, and it is conjectured to lie in CLS [80] due to the homotopy-based proof of the György-Lovász theorem [76].

1.2.2.3 CLS *as a Singular Intersection Subclass*

As we discussed, one of the motivations of Daskalakis and Papadimitriou [25] for defining CLS was to find a better upper bound for TFNP problems known to be contained in the intersection of PPAD and PLS. Their hypothesis about CLS being a proper subclass of PLS ∩ PPAD

was, however, disproved by Fearnley et al. [32]. The characterization of CLS as an intersection class suggests the following question:

Are any other classical TFNP *subclasses intersection classes?*

Surely, one can define contrived or uninteresting subclasses of TFNP that would be intersection subclasses. We are, however, interested in intersections of the well established subclasses of TFNP and, in particular, those that have a complete problem not defined in terms of Boolean circuits.[4] As a prime candidate, one would consider PPAD, which is known to be contained both in the intersection class PPAP and PPADS [53]. In Hubáček, Khaniki, and Thapen [59], we addressed this question via techniques from propositional proof complexity.

Our starting point is the well-established correspondence between the standard subclasses of TFNP and suitable propositional proof systems, due to Göös et al. [46] and earlier work. We identify a generic mechanism that forces collapses of TFNP intersections in this setting, based on the *feasible disjunction* property of a proof system [87]: whenever a disjoint CNF conjunction $F \wedge G$ has a short refutation, then either F or G has a short refutation as well (for an appropriate complexity measure). Building on known semantic characterizations of low-degree Nullstellensatz and Sherali–Adams refutations and extending them where necessary, we show that Nullstellensatz enjoys feasible disjunction for degree and Sherali–Adams enjoys feasible disjunction for a combined degree-and-magnitude measure. Combining these properties with existing lower bounds for canonical CNF families corresponding to problems complete for PPA_q, PPADS, and PLS, we obtain a robust "no-win" argument: any hypothetical equality between a nontrivial intersection $A \cap B$ and a standard class C would translate into small refutations of suitable CNF conjunctions, which is impossible by feasible disjunction and the lower bounds. In this way, we showed the following.

Theorem 1.5 (Hubáček, Khaniki, and Thapen [59])**.** *Apart from the known collapse* $\mathsf{CLS} = \mathsf{PLS} \cap \mathsf{PPAD}$*, no other nontrivial intersection of the classical subclasses* PLS, PPA, PPP, PPADS, PPAD, CLS *coincides with a standard class or with another nontrivial intersection in the black-box model.*

RELATED WORK: Rezende, Potechin, and Risse [90] recently leveraged a similar notion of Sherali–Adams proof system with bounded degree. Motivated by our separation of intersection subclasses from the standard TFNP subclasses, we introduced a simple non-trivial complete problem for PPA ∩ PPADS; similarly, Ishizuka [62] defined a non-trivial complete problem for PPA ∩ PPP.[5]

4 Some works refer to this property of a TFNP problem as *natural*. However, we avoid this terminology as it seems neither descriptive nor comprehensive enough – consider, e.g., the very "natural" problem of finding a collision in a shrinking function represented by a Boolean circuit.

5 Note that any intersection subclass $\mathcal{C} \cap \mathcal{D}$ admits a "trivial" complete problem defined by a pair of instances, one of $\mathcal{C}$ and the other of $\mathcal{D}$, asking for a solution of either one of the instances.

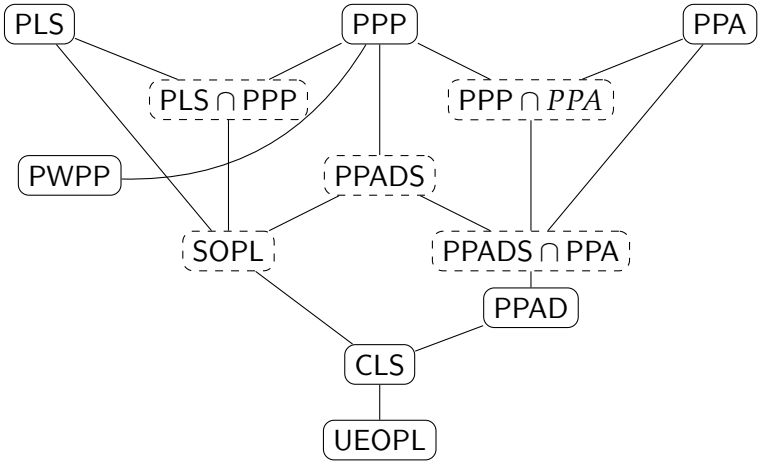

Figure 1.2: The refined partially ordered set defined by the known inclusions among subclasses of TFNP.

The independent concurrent work of Li, Pires, and Robere [75] is the closest in spirit to ours. Compared to our largely "semantic" approach, Li, Pires, and Robere [75] work "syntactically" with fixed canonical complete problems and associated Resolution-style proof systems, and they obtain a complete classification (in the black-box setting) of all non-trivial intersections among PLS, PPA, PPAD, and PPADS, together with natural proof systems whose bounded-width refutations characterize each intersection class. Their results thus complement ours: they provide explicit complete problems and proof systems for intersection classes such as PLS ∩ PPADS and PLS ∩ PPA, while our work treats a larger family of intersections (including those involving PPP and CLS) and explains, via feasible disjunction, why many potential collapses between intersection classes and the original TFNP subclasses cannot occur.

1.2.3 *The Updated Landscape*

In Figure 1.2, we present an updated picture of some of the TFNP subclasses based on the results and related work discussed in this section. The subclasses highlighted by dashing, i.e., the intersection classes and PPADS, are characterized by simple combinatorial search problems [45, 59, 62, 81]. However, to the best of our knowledge, they are not characterized by other complete problems. The remaining depicted subclasses are known to contain diverse complete problems.

1.3 CRYPTOGRAPHIC LOWER BOUNDS

The methodology and results presented in the previous section concern the relative complexity of TFNP problems. As we discussed, there was tremendous progress in characterizing subclasses of TFNP via complete problems from broad corners of computational mathematics. Additionally, oracle separations of the decision-tree analogues of the known subclasses of TFNP suggest that many of the known inclusions among various subclasse are indeed proper, or that some subclasses are incomparable (e.g., PPAD and PPADS, or PPA and PPADS). Such results, however, do not answer the fundamental question of whether TFNP contains any computationally hard problems in the original "white-box" setting. In this section, we present our result on this line of work. In Section 1.3.1, we discuss lower bounds for CLS under strong but plausible cryptographic assumptions. In Section 1.3.2, we discuss the minimal cryptographic assumptions that suffice for constructing hard-on-average total search problems.

CRYPTOGRAPHIC HARDNESS AND TFNP: We already mentioned that any family of collision-resistant hash functions gives rise to a hard-on-average TFNP problem. In fact, it is equivalent to assuming average-case hardness of the subclass PWPP. Papadimitriou [81] presented a simple argument showing PPP-hard problems cannot be solved efficiently if one-way permutations exist. Therefore, under either of the two mild cryptographic assumptions (i.e., the existence of one-way permutations or collision-resistant hashing), there are distributions of TFNP problems that cannot be solved efficiently. Importantly, the two conditional lower bounds are restricted to PPP and PWPP, and leave open the possibility of an efficient algorithm for, e.g., PPAD-complete problems. Additionally, recall that PPP and PWPP are not known to have many diverse complete problems, unlike PPAD or the other subclasses of TFNP. Therefore, the above straightforward lower bounds have, for now, only limited implications.

