2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)

Why are Proof Complexity Lower Bounds Hard?

Ján Pich
Department of Computer Science
University of Oxford
Oxford, United Kingdom
janpich@yahoo.com

Rahul Santhanam
Department of Computer Science
University of Oxford
Oxford, United Kingdom
rahul.santhanam@cs.ox.ac.uk

***Abstract*—We formalize and study the question of whether there are inherent difficulties to showing lower bounds on propositional proof complexity.**

We establish the following unconditional result: Propositional proof systems cannot efficiently show that truth tables of random Boolean functions lack polynomial size non-uniform proofs of hardness. Assuming a conjecture of Rudich, propositional proof systems also cannot efficiently show that random k-CNFs of linear density lack polynomial size non-uniform proofs of unsatisfiability. Since the statements in question assert the average-case hardness of standard NP problems (MCSP and 3-SAT respectively) against co-nondeterministic circuits for natural distributions, one interpretation of our result is that propositional proof systems are inherently incapable of efficiently proving strong complexity lower bounds in our formalization. Another interpretation is that an analogue of the Razborov-Rudich 'natural proofs' barrier holds in proof complexity: under reasonable hardness assumptions, there are natural distributions on hard tautologies for which it is infeasible to show proof complexity lower bounds for strong enough proof systems.

For the specific case of the Extended Frege (EF) propositional proof system, we show that at least one of the following cases holds: (1) EF has no efficient proofs of superpolynomial circuit lower bound tautologies for any Boolean function or (2) There is an explicit family of tautologies of each length such that under reasonable hardness assumptions, most tautologies are hard but no propositional proof system can efficiently establish hardness for most tautologies in the family. Thus, under reasonable hardness assumptions, either the Circuit Lower Bounds program toward complexity separations cannot be implemented in EF, or there are inherent obstacles to implementing the Cook-Reckhow program for EF.

***Keywords*-Computational complexity, proof complexity, circuit lower bounds, barriers**

I. INTRODUCTION

A. Motivation

Complexity theory is full of questions that are easy to state but hard to answer. The most famous of these is the P vs NP problem [14], but there are numerous others such as the NP vs coNP problem, the PSPACE vs P problem, and the BPP vs P problem. In all of these cases, despite decades of effort, very little progress has been made. Is this because complexity theory is a young field, and we have not yet had the time to develop a deep understanding of computation and its limits? Or are these problems fundamentally intractable in some sense?

Since complexity theory is the theory of intractability, it is natural to apply it to the seeming intractability of complexity-theoretic questions themselves. Since the early days of complexity theory, progress on complexity lower bounds has gone hand-in-hand with the formalization of various sorts of *barriers* to progress. In the 70s, analogies between complexity theory and recursion theory were developed, with various concepts and techniques from recursion theory being adapted to the resource-bounded setting. However, Baker, Gill & Solovay [7] observed in the late 70s that popular machine simulation and diagonalization techniques from recursion theory *relativized*, i.e., continued to work even when machines were given access to an arbitrary oracle. By giving an oracle relative to which $\mathsf{P} = \mathsf{NP}$ and another oracle relative to which $\mathsf{P} \neq \mathsf{NP}$ [7], they proved that no relativizing techniques could solve the NP vs P question. It seemed that techniques of a fundamentally different sort were required.

After this barrier result, attention shifted to a more *finitistic* setting. Rather than considering uniform machines that work for all inputs, Boolean circuits corresponding to finite functions became the object of study. It is well-known that P can be simulated by polynomial-size Boolean circuits, and therefore super-polynomial lower bounds on Boolean circuit size imply lower bounds against P.

Perhaps the hope was that circuits are simpler and more 'combinatorial' objects, and therefore more amenable to lower bounds via combinatorial and algebraic techniques. Indeed, initial results were promising. In a series of influential works [2], [21], [24], [48] applying the technique of random restrictions, super-polynomial lower bounds were shown for the Parity function against constant-depth circuits. Razborov [40] and Smolensky [47] developed the polynomial approximation technique to give lower bounds against constant-depth circuits with prime modular gates. Razborov [39] used the method of approximations to show that the Clique problem required super-polynomial size monotone circuits. This sequence of works introduced several new lower bound techniques, and it seemed that steady progress was being made toward the goal of

DOI 10.1109/FOCS.2019.00080

separating NP and P via a *Circuit Lower Bounds Program*.

Circuit Lower Bounds Program: Separate NP and P by proving super-polynomial lower bounds for functions in NP against increasingly expressive classes of Boolean circuits.

Unfortunately, the Circuit Lower Bounds Program stalled in the early 90s. Even now, almost 30 years later, we still don't know if there are explicit functions that require super-polynomial depth-two threshold circuits, or constant-depth circuits with Mod 6 gates. However, even if no breakthrough lower bounds were proved for a while, our understanding of barriers to lower bounds did evolve, due mainly to work of Razborov [41], [42] and Razborov & Rudich [45]. Razborov & Rudich [45] developed the concept of natural proofs, which captures known circuit lower bounds proved via combinatorial and algebraic techniques, and showed that under standard cryptographic assumptions, natural proofs cannot prove super-polynomial lower bounds for general Boolean circuits.

The natural proofs barrier has not proven to be as damaging to the finitistic approach to complexity as the relativization barrier was to the machine-based approach. Circuits are still a popular model, and combinatorial and algebraic techniques are still heavily used. Various methods to evade the natural proofs barrier have been proposed, though none have yet resulted in any breakthrough lower bounds for explicit functions.

The meta-mathematics of circuit lower bounds is one of our motivations for the results in this paper, but the main motivation comes from a different approach to complexity theory, via *proof complexity*. The NP vs P problem essentially asks if short proofs for mathematical theorems are easy to find. Thus logic and proof are inherent to the problem. These aspects are emphasized not just in Steve Cook's seminal paper [14] stating the problem, but already in Godel's famous letter to von Neumann in the early 1950s.

Cook and Reckhow [18] defined the notion of a *propositional proof system*: a polynomial-time computable surjective function mapping strings to tautologies. The requirement that the range of the function is the set of tautologies models the soundness of the proof system; the surjectiveness of the function models completeness; the polynomial-time computability of the function models efficient verifiability of proofs. A *proof* of a tautology ϕ with respect to a propositional proof system P is simply a string x such that $P(x) = \phi$. A natural complexity question then is how the minimum proof size grows as a function of the length of the tautology, with respect to a given propositional proof system P.

Cook and Reckhow [18] show that $\mathsf{NP} = \mathsf{coNP}$ iff there is a propositional proof system P that is *polynomially bounded*, i.e., all tautologies have polynomial-size proofs in P. Equivalently, $\mathsf{NP} \neq \mathsf{coNP}$ iff there is a *hard* sequence of tautologies for every propositional proof system P.

Just as the Circuit Lower Bounds Program aims to make progress on the NP vs P question, the work of Cook and Reckhow suggests an approach to making progress on the NP vs coNP question.

Cook-Reckhow Program: Separate NP and coNP by proving super-polynomial lower bounds on the proof size of tautologies in increasingly powerful proof systems P.

We note a couple of differences between the circuit complexity lower bound setting and the proof complexity lower bound setting. In the circuit complexity setting, it is easy to see by a counting argument that *most* Boolean functions do not have small circuits. The challenge is to show a circuit lower bound for an *explicit* Boolean function, i.e., a function in NP. In the proof complexity setting, we do not care about explicitness - we are satisfied with lower bounds for *any* sequence of tautologies. In this setting, counting arguments do not work to get non-constructive lower bounds, since the space of small proofs is comparable in size to the space of tautologies.

Despite these differences, progress on the Cook-Reckhow Program has been partly inspired by analogies with the Circuit Lower Bounds Program. Haken [23] showed super-polynomial lower bounds on the size of proofs of the Pigeonhole Principle in the Resolution proof system. Ajtai [3] obtained a significant extension of this result by showing that the Pigeonhole Principle is also super-polynomially hard for Bounded-Depth Frege - a restricted version of the standard Frege system where the lines are bounded-depth circuits. Ajtai's super-polynomial lower bound was strengthened to an exponential lower bound by [9]. Both [3] and [9] use strong versions of the random restriction technique used to prove lower bounds on bounded-depth circuits.

There has been significant effort devoted to proving lower bounds for constant-depth Frege proofs with modular counting gates - the proof-theoretic analogue of constant-depth circuits with modular counting gates - but without any success. Thus the Cook-Reckhow program is also stalled. What is even more unsatisfactory, though, is that unlike the Circuit Lower Bounds program, we do not have a good explanation for *why* progress is stalled. There is no known analogue of the natural proofs barrier for proof complexity, and indeed this is one of the main open questions asked in the survey of Beame and Pitassi [10] from the late 90s.

Our main motivation in this paper is to formalize and study the question of whether there are inherent obstacles to proving proof complexity lower bounds for strong propositional proof systems.

There are several reasons why this question is interesting. First, as described above, it is motivated by the difficulty of making progress on the Cook-Reckhow program.

Second, propositional proof systems correspond naturally to certain classes of algorithms that are used in practice for solving the Satisfiability problem. For example, the Tree Resolution proof system corresponds to branching algorithms. Showing lower bounds for a proof system P translates to constructing hard instances for the corresponding class $A(P)$ of algorithms, which is an interesting question in its own right. The question of whether proof complexity lower bounds are hard thus connects to the question of whether provably hard instances exist for various classes of algorithms.

Third, propositional proof systems can be interpreted not only *algorithmically*, as in the previous para, but also *meta-mathematically*, as a setting within which barriers to progress in complexity theory can be formalized and understood. This duality is also important in this paper, and enables us to use our ideas to show the unprovability of certain strong complexity hypotheses within a finitistic context. In the terminology of Aaronson [1], these are new 'second-generation' independence results. First generation independence results are about unprovability in logical theories such as $\mathsf{ZFC}, \mathsf{PA}$ and restrictions thereof. In this paper, on the other hand, we focus on finitistic results in the context of propositional proof complexity, where proofs always *exist* for any tautology, but the question is whether they are of a reasonable size.

We now proceed to discuss our model, and the hypotheses we consider.

B. The Setting

We would like to study the question of whether proof complexity lower bounds are inherently 'hard'. The most natural formalization of 'hardness' we can imagine in this context is that tautologies corresponding to proof complexity lower bounds are themselves hard to prove, even in strong propositional proof systems.

This formalization is inspired by the analogy with circuit complexity. In his work on complexity barriers, Razborov [41], [42] considered the proof complexity of the *circuit lower bound formulas*, which are propositional formulas $\mathtt{tt}(f_n, s)$ expressing that a Boolean function f_n on n bits given by its truth table does not have Boolean circuits of size s, for some parameter s. Intuitively the reason why this statement can be expressed by propositional formulas (which are tautologies when f is indeed hard) is that it is a universal statement, saying that *no* circuit of size at most s can compute the function corresponding to the given truth table. This can be encoded by a DNF of size $O(2^n s^3)$ where the propositional variables correspond to the bits of the circuit[1]. The main result of [41], [42] is that under standard cryptographic assumptions, no propositional proof system satisfying the 'feasible interpolation' property can efficiently prove circuit lower bound formulas corresponding to $s = n^{\omega(1)}$ for *any* Boolean function. The advantage of this result is that it applies to any Boolean function; the weakness is that it only holds for propositional proof systems with feasible interpolation, and systems such as Extended Frege and Frege are known not to have feasible interpolation under cryptographic assumptions [12], [33].

