From Proof Complexity to Circuit Complexity via Interactive Protocols

Noel Arteche
Lund University, Sweden
University of Copenhagen, Denmark

Erfan Khaniki
Institute of Mathematics of the Czech Academy of Sciences, Prague, Czech Republic

Ján Pich
University of Oxford, UK

Rahul Santhanam
University of Oxford, UK

Abstract

Folklore in complexity theory suspects that circuit lower bounds against $\mathbf{NC}^1$ or $\mathbf{P}/\mathrm{poly}$, currently out of reach, are a necessary step towards proving strong proof complexity lower bounds for systems like Frege or Extended Frege. Establishing such a connection formally, however, is already daunting, as it would imply the breakthrough separation $\mathbf{NEXP} \not\subseteq \mathbf{P}/\mathrm{poly}$, as recently observed by Pich and Santhanam [58].

We show such a connection conditionally for the Implicit Extended Frege proof system (iEF) introduced by Krajíček [45], capable of formalizing most of contemporary complexity theory. In particular, we show that if iEF proves efficiently the standard derandomization assumption that a concrete Boolean function is hard on average for subexponential-size circuits, then any superpolynomial lower bound on the length of iEF proofs implies $\#\mathbf{P} \not\subseteq \mathbf{FP}/\mathrm{poly}$ (which would in turn imply, for example, $\mathbf{PSPACE} \not\subseteq \mathbf{P}/\mathrm{poly}$). Our proof exploits the formalization inside iEF of the soundness of the sum-check protocol of Lund, Fortnow, Karloff, and Nisan [54]. This has consequences for the self-provability of circuit upper bounds in iEF. Interestingly, further improving our result seems to require progress in constructing interactive proof systems with more efficient provers.

2012 ACM Subject Classification Theory of computation → Proof complexity; Theory of computation → Circuit complexity; Theory of computation → Complexity theory and logic

Keywords and phrases proof complexity, circuit complexity, interactive protocols

Digital Object Identifier 10.4230/LIPIcs.ICALP.2024.12

Category Track A: Algorithms, Complexity and Games

Related Version *Full Version*: `https://arxiv.org/abs/2405.02232`

Funding *Noel Arteche*: This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.
Erfan Khaniki: This work was supported by the Institute of Mathematics of the Czech Academy of Sciences (RVO 67985840) and GAČR grant 19-27871X.
Ján Pich: This work received support from the Royal Society University Research Fellowship URF\R1\211106 “Proof complexity and circuit complexity: a unified approach”.

Acknowledgements Independently, Albert Atserias suggested to us to consider using interactive proof systems to derive this type of connections. We thank Pavel Pudlák for useful comments, and the anonymous reviewers for relevant comments and references. This work was done in part while the first author was visiting the University of Oxford and the Institute of Mathematics of the Czech Academy of Sciences. For the purpose of Open Access, the authors have applied a CC BY public copyright license to any Author Accepted Manuscript version arising from this submission.

51st International Colloquium on Automata, Languages, and Programming (ICALP 2024).
Editors: Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson;
Article No. 12; pp. 12:1–12:20
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

At a high level, both circuit complexity and proof complexity can be thought of as an approach towards the **P** versus **NP** question. The circuit complexity program, which met with considerable success in the 1980s, tries to prove lower bounds against gradually larger circuit classes, hoping to eventually show $\mathbf{NP} \not\subseteq \mathbf{P}/\mathrm{poly}$. Proof complexity, often identified with the so-called Cook-Reckhow program, intends to show $\mathbf{NP} \neq \mathbf{coNP}$ and, in turn, $\mathbf{P} \neq \mathbf{NP}$, by proving lower bounds against gradually more powerful proof systems for propositional logic.

While both enterprises share the motivation to study *concrete* computational models of increasing power hoping to build up techniques to attack the long-sought separations, there exist notable differences. Circuit complexity looks at deterministic models of computation, while proof complexity deals with proof systems, which are inherently non-deterministic. Furthermore, while circuit complexity has a clear end-goal (lower bounds against general Boolean circuits), it remains wide open whether the Cook-Reckhow program can be realized even in principle. It is not known whether lower bounds against strong systems like Extended Frege can imply lower bounds for every other system and, as such, one could potentially keep proving lower bounds for ever-stronger systems without ever settling whether $\mathbf{NP} \neq \mathbf{coNP}$.

The parallels between circuit complexity and proof complexity are made clearer by Frege systems. For each circuit complexity class $\mathcal{C}$, one can define the proof system $\mathcal{C}$-Frege, in which proof lines are restricted to be circuits from $\mathcal{C}$. In this setting strong systems like Frege and Extended Frege correspond to $\mathbf{NC}^1$-Frege and $\mathbf{P}$/poly-Frege, respectively, and thus the natural question arises: Can we turn explicit lower bounds for $\mathcal{C}$ circuits into lower bounds for $\mathcal{C}$-Frege systems, and vice versa?

While the question is essentially open, work on weaker systems and circuit classes has proven successful. In one direction, the method of feasible interpolation [43, 65, 50] (see [51, §17.9.1] for the history of the method) has been extensively applied to obtain proof complexity lower bounds. The framework of feasible interpolation formalizes the idea of extracting computational content from proofs: given short proofs in a given system, one can extract a small Boolean circuit in some restricted classes for a related interpolant function. Contrapositively, circuit lower bounds for such functions (often coming from unconditional results such as lower bounds against monotone circuits [64, 4, 3]), turn into lower bounds for proofs systems like Resolution [50] or Cutting Planes [61] (and conditionally for other systems, such as Polynomial Calculus or Sum-of-Squares [32]). Unfortunately, this connection breaks for stronger proof systems: already $\mathbf{AC}^0$-Frege and $\mathbf{TC}^0$-Frege are known to lack feasible interpolation properties[1] under standard cryptographic hardness assumptions [52, 13, 12], and this holds even if we allow feasible interpolation by quantum circuits [6].

In the other direction (circuit complexity from proof complexity), the theory of lifting has unveiled deep connections between proofs, circuits and communication protocols. Here, so-called query-to-communication lifting theorems translate query complexity lower bounds (corresponding to weak systems, like Resolution) into communication complexity lower bounds (e.g. [63, 53]). The latter provide restricted circuit lower bounds, such as for monotone circuits (see e.g. [27, 24, 25] and references therein). It is, however, not known how to derive non-monotone lower bounds for unrestricted Boolean circuits by lifting proof complexity lower bounds.

[1] Some of these systems are known to admit some form of interpolation by stronger computational models, see e.g. [62, 23], but we are interested in Boolean circuits.

For proper Frege systems, the connection has worked mostly in one direction, from circuits to proofs, particularly at the level of techniques. The method of random restrictions and the celebrated switching lemmas used to show constant-depth circuit lower bounds in the 1980s [26, 1, 33] were successfully transferred into $\mathbf{AC}^0$-Frege lower bounds shortly after [2, 10, 43, 8, 59, 49]. This suggests that understanding what makes proof lines large might be necessary to understand why proofs are long. Intriguingly, understanding the proof lines alone does not seem to suffice: the $\mathbf{AC}^0[p]$ lower bounds of Razborov and Smolensky [66, 68] are yet to be successfuly translated to proof complexity, with lower bounds for $\mathbf{AC}^0[2]$-Frege being one of the prominent frontier problems in the field.

The current situation seems to suggest that in order to make progress towards proof complexity lower bounds, it is *necessary* (though seemingly not sufficient) to first obtain strong enough circuit lower bounds. In particular, under this folklore belief, circuit lower bounds against $\mathbf{NC}^1$ or $\mathbf{P}/\text{poly}$, currently out or reach, would be a necessary step towards proving strong proof complexity lower bounds for systems like Frege or Extended Frege. However, the suspicion remains unproven, and no generic way of deriving explicit circuit lower bounds for unrestricted Boolean circuits from proof complexity lower bounds for concrete propositional proof systems has been discovered[2].

The first result giving such a connection under relatively conventional assumptions which are presumably weaker than the conclusion of the connection itself was presented recently by Pich and Santhanam [58]. Specifically, they showed that any superpolynomial lower bound on the length of tautologies in the Extended Frege system EF implies $\mathbf{NP} \not\subseteq \mathbf{P}/\text{poly}$ assuming hypotheses (1) and (2) below:

(1) (*Provable circuit lower bound.*) EF proves efficiently that a concrete Boolean function in $\mathbf{E}$ is average-case hard for subexponential-size circuits.

(2) (*Provable reduction of OWFs to* $\mathbf{P} \neq \mathbf{NP}$.) EF proves efficiently that a polynomial-time function transforms circuits breaking one-way functions into circuits solving SAT.

We remark that Hypothesis (1) above presupposes $\mathbf{E} \not\subseteq \mathbf{P}/\text{poly}$, which is however believed to be a significantly weaker statement than $\mathbf{NP} \not\subseteq \mathbf{P}/\text{poly}$. Alternatively, Hypotheses (1) and (2) can be replaced by a single assumption on the feasible provability of the existence of antickeckers in EF. These results remain valid even if we replace EF by an essentially arbitrary proof system simulating EF.

Crucially, improving this and related results by dropping the hypotheses is surprisingly daunting. As noted by Pich and Santhanam [58, Prop. 1], if one unconditionally establishes the implication "if S is not polynomially bounded, then $\mathbf{NP} \not\subseteq \mathbf{P}/\text{poly}$" for a concrete proof system S, then the breakthrough separation $\mathbf{NP} \not\subseteq \mathbf{SIZE}[n^k]$, for every fixed k (and $\mathbf{NEXP} \not\subseteq \mathbf{P}/\text{poly}$) follows!

In short, proving a formal connection between proof complexity and circuit complexity provably requires breakthrough circuit lower bounds! Despite this setback, one can still hope to get evidence that points at these connections, possibly by shifting some of the components of the ingredients. Namely, one may try to (a) adopt some hardness assumption, in the style of [58]; (b) conclude lower bounds weaker than $\mathbf{NP} \not\subseteq \mathbf{P}/\text{poly}$; or (c) look at non-Cook-Reckhow proof systems (such as MA proof systems or proof systems for languages beyond $\mathbf{coNP}$).