Buresh-Oppenheim [12] and Jeřábek [66] gave the most compelling conditional lower bound for a subclass of TFNP to day. By leveraging the relationship between integer factoring and computing modular square roots, Buresh-Oppenheim [12] showed a randomized reduction from the problem of factoring Blum integers (i.e., products of two strong primes of similar bit length) to the PPA-complete problem LONELY.[6] Jeřábek [66] lifted the restriction on the structure of the integers and showed that the general integer factoring problem is randomly reducible to LONELY.[7]

Abbott, Kane, and Valiant [1] outlined a different approach for cryptographic lower bounds for TFNP problems. On a high level, their approach can be seen as a cryptographic emulation of an oracle that

6 The result of Buresh-Oppenheim [12] applies to a slightly broader class of integers. See his note for the details.

7 The probabilistic aspect of both reductions is limited to the problem of finding a quadratic residue, which, under the generalized Riemann hypothesis, can be solved deterministically by exhaustive search in polynomial time.

establishes a query complexity lower bound for the TFNP problem. Recall that, in query complexity, we are interested only in the number of queries that an algorithm does to the instance and we do not restrict its running time. For example, we could consider the function inversion problem, where, given oracle access to a function f and an element y in its image, one must find a pre-image of y under f. Since an algorithm solving the function inversion problem in the query complexity model must succeed for an arbitrary function, we can leverage a truly random function to prove an exponential lower bound on the number of necessary queries. Similarly, strong query complexity lower bounds were established for various TFNP problems (see, e.g., Babichenko [5] and Savani and Stengel [92]). The query complexity model restricts the power of the algorithm for the given problem, and, thus, one can prove very strong lower bounds via rather direct arguments. However, for TFNP problems, one is given a succinct representation of the instance that, in particular, cannot be used to encode a large completely random instance commonly used to establish query complexity lower bounds. Abbott, Kane, and Valiant [1] suggested using cryptography together with ideal code obfuscation to give a succinct description of plausibly hard instances of the PPAD-complete problem End-of-Line. Bitansky, Paneth, and Rosen [10] were the first to instantiate the above template using sub-exponentially hard indistinguishability obfuscation and injective one-way functions, establishing the first conditional lower bound for PPAD under a plausible cryptographic hardness assumption.

1.3.1 *Cryptographic Lower Bounds for CLS*

Bitansky, Paneth, and Rosen [10] established their lower bound for PPAD via two independent steps:

1. a conditional lower bound for the Sink-of-Verifiable-Line problem,[8] and
2. a reduction from Sink-of-Verifiable-Line to the PPAD-complete problem End-of-Line.

The subsequent work extending their result focused predominantly on improving the cryptographic assumptions that allow proving a lower bound for Sink-of-Verifiable-Line.

In Hubáček and Yogev [58], we reduced Sink-of-Verifiable-Line to End-of-Metered-Line, and, thus, showed that the average-case hardness of Sink-of-Verifiable-Line implies conditional lower bounds for a significantly wider collection of TFNP problems.

Theorem 1.6 (Hubáček and Yogev [58])**.** End-of-Metered-Line *is at least as hard as* Sink-of-Verifiable-Line*.*

8 Sink-of-Verifiable-Line is a *promise* problem introduced by Bitansky, Paneth, and Rosen [10] to make their lower bound more modular. We do not present it in detail since it is not necessary for our high-level discussion.

As a corollary of Theorem 1.6 and the conditional lower bound for SINK-OF-VERIFIABLE-LINE from [10], we established that sub-exponentially hard indistinguishability obfuscation and injective one-way functions imply average-case hardness for any class that contains CLS. Similar corollaries follow by combining Theorem 1.6 with improved lower bounds for SINK-OF-VERIFIABLE-LINE from Garg, Pandey, and Srinivasan [39] and Komargodski and Segev [73]. Specifically, the last two works showed that a variant of indistinguishability obfuscation implied by functional encryption is sufficient to prove a lower bound for SINK-OF-VERIFIABLE-LINE. In fact, standard polynomial hardness of the considered primitives suffices (opposed to the subexponential hardness assumed by Bitansky, Paneth, and Rosen [10]).

CONDITIONAL LOWER BOUNDS WITHOUT OBFUSCATION: A natural direction for improving the conditional lower bound for CLS under indistinguishability obfuscation (or conceptually similar notions such as functional encryption) is to leverage weaker cryptographic hardness assumptions. In Choudhuri et al. [19], we were the first to show that neither indistinguishability obfuscation nor functional encryption is necessary for cryptographic lower bounds for CLS.

On a very high level, the obfuscation-based approach is based on the insight of Abbott, Kane, and Valiant [1] that a hard PPAD instance can be defined by obfuscating the successor function that on input $(i, F_k(i))$ outputs $(i+1, F_k(i+1))$, where F_k is a pseudorandom function. In this way, we can construct a line of exponential length starting at $(0, F_k(0))$ and terminating at $(2^n, F_k(2^n))$. Assuming that the obfuscated version of the above successor does not leak the key k of the pseudorandom function, the best strategy is likely to perform the 2^n steps using the successor function to reach the sink at $(2^n, F_k(2^n))$. Indeed, Bitansky, Paneth, and Rosen [10] and subsequent work showed that this intuition can be turned into a formal proof under a suitable notion of indistinguishability obfuscation (respectively functional encryption).

In Choudhuri et al. [19], we proposed a different construction of an exponential-length line. For a 3SAT formula φ on n variables, we define a successor circuit that takes as input a tuple (i, y_i, π_i), where π_i is a proof for the statement "there are exactly y_i satisfying assignments for φ among the first i assignments," and outputs the corresponding subsequent tuple $(i+1, y_{i+1}, \pi_{i+1})$. Here, by proof we mean a non-interactive cryptographic proof for the above statement with soundness against efficient cheating provers. The natural approach is to leverage the famous sum-check protocol for #SAT [95], which can be made non-interactive using the Fiat–Shamir heuristic [34]. The nontrivial part of our argument was showing that the intermediate proofs π_i can be efficiently updated. We also had to refine the search problem SINK-OF-VERIFIABLE-LINE to handle the unavoidable existence of accepting "false" proofs for incorrect values y_i', which are, however, computationally hard to construct. To this end, we introduced the RELAXED-SINK-OF-VERIFIABLE-LINE problem, which, as we showed, is still reducible to END-OF-METERED-LINE. Together, these relaxations

allowed us to rely on weaker cryptographic hardness assumptions and prove the following.

Theorem 1.7 (Choudhuri et al. [19])**.** *Assuming* #SAT $\notin$ P *and that the Fiat-Shamir heuristic is sound for Shamir's sum-check protocol, there is no polynomial-time algorithm for* End-of-Metered-Line*.*

RELATED WORK: Currently, the above two approaches are the main conceptual methods for proving lower bounds for problems in CLS under cryptographic hardness assumptions. The first follows the "obfuscation" paradigm, initiated by Abbott, Kane, and Valiant [1] and Bitansky, Paneth, and Rosen [10]. Any improvement in the assumptions required to construct the corresponding notion of indistinguishability obfuscation or functional encryption yields an improved lower bound in terms of cryptographic assumptions (see, e.g., Jain, Lin, and Sahai [64] for a recent survey on indistinguishability obfuscation).

The second, complementary line of work, initiated in Choudhuri et al. [19], proceeds via the "Fiat-Shamir" route. Subsequent works pushed the cryptographic assumptions further toward standard ones and strengthened the hardness guarantees. Jawale et al. [65] constructed SNARGs for bounded-depth computations from subexponential LWE and, by instantiating the Fiat–Shamir transform for the sum-check and GKR protocols, recovered and strengthened average-case hardness for relaxed-Sink-of-Verifiable-Line under subexponential LWE. In a similar direction,Bitansky et al. [8] showed that polynomial-time hardness of iterated squaring in RSA groups together with the Learning with Errors (LWE) assumption already suffices to give average-case hardness of relaxed-Sink-of-Verifiable-Line. They also gave a reduction from relaxed-Sink-of-Verifiable-Line to Unique-End-of-Potential-Line, establishing a conditional lower bound for UEOPL. More recently, Kalai, Lombardi, and Vaikuntanathan [69] developed delegation schemes and SNARG constructions based on decisional Diffie-Hellman-type assumptions and used them to derive average-case hardness in CLS under these classical number-theoretic assumptions.