We can define *proof lower bound formulas* for a given propositional proof system R in an analogous way. Given a formula ϕ and a parameter s, the corresponding R-proof lower bound formula $\mathsf{lb}_R(\phi, s)$ states that there is no R-proof of ϕ of size s. Just as with circuit lower bound formulas, this is a universal statement: there *no* proof for ϕ of size at most s in R. This can be encoded by a DNF of size $\mathrm{poly}(|\phi|, s)$, where the propositional variables correspond to the bits of the proof.

Now we can ask about the proof complexity of R-proof lower bound formulas. In this paper, we typically adopt a high standard for hardness - we require that the proof complexity of R-proof lower bound formulas is large for *every* propositional proof system S arguing about these lower bound formulas.

We remark that *upper bounds* on the proof complexity of R-proof lower bound formulas for certain proof systems R are implicit in previous work. A result of Cook and Pitassi [17] implies that Resolution-proof lower bound formulas corresponding to the Pigeonhole principle can be efficiently proved within Extended Frege. Bellantoni, Pitassi and Urquhart [11] give constructive proofs of constant-depth Frege lower bounds that appear to be efficiently implementable within Extended Frege. Thus some of the strongest proof complexity lower bounds we have at present seem also to be provable within standard propositional proof systems.

In contrast to these upper bound results, our emphasis is on lower bound and impossibility results, which give evidence that certain propositional proof systems are inherently hard to analyze.

Let us say a collection of R-proof complexity lower bounds is *feasibly provable* if the corresponding R-proof lower bound formulas all have short S-proofs in some propositional proof system S. Armed with this notion, we can define a feasible version of the Cook-Reckhow program.

Feasible Cook-Reckhow Program: For every propositional proof system R, show that for each $k > 0$ there is a sequence of tautologies $\{\phi_n\}$ and a propositional proof system S, $\mathsf{lb}_R(\phi_n, |\phi_n|^k)$ has polynomial-size S-proofs.

Note that our requirement here is fairly mild: we allow the propositional proof system S to depend on the propositional proof system R for which lower bounds are being shown. We even allow S to depend on the constant k corresponding

---

[1] Note that the encoding is exponentially large. More succinct encodings have been considered in recent work of Müller and Pich [36]

to the lower bound being shown. Also note that we *do not* require that S has efficient proofs that the formulas ϕ_n are tautologies. All we ask is that these formulas are indeed tautologies, and that S can show that they don't have efficient R-proofs.

Since we are interested in impossibility results, the mildness of our proof-theoretic requirements is an advantage - it makes our results stronger. However, we do insist on feasibility. Intuitively, in a finite resource-bounded world, it seems unreasonable to allow the prover unlimited resources.

The results of [17] and [11] mentioned above constitute partial progress toward the Feasible Cook-Reckhow program. If R is Resolution or Constant-Depth Frege, the feasibility condition is satisfied when S is Extended Frege and the formulas $\{\phi_n\}$ are the pigeon-hole principles.

The Feasible Cook-Reckhow Program can be interpreted as a feasible approach to proving $\mathsf{NP} \neq \mathsf{coNP}$. We wish to get evidence for the difficulty or impossibility of implementing this program. However, the mildness of our proof-theoretic requirements is an issue here. Consider any efficiently computable sequence $\{\phi_n\}$ of tautologies of increasing length, such that the sequence $\{\phi_n\}$ requires R-proofs of super-polynomial size. One way to define a proof system S in which these lower bounds are easily provable is to simply *add* the corresponding proof lower bound tautologies $\mathsf{lb}_R(\phi_n, |\phi_n|^k)$ as *axioms* to a standard propositional proof system such as Extended Frege. Since these are tautologies, the resulting system S is still sound, and since the $\{\phi_n\}$ are efficiently computable, S can be shown to be a propositional proof system in the Cook-Reckhow sense. Since the R-proof lower bound formulas corresponding to $\{\phi_n\}$ are axioms of S, they are also easily provable.

We are therefore led to consider situations where the hard tautologies are not efficiently computable. One natural situation in which this happens is if the hard tautologies are *random* in some sense. The Cook-Reckhow program aims to show a worst-case separation between NP and coNP. It is plausible that stronger *average-case* separations hold, and the hypotheses we consider in this paper correspond to such average-case separations for standard NP problems against (non-uniform) coNP over natural distributions.

The main hypothesis we consider is Rudich's Conjecture [46]. Rudich's main motivation in making this conjecture was to strengthen the 'natural proofs' barrier of Razborov and Rudich [45]. The 'natural proofs' barrier shows that under the standard cryptographic assumption that one-way functions exist, there are no dense subsets of the hard Boolean functions computable by polynomial-size circuits, where Boolean functions are represented explicitly by their truth tables. Rudich conjectured that something stronger was true: most hard functions do not even have short *proofs* of hardness that are verifiable in polynomial size, where 'short' means polynomial in the length of the truth table. Rudich originally stated his conjecture in the terminology of natural proofs; we find the reformulation below in terms of proof systems more convenient.

Rudich's Conjecture: For any proof system R verifiable in polynomial size, most Boolean functions on n bits do not have short (i.e., $\mathrm{poly}(2^n)$) size R-proofs of hardness.

Suppose we wish to generate hard tautologies for some propositional proof system R. Simply pick the truth table of a random Boolean function f_n, and consider the circuit lower bound formula $\mathtt{tt}(f_n, n^k)$ for some fixed k. These circuit lower bound formulas are tautologies with high probability because most Boolean functions are hard. Rudich's Conjecture implies that these tautologies are also hard for R (in fact, even for R that is verifiable in polynomial size) with high probability.

Another useful perspective on Rudich's Conjecture is to think of it as an average-case hardness hypothesis for the *Minimum Circuit Size Problem.* The Minimum Circuit Size Problem MCSP is a fundamental problem in NP that is believed to be intractable, but for which NP-completeness has not yet been established. MCSP asks, given the truth table of a Boolean function f_n and a parameter s, whether f_n has circuits of size at most s. Suppose we fix the parameter s as a function of n. Let us call this parameterized version of the problem $\mathsf{MCSP}[s]$. Then Rudich's Conjecture says that for a large enough constant k, $\overline{\mathsf{MCSP}[n^k]}$ does not have any dense subsets computable by nondeterministic polynomial-size circuits, where $\overline{L}$ denotes the complement of L. The conjecture that there are no dense subsets of the NO instances of $\mathsf{MCSP}[n^k]$ computable in some circuit class $\mathfrak{C}$ can be seen as an assertion of zero-error average-case hardness assumption against $\mathfrak{C}$ with respect to the uniform distribution, since almost all instances of a given length N are NO instances [25].

Our notion of feasible proofs extends in a natural way to Rudich's Conjecture, modulo a technicality about non-uniform proofs. The Cook-Reckhow Program aims to separate the uniform classes NP and coNP. Rudich's Conjecture, on the other hand, asserts hardness for an NP problem even against nondeterministic *circuits*, or equivalently proof systems verifiable in polynomial size. This non-uniformity can be modelled easily within our propositional setting using the notion of proof systems with advice due to Cook and Krajicek [16].

In analogy to the Feasible Cook-Reckhow Program, we say that Rudich's Conjecture admits feasible propositional proofs if for every propositional proof system R with polyomial advice and every constant k, there is a propositional proof system S such that for a $1 - o(1)$ fraction of Boolean functions f_n on n bits, S proves efficiently that there are no m^k size R-proofs of $\mathtt{tt}(f_n, n^k)$, where $m = |\mathtt{tt}(f_n, n^k)|$. Namely, for a significant fraction of Boolean functions f,

S can efficiently prove that there are no short R-proofs of the circuit lower bound formula corresponding to f and size parameter n^k.

The second hypothesis we consider is a nondeterministic version of Feige's Hypothesis [20]. Consider the following very natural distribution $U_{\Delta,N}$ over 3-CNFs on N variables with ΔN clauses, where $\Delta > 0$ is any constant: we pick each clause by selecting 3 literals uniformly and independently at random from the $2N$ possible literals. Feige's Hypothesis is that there is no polynomial-time algorithm that outputs 'unsatisfiable' with significant probability over the distribution $U_{\Delta,N}$ and never outputs 'unsatisfiable' on satisfiable formulas. We consider a nondeterministic version of Feige's Hypothesis conjectured by Ryan O'Donnell [25], [38].

Nondeterministic Feige's Hypothesis: For any propositional proof system R verifiable in polynomial size, with high probability over ϕ picked from $U_{\Delta,N}$, there are no polynomial size *proofs* of unsatisfiability for ϕ in R.

Just as Rudich's conjecture implies that circuit lower bound formulas corresponding to random functions are hard tautologies for any propositional proof system with high probability, Nondeterministic Feige's Hypothesis implies that random 3-CNFs with ΔN clauses are hard tautologies for any propositional proof system with high probability, for any $\Delta > 0$. Thus both hypotheses state that natural distributions on tautologies are hard.

Also, just as Rudich's conjecture is an average-case hardness hypothesis for MCSP, Nondeterministic Feige's Hypothesis is an average-case hardness hypothesis for 3-SAT against nondeterministic algorithms. Barak [8] has advocated studying Feige's hypothesis and its ilk, as offering some of our best hopes of more insight into average-case complexity.

We say that Nondeterministic Feige's Hypothesis admits feasible propositional proofs if for every propositional proof system R with polyomial advice and every constant k, there is a propositional proof system S such that with probability $1-o(1)$ over ϕ chosen from $U_{\Delta,N}$, S proves efficiently that there are no m^k size R-proofs of ϕ, where $m = |\phi|$.

C. Our Results

The main result of this paper is that we *unconditionally* rule out feasible proofs of Rudich's Conjecture.

Theorem 1. *Rudich's Conjecture does not admit feasible propositional proofs.*

In other words, there is a propositional proof system R with polynomial advice such that no propositional proof system S can prove lower bounds on the size of R-proofs for most circuit lower bound tautologies. We emphasize two aspects of this result. First, it is unconditional. Second, it rules out polynomial-size S-proofs in *every* propositional proof system S [2], even though S is allowed to depend on R.

At first glance, this might seem strange. If no propositional proof system S can efficiently prove a sequence of tautologies, then $\mathsf{NP} \neq \mathsf{coNP}$ [18]. So why doesn't our result imply a significant complexity lower bound?

The reason is that the R-proof lower bound formulas aren't necessarily tautologies with high probability. If Rudich's Conjecture is true, then they are tautologies. If Rudich's Conjecture is false, on the other hand, this isn't clear. Indeed, our proof of Theorem 1 splits into two cases: the first in which Rudich's Conjecture is true, and the second in which Rudich's Conjecture is false. If Rudich's Conjecture is false, we show using standard techniques (together with an amplification argument) that there is a propositional proof system R with polynomial advice in which most circuit lower bound tautologies for functions on n inputs have short proofs, for infinitely many n. In this case, the R-proof lower bound formulas aren't tautologies, with high probability, and hence they cannot have polynomial-size S-proofs (or indeed proofs of any size) for any sound propositional proof system S.

The crux of our proof is the argument that when Rudich's Conjecture is true, then Rudich's Conjecture does not admit feasible proofs. Thus Rudich's Conjecture is *self-defeating* in the propositional setting: its truth implies its unprovability. This is reminiscent of a comment by Scott Aaronson in his survey on independence results [1] for P vs NP about the 'bizarre self-referential nature of $\mathsf{P} \neq \mathsf{NP}$ - a conjecture that all but asserts the titanic difficulty of finding its own proof'. We show that when considering the stronger statement that is Rudich's Conjecture, this intuition can be made formal in the feasible setting.