[2] We note that the issue lies in establishing such a connection for a *concrete* system. Of course, the statement "there is a proof system S such that if S is not polynomially bounded, then $\mathbf{P} \neq \mathbf{NP}$" is true: if $\mathbf{NP} = \mathbf{coNP}$ the implication is vacuously true by taking a polynomially bounded proof system; if $\mathbf{NP} \neq \mathbf{coNP}$, then $\mathbf{P} \neq \mathbf{NP}$ and thus the statement holds for any proof system. It would be dramatically different to obtain such a connection for a concrete system.

In this style, Grochow and Pitassi [31] showed that the Ideal Proof System (IPS) does satisfy such a connection, to *algebraic* circuit complexity. Indeed, any superpolynomial lower bound in the length of proofs in $\mathsf{IPS}_{\mathbb{F}}$ implies $\mathbf{VP}_{\mathbb{F}} \neq \mathbf{VNP}_{\mathbb{F}}$. Grochow and Pitassi avoid the Pich-Santhanam barrier by means of (b) and (c) above: first, IPS is not known to be a Cook-Reckhow system, since proofs are verified by randomized machines via polynomial identity testing; second, the lower bounds are algebraic and not Boolean. Recall that while separating $\mathbf{VP}$ and $\mathbf{VNP}$ is a necessary step[3] towards $\mathbf{NP} \not\subseteq \mathbf{P}/\text{poly}$ [14], the converse is not known.

Another interesting connection has been established in the realm of quantified Boolean formulas, where the connection can be made essentially optimal. Beyersdorff, Bonacina, Chew, and Pich [11] showed that for every circuit class $\mathcal{C}$, the quantified system $\mathcal{C}$-Frege + $\forall$red is not polynomially bounded if and only if either $\mathbf{PSPACE} \not\subseteq \mathcal{C}$ or $\mathcal{C}$-Frege is not polynomially bounded. Here, $\mathcal{C}$-Frege + $\forall$red stands for the natural quantified system obtained by extending $\mathcal{C}$-Frege with a universal reduction rule, which takes care of universal quantifiers by instantiating concrete values for its variables in the hope of refuting the formula. The reason this avoids the Pich-Santhanam barrier is the disjunct in the conclusion. That is, in the context of QBF the conclusion of the Pich-Santhanam barrier becomes that that either $\mathbf{NEXP} \not\subseteq \mathbf{P}/\text{poly}$ or $\mathcal{C}$-Frege is not polynomially bounded. But this disjunction is no breakthrough, since it follows directly by a diagonalization argument anyway: if a propositional system is polynomially bounded, then $\mathbf{NEXP}$ is hard for $\mathbf{P}/\text{poly}$ [44].

Contributions

We prove a new conditional connection between proof complexity and circuit complexity, giving further evidence that strong proof complexity lower bounds require circuit lower bounds. This constitutes the first example of a natural proof system that is conditionally Cook-Reckhow and whose lower bounds imply Boolean circuit lower bounds.

The system in question is (an extension of) the Implicit Extended Frege (iEF) proof system of Krajíček [45], capable of formalizing most of contemporary complexity theory. Our result can be informally stated as follows, where $\mathsf{iEF}^{\mathrm{tt}(h)}$ stands for the proof system extending iEF by axioms $\mathrm{tt}^{\mathrm{avg}}_{1/4}(h_n, 2^{n/4})$ claiming there are no circuits of size $2^{n/4}$ approximating a concrete function h on more than a $(1/2 + 1/2^{n/4})$-fraction of the inputs.[4]

▶ **Theorem 1** (Main theorem, informal)**.** *Suppose there is a Boolean function $h \in \mathbf{NE} \cap \mathbf{coNE}$ that is hard on average for subexponential-size circuits. If the Cook-Reckhow proof system $\mathsf{iEF}^{\mathrm{tt}(h)}$ is not polynomially bounded, then $\#\mathbf{P} \not\subseteq \mathbf{FP}/\text{poly}$.*

In the theorem above one could instead consider the system $\mathsf{iEF}^{\mathrm{tt}(h)}$ for some unconditionally hard function family h that is guaranteed to exist. The only problem in this case is that we might need non-uniform advice to verify the proofs, and so the system would not be Cook-Reckhow (we refer to Cook and Krajíček [19] for a systematic treatment of non-uniform proof systems).

One can interpret our theorem as improving on the connection of Pich and Santhanam [58] from proof complexity to circuit complexity. Our result improves that of Pich and Santhanam by completely dropping their second assumption (the one about EF proving the existence of one-way functions under $\mathbf{P} \neq \mathbf{NP}$). The price to pay for these changes is two-fold:

[3] Unconditionally over finite fields, and assuming the Generalized Riemann Hypothesis for infinite fields.

[4] For technical reasons, we define $\mathsf{iEF}^{\mathrm{tt}(h)}$ using a system which is polynomially equivalent to iEF instead of iEF itself, see Definition 17.

1. we need to replace EF by the seemingly stronger Implicit Extended Frege system (iEF). Informally, iEF extends EF with an extra rule allowing us to derive a formula φ after we have derived that a truth table of a given circuit encodes an EF-proof of φ. Such a circuit is called an *implicit* proof;
2. we can conclude only $\#\mathbf{P} \not\subseteq \mathbf{FP}/\text{poly}$ from iEF lower bounds, instead of $\mathbf{NP} \not\subseteq \mathbf{P}/\text{poly}$.

One may also compare our result to that of Grochow and Pitassi [31], who showed $\mathbf{VP} \neq \mathbf{VNP}$ (and hence hardness of computing the permanent) would follow from IPS lower bounds. Like our result, the IPS proof system is only conditionally Cook-Reckhow. Indeed, IPS is a Merlin-Arthur proof system which can be derandomized[5] under standard assumptions, like $\mathbf{E}$ being hard to approximate by subexponential-size circuits. Our result is in some sense stronger in that the lower bounds obtained are Boolean rather than algebraic. However, we seem to be getting to lower bounds for the same problem as Grochow and Pitassi, since computing the permanent is both $\mathbf{VNP}$-complete and $\#\mathbf{P}$-complete.

We note that the requirement that $h \in \mathbf{NE} \cap \mathbf{coNE}$ is not strictly needed and, in fact, one can phrase the result in a more general style (as we do in the technical part) in which the connection holds for any extension of iEF by truth table formulas for any hard function. Observe, however, that iEF is a very strong proof system, with its bounded arithmetic counterpart being the theory V^1_2 (or $\mathsf{S}^1_2 + 1\text{-}\mathsf{EXP}$, in the first-order setting), and so it is plausible that iEF already proves such a circuit lower bound. For example, already EF can prove efficiently the PCP theorem [57], $\mathbf{AC}^0$, $\mathbf{AC}^0[2]$ and monotone circuit lower bounds [67, 55], or the hardness amplification producing average-case hard functions in $\mathbf{E}$ from worst-case hard functions in $\mathbf{E}$ [36]. Furthermore, iEF proves efficiently the correctness of Zhuk's algorithm from a CSP dichotomy [28, 29]. Hence, it is plausible to imagine that if circuit lower bounds are at all provable, they may well be provable already in iEF. If that turned out to be the case, then the concrete proof system in our main theorem becomes iEF itself.

▶ **Corollary 2** (Main theorem, restated). *Assume that* iEF *proves efficiently* $\mathrm{tt}^{\mathrm{avg}}_{1/4}(h_n, 2^{n/4})$ *for some function family h and each sufficiently big n. Then, if* iEF *is not polynomially bounded,* $\#\mathbf{P} \not\subseteq \mathbf{FP}/\text{poly}$.

Let us note that one cannot make big improvements to this result without hitting the Pich-Santhanam barrier that implies $\mathbf{NEXP} \not\subseteq \mathbf{P}/\text{poly}$ unconditionally: if we managed to prove Theorem 1 for a Cook-Reckhow proof system, then $\mathbf{NEXP} \not\subseteq \mathbf{P}/\text{poly}$ would follow unconditionally. On the other hand, if our final goal is to prove $\mathbf{FP} \neq \#\mathbf{P}$, then the assumption of Theorem 1 is given to us for free even for some hard $h \in \mathbf{E}$, as otherwise, if $\mathbf{E}$ can be computed by subexponential-size circuits, it is not hard to show that $\mathbf{P} \neq \mathbf{NP}$ [44].

Consequences for self-provability of circuit upper bounds

Our result has consequences for the self-provability of circuit upper bounds. Suppose that $\#\mathbf{P} \subseteq \mathbf{FP}/\text{poly}$. Then, there is a sequence of polynomial-size circuits $\{C_n\}_{n\in\mathbb{N}}$ that on input a formula φ of size n, outputs a satisfying assignment if one exists. This means that the propositional formula $\mathrm{SAT}_n(\varphi, \alpha) \to \mathrm{SAT}_n(\varphi, C_n(\varphi))$ claiming the correctness of C_n as a SAT solver is tautological (where SAT_n is the satisfiability predicate, taking a formula φ and

[5] In fact, derandomizing IPS at all by simulating it by a Cook-Reckhow system implies a non-trivial derandomization of polynomial identity testing to $\mathbf{NP}$ [30]; this, in turn, implies some circuit lower bounds, as shown by Kabanets and Impagliazzo [38].

an assignment α and evaluating the formula). But by Theorem 1, $\mathsf{iEF}^{\mathrm{tt}}$ is now polynomially bounded, and so the proof system is able to efficiently argue for the correctness of the circuits. Namely, the mere validity of the upper bound $\#\mathbf{P} \subseteq \mathbf{FP}/\mathrm{poly}$ would imply the efficient propositional provability of $\mathrm{SAT} \in \mathbf{P}/\mathrm{poly}$.