Bitansky and Gerichter [9] showed worst-case and average-case hardness for PLS via the notion of *incrementally complete* incrementally verifiable computation, under a number-theoretic assumption introduced by Kalai, Paneth, and Yang [70] and (randomized) ETH. They even obtained unconditional average-case hardness for Local-Search relative to a random oracle.

1.3.2 *Average-Case Hardness in TFNP and One-Way Functions*

When presented with the push for establishing cryptographic lower bounds for problems in TFNP, one might ask about the limits of this line of work or, equivalently,:

What is the minimal assumption implying average-case hardness in TFNP*?*

Already Megiddo and Papadimitriou [78] noticed that the worst-case hardness of TFNP problems cannot be based on the worst-case hardness of NP, unless NP = coNP.

1.3.2.1 *From Average-Case Hardness in* NP *to* TFNP

In Hubáček, Naor, and Yogev [56], we gave an almost tight answer to the above question by showing that hard-on-average TFNP problems already follow from the very weak assumption that there exists a hard-on-average language in NP and "derandomization-style" complexity assumptions.

Formally, we work in the standard framework of distributional problems (L, D), where $L \in \mathsf{NP}$ and D is an efficiently sampleable input distribution, for which we ask to decide the membership in L. Our first main result shows that any such hard-on-average decision problem gives rise to a non-uniform TFNP problem that is hard-on-average.

Theorem 1.8 (Hubáček, Naor, and Yogev [56])**.** *If there exists a hard-on-average language in* NP*, then there exists a hard-on-average problem in* TFNP/ poly*.*

The key technical challenge is to ensure totality while preserving average-case hardness. Starting from a hard-on-average pair (L, D), it is straightforward to obtain a hard-on-average *search* problem by asking for an NP witness for $L(x)$ on input $x \leftarrow D$. However, this search problem is not total: some inputs simply have no witness, and even recognizing satisfiable instances is NP-hard. Thus, we must transform (L, D) into a new pair (L', D') such that

1. every instance in the support of D' is in L' (totality), and
2. (L', D') remains hard-on-average for search.

To achieve this, we used a *reverse randomization* or "shifting" trick. Very roughly, we fix a polynomial number k of bitstrings $s_1, \ldots, s_k$ and consider the language

$$L_{s_1,\ldots,s_k}(r) = 1 \quad \iff \quad \exists i \in [k] \text{ s.t. } D(1^n; r \oplus s_i) \in L,$$

and the uniform distribution on $r \in \{0,1\}^n$. For an appropriate choice of k, a random choice of the shifts $s_1, \ldots, s_k$ makes $L_{s_1,\ldots,s_k}$ total with overwhelming probability, while a careful averaging argument shows that the associated search problem is still hard-on-average. This yields a non-uniform hard-on-average problem in TFNP/ poly by hardwiring a good choice of the shifts at each input length.

To obtain a *uniform* hard-on-average problem in TFNP, we must derandomize the choice of good shifts. The property "$s_1, \ldots, s_k$ make $L_{s_1,\ldots,s_k}$ total" can be checked by a polynomial-size Π_2-circuit, so we can use a Nisan–Wigderson style pseudorandom generator that fools Π_2-circuits to pick a small family of candidate shift-tuples, one of which is guaranteed to be good. Assuming the variant of a standard "fooling assumption" that there exists a function in E of Π_2-circuit

complexity $2^{\Omega(n)}$, such generators exist, and we can deterministically search over their images. This yields the following uniform version.

Corollary 1.9 (Hubáček, Naor, and Yogev [56])**.** *Assume there exists a function in* E *of* Π_2*-circuit complexity* $2^{\Omega(n)}$*. If there exists a hard-on-average language in* NP*, then there exists a hard-on-average problem in* TFNP*.*

A further subtlety is that the above argument initially requires the distribution D to remain hard even when given its sampling randomness (what we call "public-coin" distributions), but we show that this assumption is without loss of generality. Altogether, we obtain hard-on-average TFNP problems in Impagliazzo's world of *Pessiland* [61], i.e., under the mere existence of hard-on-average NP languages, a setting where none of the standard cryptographic primitives (not even one-way functions) are guaranteed to exist.

HARD TOTAL SEARCH PROBLEMS WITH FEW SOLUTIONS. Our generic transformation from average-case NP hardness to TFNP produces instances that may have many solutions. In the second part of Hubáček, Naor, and Yogev [56], we asked whether one can obtain structured hard-on-average TFNP problems with *few* solutions per instance, under stronger but still standard cryptographic assumptions. In particular, we considered injective one-way functions and *non-interactive witness-indistinguishable proof systems for* NP.

Our construction defines a total search problem Invert-Either whose instances consist of two images y, y' under an injective one-way function f together with a witness-indistinguishable proof π that at least one of them is in the range of f. A solution is either

- a special default string, in case π is invalid, or
- a preimage of one of y, y' under f, in case π is valid.

Perfect soundness of the proof system guarantees totality: either the proof is invalid (and the default solution is accepted), or it certifies that a valid preimage exists. Witness indistinguishability ensures that π does not leak which of the two images we will later embed as a challenge for inverting f, so any solver for Invert-Either can be turned into an inverter for f. Formally, we obtain:

Theorem 1.10 (Hubáček, Naor, and Yogev [56])**.** *If injective one-way functions and non-interactive witness-indistinguishable proof systems for* NP *exist, then there exists a hard-on-average problem in* TFNP *such that every instance has at most two solutions.*

This result shows that *Minicrypt*-style [61] assumptions (one-way functions with the currently separate assumptions for non-interactive witness-indistinguishable proofs for NP) already suffice to obtain very structured hard-on-average problems in TFNP, in sharp contrast with the earlier lower bounds for PPP or CLS that relied on stronger primitives such as collision-resistant hash functions, one-way permutations, or even indistinguishability obfuscation.

RELATED WORK: Pass and Venkitasubramaniam [83, 84] gave a related result showing that the existence of an almost-everywhere hard-on-average language in NP implies either the existence of one-way functions, or a hard-on-average TFNP problem. So, the existence of average-case hardness in TFNP is unconditional in *Pessiland*.

1.3.2.2 *Towards a Black-Box Separation of* TFNP *and One-Way Functions*

A major open question is whether one can base average-case hardness of TFNP problems on the unstructured assumption of the existence of one-way functions *without additional assumptions*. The last two works included in this thesis provide the first steps in this direction.

In Hubáček et al. [55], we initiated a systematic study of black-box barriers to basing TFNP-hardness on one-way functions. Motivated by Papadimitriou's construction of hard instances of PIGEON from one-way permutations and its extensions using collision-resistant hash functions [81], we asked whether similarly "simple" black-box constructions could exist under the weaker assumption of the existence of injective one-way functions. Formally, we considered fully black-box constructions of (even worst-case) hard TFNP problems from injective one-way functions with randomized, *f-oblivious*, and *non-adaptive* security reductions, a class that already subsumes the classical many-one reductions underlying these known constructions. Our main result shows that no such construction exists: relative to a suitable two-oracle world (f, Solve), where f is a random injective one-way function and Solve solves every TFNP instance (making TFNP easy), every reduction of this restricted type still fails to invert f with non-negligible probability.

The proof combines the Impagliazzo-Rudich framework of black-box separations with a reconstruction-style information-theoretic argument. Ourt key structural insight is the ubiquitous presence of "useless" solutions in TFNP instances constructed from (unstructured) one-way functions: for a random injective oracle f and a random challenge y, with overwhelming probability there exists a solution to any TFNP instance that never queries a preimage of y, and, hence, carries no information about the challenge from the reduction's point of view. We leverage the freedom afforded by fully black-box security to design, *for each given reduction*, a tailored TFNP breaker Solve that answers almost always with such useless solutions. Using an incompressibility argument in the spirit of Gennaro and Trevisan [42], we then showed that any reduction that could still invert f in the presence of this oracle would yield an encoding of a random permutation that is asymptotically shorter than its entropy, a contradiction. In this way, we obtain a first negative result that rules out a large and natural family of black-box constructions of TFNP-hardness from injective one-way functions.