We briefly explain the ideas of the proof. The general plan is to exploit the connections between proof complexity and pseudorandomness discovered by [42], [45]. If Rudich's conjecture is true, then given any propositional proof system R with polynomial advice, most truth table tautologies for functions on n bits are indeed hard for R. We would like to use Rudich's conjecture again to show that for any proof system S, even the R-proof lower bound formulas for the circuit lower bound tautologies based on f_n do not have short proofs in S, for most f_n.

The problem is that if S does not have short proofs of a certain formula, this can be for two different reasons. First, the formula might be a tautology but without short proofs in S. This is the good case for us - if all short proofs in S for R-lower bound formulas were for tautologies, we could use this to get a dense set of hard functions computable by small nondeterministic circuits, and thereby get a contradiction to

---

[2] In fact we rule out polynomial-size proofs even for systems S that take polynomial advice

Rudich's Conjecture. However, there is a second case: the formula is not a tautology at all, and so of course it does not have short proofs in S. This case is problematic, because now some functions f_n for which there are short S-proofs of R-lower bound formulas corresponding to f_n might be easy, and thus we don't get a dense subset of hard functions.

To overcome this problem, we use Rudich's Conjecture again, to argue that the truth table of MCSP *itself* does not have small circuits. Thus, the circuit lower bound formula with $f_n = \mathsf{MCSP}$ is indeed a tautology. Moreover, the sequence of such circuit lower bound tautologies has short proofs in an appropriately defined propositional proof system R, where these circuit lower bound tautologies are simply added as axioms to a standard proof system.

It is still not clear how this helps us. An important step conceptually is to introduce the notion of *pseudorandom tautologies*, which are analogues of pseudorandom functions in the proof complexity setting. Intuitively, pseudorandom tautologies are a collection of tautologies which are derived in a specified way from pseudorandom sets, but unlike random tautologies, have short proofs in some predefined proof system R. Thus they can be distinguished from random tautologies by having short proofs in R, just as pseudorandom functions can be distinguished from random functions in the cryptographic setting by being computable from short seeds by a polynomial-time function. For the actual proof of Theorem 1, we work with *hitting tautologies*, which correspond to hitting sets just as pseudorandom tautologies correspond to pseudorandom sets.

Inspired by an idea of Razborov [42], we show how to use Rudich's Conjecture (for a third time!) to get a collection W of hitting tautologies which are easy for the proof system R we define. Since all these hitting tautologies are easy for R, for any propositional proof system S, the soundness of S implies that there are no short S-proofs of R-lower bound formulas corresponding to these tautologies. Suppose that there were short S-proofs of R-lower bound formulas for random circuit lower bound formulas. We show how this together with the hitting property of W implies a contradiction.

We now discuss two possible interpretations of Theorem 1. The first is a metamathematical interpretation, where we understand the result as saying something about the difficulty of proving strong complexity lower bounds. If strong lower bounds such as Rudich's Conjecture hold, then Theorem 1 indicates that standard 'slice-and-measure' techniques such as those used in existing circuit complexity and proof complexity lower bounds are likely to fail in establishing such lower bounds. So either we must believe in the failure of Rudich's Conjecture, which would itself be very interesting and somewhat counterintuitive, or we should be prepared to step outside a finitistic mindset if we want to make progress on very strong complexity conjectures. Indeed, this is illustrated by our proof of Theorem 1, which is non-constructive: we show unconditionally that a certain propositional proof system R with polynomial advice exists for which it is hard to show lower bounds, but we are unable to say what R is. Such non-constructive techniques have been used before as ingredients in diagonalization-style arguments, but we are unaware of any previous examples in propositional proof complexity.

The second interpretation is more relevant to our original motivation of finding proof complexity barriers. As we mentioned, the crux of the proof of Theorem 1 is to show that if Rudich's Conjecture is true, there is a propositional proof system R (without advice) such that R-proof complexity lower bounds are hard to show in any propositional proof system S. This is an analogue of the 'natural proofs' barrier of Razborov and Rudich [45] in proof complexity. Just as the 'natural proofs' barrier says that 'feasible' circuit lower bounds for random functions are unlikely if one-way functions exist, our result says that if Rudich's Conjecture is true, there is a proof system R such that R-proof complexity lower bounds for appopriately defined 'random tautologies' are hard to prove.

Corollary 1. *If Rudich's Conjecture holds, then there is a pps R and a samplable sequence of distributions $\{D_N\}$ on formulas of length $\mathrm{poly}(N)$, such that with probability $1 - 1/N^{\omega(1)}$ over ϕ_N sampled from D_N, ϕ_N is a tautology that does not have $\mathrm{poly}(N)$ size R-proofs, but there is a constant k such that no pps S has polynomial-size proofs of $\mathsf{lb}_R(\phi_N, |\phi_N|^k)$.*

More succinctly, if random circuit lower bound tautologies are hard for propositional proof systems, it is also hard within a propositional framework to *explain why* they are hard.

It is natural to ask if Theorem 1 is a special phenomenon to do with circuit lower bounds, or is part of a more general phenomenon. We give evidence for the latter by showing a similar result for Nondeterministic Feige's Hypothesis, which on the surface seems to have nothing to do with circuit lower bounds. However, unlike Theorem 1, our result for Nondeterministic Feige's Hypothesis is conditional.

Theorem 2. *Assuming Rudich's Conjecture*[3]*, Nondeterministic Feige's Hypothesis does not admit feasible propositional proofs.*

Nondeterministic Feige's Hypothesis guarantees that random 3-CNFs of linear density are hard for any propositional proof system R. Now the question is how to leverage this to show that it is also hard to *prove* that a randomly chosen

---

[3] In fact, we don't need the full strength of Rudich's Conjecture to get the desired consequence for Nondeterministic Feige's Hypothesis. It would be enough to solve Open Problem 3 in [46], which is about 'stretching demi-bits', and seems within reach of current techniques. In the interest of minimizing hypotheses used, we state the result as above, but it is useful to keep in mind that much weaker hypotheses suffice.

3-CNF of linear density is hard for a given proof system R.

If we had a way of constructing pseudorandom or hitting tautologies based on Nondeterministic Feige's Hypothesis, an approach similar to the one for Rudich's Conjecture might work. However, we have no idea how to do this, and are forced to adopt a different strategy.

We use a connection to a hardness hypothesis we call the MKTP Hardness Hypothesis. We will not describe this hypothesis formally here, but it is analogous to Rudich's Conjecture: while Rudich's Conjecture is about most functions lacking short proofs of hardness, the MKTP Hardness Hypothesis is about most strings lacking short proofs that they are hard in the sense of KT-complexity. KT-complexity is a notion of time-bounded Kolmogorov complexity defined by Allender [4] that is closely related to circuit complexity. We leave the details to the main body of the paper and continue with a sketch of the proof.

It is shown in [25] that there is an average-case reduction from Feige's Hypothesis to the question of whether a string has high KT-complexity. We observe that this reduction translates to our setting, and with some technical work, show how to leverage it to get an implication from feasible proofs of Nondeterministic Feige's Hypothesis to feasible proofs of the MKTP Hardness Hypothesis (with parameters chosen appropriately). We then show that feasible proofs of the MKTP Hardness Hypothesis do not exist under strong enough complexity assumptions, and in particular assuming Rudich's Conjecture. This second part of the argument is analogous to the proof of Theorem 2, but there are some additional difficulties that need to be overcome.

A common theme to Theorems 1 and 2 is that the hardness of proving that strings are "random-like" has implications for barriers in proof complexity.

Theorem 2 has implications for both meta-mathematics of complexity lower bounds and for proof complexity barriers, just as with Theorem 1. The implication for proof complexity barriers is especially interesting in this case since random CNFs are a very natural class of formulas to analyze in terms of proof complexity. Similar to Corollary 1, we get from Theorem 2 that under certain hardness assumptions, namely Rudich's Conjecture and Nondeterministic Feige's Hypothesis, there is a propositional proof system R with advice such that proving lower bounds on R is hard for any propositional proof system S.

The final result we highlight in this introduction concerns the standard Extended Frege proof system. Our previous results give 'proofs of principle': under natural hardness assumptions, there are propositional proof systems that are hard to analyze. While these results show fundamental obstacles to proving strong complexity lower bounds in a feasible way, it is unclear how relevant they are to analysis of standard proof systems such as Frege and Extended Frege. We show that for these proof systems, a 'lose-lose theorem' holds: Either the Circuit Lower Bounds program cannot be implemented within the proof system, or there are inherent obstacles to implementing the Cook-Reckhow program for the proof system. We state the result for Extended Frege, but essentially the same proof also gives the analogous result for Frege.

Theorem 3. *Assume Rudich's Conjecture and that* E *does not have sub-exponential size nondeterministic circuits. Either* EF *does not efficiently prove circuit lower bound tautologies* $\mathsf{tt}(f_n, s)$ *for any sequence* $\{f_n\}$ *of functions and* $s(n) = n^{\omega(1)}$*, or there are sets* S_N *of formulas of size* N *that can be generated in time* $\mathrm{poly}(N)$*, such that for each* $c > 0$ *and all large enough* N*, most* $\phi \in S_N$ *are tautologies and require* EF*-proofs of size at least* $N^{\omega(1)}$*, and yet no propositional proof system* S *can prove in size* N^{c^2} *that for most* $\phi \in S_N$*,* ϕ *requires* EF *proofs of size at least* N^c*.*

The proof of Theorem 3 uses ideas mentioned earlier together with a complexity-theoretic derandomization technique [26], [37] to generate explicit sets of tautologies.

D. Related Work

Perhaps the most interesting recent work on the difficulty of showing proof complexity lower bounds is by Grochow and Pitassi [22]. They define a proof system called the Ideal Proof System (IPS). IPS is verifiable in randomised polynomial time, but is not known to be a propositional proof system [4]. It is shown in [22] that IPS lower bounds for DNF tautologies imply that VNP $\neq$ VP, where VNP is the algebraic complexity analogue of NP and VP is the algebraic complexity analogue of P. Since separating VNP and VP is believed to be a hard problem, this gives evidence that showing IPS lower bounds is hard. The work of [22] differs from ours in considering a proof system that is not known to be propositional, and giving evidence for the hardness of proof complexity lower bounds based on the presumed hardness of circuit complexity separations. In our work, we formalize the question of proof complexity lower bounds within propositional proof complexity, and study the inherent limitations of propositional proof systems, rather than reducing to circuit complexity questions.

In terms of the finitistic meta-mathematics of complexity separations, the work that is perhaps closest to ours in spirit is by Razborov [42]. Razborov shows that under certain complexity assumptions, no proof system P with feasible interpolation can prove *any* super-polynomial circuit complexity lower bound. Our work differs from that of Razborov in showing an unconditional result, and in considering arbitrary propositional proof systems, rather than ones with feasible interpolation. The 'feasible interpolation' condition is a fairly restrictive one, and there is evidence that Frege and EF do not have feasible interpolation [12], [32]. Rudich [46]

---

[4] However, as pointed out by an anonymous referee, by using Adleman's trick, IPS is a propositional proof system with polynomial advice, when defined over large enough fields

uses his Conjecture to extend Razborov's result to propositional proof systems that have the 'feasible disjunction' property. However, the feasible disjunction property is not well understood, and it is not clear if Frege or EF have this property. There is also work on proof complexity generators [6], [28], [29], [44] where fairly general conjectures are made about unprovability of circuit lower bounds in EF, but it is not known how to connect these conjectures to complexity assumptions. It is also important to note that our formalization is different from the ones considered in these papers - our notion of 'feasible proofs' aims to prove complexity separations implicitly rather than explicitly as done using the circuit lower bound tautologies. This implicit formulation allows us to prove negative results even for arbitrary propositional proof systems.