Outline of the proof

Our main result follows from a derandomization of the known fact that $\mathbf{coNP} \not\subseteq \mathbf{MA}$ implies $\#\mathbf{P} \not\subseteq \mathbf{FP}/\mathrm{poly}$ (see, for example, [5, Thm. 8.22]), together with a formalization of the underlying MA system in a suitable theory of bounded arithmetic. The implication holds, actually, for the MA system given by the sum-check protocol of Lund, Fortnow, Karloff, and Nisan [54] in which proofs consist of a circuit simulating the moves of the Prover in the protocol, so that given such a circuit, the Verifier can simulate the entire protocol on their own with the aid of randomness. If $\#\mathbf{P} \subseteq \mathbf{FP}/\mathrm{poly}$, then the $\#\mathbf{P}$-powerful Prover in the sum-check protocol can be replaced by a polynomial-size circuit and thus the system is a polynomially bounded Merlin-Arthur system. Clearly, lower bounds on the length of proofs in this system are exactly circuit lower bounds against $\#\mathbf{P}$.

Since $\mathbf{MA}$ can be derandomized under standard hardness assumptions, assuming, for example, that $\mathbf{E}$ is hard for subexponential-size circuits, the proof system R based on the sum-check protocol above becomes a Cook-Reckhow system such that if R is not polynomially bounded, then $\#\mathbf{P} \not\subseteq \mathbf{FP}/\mathrm{poly}$. This is almost our goal. Our task now is to replace this system by a different more standard Cook-Reckhow system S. This can be achieved by proving efficiently the reflection principle of the system R in S, which essentially amounts to proving the soundness of the sum-check protocol in S. Here, we employ a recent work of Khaniki [40], in which the soundness of the sum-check protocol was formalized in $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$.

In order to translate the formalization inside $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ into propositional logic, we need to express the soundness of the sum-check protocol by propositional formulas. This is achieved using the machinery of approximate counting of Jeřábek [37], which exploits Nisan-Wigderson generators based on a hard Boolean function.

Open problems

Improving our result seems to require significant conceptual work. Of course, simultaneously dropping the circuit lower bound assumption as well as getting the stronger separation $\mathbf{NP} \not\subseteq \mathbf{P}/\mathrm{poly}$ would already imply $\mathbf{NEXP} \not\subseteq \mathbf{P}/\mathrm{poly}$, but one may hope to improve the existing connection by improving on one of the two fronts only. Interestingly, this seems to require progress in some of the central open questions in the theory of interactive proof systems or in hardness magnification.

The power of the prover

Is it possible to strengthen the conclusion of the main theorem all the way down to $\mathbf{NP} \not\subseteq \mathbf{P}/\mathrm{poly}$? This would follow, for example, if we managed to design an interactive protocol for TAUT with a prover solving only $\mathbf{NP}$ problems and prove its correctness in iEF (unlike the current situation, where the prover is required to compute a $\#\mathbf{P}$-complete function). The general question of constructing a protocol for a language L where the prover's power is limited to $\mathbf{P}^L$ is a well-known open problem in the theory of interactive proof systems (see, for example, [5, §8.4]).

Note, of course, that the existence of such a protocol does not suffice, since its soundness must be provable inside iEF. In fact, the reason why we require iEF (or $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$) to carry out the formalization of the existing sum-check protocol is that one cannot feasibly talk about $\#\mathrm{SAT}$ directly in EF or S^1_2 (unless $\mathbf{FP} = \#\mathbf{P}$).

Hardness magnification

Is it possible to replace iEF in the main theorem by Gentzen's system G, or even by Extended Frege? One option would be to carry out the existing formalization inside EF, as mentioned above. The caveat would be, however, that we would then have to make the assumption on truth table tautologies for EF. Whether EF can prove general circuit lower bounds at all seems much less believable than for iEF, and so the plausibility of our hypotheses seems affected.

Instead, one may choose to keep everything in iEF and obtain the connection indirectly for EF via hardness magnification. Is there a natural class of formulas over which EF simulates iEF (and which are believably hard for EF)? If so, assuming hardness of these formulas for EF would imply iEF lower bounds. By our main theorem, $\#\mathbf{P} \not\subseteq \mathbf{P}/\mathrm{poly}$ would follow. To the best of our knowledge, no such type of hardness magnification is known for strong proof systems.

2 Preliminaries

We assume familiarity with the central concepts of computational complexity theory, propositional proof complexity and mathematical logic. Below we review the central concepts needed in this paper and fix some notation.

2.1 Proof complexity

Following Cook and Reckhow [22], a *propositional proof system* S for the language $\textsc{Taut}$ of propositional tautologies is a polynomial-time surjective function $S : \{0,1\}^* \to \textsc{Taut}$ taking as input a proof $\pi \in \{0,1\}^*$ and outputting $S(\pi) = \varphi$, the theorem that π proves. Soundness follows from the fact that the range is exactly $\textsc{Taut}$, and implicational completeness is guaranteed by the fact that S is surjective. We sometimes drop the term *proof* in *proof system* and use the term *system* alone to refer to a function S that is not guaranteed to be a Cook-Reckhow proof system (perhaps because it is unsound, or not deterministically computable).

We denote by $\mathsf{size}_S(\varphi)$ the *size* of the smallest S-proof of φ plus the size of φ. A proof system S is *polynomially-bounded* if for every $\varphi \in \textsc{Taut}$, $\mathsf{size}_S(\varphi) \leq |\varphi|^{O(1)}$. We say that a proof system S *polynomially simulates* a system Q, written $S \geq Q$, if for every $\varphi \in \textsc{Taut}$, $\mathsf{size}_S(\varphi) \leq \mathsf{size}_Q(\varphi)^{O(1)}$. Note that the notion of size and the definition of simulation do not exploit the soundness requirement of Cook-Reckhow systems. In particular, an unsound system can be polynomially bounded and simulate every other system. In some cases simulations hold only for some set T of tautologies, such as the set of tautologies written as 3DNFs, and not for all formulas, and then we say that S polynomially simulates Q over T. Given a family $\{\varphi_n\}_{n\in\mathbb{N}}$ of propositional tautologies, we write $S \vdash \varphi_n$ whenever $\mathsf{size}_S(\varphi_n) \leq |\varphi_n|^{O(1)}$.

2.1.1 Frege systems

Proof complexity studies a wide variety of proof systems. The most important ones for us are *Frege systems*. A Frege system is a finite set of axiom schemas and inference rules that are sound and implicationally complete for the language of propositional tautologies built from the Boolean connectives negation ($\neg$), conjunction ($\wedge$), and disjunction ($\vee$). A Frege proof is a sequence of formulas where each formula is obtained by either substitution of an axiom schema or by application of an inference rule on previously derived formulas. The specific choice of rules does not affect proof size up to polynomial factors, as long as there are only finitely many rules and these are sound and implicationally complete. Indeed, Frege systems polynomially simulate each other [51, Thm. 4.4.13]. Alternatively, one may choose to think of Frege systems as some variant of Natural Deduction or the Sequent Calculus for classical propositional logic.

Particularly important for us is the Extended Frege (EF) system, in which proof lines can be Boolean circuits and not just formulas, which would allow in principle for more succinct proofs. We shall often consider extensions of Extended Frege by sets of additional axioms. For a set $A \subseteq \textsc{Taut}$ of tautologies recognizable in polynomial time, the system $\mathsf{EF} + A$ refers to Extended Frege extended with substitution instances of any formula in A. Note that if A were to contain contingent formulas, then $\mathsf{EF} + A$ would not be sound; in particular, it would not be a Cook-Reckhow system, though it would be polynomially bounded.

A useful property of EF is the fact that $\mathsf{EF} + \mathrm{Ref}_S \geq S$ for every propositional system S [47]. Here Ref_S is the sequence of tautologies encoding the *reflection principle for* S, which states that S is sound. Namely, $\mathrm{Ref}_S \coloneqq \{\mathrm{Ref}_{S,n,m}\}_{n,m\in\mathbb{N}}$ where the formulas $\mathrm{Ref}_{S,n,m} \coloneqq \mathrm{Prf}_{S,n,m}(\pi,\varphi) \to \mathrm{Sat}_{n,m}(\varphi,\alpha)$ encode the soundness of S, and φ is a formula of size n, π is a purported S-proof of size m and α is an assignment to the variables in φ, which are all encoded by free variables. The formula $\mathrm{Prf}_{S,n,m}$ encodes that π is a correct S-proof of φ, and $\mathrm{Sat}_{n,m}(\varphi,\alpha)$ encodes the standard satisfaction relation for propositional formulas. Alternatively, one may exploit the same relation with respect to the *consistency* of S, $\mathrm{Con}_S \coloneqq \{\mathrm{Con}_{S,m}\}_{m\in\mathbb{N}}$, where $\mathrm{Con}_{S,m} \coloneqq \neg\,\mathrm{Prf}_{S,1,m}(\pi,\bot)$ and π encodes a purported proof of size m.

2.1.2 Quantified propositional systems

It is often convenient to operate on systems capable of reasoning with *quantified* Boolean formulas, where the quantification ranges over $\{0,1\}$. We denote by Σ_i^q (respectively, Π_i^q) the class of quantified Boolean formulas with i alternations between existential and universal quantifiers, starting with an existential (respectively, universal) one.

We are particularly interested in Gentzen's Sequent Calculus for quantified propositional logic. The system extends the usual propositional Sequent Calculus by four new rules to handle quantifiers (see [51, Def. 4.1.2] for a formal definition of the rules). We denote this system by G, and by G^* its tree-like counterpart. The system G_i, for $i \in \mathbb{N}$, corresponds to G where the quantified formulas appearing in the sequents can only be in the class $\Sigma_i^q \cup \Pi_i^q$. The tree-like counterpart of G_i is naturally denoted G_i^*. It is useful to know that EF and G_1^* are polynomially equivalent with respect to Π_1^q formulas [51, Thm. 4.1.3].