In the subsequent work of Folwarczný et al. [37], we strengthened and simplified the above separation. Again, we constructed a pair of oracles (f, Solve) such that f is an injective one-way function and Solve can solve any TFNP instance, yet f remains one-way against a broad

class of *single-query* algorithms that interact with Solve. In contrast to our fully black-box setting, this negative result applies already to *semi*-black-box reductions, where the reduction is allowed to depend on the behaviour of Solve, and it removes the f-obliviousness and non-adaptivity restrictions, at the cost of only ruling out reductions that make a single query to the TFNP oracle and do not query f before that query. Technically, we replaced the information-theoretic reconstruction machinery of Hubáček et al. [55] by a more direct combinatorial analysis of the interaction between the reduction and the TFNP solver, yielding a cleaner proof that nonetheless captures a strictly larger class of reductions.

1.4 CONCLUSION

In this habilitation thesis, we have presented our main results on the computational complexity of total NP search problems and discussed closely related and subsequent work. We conclude with some timely open problems related to this line of research.

NEW STRUCTURAL LENS: As we have seen throughout this habilitation thesis, recent work has exemplified the intuitively strong connection between subclasses of TFNP and propositional proof systems (see, e.g., [46]). Thapen [99] recently proposed an alternative view of TFNP through "higher" complexity classes such as PSPACE or parity P. Fleshing out his approach so that it can address most of the current landscape of TFNP and understanding its implications seems to be an exciting research direction.

SIMPLE HARD DISTRIBUTIONS: The cryptographic lower bounds for TFNP subclasses such as PPAD that we discussed in Section 1.3.1 are conceptually very interesting, but they are of limited use if one wishes to construct a hard instance of "practical" size, e.g., to obtain a challenging game for a state-of-the-art equilibrium solver. In theory, one could start from an instance of some cryptographic hardness assumption and follow the chain of polynomial-time reductions to construct, for example, an instance of the PPAD-complete problem NASH. However, the accumulated overhead of these reductions makes the final instance prohibitively large. Thus, a well-understood central open problem in the area—namely, to give a direct construction of a finite multiplayer game in which computing a Nash equilibrium would immediately amount to breaking a standard cryptographic assumption—is still unresolved. To the best of our knowledge, analogous issues arise for every subclass of TFNP other than PPP and PWPP, where the connection to basic cryptographic primitives is more straightforward and the blow-up in known reductions is less of an obstacle. On the other hand, neither PPP nor PWPP are known to have a complete problem represented by an explicit game or graph.

HARDNESS RELATIVE TO A RANDOM ORACLE: There is an intermediate type of lower-bound argument for TFNP problems that has not been significantly explored in the literature so far. Specifically, constructing hard instances of total search problems relative to a random oracle. On the one hand, a random oracle is strongly collision resistant, and thus it immediately gives a lower bound for PWPP and any class containing PWPP. On the other hand, Bitansky and Gerichter [9] gave a complex construction of a hard PLS instance relative to a random oracle. The intermediate region, e.g., lower bounds for PPA or *simple* lower bounds for PLS relative to a random oracle, has not been explored. Note that such arguments are conceptually stronger than query-complexity lower bounds, which rely on *structured* oracles implementing hard instances that enable direct adversarial arguments. However, they should be technically

easier to establish than cryptographic lower bounds, which need to hold in the challenging white-box setting.

TOTAL SEARCH PROBLEMS WITH UNIQUE SOLUTIONS: The complexity class TFUP of TFNP problems with *syntactically* guaranteed uniqueness of solution for every instance, i.e., the search analogue of the class UP ∩ coUP, is yet to be properly studied. Currently, we know only a few TFUP problems that are not polynomial-time solvable, such as ARRIVAL, the discrete logarithm problem in specific groups, integer factoring, and some games on graphs such as parity games [68], (discounted) mean-payoff games [68, 102], simple stochastic games [15, 21], or quantitative games with interval/exact-value objectives [60]. Identifying additional natural TFUP problems with unresolved complexity would clearly be interesting.

On the structural side, Harsha, Mitropolsky, and Rosen [50] studied downward self-reducibility in the context of TFNP and (among other results) showed that any downward self-reducible TFUP problem lies in UEOPL. Gajulapalli et al. [38] recently refined the downward self-reducibility in TFNP framework, and, additionally, showed that ARRIVAL and some of the other TFUP problems known to be contained in UEOPL are downward self-reducible. Clarifying the precise significance of downward self-reducibility for TFUP remains open.

Finally, Homan and Thakur [54] showed that *worst-case* one-way permutations are equivalent to *worst-case* hardness of TFUP. The containment of the discrete logarithm problem in TFUP clearly suggests exploring the connection between *average-case* hardness in TFUP and *cryptographic* one-way permutations.

BIBLIOGRAPHY

[1] Tim Abbott, Daniel Kane, and Paul Valiant. *On Algorithms for Nash Equilibria*. Technical Report. Massachusetts Institute of Technology (MIT), 2004. URL: http://web.mit.edu/tabbott/Public/final.pdf.

[2] David Aldous. "Minimization Algorithms and Random Walk on the d-Cube." In: *The Annals of Probability* 11.2 (1983), pp. 403–413. DOI: 10.1214/aop/1176993605.

[3] Noga Alon. "Splitting necklaces." In: *Advances in Mathematics* 63.3 (1987), pp. 247–253.

[4] David Auger, Pierre Coucheney, and Loric Duhaze. "Polynomial Time Algorithm for ARRIVAL on Tree-Like Multigraphs." In: *47th International Symposium on Mathematical Foundations of Computer Science, MFCS 2022, August 22-26, 2022, Vienna, Austria*. Ed. by Stefan Szeider, Robert Ganian, and Alexandra Silva. Vol. 241. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 12:1–12:14. DOI: 10.4230/LIPICS.MFCS.2022.12. URL: https://doi.org/10.4230/LIPIcs.MFCS.2022.12.

[5] Yakov Babichenko. "Query Complexity of Approximate Nash Equilibria." In: *J. ACM* 63.4 (2016), 36:1–36:24. DOI: 10.1145/2908734. URL: https://doi.org/10.1145/2908734.

[6] Frank Ban, Kamal Jain, Christos H. Papadimitriou, Christos-Alexandros Psomas, and Aviad Rubinstein. "Reductions in PPP." In: *Inf. Process. Lett.* 145 (2019), pp. 48–52. DOI: 10.1016/J.IPL.2018.12.009. URL: https://doi.org/10.1016/j.ipl.2018.12.009.

[7] Paul Beame, Stephen A. Cook, Jeff Edmonds, Russell Impagliazzo, and Toniann Pitassi. "The Relative Complexity of NP Search Problems." In: *J. Comput. Syst. Sci.* 57.1 (1998), pp. 3–19. DOI: 10.1006/JCSS.1998.1575. URL: https://doi.org/10.1006/jcss.1998.1575.

[8] Nir Bitansky, Arka Rai Choudhuri, Justin Holmgren, Chethan Kamath, Alex Lombardi, Omer Paneth, and Ron D. Rothblum. "PPAD is as Hard as LWE and Iterated Squaring." In: *Theory of Cryptography - 20th International Conference, TCC 2022, Chicago, IL, USA, November 7-10, 2022, Proceedings, Part II*. Ed. by Eike Kiltz and Vinod Vaikuntanathan. Vol. 13748. Lecture Notes in Computer Science. Springer, 2022, pp. 593–622. DOI: 10.1007/978-3-031-22365-5_21. URL: https://doi.org/10.1007/978-3-031-22365-5_21.