We focus in this paper on the setting of propositional proof complexity. There is a lot of work on independence results in the setting of bounded arithmetic, which can be thought of as a uniform version of propositional proof complexity. For example, it is known unconditionally that Cook's theory PV_1 cannot prove super-polynomial lower bounds on EF (which is the propositional proof system analogue of PV_1) [13], [19], [32]. Some of our results in this paper have analogues in the bounded arithmetic setting, which we plan to explore in a separate work.

II. PRELIMINARIES

$\mathcal{F}_n$ denotes the set of all Boolean functions on n inputs.

A. Proof complexity

Propositional proof systems were defined by Cook and Reckhow as surjective polynomial-time computable functions from $\{0,1\}^*$ to the set of tautologies (represented as strings). We use an equivalent notion that is better suited to defining proof systems with advice. Let TAUT denote the set of strings over the binary alphabet that encode DNF tautologies, under some standard efficient encoding of formulas as strings. Given this standard encoding of formulas as strings, for any formula ϕ, $|\phi|$ denotes the length of the encoding of ϕ, in bits.

Definition 1. *A propositional proof system (pps) is a polynomial-time computable relation* $R(\cdot,\cdot)$ *such that for each* $x \in \{0,1\}^*$, $x \in \mathsf{TAUT}$ *iff there exists* $y \in \{0,1\}^*$ *such that* $R(x,y)$ *holds. Given* $x \in \mathsf{TAUT}$, *any* y *for which* $R(x,y)$ *holds is called an* R*-proof of* x. *A pps* R *is polynomially bounded (p-bounded) if there exists a polynomial* p *such that for each* $x \in \mathsf{TAUT}$, *there is an* R*-proof of* x *of size at most* $p(|x|)$.

Proposition 1. $\mathsf{NP} = \mathsf{coNP}$ *iff there exists a p-bounded pps.*

Definition 2. *Given* $a : \mathbb{N} \to \mathbb{N}$, *a propositional proof system with* a *bits of advice is a relation* $R(x,y,z)$ *computable in polynomial time such that for each* $n \in \mathbb{N}$, *there is* $w_n \in \{0,1\}^*$ *of length* $a(n)$ *satisfying the following condition: for each* $x \in \{0,1\}^n$, $x \in \mathsf{TAUT}$ *iff there exists* $y \in \{0,1\}^*$ *such that* $R(x,y,w_n)$ *holds. We call an advice string* w_n *good for* R *if it satisfies the preceding condition, and we call an advice sequence* $\{w_i\}$ *good for* R *if for each* n, w_n *is good for* R. *Given* $x \in \mathsf{TAUT}$ *and advice string* $w_{|x|}$, *any* y *for which* $R(x,y,w_{|x|})$ *holds is called an* R*-proof of* x *with advice* $w_{|x|}$. *A pps* R *with advice is p-bounded if there exists an advice sequence* $\{w_i\}$ *good for* R *and a polynomial* p *such that for each* $x \in \mathsf{TAUT}$, *there is an* R*-proof of* x *with advice* $w_{|x|}$ *of size at most* $p(|x|)$.

Proposition 2. $\mathsf{NP} \subseteq \mathsf{coNP/poly}$ *iff there exists a p-bounded pps with polynomial advice.*

Let P, Q be propositional proof systems. P *simulates* Q if there is a polynomial p such that whenever there is an s-size Q-proof of ϕ, there is a $p(s)$-size P-proof of ϕ. P is *optimal* if it simulates every propositional proof system.

P admits *instantiation property* if whenever there is an s-size P-proof of $\phi(x_1,\ldots,x_n)$, P proves each instance of ϕ, i.e. $\phi(a_1,\ldots,a_n)$ where $a_i \in \{0,1\}^n$, by a proof of size s.

B. Formalizing Lower Bounds in the Propositional Setting

Circuit lower bounds. An $s(n)$-size circuit lower bound for a function $f \in \mathcal{F}_n$ can be expressed by a $2^{O(n)}$-size propositional formula $\mathsf{tt}(f,s)$,

$$\bigvee_{y \in \{0,1\}^n} f(y) \neq C(y)$$

where the formula $f(y) \neq C(y)$ says that a circuit C represented by $poly(s)$ variables does not output $f(y)$ on input y.

Definition 3. *Given a Boolean function* $f \in \mathcal{F}_n$ *and size parameter* s, $\mathsf{tt}(f,s)$ *is a propositional DNF formula of size* $\tilde{O}(2^n s^3)$ *over* $\tilde{O}(s)$ *variables expressing that* f *does not have Boolean circuits of size* s.

More explicitly, $\mathsf{tt}(f,s)$ is defined by taking an OR over all $y \in \{0,1\}^n$ of the predicate $f(y) \neq C(y)$, and expressing $f(y) \neq C(y)$ as a DNF formula of size $\tilde{O}(s^3)$ over $\tilde{O}(s)$ propositional variables using standard techniques.

Definition 4. *Given pps* R, *propositional formula* ϕ *and size function* $s : \mathbb{N} \to \mathbb{N}$, $\mathsf{lb}_R(\phi,s)$ *is a propositional DNF formula of size* $\mathrm{poly}(|\phi|+s)$ *over* $\mathrm{poly}(|\phi|+s)$ *variables expressing that there is no* R*-proof of* ϕ *having size* s.

More explicitly, the formula $\mathsf{lb}_R(\phi,s)$ contains s variables $y_1,\ldots,y_s$ encoding R-proofs of length s and $\mathrm{poly}(|\phi|+s)$ auxiliary variables encoding the computation of the relation R, to verify that $y_1,\ldots,y_s$ does not constitute an R-proof of ϕ.

We extend the above definition in a natural way to ppses with advice. In the process, we overload the notation lb: the number of parameters tells us whether we are dealing with standard ppses, or ppses with advice.

Definition 5. *Given pps R with advice function $a : \mathbb{N} \to \mathbb{N}$, propositional formula ϕ, size function $s : \mathbb{N} \to \mathbb{N}$ and advice string w of length $a(|\phi|)$, $\mathsf{lb}_R(\phi, s, w)$ is a propositional DNF formula of size $\mathrm{poly}(|\phi| + s)$ over $\mathrm{poly}(|\phi| + s)$ variables expressing that there is no R-proof with advice w of ϕ having size s.*

C. Rudich's Conjecture and Nondeterministic Feige's Hypothesis

Given a language $L \subseteq \{0,1\}^*$, the n-slice L_n of L is $L \cap \{0,1\}^n$. Given a function $\epsilon : \mathbb{N} \to [0,1]$, we say that a language $L \subseteq \{0,1\}^*$ is ϵ-sparse if for each large enough n, $|L_n|/2^n \leq \epsilon(n)$. L is ϵ-dense if for each large enough n, $|L_n|/2^n > \epsilon(n)$. Note that ϵ-density is not simply the negation of ϵ-sparsity.

We say that a function $f : \mathbb{N} \to \mathbb{N}$ is poly-constructible if there is a Turing machine transducer that, on input 1^n, halts with $f(n)$ on its output tape within $\mathrm{poly}(n)$ steps.

Given a language L, $\overline{L}$ denotes the complement of L.

We will be working with two fundamental NP problems - MCSP (Minimum Circuit Size Problem) and 3-SAT (Satisfiability Problem for 3-CNFs).

Given a string y of length N, $\mathrm{fn}(y)$ is the Boolean function on $\lfloor \log(N) \rfloor$ bits whose truth table is the $2^{\lfloor \log(N) \rfloor}$-bit initial prefix of y. Conversely, given a Boolean function f on n bits, $\mathrm{tt}(f)$ is the truth table of f. Given a Boolean circuit C, f_C denotes the Boolean function computed by C, and $< C >$ denotes the standard encoding of C as a string.

Definition 6. *Given a size function $s : \mathbb{N} \to \mathbb{N}$, $\mathsf{MCSP}[s]$ is the set of strings y such that $\mathrm{fn}(y)$ has Boolean circuits of size at most $s(n)$.*

Note that for any poly-constructible s, $\mathsf{MCSP}[s]$ is in NP, simply by guessing a circuit of size at most $s(n)$ and checking in polynomial time (in the truth table size) that it computes $\mathrm{fn}(y)$. It is also worth pointing out that in the above definition, s is measured as a function of the length n of the input to the circuit, rather than as a function of the length N of the input to the problem, which is exponentially large in n.

When we are working with MCSP, by default we will use N to refer to the input size of the problem, and $n = \lfloor \log(N) \rfloor$ to refer to the input size of the function encoded by the input to MCSP.

We consider two hardness assumptions, which concern the average-case hardness of MCSP and 3-SAT respectively against non-deterministic circuits over natural distributions. The first of these assumptions has been considered before by Rudich [46], while the second is a nondeterministic variant of a hypothesis originally considered by Feige [20] in the context of hardness of approximation. This nondeterministic variant has been conjectured by O'Donnell [25], [38]. We state each of these hardness assumptions in turn.

Rudich's Conjecture: There is a constant $c > 0$ such that for every language L, if $L \subseteq \overline{\mathsf{MCSP}[n^c]}$ and $L \in \mathsf{NSIZE}(\mathrm{poly})$, then L is $1/N^{\omega(1)}$-sparse.

In other words, any set in $\mathsf{NSIZE}(\mathrm{poly})$ consisting only of truth tables of hard Boolean functions (where 'hard' means that the circuit complexity is greater than n^k) is sub-polynomially sparse. Rudich's conjecture was originally stated in terms of the notion of natural proofs, but the formulation above is equivalent.

Next, we state the nondeterministic variant of Feige's hypothesis. Given a positive constant Δ, $U_{\Delta,N}$ is the distribution over 3-CNFs on N variables obtained by picking $\lceil \Delta n \rceil$ 3-clauses independently at random, where each 3-clause is picked by choosing 3 literals uniformly and independently at random from the set of $2N$ literals over N variables.

Nondeterministic Feige's Hypothesis: For any constant $\Delta > 0$, for every language L, if $L \subseteq \overline{3-\mathsf{SAT}}$ and $L \in \mathsf{NSIZE}(\mathrm{poly})$, then $\Pr_{\phi \sim U_{\Delta,N}}[L(\phi) = 1] = o(1)$.

As stated, this hypothesis is about the hardness of proving unsatisfiability. Since we are concerned with proof systems for tautologies in this paper, we will simply interprets a proof that $\overline{\phi}$ is a tautology as a proof that ϕ is unsatisfiable, where $\overline{\phi}$ is the complement of ϕ obtained by using De Morgan's laws. Thus $\overline{\phi}$ is a 3-DNF iff ϕ is a 3-CNF.

Feige [20] also considered a weaker, but more robust, version which asserts the hardness of proving that formulas are far from satisfiable. We choose the version stated above because it is more convenient from the point of view of proof complexity; however, our results can also be adapted to deal with the weaker variant.

We now define what it means to have feasible propositional proofs of these two hardness assumptions.