2.1.3 Implicit proof systems

Implicit proof systems constitute a systematic way of obtaining, for every proof system S, a potentially stronger system S', and were introduced by Krajíček [45]. The essential idea is to encode a given proof in the system S as a multi-output Boolean circuit taking as input a number i in binary and outputting the i-th step of the proof. More formally, given propositional proof systems S and Q, a proof of a tautology φ in the *implicit system* $[S, Q]$ is a pair (π, C) consisting of a proof and a circuit, such that the truth table of C encodes a valid Q-proof of φ (the *implicit* proof), while π is an *explicit* S-proof of the formula $\mathrm{Correct}_Q(\varphi, C)$, which is the formula stating that the truth table of C is a correct Q-proof of φ. If S and Q are Cook-Reckhow proof systems, then so is $[S, Q]$.

For a system S, the implicit system $[S, S]$ is denoted by $\mathsf{i}S$. In particular, we shall work with the *Implicit Extended Frege* proof system, $\mathsf{iEF} := [\mathsf{EF}, \mathsf{EF}]$. The system iEF is particularly strong, and it can in fact simulate all of Gentzen's G with respect to propositional tautologies [45, Cor. 2.4].

2.2 Bounded arithmetic

Our proofs exploit the connections between propositional proof complexity and theories of bounded arithmetic. Below we cover the essential preliminaries, which should be accessible to any reader with basic knowledge of first-order logic.

2.2.1 The theories S^1_2 and $\mathsf{S}^1_2 + 1\text{-}\mathsf{EXP}$

Theories of bounded arithmetic capture various levels of feasible reasoning and act as a uniform counterpart of propositional systems. Intuitively, feasibility is achieved by restricting the complexity of formulas over which one can apply general reasoning schemes like induction.

The central theory for us is Buss's S^1_2, which we think of as corresponding to polynomial-time reasoning. In this context, we work over the first-order language of bounded arithmetic, $\mathcal{L}_{\mathrm{BA}} := \{0, S, +, \cdot, <, |x|, \lfloor x/2 \rfloor, x\#y\}$, which extends the language of Peano Arithmetic by the symbols $|x|$, $\lfloor x/2 \rfloor$ and $x\#y$. The standard interpretation of $\lfloor x/2 \rfloor$ is clear. The notation $|x|$ denotes the length of the binary encoding of the number x, $\lceil \log(x+1) \rceil$, while the *smash symbol* $x\#y$ stands for $2^{|x|\cdot|y|}$.

The definition of *bounded formulas* is analogous to the bounded quantification one encounters in the polynomial hierarchy. For a quantifier $Q \in \{\exists, \forall\}$ and a term t in the language of bounded arithmetic, a formula of the form $Qx < t.\varphi(x)$ stands for either $\forall x.(x < t \to \varphi(x))$ or $\exists x.(x < t \wedge \varphi(x))$. These are called *bounded quantifiers*. Whenever the bounded quantifier is of the form $Q < |s|$ for some term s, we talk about *sharply bounded quantifiers*. The hierarchy of *bounded formulas* consists of the classes Σ^b_n and Π^b_n, for $n \geq 1$, which are defined by counting the alternations of bounded quantifiers ignoring the sharply bounded ones, starting with an existential (respectively, universal) one. The class Δ^b_n consists of all formulas that admit an equivalent definition in both Σ^b_n and Π^b_n. In particular, the class Δ^b_0 stands for all formulas with sharply bounded quantifiers only.

The theory S^1_2 of Buss [15] extends Robinson's arithmetic Q by some basic axioms for the new function symbols and the polynomial induction scheme (PInd) for Σ^b_1-formulas: for every $\varphi \in \Sigma^b_1$, the theory contains the axiom

$$\varphi(0) \wedge \forall x(\varphi(\lfloor x/2 \rfloor) \to \varphi(x)) \to \forall x \varphi(x). \tag{PInd}$$

An alternative system intended to capture polynomial-time reasoning is Cook's equational theory PV [21]. In the formalism of PV one has some basic function symbols and introduces new ones recursively by composition and limited recursion on notation, in the style of Cobham's functional definition of $\mathbf{FP}$ [17]. In this way, the function symbols obtained in PV are precisely those of all polynomial-time functions over the naturals. The first-order version of PV is PV_1 [48, 16, 18]. Without loss of generality, we shall work in the theory $\mathsf{S}^1_2(\mathsf{PV})$, which is the theory S^1_2 in the language of bounded arithmetic extended by all PV function symbols, meaning that we have a fresh symbol for each function in **FP**, and induction is now available for all $\Sigma^b_1(\mathsf{PV})$ formulas. We abuse notation and refer to this directly as S^1_2.

While S^1_2 is able to formalize a significant amount of complexity theory and some mathematics, it suffers from the drawback of being unable to even state the existence of exponentially large objects. For certain more elaborate arguments we shall work instead inside $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$, which patches this issue. We follow here the definition of the theory given by Krajíček [45, Cor. 2.2]: we write $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP} \vdash \forall x \varphi(x)$ for some arithmetic formula φ if there exists a term t such that

$$\mathsf{S}^1_2 \vdash \forall x \forall y (t(x) \leq |y| \rightarrow \varphi(x)).$$

The definition is somewhat indirect and may be hard to grasp at first glance. Intuitively, it allows one to derive properties about x under the assumption that $y = 2^x$ exists.

The theory S^1_2 corresponds to polynomial-time computations in the sense that the provably total relations in S^1_2 are precisely the polynomial-time-computable ones. The same relation holds for $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ and the complexity class **EXP**.

2.2.2 Approximate counting

Many of the formalizations carried out in bounded arithmetic require the ability to count. In some cases, small sets can be counted *exactly*, but one often requires more sophisticated machinery for *approximate counting*, needed to formalize many probabilistic arguments.

For $a \in \mathbb{N}$, a *bounded definable set* is a set of naturals $X = \{x < a \mid \varphi(x)\} \subseteq [0, a)$, where $\varphi \in \Sigma^b_\infty$ is some arithmetic formula. For $X \subseteq a$ and $Y \subseteq b$, we define $X \times Y := \{bx + y \mid x \in X, y \in Y\} \subseteq ab$ and $X \dot\cup Y := X \cup \{y + a \mid y \in Y\} \subseteq a + b$. Rational numbers are assumed to be represented by pairs of integers in the natural way. We also use the unfortunate but standard *Log-notation* widespread in bounded arithmetic, by which $n \in \mathrm{Log}$ stands for the formula $\exists x(n = |x|)$ and $n \in \mathrm{LogLog}$ stands for $\exists x(n = ||x||)$.

Intuitively, from the point of view of the theory, numbers in Log are "small" numbers. For a circuit $C : 2^k \rightarrow 2$, where we adopt the set-theoretic custom of identifying $\{0, 1\}$ with the number 2, we can consider the bounded definable set $X_C := \{x < 2^k \mid C(x) = 1\}$, and ask about the task of counting the size of X_C.

There exists a PV-function $\mathrm{Count}(C, y) = |X_C \cap |y||$. This means that if $2^k \in \mathrm{Log}$, then one can do *exact counting* of $|X_C|$ efficiently. We use the notation $\Pr_{x<|y|}[C(x) = 1] \leq z/w$ for the PV-relation $w \cdot \mathrm{Count}(C, y) \leq |y| \cdot z$.

If $2^k \notin \mathrm{Log}$, exact counting becomes problematic. To avoid this, Jeřábek [36, 37] systematically developed the theory APC_1 capturing probabilistic polynomial-time reasoning by means of approximate counting. The theory APC_1 is defined as $\mathsf{PV}_1 + \mathrm{dWPHP}(\mathsf{PV})$ where $\mathrm{dWPHP}(\mathsf{PV})$ stands for the *dual (surjective) pigeonhole principle* for all PV-functions. That is, the set of all formulas

$$x > 0 \rightarrow \exists v < x(|y| + 1). \forall u < x|y|.\ f(u) \neq v, \tag{dWPHP}$$

where f is a PV-function which might involve other parameters not explicitly shown.

We write $C : X \twoheadrightarrow Y$ if C is a surjective mapping from X to Y. Let $X, Y \subseteq 2^n$ be definable sets, and $\epsilon \leq 1$. The size of X is *approximately less than the size of* Y *with error* ϵ, written as $X \preceq_\epsilon Y$, if there exists a circuit C, and $v \neq 0$ such that

$$C : v \times (Y \dot\cup \epsilon 2^n) \twoheadrightarrow v \times X.$$

In this context, the notation $X \approx_\epsilon Y$ stands for $X \preceq_\epsilon Y$ and $Y \preceq_\epsilon X$. As with exact counting, the notation $\Pr_{x<y}[C(x) = 1] \circ_\epsilon z/w$ stands for $w \cdot (X_C \cap y) \circ_\epsilon y \cdot z$, for $\circ \in \{\preceq, \approx\}$. Since a number s is identified with the interval $[0, s)$, $X \preceq_\epsilon s$ means that the size of X is at most s with error ϵ.

The definition of $X \preceq_\epsilon Y$ is an unbounded $\exists\Pi^b_2$ formula even if X and Y are defined by circuits, so it cannot be used freely in bounded induction. This problem can be solved by working in $\mathsf{sHARD}^{\mathsf{A}}$, defined as the relativized theory $\mathsf{S}^1_2(\alpha)$ extended with axioms postulating that $\alpha(x)$ is a truth table of a function on $||x||$ variables hard on average for circuits of size $2^{||x||/4}$. In $\mathsf{sHARD}^{\mathsf{A}}$ there is a $\mathsf{PV}(\alpha)$ function Size approximating the size of any set $X \subseteq 2^n$ defined by a circuit C so that $X \approx_\epsilon \mathrm{Size}(\alpha, C, 2^n, 2^{\epsilon^{-1}})$ for $\epsilon^{-1} \in \mathrm{Log}$ (by combination of [37, Lemma 2.14] and [35, Cor. 3.6]).

The following key definition allows us to express that a function is indeed hard on average.