[9] Nir Bitansky and Idan Gerichter. "On the Cryptographic Hardness of Local Search." In: *11th Innovations in Theoretical Computer Science Conference, ITCS 2020, January 12-14, 2020, Seattle, Washington, USA*. Ed. by Thomas Vidick. Vol. 151. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, 6:1–6:29. DOI: 10.4230/LIPICS.ITCS.2020.6. URL: https://doi.org/10.4230/LIPIcs.ITCS.2020.6.

[10] Nir Bitansky, Omer Paneth, and Alon Rosen. "On the Cryptographic Hardness of Finding a Nash Equilibrium." In: *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*. Ed. by Venkatesan Guruswami. IEEE Computer Society, 2015, pp. 1480–1498. DOI: 10.1109/FOCS.2015.94. URL: https://doi.org/10.1109/FOCS.2015.94.

[11] Romain Bourneuf, Lukáš Folwarczný, Pavel Hubáček, Alon Rosen, and Nikolaj Ignatieff Schwartzbach. "PPP-Completeness and Extremal Combinatorics." In: *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*. Ed. by Yael Tauman Kalai. Vol. 251. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 22:1–22:20. DOI: 10.4230/LIPICS.ITCS.2023.22. URL: https://doi.org/10.4230/LIPIcs.ITCS.2023.22.

[12] Joshua Buresh-Oppenheim. "On the TFNP Complexity of Factoring." Unpublished Note. 2006. URL: http://www.cs.toronto.edu/~bureshop/factor.pdf.

[13] Arthur Cayley. "A theorem on trees." In: *Quarterly Journal of Mathematics* 23 (1889), pp. 376–378.

[14] L. Sunil Chandran, Yun Kuen Cheung, and Davis Issac. "Spanning Tree Congestion and Computation of Generalized Győri-Lovász Partition." In: *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, Prague, Czech Republic, July 9-13, 2018*. Ed. by Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella. Vol. 107. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, 32:1–32:14. DOI: 10.4230/LIPICS.ICALP.2018.32. URL: https://doi.org/10.4230/LIPIcs.ICALP.2018.32.

[15] Krishnendu Chatterjee and Nathanaël Fijalkow. "A Reduction from Parity Games to Simple Stochastic Games." In: *Proceedings of GandALF 2011*. Vol. 54. 2011, pp. 74–86. DOI: 10.4204/EPTCS.54.6.

[16] Xi Chen and Xiaotie Deng. "On algorithms for discrete and approximate Brouwer fixed points." In: *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*. Ed. by Harold N. Gabow and Ronald Fagin. ACM, 2005, pp. 323–330. DOI: 10.1145/1060590.1060638. URL: https://doi.org/10.1145/1060590.1060638.

[17] Xi Chen and Xiaotie Deng. "On the complexity of 2D discrete fixed point problem." In: *Theor. Comput. Sci.* 410.44 (2009), pp. 4448–4456. DOI: 10.1016/J.TCS.2009.07.052. URL: https://doi.org/10.1016/j.tcs.2009.07.052.

[18] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. "Settling the complexity of computing two-player Nash equilibria." In: *J. ACM* 56.3 (2009), 14:1–14:57. DOI: 10.1145/1516512.1516516. URL: https://doi.org/10.1145/1516512.1516516.

[19] Arka Rai Choudhuri, Pavel Hubáček, Chethan Kamath, Krzysztof Pietrzak, Alon Rosen, and Guy N. Rothblum. "Finding a Nash equilibrium is no easier than breaking Fiat-Shamir." In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019, Phoenix, AZ, USA, June 23-26, 2019*. 2019, pp. 1103–1114. DOI: 10.1145/3313276.3316400. URL: https://doi.org/10.1145/3313276.3316400.

[20] Anne Condon. "The Complexity of Stochastic Games." In: *Information and Computation* 96.2 (1992), pp. 203–224. DOI: 10.1016/0890-5401(92)90048-K.

[21] Anne Condon. "The Complexity of Stochastic Games." In: *Information and Computation* 96.2 (1992), pp. 203–224. DOI: 10.1016/0890-5401(92)90048-K.

[22] Thomas M. Cover. "Enumerative source encoding." In: *IEEE Transactions on Information Theory* 19.1 (1973), pp. 73–77. DOI: 10.1109/TIT.1973.1054929.

[23] Ivan Damgård. "Collision Free Hash Functions and Public Key Signature Schemes." In: *Advances in Cryptology - EUROCRYPT '87, Workshop on the Theory and Application of of Cryptographic Techniques, Amsterdam, The Netherlands, April 13-15, 1987, Proceedings*. Ed. by David Chaum and Wyn L. Price. Vol. 304. Lecture Notes in Computer Science. Springer, 1987, pp. 203–216. DOI: 10.1007/3-540-39118-5_19. URL: https://doi.org/10.1007/3-540-39118-5_19.

[24] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. "The Complexity of Computing a Nash Equilibrium." In: *SIAM J. Comput.* 39.1 (2009), pp. 195–259. DOI: 10.1137/070699652. URL: https://doi.org/10.1137/070699652.

[25] Constantinos Daskalakis and Christos H. Papadimitriou. "Continuous Local Search." In: *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*. Ed. by Dana Randall. SIAM, 2011, pp. 790–804. DOI: 10.1137/1.9781611973082.62. URL: https://doi.org/10.1137/1.9781611973082.62.

[26] Xiaotie Deng, Jack R. Edmonds, Zhe Feng, Zhengyang Liu, Qi Qi, and Zeying Xu. "Understanding PPA-completeness." In: *J. Comput. Syst. Sci.* 115 (2021), pp. 146–168. DOI: 10.1016/J.JCSS.

2020.07.006. URL: https://doi.org/10.1016/j.jcss.2020.07.006.

[27] Jérôme Dohrau, Bernd Gärtner, Manuel Kohler, Jiří Matoušek, and Emo Welzl. "ARRIVAL: A zero-player graph game in NP∩coNP." In: *A Journey Through Discrete Mathematics: A Tribute to Jiří Matoušek*. Ed. by Martin Loebl, Jaroslav Nešetřil, and Robin Thomas. Springer International Publishing, 2017, pp. 367–374.

[28] Ömer Egecioglu and Alastair King. "Random walks and Catalan factorization." In: (1999).

[29] Paul Erdős, Chao Ko, and Richard Rado. "Intersection theorems for systems of finite sets." In: *The Quarterly Journal of Mathematics* 12.1 (Jan. 1961), pp. 313–320. ISSN: 0033-5606. DOI: 10.1093/qmath/12.1.313. eprint: https://academic.oup.com/qjmath/article-pdf/12/1/313/7288156/12-1-313.pdf. URL: https://doi.org/10.1093/qmath/12.1.313.

[30] Alex Fabrikant, Christos H. Papadimitriou, and Kunal Talwar. "The complexity of pure Nash equilibria." In: *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*. Ed. by László Babai. ACM, 2004, pp. 604–612. DOI: 10.1145/1007352.1007445. URL: https://doi.org/10.1145/1007352.1007445.

[31] John Fearnley, Martin Gairing, Matthias Mnich, and Rahul Savani. "Reachability Switching Games." In: *Log. Methods Comput. Sci.* 17.2 (2021). URL: https://lmcs.episciences.org/7392.

[32] John Fearnley, Paul Goldberg, Alexandros Hollender, and Rahul Savani. "The Complexity of Gradient Descent: CLS = PPAD ∩ PLS." In: *J. ACM* 70.1 (2023), 7:1–7:74. DOI: 10.1145/3568163. URL: https://doi.org/10.1145/3568163.

[33] John Fearnley, Spencer Gordon, Ruta Mehta, and Rahul Savani. "Unique end of potential line." In: *J. Comput. Syst. Sci.* 114 (2020), pp. 1–35. DOI: 10.1016/J.JCSS.2020.05.007. URL: https://doi.org/10.1016/j.jcss.2020.05.007.