Definition 7. *We say that Rudich's conjecture admits feasible propositional proofs if for every large enough integer $d > 0$ and every pps R with polynomial advice, there is a pps S, such that if $\{w_m\}$ is a sequence of poly-sized advice strings good for R, then for all large enough n, for a $1 - o(1)$ fraction of Boolean functions $f_n \in \mathcal{F}_n$, there are polynomial-sized S-proofs of $\mathsf{lb}_R(\mathsf{tt}(f_n, n^d), m^d, w_m)$, where $m = |\mathsf{tt}(f_n, n^d)|$.*

To clarify this definition, note that Rudich's conjecture states that there is a constant $c > 0$ such that for every constant $d > 0$, there are no non-uniform proofs of size m^d of $\mathsf{tt}(f_n, n^c)$ (where $m = |\mathsf{tt}(f_n, n^c)|$) for at least a

$1 - 1/N^{\omega(1)}$ fraction of Boolean functions f_n over $n = \lfloor \log(N) \rfloor$ bits, when N is large enough. This is equivalent to saying that for every large enough constant $d > 0$, there are no non-uniform proofs of size m^d of $\mathsf{tt}(f_n, n^d)$ for most f_n, since a lower bound of size n^d for $d \geq c$ implies a lower bound of size n^c. We say that the conjecture admits feasible proofs if given a fixed pps R with polynomial advice, there is a pps S that efficiently proves lower bounds on the size of the R-proofs for random truth table tautologies, thereby feasibly giving evidence that R does not witness a refutation of Rudich's conjecture. Note that we allow S to depend on R, and moreover we only require S to provide efficient proofs of lower bounds on R-proof size for $1-o(1)$ fraction of truth-table tautologies, even though Rudich's conjecture implies more strongly that all but a negligible fraction of truth table tautologies require large R-proofs.

Definition 8. *We say that Nondeterministic Feige's Hypothesis admits feasible propositional proofs if for every $\Delta > 0$, for every pps R with polynomial advice and for every integer $d > 0$, there is a pps S, such that if $\{w_m\}$ is a sequence of poly-size advice strings good for R, then for all large enough N, with probability $1 - o(1)$ over ϕ sampled from $U_{\Delta,N}$, there are polynomial-sized S-proofs of $\mathsf{lb}_R(\overline{\phi}, m^d, w_m)$, where $m = |\overline{\phi}|$.*

D. Pseudorandomness

Definition 9. *For fixed integers N and t and a parameter $\epsilon \geq 0$, we say that $H_N \subseteq \{0,1\}^N$ is an ϵ-hitting set against size t (resp. nondeterministic size t) if $H_N \cap S \neq \emptyset$ for every ϵ-dense $S \subseteq \{0,1\}^N$ computable by circuits (resp. nondeterministic circuits) of size t. We say that $H_N \subseteq \{0,1\}^N$ is an ϵ-discrepancy set against size t if for every circuit C of size t on N bits, $|\Pr_{x\sim U_N}[C(x)=1] - \Pr_{y\in H_N}[C(y)=1]| \leq \epsilon$.*

Given an integer s, we say that an ϵ-hitting set (resp. an ϵ-discrepancy set) H_N is s-succinct if for each $y \in H_N$, $\mathrm{fn}(y)$ has circuits of size at most s.

Proposition 3. *Let $s : \mathbb{N} \to \mathbb{N}$ and $t : \mathbb{N} \to \mathbb{N}$ be size functions, and let $\epsilon : \mathbb{N} \to \mathbb{R}$ be non-negative. For any $N \in \mathbb{N}$, there are $s(n)$-succinct $\epsilon(N)$-hitting sets over N-bit strings against non-deterministic size $t(N)$ iff $\overline{\mathsf{MCSP}[s]}$ has no $\epsilon(N)$-dense subsets in non-deterministic size $t(N)$.*

Definition 10. *Let $\ell : \mathbb{N} \to \mathbb{N}$ satisfy $\ell(N) \leq N$ for each n, and let $t : \mathbb{N} \to \mathbb{N}$ be a size function. Let $\epsilon : \mathbb{N} \to \mathbb{R}$ be non-negative. We say that a sequence of functions $\{G_N\}$, where each $G_N : \{0,1\}^{\ell(N)} \to \{0,1\}^N$ is a complexity-theoretic ϵ-PRG with seed length $\ell(N)$ against size t (resp. nondeterministic size s) if there is a Turing machine which given 1^N and $z \in \{0,1\}^{\ell(N)}$ computes $G_N(z)$ in time $2^{O(\ell(N))}$, and moreover for each large enough N, the range of G_N is an $\epsilon(N)$-discrepancy set against size $t(N)$ (resp. non-deterministic size $t(N)$).*

Theorem 4. *[26] If there is a constant $\gamma > 0$ such that $\mathsf{E} \not\subseteq \mathsf{io-NSIZE}(2^{\gamma n})$, then there is a complexity-theoretic $1/N$-PRG with seed length $O(\log(N))$ against nondeterministic size N.*

We next define the crucial concept of R-easy hitting and pseudorandom tautologies.

Definition 11. *Let R be a pps, $\epsilon : \mathbb{N} \to \mathbb{R}$ be non-negative, and $s : \mathbb{N} \to \mathbb{N}$ be a size function. Let $\mathfrak{C}$ be a circuit class. We say W is a set of ϵ-pseudorandom (resp. ϵ-hitting) tautologies against $\mathfrak{C}$ that is s-easy for R if there is a polynomial-time computable function $f : \{0,1\}^* \to \{0,1\}^*$ and a sequence $\{H_N\}$, where for large enough n, $H_N \subseteq \{0,1\}^*$ is an $\epsilon(N)$-discrepancy (resp. $\epsilon(N)$-hitting) set against $\mathfrak{C}$, and moreover:*

1) $W = \bigcup_N f(H_N)$
2) *For all but finitely many $\phi \in W$, we have that $\phi \in \mathsf{TAUT}$*
3) *Each tautology $\phi \in W$ has R-proofs of size $s(|\phi|)$.*
4) $\Pr_{y\sim U_N}[f(y) \in \mathsf{TAUT}] = 1 - 1/N^{\omega(1)}$
5) $\Pr_{y\sim U_N}[f(y)$ *has R-proofs of size* $s(|f(y)|)] = o(1)$

On occasion, we will use the above definition for an R that is a pps with advice. In the default case that the parameter s is polynomially bounded, we will simply call the corresponding pseudorandom or hitting tautologies R-easy.

Intuitively, the above definition captures the notion of tautologies that can be computed efficiently (using the function f) from hitting or pseudorandom sets and have short proofs in R, while if f is applied to random strings, the resulting formulas are tautologies with very high probability but lack short R-proofs with significant probability.

E. Kolmogorov Complexity

KT-complexity was proposed in [4], as a variant of Levin's notion of time-bounded Kolmogorov complexity that is closely connected to circuit complexity. Indeed, it is known that $\mathsf{KT}(\mathrm{tt}(f))$ and the minimum circuit size of f are polynomially-related to each other. Fix a universal random-access Turing machine U that simulates all Turing machines efficiently. Informally, the KT-complexity of a string y is the minimum of $|d| + t$, where d is a string for describing y implicitly and t is the time it takes to output y. More formally, we have the definition below, where U^d denotes the Turing machine U with random access to the string d:

Definition 12. *Let $y = y_1 \cdots y_N \in \{0,1\}^N$. The KT-complexity of y is defined as follows.*

$$\mathsf{KT}(y) := \min\{|d|+t \mid U^d(i) = y_i \text{ in } t \text{ steps for any } 1 \leq i \leq N+1\}.$$

Here, y_{N+1} is defined as $\perp$ (a stop symbol).

We have the following simple facts about KT- complexity, in analogy to corresponding facts about the standard notion of Kolmogorov complexity.

Proposition 4. 1) *For any string y,* $\mathsf{KT}(y) \leq |y| + O(\log(|y|))$
2) *Given any non-negative integer r, for each n, for at least $1 - 1/2^r$ fraction of strings y of length n,* $\mathsf{KT}(y) \geq |y| - r$.

The first item follows by using y as its own description, and the second item follows from a straightforward counting argument.

The following computational problem is naturally associated with KT-complexity.

Definition 13. *Given a size function $s : \mathbb{N} \to \mathbb{N}$, $\mathsf{MKTP}[s]$ is the set of strings y such that $\mathsf{KT}(y) \leq s(|y|)$.*

Note that for any poly-constructible s, $\mathsf{MKTP}[s]$ is in NP, simply by guessing a string d and a number t such that $|d| + t \leq s(N)$, and checking that $U^d(i) = y_i$ in t steps for each $1 \leq i \leq N + 1$. Note also that unlike in the case of MCSP, we measure s as a function of the length N of the input string.

The following lemma bounds the KT complexity of satisfiable 3-CNFs.

Lemma 1. *Let $\{\phi_n\}$ be any sequence of satisfiable 3-CNFs, such that $|\phi_n| = n$ for each n. Then there is a constant $\delta > 0$ such that for large enough n, $\mathsf{KT}(\phi_n) \leq (1 - \delta)n$.*

We consider a family of hypotheses parameterized by a size function $s : \mathbb{N} \to \mathbb{N}$ such that $s(N) = N - \omega(\log(N))$.

$\mathsf{MKTP}[s]$ **Hardness Hypothesis:** If $L \subseteq \overline{\mathsf{MKTP}[s]}$ and $L \in \mathsf{NSIZE}(\text{poly})$, then L is $o(1)$-sparse.

For $s \leq s'$, it is obvious that the $\mathsf{MKTP}[s]$ Hardness Hypothesis implies the $\mathsf{MKTP}[s']$ Hardness Hypothesis, thus the hypothesis is more believable for larger s. In fact, for any $s = \log(N)^{\omega(1)}$, Rudich's Conjecture implies the corresponding hardness hypothesis for MKTP.

Proposition 5. *Let $s(N) = \log(N)^{\omega(1)}$ be any size function. Rudich's Conjecture implies the* $\mathsf{MKTP}[s]$ *Hardness Hypothesis.*

Proof: Follows immediately from the standard fact that for any string x, $KT(x)$ is upper bounded by a fixed polynomial in the circuit size of $\text{fn}(x)$. ■

The following proposition, giving a connection between the $\mathsf{MKTP}[s]$ Hardness Hypothesis for a given s and hitting sets composed of strings with low KT-complexity, is completely analogous to Proposition 3.

Proposition 6. *Let $s : \mathbb{N} \to \mathbb{N}$ and $t : \mathbb{N} \to \mathbb{N}$ be size functions, and let $\epsilon : \mathbb{N} \to \mathbb{R}$ be non-negative. For any $N \in \mathbb{N}$, there is an $\epsilon(N)$-hitting set $H_N \subseteq \mathsf{MKTP}[s]$ over N-bit strings against non-deterministic size $t(N)$ iff $\overline{\mathsf{MKTP}[s]}$ has no $\epsilon(N)$-dense subsets in non-deterministic size $t(N)$.*

We next define certain formulas expressing that a string is random in the sense of KT-complexity.

Definition 14. *Given a string y of length N and size parameter s, $\mathsf{random}(y, s)$ is a propositional DNF formula of size $\tilde{O}(Ns^3)$ over $\tilde{O}(s)$ variables expressing that y does not have KT-complexity at most s.*

More explicitly, $\mathsf{random}(y, s)$ is defined by taking an OR over all $i \in [N + 1]$ of the predicate $y_i \neq U^d_{s-|d|}(i)$, where $U^d_{t'}(i)$ denotes the simulation of U on i for t' steps, and expressing $y_i \neq U^d_{s-|d|}(i)$ as a DNF formula of size $\tilde{O}(s^3)$ over $\tilde{O}(s)$ propositional variables using standard techniques.