▶ **Definition 3** ($\mathrm{Hard}^{\mathrm{A}}_\epsilon(f)$, in PV_1 [37]). *Let $f : 2^k \to 2$ be a truth table of a Boolean function with k inputs (with f encoded as a string of 2^k bits, and hence with $k \in \mathrm{LogLog}$). We say that f is* average-case ϵ-hard, *written as $\mathrm{Hard}^A_\epsilon(f)$, if for every circuit C of size at most $2^{\epsilon k}$,*

$$|\{u < 2^k \mid C(u) = f(u)\}| < (1/2 + 2^{-\epsilon k})2^k.$$

Note that $\mathrm{Hard}^A_\epsilon(f)$ is Π^b_1-definable in PV_1.

We write $\mathrm{tt}^{\mathrm{avg}}_\epsilon(f_k, 2^{\epsilon k}) := ||\mathrm{Hard}^{\mathrm{A}}_\epsilon(f)||_m$ for the propositional translation (see Section 2.2.3) of the formula $\mathrm{Hard}^{\mathrm{A}}_\epsilon(f)$ above and an appropriately chosen parameter m depending on k and ϵ. We also consider the polynomial-time function $\mathrm{CorrectFracTT}^\delta(s, n, C, f)$, that checks whether f is a string of length 2^n, C encodes a circuit of size at most s, and finally verifies whether the fraction of accepted inputs is larger than $(1/2 + 2^{-\delta n})2^n$.

The theory APC_1 is strong enough to show that hard-on-average functions do exist.

▶ **Proposition 4** (Jeřábek [35]). *For every rational constant $\epsilon < 1/3$, there exists a constant c such that APC_1 proves that for every $k \in \mathrm{LogLog}$ such that $k \geq c$, there exist a function $f : 2^k \to 2$ that is average-case ϵ-hard.*

The theory S^1_2 can be relativized to $\mathsf{S}^1_2(\alpha)$. This means, in particular, that the language of $\mathsf{S}^1_2(\alpha)$, denoted also $\mathsf{S}^1_2(\alpha)$, contains symbols for all polynomial-time machines with access to the oracle α.

▶ **Definition 5** ($\mathsf{sHARD}^{\mathsf{A}}$ [35]). *The theory sHARD^A is an extension of the theory $\mathsf{S}^1_2(\alpha)$ by the axioms stating*

1. *the number $\alpha(x)$ encodes the truth table of a Boolean function in $||x||$ variables;*
2. *$x \geq c \to \mathrm{Hard}^A_{1/4}(\alpha(x))$, where c is the constant from the previous proposition;*
3. *$||x|| = ||y|| \to \alpha(x) = \alpha(y)$.*

The key technical tool from the framework of approximate counting is the following theorem by Jeřábek.

▶ **Theorem 6** (Jeřábek [37]). *There is a $\mathsf{PV}(\alpha)$-function Size such that sHARD^A proves that if $X \subseteq 2^n$ is definable by a circuit C, then $X \approx_\epsilon \mathrm{Size}(\alpha, C, 2^n, e)$, where $\epsilon = |e|^{-1}$.*

For a circuit $C : 2^n \to 2$, we introduce the notation

$$\Pr_{x<y}[C(x) = 1] \preceq^f_\epsilon \frac{z}{w}$$

to mean $w \cdot \mathrm{Size}(f, C, 2^n, e) \leq y \cdot z$, where $\epsilon = |e|^{-1}$.

2.2.3 Correspondences and propositional translations

While our formalizations are comfortably carried out in the first-order theories presented above, we are able to transfer our results back to propositional logic thanks to the existence of *propositional translations*. Following Krajíček [51], we say that a theory T *corresponds* to a propositional proof system S if (i) T can prove the soundness of S and (ii) every universal consequence $\forall x \varphi(x)$ of T, where φ is quantifier-free, admits polynomial-size proofs in S when grounded into a sequence of propositional formulas. Pudlák alternatively says that S is the *weak system* of the theory T [62]. More formally, for such a universal formula φ, we denote by $||\varphi||_n$ the propositional translation for models of size n. Sometimes we abuse the notation and write $||\varphi||$ dropping the subscript n. We refer the reader to standard texts like those of Krajíček [51] or Cook and Nguyen [20] for formal definitions of the translation.

The key fact for us is that universal theorems of S^1_2 admit short propositional proofs in Extended Frege. More importantly, $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ corresponds to Implicit Extended Frege.

▶ **Theorem 7** (Correspondence of $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ and iEF [45, Thm. 2.1]). *The proof system* iEF *corresponds to* $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$. *That is,*

(i) *the theory* $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ *proves the soundness of* iEF*;*
(ii) *whenever a* $\forall\Pi^b_1$*-sentence* $\forall x \varphi(x)$ *is provable in* $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$*, there are polynomial-size* iEF*-proofs of the sequence of tautologies* $\{||\varphi||_n\}_{n\in\mathbb{N}}$*;*
(iii) *if* $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ *proves the soundness of some propositional system* S*, then* iEF $\geq S$*.*

The translation also works for formulas beyond $\forall\Pi^b_1$ as long as we translate into a quantified propositional system. The definition of the translation is straightforward, and we note that Σ^b_1-consequences of S^1_2 translated as Σ^q_1 formulas admit polynomial-size proofs in G^*_1.

▶ **Theorem 8** (Correspondence of S^1_2 and G^*_1 [47]). *Whenever a* $\forall\Sigma^b_1$*-sentence* $\forall x \exists y \leq t.\varphi(x, y)$ *is provable in* S^1_2*, there are polynomial-size proofs of the sequence of* Σ^q_1*-formulas obtained by the translation,* $\{||\exists x \varphi(x, y)||_n\}_{n\in\mathbb{N}}$*, in* G^*_1*.*

2.3 Interactive proof systems and the sum-sheck protocol

While our focus is on propositional proof systems in the sense of Cook and Reckhow, our work exploits relations to more lax notions of provability. Following Babai [7], an *Merlin-Arthur proof system* or *Merlin-Arthur protocol* for a language $L \subseteq \{0,1\}^*$ is a polynomial-time function S together with some constant c such that the two following properties are satisfied for every $x \in \{0,1\}^*$. Namely,

1. if $x \in L$, then there exists some $\pi \in \{0,1\}^*$ such that $\Pr_{r\in\{0,1\}^{(|x|+|\pi|)^c}}[S(x, \pi, r) = 1] = 1$;
2. if $x \notin L$, then for every $\pi \in \{0,1\}^*$, $\Pr_{r\in\{0,1\}^{(|x|+|\pi|)^c}}[S(x, \pi, r) = 1] < 1/3$.

The first condition formalizes *completeness*, while the second corresponds to *soundness*. The complexity class **MA** contains all languages that admit a polynomially-bounded Merlin-Arthur protocol, meaning that there exists a constant d such that the completeness guarantee is strengthened to proofs $\pi \in \{0,1\}^{|x|^d}$. One should think of MA proof systems as Cook-Reckhow systems where the verifier is randomized and may thus accept some incorrect proofs with small probability.

We recall that, under the standard derandomization assumption that there exists a Boolean function family in **E** that is wort-case hard for subexponential-size circuits, every Merlin-Arthur system derandomizes into a Cook-Reckhow system and, in particular, **MA** = **NP** [56, 34].

Our proofs rely on a particular interactive protocol, the *Sum-Check Protocol* of Lund, Fortnow, Karloff, and Nisan [54] for the language of unsatisfiable 3CNFs. Unlike Merlin-Arthur protocols, this is an interactive protocol running for multiple rounds between a Prover and a Verifier, before the Verifier makes a decision. We now recall the details of the protocol.

The Sum-Check Protocol [54]

The protocol considers a 3CNF $\varphi(x_1, \dots, x_n)$ over m clauses, known to both the Verifier and the Prover.

1. The Prover generates a prime number[6] $p \in (2^{2n^3+n}, 2^{(2n^3+n)^{c_p}}]$ together with a Pratt certificate[7] on the primality of p and sends them to the Verifier, who checks for correctness of the certificate, and aborts if incorrect.
2. The Prover and the Verifier arithmetize φ into a polynomial $P_\varphi(x_1, \dots, x_n)$ of degree at most $3m$ over $\mathbb{F}_p$ in the usual way: a clause like $(x \vee \neg y \vee z)$ is turned into $1-(1-x)y(1-z)$, and one then takes the product of all such arithmetized clauses. In this way, for all $x \in \{0,1\}^n$, $\varphi(x) = 1$ if and only if $P_\varphi(x) = 1$.
3. The Verifier sets $(a_1, \dots, a_n) := (0, \dots, 0)$, $Q_0(a_0) := 0$ and for $i \in \{1, \dots, n\}$, the following interaction is carried out:
   a. Leaving x_i free, the Prover computes the coefficients of the following univariate polynomial over $\mathbb{F}_p$, $Q_i(x_i) := \sum_{x_{i+1}\in\{0,1\}} \cdots \sum_{x_n\in\{0,1\}} P_\varphi(a_1, \dots, a_{i-1}, x_i, x_{i+1}, \dots, x_n)$ and sends the $O(m)$ coefficients of Q_i to the Verifier.
   b. The Verifier checks whether $Q_i(0) + Q_i(1) = Q_{i-1}(a_{i-1})$. If the check fails, the Verifier rejects. Otherwise, it samples a random $a_i \in \mathbb{F}_p$ and sends it to the Prover.
   c. In the final round, instead of sending a_n to the Prover, the Verifier checks whether $P_\varphi(a_1, \dots, a_n) = Q_n(a_n)$ and accepts or rejects based on this.

3 Main result

Our proof exploits the known fact that if $\#\mathbf{P} \subseteq \mathbf{FP}/\mathrm{poly}$, then $\mathbf{coNP} \subseteq \mathbf{MA}$. Indeed, if $\#\mathbf{P}$ has small circuits one can provide polynomial-size circuits that simulate the Prover's movements in the Sum-Check protocol for UNSAT, since one can consider the MA proof system in which Arthur receives from Merlin a circuit claiming to be the circuit that the Prover used to carry out their strategy, and with the aid of randomness, Arthur can execute this on his own and decide based on the outcome of this simulation.

Let us make this formal.