[34] Amos Fiat and Adi Shamir. "How to Prove Yourself: Practical Solutions to Identification and Signature Problems." In: *Advances in Cryptology - CRYPTO '86, Santa Barbara, California, USA, 1986, Proceedings*. Ed. by Andrew M. Odlyzko. Vol. 263. Lecture Notes in Computer Science. Springer, 1986, pp. 186–194. DOI: 10.1007/3-540-47721-7_12. URL: https://doi.org/10.1007/3-540-47721-7_12.

[35] Aris Filos-Ratsikas and Paul W. Goldberg. "The Complexity of Necklace Splitting, Consensus-Halving, and Discrete Ham Sandwich." In: *SIAM J. Comput.* 52.on (2019), STOC19–200–STOC19–268. DOI: 10.1137/20M1312678. URL: https://doi.org/10.1137/20m1312678.

[36] Noah Fleming, Stefan Grosser, Toniann Pitassi, and Robert Robere. "Black-Box PPP Is Not Turing-Closed." In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*. Ed. by Bojan Mohar, Igor Shinkar, and Ryan O'Donnell. ACM, 2024, pp. 1405–1414. DOI: 10.1145/3618260.3649769. URL: https://doi.org/10.1145/3618260.3649769.

[37] Lukáš Folwarczný, Mika Göös, Pavel Hubáček, Gilbert Maystre, and Weiqiang Yuan. "One-Way Functions vs. TFNP: Simpler and Improved." In: *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*. Ed. by Venkatesan Guruswami. Vol. 287. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, 50:1–50:14. DOI: 10.4230/LIPICS.ITCS.2024.50. URL: https://doi.org/10.4230/LIPIcs.ITCS.2024.50.

[38] Karthik Gajulapalli, Surendra Ghentiyala, Zeyong Li, and Sidhant Saraogi. "Downward self-reducibility in the total function polynomial hierarchy." In: *CoRR* abs/2507.19108 (2025). DOI: 10.48550/ARXIV.2507.19108. arXiv: 2507.19108. URL: https://doi.org/10.48550/arXiv.2507.19108.

[39] Sanjam Garg, Omkant Pandey, and Akshayaram Srinivasan. "Revisiting the Cryptographic Hardness of Finding a Nash Equilibrium." In: *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*. Ed. by Matthew Robshaw and Jonathan Katz. Vol. 9815. Lecture Notes in Computer Science. Springer, 2016, pp. 579–604. DOI: 10.1007/978-3-662-53008-5_20. URL: https://doi.org/10.1007/978-3-662-53008-5_20.

[40] Bernd Gärtner, Thomas Dueholm Hansen, Pavel Hubáček, Karel Král, Hagar Mosaad, and Veronika Slívová. "ARRIVAL: Next Stop in CLS." In: *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*. 2018, 60:1–60:13. DOI: 10.4230/LIPICS.ICALP.2018.60. URL: https://doi.org/10.4230/LIPIcs.ICALP.2018.60.

[41] Bernd Gärtner, Sebastian Haslebacher, and Hung P. Hoang. "A Subexponential Algorithm for ARRIVAL." In: *48th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference)*. Ed. by Nikhil Bansal, Emanuela Merelli, and James Worrell. Vol. 198. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 69:1–69:14. DOI: 10.4230/LIPICS.ICALP.2021.69. URL: https://doi.org/10.4230/LIPIcs.ICALP.2021.69.

[42] Rosario Gennaro and Luca Trevisan. "Lower Bounds on the Efficiency of Generic Cryptographic Constructions." In: *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, Redondo Beach, California, USA, November 12-14, 2000*. IEEE

Computer Society, 2000, pp. 305–313. DOI: 10.1109/SFCS.2000.892119. URL: https://doi.org/10.1109/SFCS.2000.892119.

[43] Paul W. Goldberg. *Introduction to Total Search Problems*. Invited talk at the STOC 2025 workshop "Total Search Problems in TCS — Complexity, Cryptography, Combinatorics, and more". Slides. Prague, Czech Republic, June 2025. URL: https://www.cs.ox.ac.uk/people/paul.goldberg/slides/pwg-stoc-tfnp.pdf.

[44] Paul W. Goldberg and Christos H. Papadimitriou. "TFNP: An Update." In: *Algorithms and Complexity - 10th International Conference, CIAC 2017, Athens, Greece, May 24-26, 2017, Proceedings*. Ed. by Dimitris Fotakis, Aris Pagourtzis, and Vangelis Th. Paschos. Vol. 10236. Lecture Notes in Computer Science. 2017, pp. 3–9. DOI: 10.1007/978-3-319-57586-5_1. URL: https://doi.org/10.1007/978-3-319-57586-5_1.

[45] Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao. "Further Collapses in TFNP." In: *SIAM J. Comput.* 53.3 (2024), pp. 573–587. DOI: 10.1137/22M1498346. URL: https://doi.org/10.1137/22m1498346.

[46] Mika Göös, Alexandros Hollender, Siddhartha Jain, Gilbert Maystre, William Pires, Robert Robere, and Ran Tao. "Separations in Proof Complexity and TFNP." In: *J. ACM* 71.4 (2024), 26:1–26:45. DOI: 10.1145/3663758. URL: https://doi.org/10.1145/3663758.

[47] Mika Göös, Pritish Kamath, Robert Robere, and Dmitry Sokolov. "Adventures in Monotone Complexity and TFNP." In: *10th Innovations in Theoretical Computer Science Conference, ITCS 2019, January 10-12, 2019, San Diego, California, USA*. Ed. by Avrim Blum. Vol. 124. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, 38:1–38:19. DOI: 10.4230/LIPICS.ITCS.2019.38. URL: https://doi.org/10.4230/LIPIcs.ITCS.2019.38.

[48] Mika Göös, Pritish Kamath, Katerina Sotiraki, and Manolis Zampetakis. "On the Complexity of Modulo-q Arguments and the Chevalley-Warning Theorem." In: *35th Computational Complexity Conference, CCC 2020, Saarbrücken, Germany (Virtual Conference), July 28-31, 2020*. Ed. by Shubhangi Saraf. Vol. 169. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, 19:1–19:42. DOI: 10.4230/LIPICS.CCC.2020.19. URL: https://doi.org/10.4230/LIPIcs.CCC.2020.19.

[49] Ervin Győri. "On division of graphs to connected subgraphs." In: *Combinatorics*. Ed. by András Hajnal and Vera T. Sós. Vol. 18. Colloquia Mathematica Societatis János Bolyai. Proc. Fifth Hungarian Combinatorial Colloquium, Keszthely, 1976. Amsterdam: North-Holland, 1978, pp. 485–494.

[50] Prahladh Harsha, Daniel Mitropolsky, and Alon Rosen. "Downward Self-Reducibility in TFNP." In: *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*. Ed. by Yael Tauman Kalai.

Vol. 251. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 67:1–67:17. DOI: 10.4230/LIPICS.ITCS.2023.67. URL: https://doi.org/10.4230/LIPIcs.ITCS.2023.67.

[51] Sebastian Haslebacher. "ARRIVAL: Recursive Framework & ℓ_1-Contraction." In: *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, July 8-11, 2025, Aarhus, Denmark*. Ed. by Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis. Vol. 334. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025, 95:1–95:17. DOI: 10.4230/LIPICS.ICALP.2025.95. URL: https://doi.org/10.4230/LIPIcs.ICALP.2025.95.

[52] Michael D. Hirsch, Christos H. Papadimitriou, and Stephen A. Vavasis. "Exponential lower bounds for finding Brouwer fix points." In: *J. Complex.* 5.4 (1989), pp. 379–416. DOI: 10.1016/0885-064X(89)90017-4. URL: https://doi.org/10.1016/0885-064X(89)90017-4.

[53] Alexandros Hollender. "The classes PPA-k: Existence from arguments modulo k." In: *Theor. Comput. Sci.* 885 (2021), pp. 15–29. DOI: 10.1016/J.TCS.2021.06.016. URL: https://doi.org/10.1016/j.tcs.2021.06.016.