Definition 15. *Let $s : \mathbb{N} \to \mathbb{N}$ be such that $s(N) \leq N$ for all N. We say that the* $\mathsf{MKTP}[s]$ *Hardness Hypothesis admits feasible propositional proofs if for every large enough integer $d > 0$ and every pps R with polynomial advice, there is a pps S, such that if $\{w_m\}$ is a sequence of poly-sized advice strings good for R, then for all large enough N, for a $1 - o(1)$ fraction of strings y of length N, there are polynomial-sized S-proofs of $\mathsf{lb}_R(\mathsf{random}(y, s(N)), m^d, w_m)$, where $m = |\mathsf{random}(y, s(N))|$.*

III. IMPOSSIBILITY OF FEASIBLY PROVING RUDICH'S CONJECTURE

Lemma 2. *If Rudich's Conjecture holds, then there is a pps R such that for any k, there is a collection of R-easy $1/N^k$-hitting tautologies against nondeterministic size N^k.*

Proof: Suppose Rudich's Conjecture holds. Then there is a constant c such that any $\mathsf{NSIZE}(\text{poly})$ subset of $\overline{\mathsf{MCSP}[n^c]}$ is $1/N^{\omega(1)}$-sparse. We assume wlog that $c > 1$. Since $\overline{\mathsf{MCSP}[n^c]}$ contains almost all strings of length N for any $N > 0$, it follows that $\mathsf{MCSP}[n^c]$ does not have circuits of polynomial size almost everywhere.

We define a sequence $\{z_N\}$ of N-bit strings as follows. If $N = 2^n$ for some non-negative integer n, then z_N is defined to be the truth table of $\mathsf{MCSP}[m^c]$ on inputs of length n. If not, then $z_N = z_{2^{\lfloor \log(N) \rfloor}} 0^{N - 2^{\lfloor \log(N) \rfloor}}$. By the observation in the previous para, we have that for any polynomially bounded function t, for all large enough N, $\text{fn}(z_N)$ does not have circuits of size $t(n)$.

Next we use the strings $\{z_N\}$ to define a sequence of hitting sets. Since Rudich's Conjecture holds, by Proposition 3, we have that for any k and each large enough N, there is a n^c-succinct $1/N^k$-hitting set $H_N \subseteq \{0,1\}^n$ against nondeterministic size N^k.

Consider the sequence of sets $\{H'_N\}$ defined by $H'_N = \{y \oplus z_N | y \in H_N\}$ for each N. We claim that $\{H'_N\}$ is also a sequence of $1/N^k$-hitting sets against nondeterministic size N^k. Indeed, if not, for infinitely many N, there must be a $1/N^k$-dense subset S'_N of $\{0,1\}^N$ with N^k size nondeterministic circuits such that $S'_N \cap H'_N = \emptyset$. Define the subsets $S_N = y \oplus z_N | y \in S'_N$. It is clear that for each N such that S'_N has N^k size nondeterministic circuits, so must

S_N, simply by negating the i'th input variable in the circuit for S'_N iff $z_i = 1$. Also, S'_N has the same density as S_N, as it is just a linear translate of S_N. Now, by definition of H'_N, we have that $H'_N \cap S'_N = \emptyset$ iff $H_N \cap S_N = \emptyset$. Thus we have that for infinitely many N, there is an $1/N^k$-dense subset S_N of N-bit strings with N^k size nondeterministic circuits such that $S_N \cap H_N = \emptyset$, contradicting the assumption that H_N is an $1/N^k$-hitting set against nondeterministic size N^k for all large enough N.

Now we define our candidate collection of $1/N^k$-hitting tautologies against nondeterministic size N^k, and a pps R such that these hitting tautologies all have short proofs in R. We argue that if Rudich's Conjecture holds, all the conditions in Definition 11 are satisfied, and therefore our candidate collection is indeed hitting.

Define the poly-time computable function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ by $f(x) = \mathsf{tt}(\mathrm{fn}(x), \lfloor \log(|x|) \rfloor^c)$. Let $W = \bigcup_N f(H'_N)$. W is our candidate set of $1/N^k$-hitting tautologies against nondeterministic size N^k.

Condition (i) in Definition 11 is satisfied trivially because of the way we define W, given that $\{H'_N\}$ is a sequence of $1/N^k$-hitting sets against nondeterministic size N^k. To see that condition (ii) holds, note that $\mathrm{fn}(z_N)$ does not have circuits of size $2n^c + 3$ for large enough N, where $n = \lfloor \log(N) \rfloor$. Also we have that each $y \in H_N$ is n^c-succinct. It follows that for each $y \in H_N$, $y \oplus z_N$ does not have circuits of size n^c for large enough N, since $\mathrm{fn}(y \oplus z_N)$ can be computed by combining circuits for $\mathrm{fn}(y)$ and $\mathrm{fn}(z_N)$ with one additional XOR operation (which can be implemented with 3 AND/OR gates). Thus, for each $x \in H'_N$, when N is large enough, $\mathrm{fn}(x)$ does not have circuits of size $\lfloor \log(|x|) \rfloor^c$, which implies that $f(x)$ is a tautology, using Definition 14.

To show that condition (iv) holds, simply note that at most $2^{\mathrm{polylog}(N)}$ strings y of length N satisfy the condition that $\mathrm{fn}(y)$ has circuits of size at most $\lfloor \log(N) \rfloor^c$, and hence with overwhelming probability over y chosen uniformly at random from strings of length N, $f(y)$ is a tautology.

In order to show conditions (iii) and (v) hold, we define an appropriate pps R. For any $u, v \in \{0,1\}^*$, $R(u,v) = 1$ iff (a) $v = 01^{2^{|u|}}$ and $u \in \mathsf{TAUT}$ or (b) $v = 1 < C >$ for some Boolean circuit C on n bits of size at most n^c, and $u = \mathsf{tt}(\mathrm{fn}(z_N \oplus y), n^c)$ for some y such that $\mathrm{fn}(y) = f_C$ and $N \geq N_0$, where N_0 is a fixed constant such that $\mathrm{fn}(z_N)$ does not have circuits of size $2\lfloor \log(N) \rfloor^c + 3$ for any $N \geq N_0$. Intuitively, each tautology has its standard exponential-size truth-table proof in R, but in addition, the hitting tautologies have short proofs encoded by circuits succinctly representing the corresponding elements of the hitting sets H'_N.

We now formally argue that R is a pps. We need to argue that R is complete, sound and polynomial time computable. For completeness, note that each tautology u has a proof v of size $2^{|u|}+1$, by definition of R. For soundness, there are two cases to be considered. In case (a) of the definition of R, we have that $u \in \mathsf{TAUT}$, and hence this case is sound. In case (b), u is accepted only if it is a truth-table formula expressing that $\mathrm{fn}(z_N \oplus y)$ does not have circuits of size n^c, the proof encodes a circuit of n^c accepting $\mathrm{fn}(y)$, and moreover $N \geq N_0$ satisfies that $\mathrm{fn}(z_N)$ does not have circuits of size $2n^c + 3$. It is clear that in this case, u is accepted only if it is a tautology, since the upper bound on the circuit complexity of $\mathrm{fn}(y)$ witnessed by C and the lower bound on the circuit complexity of $\mathrm{fn}(z_N)$ guaranteed by N being large enough together imply that $\mathrm{fn}(z_N \oplus y)$ does not have circuits of size n^c.

To show that R is polynomial-time computable, define a poly-time Turing machine M that decides R by operating as follows. On input (u, v), it first checks if v begins with a 0. If so, it first checks that $v = 01^{2^{|u|}}$. If not, it rejects. If yes, it uses brute force search over all possible assignments to u to check that u is a tautology. If this brute force search succeeds, it accepts, otherwise it rejects. If v begins with 1, M checks that $v = 1 < C >$ for some Boolean circuit C, say on n bits. It then checks that $u = \mathsf{tt}(\mathrm{fn}(z_N \oplus y), n^c)$ for some y such that $\mathrm{fn}(y) = f_C$ and $N \geq N_0$, where N_0 is a constant hardwired into M. Note that since $\mathsf{MCSP}[m^c]$ is in deterministic time $2^{\mathrm{polylog}(n)}$ by brute-force search, where n is the input length for $\mathsf{MCSP}[m^c]$, we have that z_N can be computed in time $\mathrm{poly}(N)$ deterministically, by computing the truth table of $\mathsf{MCSP}[m^c]$ on input length $n = \lfloor \log(N) \rfloor$ in time $2^{n+\mathrm{polylog}(n)} = \mathrm{poly}(N)$. Once z_N is computed in polynomial time, M can compute y in polynomial time and verify that $y = f_C$. Irrespective of the format of v, M halts in polynomial time, and decides R.

Finally, we establish that our candidate set W of hitting tautologies satisfies conditions (iii) and (v) in Definition 11 with respect to R. To see that W is R-easy, note that every tautology in W is of the form $\mathsf{tt}(\mathrm{fn}(y \oplus z_N), n^c)$ for some $y \in H_N$. Since H_N is n^c-succinct, it holds for each $y \in H_N$ that $\mathrm{fn}(y)$ has circuits of size at most n^c, hence there is some circuit C of size at most n^c such that $f_C = \mathrm{fn}(y)$. But then, by definition of R, $1 < C >$ is a valid proof of $u = \mathsf{tt}(\mathrm{fn}(y \oplus z_N), n^c)$ when $N \geq N_0$, and $v = 01^{2^{|u|}}$ is a valid proof of u when $N < N_0$. Since $| < C > | = \mathrm{poly}(n) << N$, we have that tautologies in W have polynomially bounded R-proofs. Indeed, for large enough N, there are proofs of size at most N.

To establish condition (v), we again use the assumption that Rudich's conjecture holds. Suppose, for the sake of contradiction, that there is a constant $\gamma > 0$ and positive integer e such that for infinitely many N, $\Pr_{y \sim U_N}[f(y)$ has R-proofs of size at most $|f(y)|^e] \geq \gamma$. We show how to refute Rudich's conjecture by constructing $L \subseteq \overline{\mathsf{MCSP}[n^c]}$ such that $L \in \mathsf{NSIZE}(\mathrm{poly})$ but L is not $1/N^{\omega(1)}$-sparse. Simply define $L(y) = 1$ iff $f(y)$ has R-proofs of size at most $|f(y)|^e$. By soundness of R, $L \subseteq \overline{\mathsf{MCSP}[n^c]}$. Using the fact that r is a pps, we have that $L \in \mathsf{NP}$. By assumption, there are infinitely many N for which the fraction of strings y of length N in L is at least γ, hence L is not $1/N^{\omega(1)}$-sparse. Contradiction. ∎

The pps R given by the proof of Lemma 2 does not have a natural form, but as we observe in Section V, we can take R to be Extended Frege with circuit lower bound axioms, and this has implications our ability to show proof complexity lower bounds for Extended Frege.

Lemma 3. *If Rudich's Conjecture holds, then Rudich's Conjecture does not admit feasible propositional proofs.*

Proof: By Lemma 2, under the assumption that Rudich's Conjecture holds, there is a pps R such that for any k, there is a collection W of R-easy $1/N^k$-hitting tautologies against nondeterministic size N^k.