▶ **Definition 9** (The SC proof system). *Let $V(p, u, \varphi, C, r)$ be the polynomial-time function carrying out the simulation of the Sum-Check protocol. Namely, p is intended to be a prime in $(2^{2n^3+n}, 2^{(2n^3+n)^{c_p}}]$, u a Pratt certificate for p, φ a 3CNF over n variables, r a string of random bits, and C a multi-output circuit providing the Prover's responses in the interactions with the Verifier in the Sum-Check protocol.*

The Sum-Check Proof System*, denoted by* SC*, is a Merlin-Arthur proof system for proving 3DNF tautologies. An* SC *proof of φ is a tuple $\langle p, u, C\rangle$ such that p is indeed a prime in the interval above, correctly certified by the Pratt certificate u, and such that $\Pr_{r\in\mathbb{F}_p^n}\left[V(p, u, \neg\varphi, C, r) = 1\right] = 1$.*

[6] The constant c_p in the exponent comes from the formalization of the soundness of the sum-check protocol inside $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ in a recent work of Khaniki [40]; while we do not need such details in our proofs, we leave it here to be faithful to the formalization.

[7] A *Pratt certificate* is a succinct witness for primality checkable in polynomial time [60]. The details are not relevant for our results, but it is important that the Verifier can be convinced of p being a prime.

The following is just a rephrasing of the fact that $\#\mathbf{P} \subseteq \mathbf{FP}/\text{poly}$ implies $\mathbf{coNP} \subseteq \mathbf{MA}$, in terms of the Merlin-Arthur system SC. We omit the proof, which can be be found in standard texts (see e.g. [5, Thm. 8.22]).

▶ **Lemma 10.** *If $\#\mathbf{P} \subseteq \mathbf{FP}/\text{poly}$, then* SC *is polynomially bounded over 3DNF tautologies.*

Our goal is to extend the previous lemma from SC to the concrete and natural Cook-Reckhow system Implicit Extended Frege. The idea again is that iEF (or rather its first-order counterpart, $\mathsf{S}^1_2 + \text{1-EXP}$) can prove the soundness of this system and thus simulate it. We shall then derandomize the SC protocol inside iEF. Fortunately for us, the soundness of the Sum-Check protocol was recently proven by Khaniki in the right theory of bounded arithmetic.

▶ **Theorem 11** (Soundness of the sum-check protocol [39, Thm. 15.3]). *There are constants $c, k \in \mathbb{N}$ such that S^1_2 proves the following sentence: for every n, φ, a, p, u, C, if it holds that (i) φ is a 3CNF in n variables where $n \geq c$, and (ii) $\varphi(a) = 1$ and, (iii) $2^{2n^3+n} < p \leq 2^{(2n^3+n)^{c_p}}$ and, (iv) $n^k \in \mathrm{Log\,Log}$, then*

$$\Pr_{r\in\mathbb{F}_p^n}\big[V(p,u,\varphi,C,r)=1\big] \leq \frac{n\binom{2n}{3}}{p}.$$

We can now formalize the soundness of the SC proof system from Definition 9. The arguments that follow are a concrete application of more sophisticated techniques employed by Khaniki [40, 39], who has studied interactive protocols in the context of defining new jump operators in proof complexity.

▶ **Definition 12** (The Sound_c(SC) formula). *We denote by Sound_c(SC) the following $\forall\Sigma^b_1$ sentence, claiming the soundness of SC: for all φ, a, p, u, C, f, where $|\varphi| > c$, there is a circuit D of size $\leq \lceil |f|^{1/4} \rceil$ such that if*

$$\neg\left(\Pr_{r\in\mathbb{F}_p^n}[V(p,u,\neg\varphi,C,r)=1] \preceq^f_\epsilon \frac{3}{8}\right)$$

holds, then at least one of the following conditions holds:

(i) $|f| \neq |C|^{k_a} + k'_a$ *or,*
(ii) $\mathrm{CorrectFracTT}^{1/4}(\lceil |f|^{1/4} \rceil, ||f||, D, f) = 1$ *or,*
(iii) $p \notin (2^{2n^3+n}, 2^{(2n^3+n)^{c_p}}]$ *or,*
(iv) $\varphi(a) = 1$,

where k_a, k'_a are the constants from Theorem 6 making sure that Size *function works properly, $\epsilon = 1/16$ and n is the number of variables of φ. In the definition of the displayed probability, we assume that $y = p^n$ and that the circuit defining the set of strings accepted by V rejects all $r \geq p^n$.*

Note that even if V accepts with probability 1 on a given input, the approximating probability from Definition 12 can be significantly smaller because of the difference between p^n and the input-size of the circuit in the input of the Size function. Another relevant point is that for each $C, 2^n, e$, the function $\mathrm{Size}(\alpha, C, 2^n, e)$ calls α only once. In fact, it calls $\alpha(x)$ on an input x which depends only on $|C|, n, |e|$. This is needed for the formula Sound_c(SC) to be well-defined.

It now suffices to verify that the encoding of the soundness of SC is indeed provable in $\mathsf{S}^1_2 + \text{1-EXP}$.

▶ **Proposition 13** (Soundness of SC inside $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$)**.** *There is a constant* $c \in \mathbb{N}$ *such that* $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP} \vdash \mathrm{Sound}_c(\mathsf{SC})$.

Proof. Let $c \in \mathbb{N}$ be a big enough constant that can be computed from the rest of the argument and

$$\mathrm{Sound}_c(\mathsf{SC}) := \forall \varphi, a, p, u, C, f \exists D \Phi(\varphi, a, p, u, C, f, D)$$

the soundness formula in Definition 12 above. Let φ be a 3DNF in n variables such that $|\varphi| > c$, and consider a, p, u, C, f. Then the following cases can happen:

(a) If $|f| \neq |C|^{k_a} + k'_a$ or $p \notin (2^{2n^3+n}, 2^{(2n^3+n)^{cp}}]$, then $\Phi(\varphi, a, p, u, C, f, 0)$ is trivially true.

(b) If there is a circuit D of size $\leq \lceil |f|^{1/4} \rceil$ such that $\mathrm{CorrectFracTT}^{1/4}(\lceil |f|^{1/4} \rceil, ||f||, D, f) = 1$, then $\Phi(\varphi, a, p, u, C, f, D)$ is trivially true.

(c) If the previous cases do not happen and moreover

$$\neg \left(\Pr_{r \in \mathbb{F}_p^n} [V(p, u, \neg\varphi, C, r) = 1] \preceq^f_\epsilon \frac{3}{8} \right)$$

holds, then we have that $8 \cdot \mathrm{Size}(f, C^*, 2^m, e) > 3p^n$, where m is the smallest integer such that $2^m \geq p^n$, $\epsilon := |e|^{-1}$ and $C^*(r) := V(p, u, \neg\varphi, C, r)$. By the assumption $\mathrm{Hard}^{\mathrm{A}}_{1/4}(f)$ holds and by the fact that we are over S^1_2 and we can use f as a parameter in polynomial induction for Σ^b_1 formulas, we can do approximate counting using Theorem 6. Hence there is a circuit G and some $v \leq \mathrm{poly}(m\epsilon^{-1}|C^*|)$ such that

$$G : v \times (X_{C^*} \,\dot\cup\, \epsilon 2^m) \twoheadrightarrow v \times \mathrm{Size}(f, C^*, 2^m, e).$$

As we work in $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ and G is surjective, we can find a subset $A \subseteq v \times (X_{C^*} \,\dot\cup\, \epsilon 2^m)$ such that G restricted to A is a one-to-one function from A to $v \times \mathrm{Size}(f, C^*, 2^m, e)$. Now we can apply exact counting (as we have $\mathsf{1\text{-}EXP}$) and show that

$$\mathrm{Size}(f, C^*, 2^m, e) \leq |X_{C^*}| + \epsilon 2^m.$$

By the fact that $8 \cdot \mathrm{Size}(f, C^*, 2^m, e) > 3p^n > 3 \cdot 2^m/2$, we have $2^m/8 < |X_{C^*}|$. Now if $\varphi(a) = 0$, by Theorem 11 we get

$$\Pr_{r \in \mathbb{F}_p^n} \left[V(p, u, \neg\varphi, \pi, r) = 1 \right] \leq \frac{n\binom{2n}{3}}{p}.$$

Note that $|\varphi| > c$ which implies that n is big enough and as $p > 2^{2n^3+n}$ we get that $n\binom{2n}{3}/p \leq 1/8$, which implies

$$\Pr_{r \in \mathbb{F}_p^n} \left[V(p, u, \neg\varphi, \pi, r) = 1 \right] \leq \frac{1}{8}.$$

As C^* rejects all $r \geq p^n$, this implies that $|X_{C^*}| \leq 2^m/8$ which leads to a contradiction, so $\varphi(a) = 1$. ◀

The main technical issue now is that $\mathrm{Sound}_c(\mathsf{SC})$ is a $\forall\Sigma^b_1$ sentence that does not translate into a propositional formula that iEF can reason about. Instead, we shall work on a quantified propositional system. For this to make sense we need to know the quantified propositional proof system associated with $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$.

We invoke the following known **TFNP** characterization of the Σ^b_1 consequences of $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$, which identifies a "complete" Σ^b_1 sentence Ψ such that any other Σ^b_1 consequence of $\mathsf{S}^1_2 + \mathsf{1\text{-}EXP}$ reduces to it in G^*_1.

▶ **Theorem 14** ([42, 41, 46, 9]). *There is a $\forall\Sigma^b_1$ sentence $\Psi \coloneqq \forall x \exists y \psi(x,y)$ (the bound on y is implicit in ψ) such that the following statements are true:*

(i) $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP} \vdash \forall x \exists y \psi(x,y)$;

(ii) *for any $\forall\Sigma^b_1$ sentence $\forall x\exists y \alpha(x,y)$ such that $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP} \vdash \forall x\exists y\alpha(x,y)$, there are* PV *functions f and g such that $\mathsf{S}^1_2 \vdash \forall x, y(\psi(f(x),y) \to \alpha(x, g(x,y)))$.*

In what follows, we shall work with Gentzen's system G extended with the propositional translation of the sentence Ψ in the theorem above. We denote this system by $\mathsf{G}_{\mathsf{EXP}} \coloneqq \mathsf{G}^*_1 + ||\Psi||$ and use the following key properties about it. The full version of the paper contains explicit proofs of each of these items, but we shall omit these here.