[54] Christopher M. Homan and Mayur Thakur. "One-way permutations and self-witnessing languages." In: *J. Comput. Syst. Sci.* 67.3 (2003), pp. 608–622. DOI: 10.1016/S0022-0000(03)00068-0. URL: https://doi.org/10.1016/S0022-0000(03)00068-0.

[55] Pavel Hubáček, Chethan Kamath, Karel Král, and Veronika Slívová. "On Average-Case Hardness in TFNP from One-Way Functions." In: *Theory of Cryptography - 18th International Conference, TCC 2020, Durham, NC, USA, November 16-19, 2020, Proceedings, Part III*. Ed. by Rafael Pass and Krzysztof Pietrzak. Vol. 12552. Lecture Notes in Computer Science. Springer, 2020, pp. 614–638. DOI: 10.1007/978-3-030-64381-2_22. URL: https://doi.org/10.1007/978-3-030-64381-2_22.

[56] Pavel Hubáček, Moni Naor, and Eylon Yogev. "The Journey from NP to TFNP Hardness." In: *Proceedings of the Eighth Annual Conference on Innovations in Theoretical Computer Science, ITCS 2017, Berkeley, USA, Simons Institute, January 9-11*. 2017. DOI: 10.4230/LIPICS.ITCS.2017.60. URL: https://doi.org/10.4230/LIPIcs.ITCS.2017.60.

[57] Pavel Hubáček and Jan Václavek. "On Search Complexity of Discrete Logarithm." In: *46th International Symposium on Mathematical Foundations of Computer Science, MFCS 2021, August 23-27, 2021, Tallinn, Estonia*. Ed. by Filippo Bonchi and Simon J. Puglisi. Vol. 202. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 60:1–60:16. DOI: 10.4230/LIPICS.MFCS.2021.60. URL: https://doi.org/10.4230/LIPIcs.MFCS.2021.60.

[58] Pavel Hubáček and Eylon Yogev. "Hardness of Continuous Local Search: Query Complexity and Cryptographic Lower Bounds." In: *SIAM J. Comput.* 49.6 (2020), pp. 1128–1172. DOI: 10.1137/17M1118014. URL: https://doi.org/10.1137/17M1118014.

[59] Pavel Hubáček, Erfan Khaniki, and Neil Thapen. "TFNP Intersections Through the Lens of Feasible Disjunction." In: *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*. Ed. by Venkatesan Guruswami. Vol. 287. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, 63:1–63:24. DOI: 10.4230/LIPICS.ITCS.2024.63. URL: https://doi.org/10.4230/LIPIcs.ITCS.2024.63.

[60] Paul Hunter and Jean-Francois Raskin. "Quantitative Games with Interval Objectives." In: *34th International Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2014)*. Vol. 29. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2014, pp. 365–377. DOI: 10.4230/LIPIcs.FSTTCS.2014.365.

[61] Russell Impagliazzo. "A Personal View of Average-Case Complexity." In: *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*. IEEE Computer Society, 1995, pp. 134–147. DOI: 10.1109/SCT.1995.514853. URL: https://doi.org/10.1109/SCT.1995.514853.

[62] Takashi Ishizuka. "On the complexity of some restricted variants of Quotient Pigeon and a weak variant of Kőnig." In: *Inf. Process. Lett.* 190 (2025), p. 106574. DOI: 10.1016/J.IPL.2025.106574. URL: https://doi.org/10.1016/j.ipl.2025.106574.

[63] Takashi Ishizuka. "The PPP-completeness of the Ward–Szabó theorem." In: (Aug. 2025). arXiv:2507.23345 (v2). DOI: 10.48550/arXiv.2507.23345. arXiv: 2507.23345 [cs.CC]. URL: https://arxiv.org/abs/2507.23345.

[64] Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability Obfuscation." In: *Proceedings of the International Congress of Mathematicians 2022*. Vol. 6. EMS Press, 2022, pp. 4762–4798. DOI: 10.4171/ICM2022/86.

[65] Ruta Jawale, Yael Tauman Kalai, Dakshita Khurana, and Rachel Yun Zhang. "SNARGs for bounded depth computations and PPAD hardness from sub-exponential LWE." In: *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*. Ed. by Samir Khuller and Virginia Vassilevska Williams. ACM, 2021, pp. 708–721. DOI: 10.1145/3406325.3451055. URL: https://doi.org/10.1145/3406325.3451055.

[66] Emil Jeřábek. "Integer factoring and modular square roots." In: *J. Comput. Syst. Sci.* 82.2 (2016), pp. 380–394. DOI: 10.1016/J.JCSS.2015.08.001. URL: https://doi.org/10.1016/j.jcss.2015.08.001.

[67] David S. Johnson, Christos H. Papadimitriou, and Mihalis Yannakakis. "How Easy is Local Search?" In: *J. Comput. Syst. Sci.* 37.1 (1988), pp. 79–100. DOI: 10.1016/0022-0000(88)90046-3. URL: https://doi.org/10.1016/0022-0000(88)90046-3.

[68] Marcin Jurdziński. "Deciding the Winner in Parity Games is in UP ∩ co-UP." In: *Information Processing Letters* 68.3 (1998), pp. 119–124. DOI: 10.1016/S0020-0190(98)00150-1.

[69] Yael Tauman Kalai, Alex Lombardi, and Vinod Vaikuntanathan. "SNARGs and PPAD Hardness from the Decisional Diffie-Hellman Assumption." In: *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part II*. Ed. by Carmit Hazay and Martijn Stam. Vol. 14005. Lecture Notes in Computer Science. Springer, 2023, pp. 470–498. DOI: 10.1007/978-3-031-30617-4_16. URL: https://doi.org/10.1007/978-3-031-30617-4_16.

[70] Yael Tauman Kalai, Omer Paneth, and Lisa Yang. "How to Delegate Computations Publicly." In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC 2019)*. Phoenix, AZ, USA: ACM, June 2019, pp. 1115–1124. DOI: 10.1145/3313276.3316411. URL: https://dl.acm.org/doi/10.1145/3313276.3316411.

[71] Karthik C. S. "Did the Train Reach its Destination: The Complexity of Finding a Witness." In: *CoRR* abs/1609.03840 (2016). arXiv: 1609.03840. URL: http://arxiv.org/abs/1609.03840.

[72] Ilan Komargodski, Moni Naor, and Eylon Yogev. "White-Box vs. Black-Box Complexity of Search Problems: Ramsey and Graph Property Testing." In: *J. ACM* 66.5 (2019), 34:1–34:28. DOI: 10.1145/3341106. URL: https://doi.org/10.1145/3341106.

[73] Ilan Komargodski and Gil Segev. "From Minicrypt to Obfustopia via Private-Key Functional Encryption." In: *J. Cryptol.* 33.2 (2020), pp. 406–458. DOI: 10.1007/S00145-019-09327-X. URL: https://doi.org/10.1007/s00145-019-09327-x.

[74] Jan Krajíček. "Structured pigeonhole principle, search problems and hard tautologies." In: *J. Symb. Log.* 70.2 (2005), pp. 619–630. DOI: 10.2178/jsl/1120224731. URL: https://doi.org/10.2178/jsl/1120224731.

[75] Yuhao Li, William Pires, and Robert Robere. "Intersection Classes in TFNP and Proof Complexity." In: *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*. Ed. by Venkatesan Guruswami. Vol. 287. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum

für Informatik, 2024, 74:1–74:22. DOI: 10.4230/LIPICS.ITCS.2024.74. URL: https://doi.org/10.4230/LIPIcs.ITCS.2024.74.

[76] László Lovász. "A homology theory for spanning trees of a graph." In: *Acta Mathematica Academiae Scientiarum Hungaricae* 30.3–4 (1977), pp. 241–251. DOI: 10.1007/BF01896190. URL: https://doi.org/10.1007/BF01896190.