Assume, for the sake of contradiction, that Rudich's Conjecture admits feasible propositional proofs. This means that for each pps R and integer $d > 0$, there is a pps S, such that for all large enough N, for a $1 - o(1)$ fraction of strings y of length N, there are polynomial-sized S-proofs of $\mathsf{lb}_R(\mathsf{tt}(\mathrm{fn}(y), \lfloor \log(N) \rfloor^d), m^d)$, where $m = |\mathsf{tt}(f_n, n^d)|$.

Consider the pps S that corresponds to the pps R from Lemma 2, with d chosen to be the constant c from Rudich's Conjecture, as in the proof of Lemma 2. We have that for large enough N, tautologies from W of size N have R-proofs of size at most N, and hence of size at most N^c, assuming wlog (as in the proof of Lemma 2) that $c > 1$. On the other hand, even though $f(y)$ is a tautology with all but negligible probability for y a randomly chosen N-bit string, it follows from the fact that W is a collection of hitting tautologies that there are N^c-size R-proofs for $f(y)$ with probability $o(1)$.

Now we use the assumption about S to contradict the hitting property of W. Let b be a constant such that for a $1 - o(1)$ fraction of strings y of length N, there are r^b-size S-proofs of $\mathsf{lb}_R(\mathsf{tt}(\mathrm{fn}(y), \lfloor \log(N) \rfloor^c), m^c)$, where $m = |\mathsf{tt}(f_n, n^c)|$, and $r = |\mathsf{lb}_R(\mathsf{tt}(\mathrm{fn}(y), \lfloor \log(N) \rfloor^c), m^c)|$. Let a be a constant such that $r^b < N^a$ - such a constant exists because m is bounded by a fixed polynomial in N and r is bounded by a fixed polynomial in m. Let q be a constant such that $S(u, v)$ is decidable in time at most $(|u| + |v|)^q$ for large enough u, v, and let k be a constant such that $k > aq$.

Define the language L_S as follows: $y \in L_S$ iff there are N^a-size S-proofs of $\mathsf{lb}_R(\mathsf{tt}(\mathrm{fn}(y), \lfloor \log(N) \rfloor^c), m^c)$. By assumption on S, for large enough N, at least a $1 - o(1)$ fraction of strings y of length N are in L_S. Moreover, by our choice of the parameter k, L_S is decidable in $\mathsf{NSIZE}(N^k)$. This can be done simply by computing the formula $\mathsf{lb}_R(\mathsf{tt}(\mathrm{fn}(y), \lfloor \log(N) \rfloor^c), m^c)$, guessing an N^a size S-proof for it, and then verifying that the proof is correct with circuits of size at most N^k.

Now we use the collection W of R-easy $1/N^k$-hitting tautologies against nondeterministic size N^k given by Lemma 2 to derive a contradiction. As in the proof of Lemma 2, let $\{H'_N\}$ be the sequence of hitting sets associated with W, and let $f(x) = \mathsf{tt}(\mathrm{fn}(x), \lfloor \log(|x|) \rfloor^c)$ be the poly-time computable function associated with W. Since W is $1/N^k$-hitting against nondeterministic size N^k, and L_S is a $1 - o(1)$-dense set computable by nondeterministic circuits of size at most N^k, there must be an infinite sequence of strings $\{y_N\}$, where each $y_N \in H'_N$, such that for large enough N, $y_N \in L_S$. But since W is R-easy, we have that for large enough N, $\mathsf{lb}_R(f(y_N), |f(y_N)|^c)$ is false, and by soundness of S, it does not have polynomial-size S-proofs, or indeed S-proofs of any size. This contradicts the inference that y_N is in L_S for large enough S, since for each large enough string $z \in L_S$, there are polynomial-sized S-proofs of $\mathsf{lb}_R(f(z), |f(z)|^c)$. ∎

The following is a restatement of Theorem 1.

Theorem 5. *Rudich's Conjecture does not admit feasible propositional proofs.*

Proof: Suppose Rudich's Conjecture holds. Then Lemma 3 gives us the desired conclusion.

Now suppose Rudich's Conjecture fails, and that Rudich's Conjecture admits feasible propositional proofs. We show that the failure of Rudich's conjecture implies that there exists a pps R with polynomially bounded advice such that for each $d > 0$ and infinitely many n, for an $\Omega(1)$ fraction of Boolean functions $f \in \mathcal{F}_n$, there are polynomial-size R-proofs of $\mathsf{tt}(f, n^d)$. We then show how to use the pps R with advice to contradict the assumption that Rudich's Conjecture admits feasible propositional proofs.

If Rudich's Conjecture fails, then for all constants $c > 0$, there is a constant $a > 0$ and a language L_c such that $L_c \subseteq \overline{\mathsf{MCSP}[n^c]}$ and $L_c \in \mathsf{NSIZE}(\mathrm{poly})$, but L_c is $1/N^a$-dense for infinitely many input lengths N. We show that in this case, it is also true that for all constants $c' > 0$, there is a constant $\gamma > 0$ and a language L' such that $L' \subseteq \overline{\mathsf{MCSP}[n^{c'}]}$ and $L' \in \mathsf{NSIZE}(poly)$, but L' is γ-dense for infinitely many input lengths N'.

L' is defined only on input lengths of the form $N' = N^{a+1}$, where N is a large enough positive integer. Let y be a given input of length N' for such an input length N'. L' interprets its input y as a sequence of N^a consecutive input blocks $y_i, i = 1 \ldots N^a$, where each y_i has length N. L' accepts y iff $L_{c'+1}$ accepts y_i for some input block y_i.

We argue that for each large enough input length y, if $y \in L'$, $\mathrm{fn}(y)$ does not have Boolean circuits of size $(n')^{c'}$, where $n' = \lfloor \log(N') \rfloor$. Indeed, such a y is in L' iff y_i is in L for some i, but this implies that $\mathrm{fn}(y_i)$ does not have Boolean circuits of size $n^{c'+1}$, by the assumption about L, where $n = \lfloor \log(N) \rfloor$. Since y is the concatenation of N^a equal size blocks y_i, this implies that $\mathrm{fn}(y)$ does not have Boolean circuits of size $n^{c'+1}$, which is at least $(n')^{c'}$ for large enough n, as $n' = O(n)$.

Next we argue that L' is γ-dense for infinitely many N', for some constant $\gamma > 0$. Indeed, for any large enough N' of the form N^{a+1} such that $L_{c'+1}$ is $1/N^a$-dense on inputs of length N, we have that the fraction of strings y of length N' in L' is at least $1 - (1 - 1/N^a)^{N^a} \geq 1 - 1/e$. This is because $y \in L'$ iff some block $y_i \in L$, and each block y_i independently has probability at least $1/N^a$ of being in L.

Finally we argue that $L' \in \mathsf{NSIZE}(\text{poly})$. Indeed, this follows directly from the fact that $L \in \mathsf{NSIZE}(poly)$, and that deciding membership of $y \in L'$ reduces to breaking y up into blocks y_i and checking if at least one y_i is in L. Indeed, if L has nondeterministic circuits of size at most N^q for some constant q, L' has nondeterministic circuits of size at most $(N')^{1+q/a}$.

Thus we have established our claim about the existence of L' corresponding to each constant $c' > 0$. We fix such an L' as follows: choose a $c > 3$ for which $L_c \in \mathsf{NSIZE}(N^q)$ for some q, and assume wlog that L_c is $1/N^a$-dense for some $a \geq q$. Let L' be the language given by the above construction corresponding to $c' = c - 1$. Then we have that $L' \subseteq \overline{\mathsf{MCSP}[n^{c-1}]}$ and L' has nondeterministic circuits of size at most $(N')^2$. Moreover L' is $1 - 1/e$-dense for infinitely many input lengths N'. Since L' has nondeterministic circuits of size at most $(N')^2$, there is a nondeterministic machine M running in time at most $(N')^2 \operatorname{polylog}(N')$ and using at most $(N')^2 \operatorname{polylog}(N)$ bits of advice deciding L'.

Now we define a pps R with advice corresponding to L'. R has at most N^2 bits of advice, and is defined as follows: $R(x, y, z) = 1$ iff (a) $y = 1^{2^{|x|}}$ and $x \in \mathsf{TAUT}$, or (b) if x is of the form $\mathsf{tt}(\text{fn}(u), n^2)$ for some string u such that $n = \lfloor \log(|u|) \rfloor$ and M accepts u with witness y and advice z.

Let $\{w_m\}$ be a good sequence of advice strings corresponding to the correct advice for the nondeterministic machine M. We claim that R is a pps such that $\{w_m\}$ is good for R. Indeed, any tautology ϕ has a trivial truth-table proof in R, even without advice. To argue soundness, note that in case (b), if M accepts u with witness y and correct advice z, then $u \in L'$ and therefore $\text{fn}(u)$ does not have circuits of size $n^{c-1} \geq n^2$. Hence $\mathsf{tt}(\text{fn}(u), n^2)$ is indeed a tautology.

We claim that for infinitely many N, for at least a fraction at least $1-1/e$ of strings u of length N, $\mathsf{tt}(\text{fn}(u), n^2)$ has R-proofs of size at most m^2, where $m = |\mathsf{tt}(\text{fn}(u), n^2)$. Indeed, L' is $1-1/e$-dense for infinitely many N, and for each such N, the nondeterministic machine M accepts u with some witness y and good advice w_m in quadratic time, giving quadratic size R-proofs of $\mathsf{tt}(\text{fn}(u), n^2)$, as per definition of the pps R.

Now we use the assumption that Rudich's Conjecture admits feasible propositional proofs. This implies that for every pps R with polynomial advice, there is a pps S, such that if $\{w_m\}$ is a sequence of poly-sized advice strings good for R, then for all large enough n, for a $1 - o(1)$ fraction of Boolean functions $f_n \in \mathcal{F}_n$, there are polynomial-sized S-proofs of $\mathsf{lb}_R(\mathsf{tt}(f_n, n^2), m^2, w_m)$, where $m = |\mathsf{tt}(f_n, n^d)|$. Let R be the pps defined above. Then by the soundness of the corresponding pps S given by the assumption that Rudich's Conjecture admits feasible propositional proofs, it follows that for all large enough n, for a $1 - o(1)$ fraction of Boolean functions $f_n \in \mathcal{F}_n$, $\mathsf{lb}_R(\mathsf{tt}(f_n, n^2), m^2, w_m)$ is true, and hence for all large enough N, for a $1 - o(1)$ fraction of strings u of length N, $\mathsf{tt}(\text{fn}(u), n^2)$ does not have quadratic size R-proofs with good advice w_m, contradicting the statement at the end of the previous para. ■

In fact, the proof of Theorem 5 gives that Rudich's Conjecture does not even admit feasible *non-uniform* propositional proofs, i.e., the impossibility result extends to ppses S that use polynomial advice.

Lemma 3 implies that there is a proof complexity-theoretic analogue of the Razborov-Rudich "natural proofs" barrier for some pps R under Rudich's conjecture. Namely, there is a samplable sequence of distributions under which a random formula is a tautology that requires large R-proofs, yet no pps S can prove this efficiently.

The following is a restatement of Corollary 1.

Corollary 2. *If Rudich's Conjecture holds, then there is a pps R and a samplable sequence of distributions $\{D_N\}$ on formulas of length* $\text{poly}(N)$*, such that with probability $1 - 1/N^{\omega(1)}$ over ϕ_N sampled from D_N, ϕ_N is a tautology that does not have* $\text{poly}(N)$ *size R-proofs, but there is a constant k such that no pps S has polynomial-size proofs of* $\mathsf{lb}_R(\phi_N, |\phi_N|^k)$.