▶ **Corollary 15.** *The following statements about $\mathsf{G}_{\mathsf{EXP}}$ hold:*

(i) $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP} \vdash \Sigma^q_1\text{-}\mathrm{Ref}(\mathsf{G}_{\mathsf{EXP}})$*, i.e. the reflection principle for $\mathsf{G}_{\mathsf{EXP}}$ and Σ^q_1 formulas is provable in $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP}$;*

(ii) *for every $\forall\Sigma^b_1$-sentence $\forall x\exists y\alpha(x,y)$, if $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP} \vdash \forall x\exists y\alpha(x,y)$, then there are polynomial-size $\mathsf{G}_{\mathsf{EXP}}$-proofs of the sequence of Σ^q_1-tautologies $\{||\exists y \alpha(y)||_n\}_{n\in\mathbb{N}}$;*

(iii) *if $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP}$ proves the soundness of a propositional proof system S, then $\mathsf{G}_{\mathsf{EXP}} \geq S$.*

Let us observe that $\mathsf{G}_{\mathsf{EXP}}$ is in fact equivalent to iEF.

▶ **Lemma 16.** *The proof systems $\mathsf{iEF}, \mathsf{EF} + \mathrm{Ref}_{\mathsf{iEF}}$ and $\mathsf{G}_{\mathsf{EXP}}$ are polynomially equivalent over propositional tautologies.*

Proof. By item (iii) of Corollary 15 and item (iii) Theorem 7, iEF and $\mathsf{G}_{\mathsf{EXP}}$ polynomially simulate each other. As mentioned in Section 2.1.1, $\mathsf{EF} + \mathrm{Ref}_{\mathsf{iEF}} \geq \mathsf{iEF}$. It is also easy to see that $\mathsf{S}^1_2 + \text{1-}\mathsf{EXP}$ proves the soundness of $\mathsf{EF} + \mathrm{Ref}_{\mathsf{iEF}}$, which by item (iii) of Theorem 7 gives us $\mathsf{iEF} \geq \mathsf{EF} + \mathrm{Ref}_{\mathsf{iEF}}$. ◀

We are now ready to define the extension of iEF for which our main theorem holds. Recall that the propositional formulas $\mathrm{tt}^{\mathrm{avg}}_{1/4}(h_n, 2^{n/4})$ were defined in Section 2.2.2 and state the average-case hardness of a Boolean function h_n represented as a truth table.

▶ **Definition 17** (The systems $\mathsf{iEF}^{\mathrm{tt}}$). *Let $h = \{h_n\}_{n\in\mathbb{N}}$ be some family of Boolean functions, and let $n_0 \in \mathbb{N}$. We denote by $\mathsf{iEF}^{\mathrm{tt}(h,n_0)} \coloneqq \mathsf{G}_{\mathsf{EXP}} + \{\mathrm{tt}^{avg}_{1/4}(h_n, 2^{n/4})\}_{n\geq n_0}$ the system that extends $\mathsf{G}_{\mathsf{EXP}}$ by the axioms claiming the hardness of h_n, for $n \geq n_0$.*

Note that $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ is a family of proof systems, parameterized by a Boolean function family h and some threshold parameter n_0. Observe that depending on the choice of h and n_0, the system $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ may not be a Cook-Reckhow system: if h is not a hard function, or n_0 is not large enough, we will be adding axioms which are not tautologies; and even if h is hard and n_0 is large enough, the system may require advice to verify the proofs. As we shall see, however, these degenerate instantiations of $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ are not a problem.

What is more important, the systems $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$, regardless of their consistency, always simulate SC.

▶ **Lemma 18.** *Let h be family of Boolean functions and let $n_0 \in \mathbb{N}$. The system $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ polynomially simulates* SC *over 3DNF tautologies.*

Proof. If the system $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ is unsound because the added axioms are not tautologies, then the system is polynomially bounded and simulates every other proof system. So suppose the added axioms are indeed tautologies, meaning that the function h is indeed hard on average.

Let φ_1 be a 3DNF in n_1 variables and $\langle p_1, u_1, C_1\rangle$ be a SC-proof of φ_1. This means

$$2^{2n_1^3+n_1} < p_1 \le 2^{(2n_1^3+n_1)^{c_p}} \wedge \Pr_{r\in\mathbb{F}_{p_1}^{n_1}}\left[V(p_1,u_1,\neg\varphi_1,C_1,r)=1\right]=1.$$

Note that by Theorem 11 and Corollary 15, there are PV functions l, g such that

$$\mathsf{S}^1_2 \vdash \forall\varphi,a,p,u,C,f\,(\psi(l(\varphi,a,p,u,C,f),y)\to\Phi(\varphi,a,p,u,C,f,g(\varphi,a,p,u,C,f,y)))\,,$$

where $\mathrm{Sound}_c(\mathsf{SC}) \coloneqq \forall\varphi,a,p,u,C,f\exists D\Phi(\varphi,a,p,u,C,f,D)$. Let $s \coloneqq |\langle p,u,C\rangle|$. Then by Theorem 8 there is a $s^{O(1)}$-size G_1^*-proof of

$$||\forall\varphi,a,p,u,C,f\,(\psi(l(\varphi,a,p,u,C,f),y)\to\Phi(\varphi,a,p,u,C,f,g(\varphi,a,p,u,C,f,y)))\,||_{s'},$$

where $s' \coloneqq \mathrm{poly}(s)$. Let us rewrite the previous quantified propositional formula as $||\Psi'||\to||\Phi'||$ with the right range of parameters such that p_1,u_1,φ_1,C_1 are substituted in the formula in their corresponding places. Now we take the substitution instance $\mathrm{tt}^{\mathrm{avg}}_{1/4}(h_{n'},2^{n'/4})$ where $|h_{n'}| \coloneqq |C_1|^{k_a}+k'_a$ and we substitute $h_{n'}$ to the variables corresponding to f and therefore the disjunct which corresponds to CorrectFracTT disappears from $||\Phi'||$ when we apply the rules of G_1^*. Moreover, it is not hard to verify that after the substitutions every other disjunct which corresponds to subformulas of $\mathrm{Sound}_c(\mathsf{SC})$ from Definition 12 disappears except φ_1. So what we have is G_1^*-proof of $||\Psi''||(\bar{x},\bar{y})\to\varphi_1(\bar{x})$ ($\bar{x}$ and $\bar{y}$ are disjoint variables) where $||\Psi''||$ is a substitution instance of $||\Psi'||$. Since we are working in $\mathsf{G}_{\mathsf{EXP}}$, we have the substitution instance $\exists\bar{y}||\Psi''||(\bar{x},\bar{y})$ and therefore using the rules of G_1^* we get a short $\mathsf{G}_{\mathsf{EXP}}$-proof of $\varphi_1(\bar{x})$. ◀

Our main theorem now easily follows.

▶ **Theorem 19** (Main theorem). *Let h be a family of Boolean functions and let $n_0\in\mathbb{N}$. If the system $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ is not polynomially bounded, then $\#\mathbf{P}\not\subseteq\mathbf{FP}/\mathrm{poly}$.*

Proof. By Lemma 18 above, for every choice of h and n_0, the system $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ polynomially simulates SC, so if $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ is not polynomially bounded, then SC is not either. Then, by the contrapositive of Lemma 10, $\#\mathbf{P}\not\subseteq\mathbf{FP}/\mathrm{poly}$. ◀

As discussed, depending on the choice of h and n_0, the system $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ may not be sound and thus possibly not Cook-Reckhow. However, for any fixed choice of a uniform candidate hard function, the system is concrete and exhibits the desired connection that proof complexity lower bounds for it imply strong circuit lower bounds. In particular, if there exist functions in $\mathbf{NE}\cap\mathbf{coNE}$ average-case hard for subexponential-size circuits, then we recover the version of the theorem presented in the introduction (Theorem 1).

We note that there is the possibility that iEF, given its strength, already proves such strong circuit lower bounds for some Boolean function. It is thus worth to mention the following corollary.

▶ **Corollary 20.** *Suppose there exists a sequence of Boolean functions $\{h_n\}_{n\in\mathbb{N}}$ for which iEF has polynomial-size proofs of the formula family $\{\mathrm{tt}^{\mathrm{avg}}_{1/4}(h_n,2^{n/4})\}_{n\ge n_0}$ for some sufficiently large $n_0\in\mathbb{N}$. If iEF is not polynomially bounded, then $\#\mathbf{P}\not\subseteq\mathbf{FP}/\mathrm{poly}$.*

Proof. If there is such a function h and threshold n_0, then $\mathsf{iEF}^{\mathrm{tt}(h,n_0)}$ is polynomially equivalent to iEF itself, so by Theorem 19 the corollary follows. ◀

References

1 Miklós Ajtai. Σ_1^1-formulae on finite structures. *Annals of Pure and Applied Logic*, 24(1):1–48, 1983.

2 Miklós Ajtai. The complexity of the pigeonhole principle. *Combinatorica*, 14:417–433, 1994.

3 Noga Alon and Ravi B Boppana. The monotone circuit complexity of Boolean functions. *Combinatorica*, 7:1–22, 1987.

4 Aleksandr Egorovich Andreev. A method for obtaining lower bounds on the complexity of individual monotone functions. In *Doklady Akademii Nauk*, volume 282(5), pages 1033–1037. Russian Academy of Sciences, 1985.

5 Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

6 Noel Arteche, Gaia Carenini, and Matthew Gray. Quantum automating $\mathbf{TC}^0$-Frege is LWE-hard, 2024. arXiv:2402.10351.

7 László Babai. Trading group theory for randomness. In *Proceedings of the seventeenth annual ACM symposium on Theory of computing*, pages 421–429, 1985.

8 Paul Beame, Russell Impagliazzo, Jan Krajíček, Toniann Pitassi, Pavel Pudlák, and Alan Woods. Exponential lower bounds for the pigeonhole principle. In *Proceedings of the Twenty-Fourth Annual ACM Symposium on Theory of Computing*, pages 200–220, 1992.