[77] Willem Mantel. "Problem 28 (Solution by H. Gouwentak, W. Mantel, J. Teixeira de Mattes, F. Schuh and W. A. Wythoff)." In: *Wiskundige Opgaven* 18 (1907), 60–61.

[78] Nimrod Megiddo and Christos H. Papadimitriou. "On Total Functions, Existence Theorems and Computational Complexity." In: *Theor. Comput. Sci.* 81.2 (1991), pp. 317–324. DOI: 10.1016/0304-3975(91)90200-L. URL: https://doi.org/10.1016/0304-3975(91)90200-L.

[79] Ruta Mehta. "Constant Rank Two-Player Games are PPAD-hard." In: *SIAM J. Comput.* 47.5 (2018), pp. 1858–1887. DOI: 10.1137/15M1032338. URL: https://doi.org/10.1137/15M1032338.

[80] Frédéric Meunier. *Lecture at the ICALP 2022 Workshop: Recent Advances on Total Search Problems*. Invited lecture, ICALP 2022 workshop "Recent Advances on Total Search Problems". Slides. July 2022. URL: https://cermics.enpc.fr/~meuniefr/TalkTFNP.pdf.

[81] Christos H. Papadimitriou. "On the Complexity of the Parity Argument and Other Inefficient Proofs of Existence." In: *J. Comput. Syst. Sci.* 48.3 (1994), pp. 498–532. DOI: 10.1016/S0022-0000(05)80063-7. URL: https://doi.org/10.1016/S0022-0000(05)80063-7.

[82] Amol Pasarkar, Christos H. Papadimitriou, and Mihalis Yannakakis. "Extremal Combinatorics, Iterated Pigeonhole Arguments and Generalizations of PPP." In: *14th Innovations in Theoretical Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge, Massachusetts, USA*. Ed. by Yael Tauman Kalai. Vol. 251. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 88:1–88:20. DOI: 10.4230/LIPICS.ITCS.2023.88. URL: https://doi.org/10.4230/LIPIcs.ITCS.2023.88.

[83] Rafael Pass and Muthuramakrishnan Venkitasubramaniam. "Is it Easier to Prove Theorems that are Guaranteed to be True?" In: *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2020, pp. 1255–1267. DOI: 10.1109/FOCS46700.2020.00119. URL: https://doi.org/10.1109/FOCS46700.2020.00119.

[84] Rafael Pass and Muthuramakrishnan Venkitasubramaniam. "Average-case Complexity Through the Lens of Interactive Puzzles." In: *SIGACT News* 52.1 (2021). Guest Column, pp. 47–

69. DOI: 10.1145/3457588.3457598. URL: https://dl.acm.org/doi/10.1145/3457588.3457598.

[85] Svatopluk Poljak, Daniel Turzík, and Pavel Pudlák. "Extensions of k-subsets to $k+1$-subsets – existence versus constructability." In: *Commentationes Mathematicae Universitatis Carolinae* 23.2 (1982), pp. 337–349.

[86] Heinz Prufer. "Neuer Beweis eines Satzes über Permutationen." In: *Archiv der Mathematischen Physik* 27 (1918), pp. 742–744.

[87] Pavel Pudlák. "On reducibility and symmetry of disjoint NP pairs." In: *Theor. Comput. Sci.* 295 (2003), pp. 323–339. DOI: 10.1016/S0304-3975(02)00411-5. URL: https://doi.org/10.1016/S0304-3975(02)00411-5.

[88] Pavel Pudlák. "On the complexity of finding falsifying assignments for Herbrand disjunctions." In: *Arch. Math. Log.* 54.7-8 (2015), pp. 769–783. DOI: 10.1007/s00153-015-0439-6. URL: https://doi.org/10.1007/s00153-015-0439-6.

[89] Frank P. Ramsey. "On a Problem of Formal Logic." In: *Proceedings of the London Mathematical Society* s2-30.1 (Jan. 1930), pp. 264–286. ISSN: 0024-6115. DOI: 10.1112/plms/s2-30.1.264. eprint: https://academic.oup.com/plms/article-pdf/s2-30/1/264/4509556/s2-30-1-264.pdf. URL: https://doi.org/10.1112/plms/s2-30.1.264.

[90] Susanna F. de Rezende, Aaron Potechin, and Kilian Risse. "Clique Is Hard on Average for Sherali-Adams with Bounded Coefficients." In: *CoRR* abs/2404.16722 (2024). DOI: 10.48550/ARXIV.2404.16722. arXiv: 2404.16722. URL: https://doi.org/10.48550/arXiv.2404.16722.

[91] Alexander Russell. "Necessary and Sufficient Condtions for Collision-Free Hashing." In: *J. Cryptol.* 8.2 (1995), pp. 87–100. DOI: 10.1007/BF00190757. URL: https://doi.org/10.1007/BF00190757.

[92] Rahul Savani and Bernhard von Stengel. "Exponentially Many Steps for Finding a Nash Equilibrium in a Bimatrix Game." In: *45th Symposium on Foundations of Computer Science (FOCS 2004), 17-19 October 2004, Rome, Italy, Proceedings*. IEEE Computer Society, 2004, pp. 258–267. DOI: 10.1109/FOCS.2004.28. URL: https://doi.org/10.1109/FOCS.2004.28.

[93] Alejandro A. Schäffer and Mihalis Yannakakis. "Simple Local Search Problems That are Hard to Solve." In: *SIAM J. Comput.* 20.1 (1991), pp. 56–87. DOI: 10.1137/0220004. URL: https://doi.org/10.1137/0220004.

[94] Dominik Scheder and Johannes Tantow. "PLS-Completeness of String Permutations." In: *33rd Annual European Symposium on Algorithms, ESA 2025, Warsaw, Poland, September 15-17, 2025*. Ed. by Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman. Vol. 351. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025, 56:1–56:14. DOI: 10.4230/LIPICS.ESA.

2025.56. URL: https://doi.org/10.4230/LIPIcs.ESA.2025.56.

[95] Adi Shamir. "IP = PSPACE." In: *J. ACM* 39.4 (1992), pp. 869–877. DOI: 10.1145/146585.146609. URL: https://doi.org/10.1145/146585.146609.

[96] Lloyd S. Shapley. "Stochastic Games." In: *Proceedings of the National Academy of Sciences* 39.10 (1953), pp. 1095–1100. DOI: 10.1073/pnas.39.10.1095.

[97] Katerina Sotiraki, Manolis Zampetakis, and Giorgos Zirdelis. "PPP-Completeness with Connections to Cryptography." In: *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*. Ed. by Mikkel Thorup. IEEE Computer Society, 2018, pp. 148–158. DOI: 10.1109/FOCS.2018.00023. URL: https://doi.org/10.1109/FOCS.2018.00023.

[98] Emanuel Sperner. "Ein Satz über Untermengen einer endlichen Menge." In: *Mathematische Zeitschrift* 27.1 (1928), pp. 544–548.

[99] Neil Thapen. "How to fit large complexity classes into TFNP." In: *CoRR* abs/2412.09984 (2024). DOI: 10.48550/ARXIV.2412.09984. arXiv: 2412.09984. URL: https://doi.org/10.48550/arXiv.2412.09984.

[100] Paul Turán. "On an extremal problem in graph theory (in Hungarian)." In: *Matematikai és Fizikai Lapok* 48 (1941), pp. 436–452. ISSN: 0020-0190. DOI: https://doi.org/10.1016/j.ipl.2018.12.009. URL: https://www.sciencedirect.com/science/article/pii/S0020019019300018.

[101] C. Ward and S. Szabó. "On swell-colored complete graphs." eng. In: *Acta Mathematica Universitatis Comenianae. New Series* 63.2 (1994), pp. 303–308. URL: http://eudml.org/doc/118694.

[102] Uri Zwick and Mike S. Paterson. "The Complexity of Mean Payoff Games on Graphs." In: *Theoretical Computer Science* 158.1–2 (1996), pp. 343–359. DOI: 10.1016/0304-3975(95)00188-3.