9 Arnold Beckmann and Sam Buss. The **NP** search problems of Frege and Extended Frege proofs. *ACM Transactions on Computational Logic (TOCL)*, 18(2):1–19, 2017.

10 Stephen Bellantoni, Toniann Pitassi, and Alasdair Urquhart. Approximation and small-depth Frege proofs. *SIAM Journal on Computing*, 21(6):1161–1179, 1992.

11 Olaf Beyersdorff, Ilario Bonacina, Leroy Chew, and Jan Pich. Frege systems for quantified boolean logic. *Journal of the ACM (JACM)*, 67(2):1–36, 2020.

12 Maria Luisa Bonet, Carlos Domingo, Ricard Gavalda, Alexis Maciel, and Toniann Pitassi. Non-automatizability of bounded-depth Frege proofs. *computational complexity*, 13:47–68, 2004.

13 Maria Luisa Bonet, Toniann Pitassi, and Ran Raz. On interpolation and automatization for Frege systems. *SIAM Journal on Computing*, 29(6):1939–1967, 2000.

14 Peter Bürgisser. Completeness and reduction in algebraic complexity theory. *Algorithms and Computation in Mathematics*, 2000.

15 Samuel R Buss. *Bounded arithmetic*. Princeton University, 1985.

16 Samuel R Buss. Relating the bounded arithmetic and polynomial time hierarchies. *Annals of Pure and Applied Logic*, 75(1-2):67–77, 1995.

17 A Cobham. The intrinsic computational difficulty of functions. In *Proc. 1964 Congress for Logic, Methodology, and the Philosophy of Science*, pages 24–30. North-Holland, 1964.

18 Stephen Cook. Relating the provable collapse of $\mathbf{P}$ to $\mathbf{NC}^1$ and the power of logical theories. In *Proof Complexity and Feasible Arithmetics*, pages 73–91, 1996.

19 Stephen Cook and Jan Krajíček. Consequences of the provability of $\mathbf{NP} \subseteq \mathbf{P}/\text{poly}$. *The Journal of Symbolic Logic*, 72(4):1353–1371, 2007.

20 Stephen Cook and Phuong Nguyen. *Logical Foundations of Proof Complexity*. Cambridge University Press, 2010.

21 Stephen A. Cook. Feasibly constructive proofs and the propositional calculus. In *Proceedings of the Seventh Annual ACM Symposium on Theory of Computing*, pages 83–97, 1975.

22 Stephen A. Cook and Robert A. Reckhow. The relative efficiency of propositional proof systems. *Logic, Automata, and Computational Complexity*, 1979.

23 Ben Davis and Robert Robere. Colourful TFNP and Propositional Proofs. In *38th Computational Complexity Conference (CCC 2023)*, volume 264 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 36:1–36:21, 2023. doi:10.4230/LIPIcs.CCC.2023.36.

24 Susanna De Rezende, Or Meir, Jakob Nordström, Toniann Pitassi, Robert Robere, and Marc Vinyals. Lifting with simple gadgets and applications to circuit and proof complexity. In *61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 24–30, 2020.

25 Susanna F de Rezende, Mika Göös, and Robert Robere. Proofs, circuits, and communication. *ACM SIGACT News*, 53(1):59–82, 2022.

26 Merrick Furst, James B Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, 1984.

27 Ankit Garg, Mika Göös, Pritish Kamath, and Dmitry Sokolov. Monotone circuit lower bounds from resolution. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 902–911, 2018.

28 Azza Gaysin. Proof complexity of CSP. *arXiv preprint*, 2022. `arXiv:2201.00913`.

29 Azza Gaysin. Proof complexity of universal algebra in a CSP dichotomy proof. *arXiv preprint*, 2024. `arXiv:2403.06704`.

30 Joshua A Grochow. Polynomial identity testing and the Ideal proof system: PIT is in **NP** if and only if IPS can be p-simulated by a Cook-Reckhow proof system. *arXiv preprint*, 2023. `arXiv:2306.02184`.

31 Joshua A Grochow and Toniann Pitassi. Circuit complexity, proof complexity, and polynomial identity testing: The ideal proof system. *Journal of the ACM (JACM)*, 65(6):1–59, 2018.

32 Tuomas Hakoniemi. Feasible interpolation for Polynomial Calculus and Sums-of-Squares. In *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, 2020.

33 John Håstad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, pages 6–20, 1986.

34 Russell Impagliazzo and Avi Wigderson. **P** = **BPP** if **E** requires exponential circuits: Derandomizing the XOR lemma. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of computing*, pages 220–229, 1997.

35 Emil Jeřábek. Dual weak pigeonhole principle, Boolean complexity, and derandomization. *Annals of Pure and Applied Logic*, 129(1-3):1–37, 2004.

36 Emil Jeřábek. *Weak pigeonhole principle, and randomized computation.* PhD thesis, Faculty of Mathematics and Physics, Charles University, Prague, 2005.

37 Emil Jeřábek. Approximate counting in bounded arithmetic. *The Journal of Symbolic Logic*, 72(3):959–993, 2007.

38 Valentine Kabanets and Russell Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. *Computational Complexity*, 13(1/2):1–46, 2004.

39 Erfan Khaniki. *(Im)possibilty results in Proof Complexity and Arithmetic.* PhD thesis, Faculty of Mathematics and Physics, Charles University, Prague, 2023. URL: `https://dspace.cuni.cz/handle/20.500.11956/187614`.

40 Erfan Khaniki. Jump operators, interactive proofs, and proof complexity generators, 2023. Unpublished preprint.

41 Leszek Aleksander Kołodziejczyk, Phuong Nguyen, and Neil Thapen. The provably total **NP** search problems of weak second order bounded arithmetic. *Annals of Pure and Applied Logic*, 162(6):419–446, 2011.

42 Jan Krajíček. Exponentiation and second-order bounded arithmetic. *Annals of Pure and Applied Logic*, 48(3):261–276, 1990.

43 Jan Krajíček. Lower bounds to the size of constant-depth propositional proofs. *The Journal of Symbolic Logic*, 59(1):73–86, 1994.

44 Jan Krajíček. Diagonalization in proof complexity. *Fundamenta Mathematicae*, 182:181–192, 2004.

45 Jan Krajíček. Implicit proofs. *The Journal of Symbolic Logic*, 69(2):387–397, 2004.

46 Jan Krajíček. Consistency of circuit evaluation, Extended Resolution and total **NP** search problems. In *Forum of Mathematics, Sigma*, volume 4, page e15. Cambridge University Press, 2016.

47 Jan Krajíček and Pavel Pudlák. Quantified propositional calculi and fragments of bounded arithmetic. *Zeitschrift für mathematische Logik und Grundlagen der Mathematik*, 36(1):29–46, 1990.

48 Jan Krajíček, Pavel Pudlák, and Gaisi Takeuti. Bounded arithmetic and the polynomial hierarchy. *Annals of Pure and Applied Logic*, 52(1-2), 1991.

49 Jan Krajíček, Pavel Pudlák, and Alan Woods. An exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle. *Random Structures & Algorithms*, 7(1):15–39, 1995.

50 Jan Krajíček. Interpolation theorems, lower bounds for proof systems, and independence results for bounded arithmetic. *The Journal of Symbolic Logic*, 62(2):457–486, 1997.

51 Jan Krajíček. *Proof Complexity*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2019. doi:10.1017/9781108242066.

52 Jan Krajíček and Pavel Pudlák. Some consequences of cryptographical conjectures for S_2^1 and EF . *Information and Computation*, 140(1):82–94, 1998.

53 Shachar Lovett, Raghu Meka, Ian Mertz, Toniann Pitassi, and Jiapeng Zhang. Lifting with sunflowers. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*, 2022.

54 Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. Algebraic methods for interactive proof systems. *Journal of the ACM (JACM)*, 39(4):859–868, 1992.

55 Moritz Müller and Ján Pich. Feasibly constructive proofs of succinct weak circuit lower bounds. *Annals of Pure and Applied Logic*, 171(2):102735, 2020.

56 Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, 1994.

57 Ján Pich. Logical strength of complexity theory and a formalization of the PCP theorem in bounded arithmetic. *Logical Methods in Computer Science*, 11, 2015.

58 Jan Pich and Rahul Santhanam. Towards $\mathbf{P} \neq \mathbf{NP}$ from Extended Frege lower bounds. *arXiv preprint*, 2023. arXiv:2312.08163.

59 Toniann Pitassi, Paul Beame, and Russell Impagliazzo. Exponential lower bounds for the pigeonhole principle. *Computational complexity*, 3:97–140, 1993.

60 Vaughan R Pratt. Every prime has a succinct certificate. *SIAM Journal on Computing*, 4(3):214–220, 1975.

61 Pavel Pudlák. Lower bounds for resolution and cutting planes proofs and monotone computations. *The Journal of Symbolic Logic*, 62(3):981–998, 1997.

62 Pavel Pudlák. Reflection principles, propositional proof systems, and theories, 2020. arXiv:2007.14835.

63 Ran Raz and Pierre McKenzie. Separation of the monotone $\mathbf{NC}$ hierarchy. In *Proceedings 38th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 234–243. IEEE, 1997.

64 Alexander Razborov. Lower bounds on the monotone complexity of some Boolean function. In *Soviet Math. Dokl.*, volume 31, pages 354–357, 1985.

65 Alexander Razborov. Unprovability of lower bounds on circuit size in certain fragments of bounded arithmetic. *Izvestiya: mathematics*, 59(1):205, 1995.

66 Alexander A. Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Mathematical Notes of the Academy of Sciences of the USSR*, 41(4):333–338, 1987.

67 Alexander A Razborov. Bounded arithmetic and lower bounds in boolean complexity. In *Feasible Mathematics II*, pages 344–386. Springer, 1995.

68 Roman Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *Proceedings of the Nineteenth Annual ACM Symposium on the Theory of Computing*, pages 77–82, 1987.