Proof: This follows from Lemma 3, by using the same pps R as in the proof of the Lemma, and the samplable sequence of distributions $\{D_N\}$ defined by choosing $f_n \in \mathcal{F}_n$ at random and outputting the formula $\mathsf{tt}(f_n, n^k)$. ■

IV. Implausibility of Feasibly Proving Nondeterministic Feige's Hypothesis

The following lemma giving an average-case reduction from SAT to MCSP is based on Theorem 38 in [25], but the proof is slightly different, as we need to be careful about the output length of the reduction.

Lemma 4. *For any large enough integer $\Delta > 0$, there is a polynomial-time computable function $g : \{0,1\}^* \to \{0,1\}^*$ satisfying the following conditions whenever N is large enough and a power of* 2*:*

1) *For any 3-CNF ϕ on N variables with ΔN clauses,* $|g(\phi)| = 3\Delta N(\log(2N))$
2) *g is poly-time invertible*
3) *For at least a $(1-o_N(1))$ fraction of strings y of length KN, $y = g(\phi)$ for some 3-CNF ϕ on N variables with ΔN clauses*

4) *If ϕ is a satisfiable 3-CNF on N variables with ΔN clauses, then $KT(g(\phi)) \leq |g(\phi)| - \Delta N/12$*

Proof: When N is a power of 2, picking a formula ϕ from $U_{\Delta,N}$ corresponds to picking a string z_ϕ of length $3\Delta N(\log(2N))$ at random - imagine z to be made up of ΔN equal size blocks each of length $3\log(2N)$, and each block to represent a sequence of 3 literals, each described by $\log(2N)$ bits. We simply set $g(\phi) = z_\phi$. The first condition is obviously satisfied. The second condition follows from the bijection between 3-CNFs over N variables with ΔN clauses and strings of length $3\Delta N(\log(2N)$ described above. The third condition follows from the simple form of the bijection.

For the fourth condition, we show that for any satisfiable 3-CNF ϕ, we can represent z_ϕ by a string z'_ϕ of length $3\Delta N\log(2N)-\Delta N/6+O(1)$ together with an N-bit string x encoding a satisfiying assignment to ϕ, such that $g(\phi)$ can be recovered quickly from these two strings.

We now think of the formula ϕ as composed of segments of 6 clauses each (with possibly a few clauses left over at the end). Each segment of 6 clauses $C_1 \ldots C_6$ is represented in z'_ϕ by the sequence of variables occurring in each clause (which costs $3\log(N)$ bits per clause, hence $18\log(N)$ bits in total), together with a string in $[7]^6$, where the ith character in the string represents the position of the clause C_i in the lexicographic ordering of all clauses on the sequence of variables in C_i which are satisfied by x. Since $6\lceil\log(7)\rceil < 6\lceil\log(8)\rceil$, this allows us to save 1 bit in our representation for each segment, and hence $\Delta N/6-O(1)$ bits in total. Moreover, the jth clause in ϕ can be reconstructed in time $O(\log(N))$ from z'_ϕ together with x, simply by identifying the segment corresponding to the jth clause, and then using x and the relevant portion of z'_ϕ of size $O(\log(N))$ to determine whether each of the literals in the jth clause occurs complemented or uncomplemented. This involves reading at most $O(\log(N))$ bits from x, and can be done in $O(\log(N))$ time.

Thus the KT-complexity of $g(\phi)$ for satisfiable ϕ is at most $3\Delta N\log(2N)-\Delta N/6+O(1)+N+O(\log(N)$, which is at most $3\Delta N\log(2N)-\Delta N/12$ when Δ and N are large enough. ■

Lemma 5. *If Nondeterministic Feige's Hypothesis has feasible propositional proofs, then there is $\gamma > 0$ such that the* $\mathsf{MKTP}[N-\gamma N/\log(N)]$ *Hardness Hypothesis admits feasible propositional proofs.*

Proof: Let $\Delta > 0$ be a large enough integer for the proof of Lemma 4 to go through, and let $\gamma < 1/(200\Delta)$ be any constant. Suppose that for every pps R' with polynomial advice and every $d > 0$, there is a pps S' such that for all large enough n, with probability $1-o(1)$ over ϕ sampled from $U_{\Delta,n}$, there are polynomial-sized S'-proofs of $\mathsf{lb}_{R'}(\overline{\phi}, m^d, w_m)$, where $m = |\overline{\phi}|$ and w_m is a good advice string for R' at length m.

We show that for every pps R with polynomial advice and every $d > 0$, there is a pps S such that for all large enough N, with probability $1-o(1)$ over $y \sim U_N$, there are polynomial-size S-proofs of $\mathsf{lb}_R(\mathsf{random}(y, N-\gamma N/\log(N)), m^d, w_m)$, where $m = |\mathsf{random}(y, N-\gamma N/\log(N))|$ and w_m is a good advice string for R at length m.

Given a string y of length N, define $trunc(y)$ as the largest prefix of y of length $3\Delta N'\log(2N')$ for some N' that is a power of 2. Note that $|trunc(y)| \geq |y|/4$.

Fix a pps R with good advice $\{w_m\}$ satisfying the criterion in the previous para. We define a pps R' with polynomial advice as follows. $R'(u,v,w) = 1$ iff either (a) $v =< v', y' >$ where v' is an R-proof of $\mathsf{random}(y, N-\gamma N/\log(N))$ for large enough N with advice w_m (where $m = |\mathsf{random}(y, N-\gamma N/\log(N))|$), for $y = g(\overline{u})y'$ and $g(\overline{u}) = trunc(y)$ (where g is the poly-time computable function from Lemma 4), and moreover w is the concatenation of advice strings w_m for all m corresponding to y such that $g(\overline{u}) = trunc(y)$ (b) $v = 1^{2^{|u|}}$ and u is a tautology.

We argue that R' is indeed a pps with polynomial advice. Completeness follows from item (b) of the definition. We show that soundness follows from soundness of R and Lemma 4. Indeed, item (b) of the definition never causes issues with soundness, so we only need to argue about item (a). Suppose $v =< v', y' >$ where v' is an R-proof of $\mathsf{random}(y, N-\gamma N/\log(N))$, and moreover we have $y = g(\overline{u})y'$ and $g(\overline{u}) = trunc(y)$. Since R is sound and w_m is good advice for R, we have that $\mathsf{KT}(y) > N-\gamma N/\log(N)$. Since $y = g(\overline{u})y'$ and $g(\overline{u}) = trunc(y)$, we have that $|g(\overline{u})| \geq |y|/4$. Since $\mathsf{KT}(y) > N-\gamma N/\log(N)$, it must be case that $\mathsf{KT}(g(\overline{u})) > |g(\overline{u})| - 5\gamma|g(\overline{u})|/\log(|g(\overline{u})|)$, since otherwise we could compress y to contradict the lower bound on its KT-complexity by giving y' explicitly together with the compressed representation of $g(\overline{u})$ corresponding to its presumed small KT-complexity.

Let N' be such that $|g(\overline{u})| = 3\Delta N'\log(2N')$. Since $\mathsf{KT}(g(\overline{u})) > |g(\overline{u})| - \gamma|g(\overline{u})|/\log(|g(\overline{u})|)$, we have that $\mathsf{KT}(g(\overline{u})) > |g(\overline{u})| - 16\gamma\Delta N'$. Thus $\mathsf{KT}(g(\overline{u})) > |g(\overline{u})| - N'/12$, using the fact that $\gamma\Delta < 1/200$, and this implies by Lemma 4 that $\overline{u}$ is unsatisfiable. Hence u is a tautology, establishing the soundness of item (a).

Finally, we need to argue polynomial-time decidability of R. Checking item (b) can clearly be done in polynomial time. This is also the case with item (a), since y' and y can be computed in polynomial time from u and v, and since R is polynomial-time decidable. Moreover, we need to check that w_m occurs as a substring of w in the appropriate location, but this is easy to do.

Now note that if $\mathsf{random}(y, N-\gamma N/\log(N))$ has polynomial-size R'-proofs for y such that $trunc(y) = g(\overline{\phi})$, then ϕ has polynomial-size R-proofs. This follows from item (a) of the definition of R'. Let c be a constant such that if

$\mathsf{random}(y, N-\gamma N/\log(N)$ has m^d-size R-proofs for large enough N, then $\overline{\phi}$ has ℓ^{cd}-size R'-proofs for large enough $\ell = |\phi|$ when $trunc(y) = g(\overline{\phi})$.

We want to show that for every pps R with advice and every $d > 0$, there is a pps S such that for all large enough N, with probability $1 - o(1)$ over $y \sim U_N$, there are polynomial-size S-proofs of $\mathsf{lb}_R(\mathsf{random}(y, N - \gamma N/\log(N)), m^d, w_m)$, where $m = |\mathsf{random}(y, N - \gamma N/\log(N))|$ and w_m is a good advice string for R at length m. Let S' be a pps such that for all large enough n, with probability $1 - o(1)$ over ϕ sampled from $U_{\Delta,n}$, there are polynomial-size S'-proofs of $\mathsf{lb}_{R'}(\overline{\phi}, \ell^{cd}, w_\ell)$, where $\ell = |\overline{\phi}|$.

We define S based on S' as follows. $S(u, v) = 1$ iff either (a) $u = \mathsf{lb}_R(\mathsf{random}(y, N - \gamma N/\log(N)), m^d, w_m)$ for some y of length N and $m = |\mathsf{random}(y, N - \gamma N/\log(N))|$, and moreover v is an S'-proof of $\mathsf{lb}_{R'}(\overline{g^{-1}(trunc(y))}, \ell^{cd}, w)$ for some w that contains w_m as a substring in the appropriate location (note that a good advice string w for R' is the concatenation of all good advice strings w_m for m = $|\mathsf{random}(y, N - \gamma N/\log(N))|$, where $trunc(y) = g(\phi)$), or (b) $v = 1^{2^{|u|}}$ and u is a tautology.

Completeness of S follows immediately from item (b) of the definition of S. Soundness follows the fact that if R' does not have ℓ^{cd}-size proofs that ϕ is a tautology, then R does not have m^d-size proofs that $y = g(\overline{\phi})y'$ is KT-random for any y such that $g(\overline{\phi}) = trunc(y)$. Polynomial-time verifiability of S is clear from the definition.

We still need to argue that for all large enough N, with probability $1 - o(1)$ over $y \sim U_N$, there are polynomial-size S-proofs of $\mathsf{lb}_R(\mathsf{random}(y, N - \gamma N/\log(N)), m^d, w_m)$ where w_m is good advice for R. We have by assumption that for all large enough n (and in particular for n a power of two), with probability $1 - o(1)$ over ϕ chosen from $U_{n,\Delta}$, there are polynomial-size S'-proofs of $\mathsf{lb}_{R'}(\overline{\phi}, \ell^{cd}, w_\ell)$. But this follows by item (a) of the definition of S since the polynomial-size S'-proofs v of $\mathsf{lb}_{R'}(\overline{g^{-1}(trunc(y))}, \ell^{cd}, w)$ also function as polynomial-size S-proofs of $\mathsf{lb}_R(\mathsf{random}(y, N - \gamma N/\log(N)), m^d, w_m)$, and by the third item of Lemma 4, this happens for at least a $1 - o(1)$ fraction of strings y of length N for N large enough. ■

Theorem 6. *For each $1/2 > \gamma > 0$, there is $\delta > 0$ such that if the $\mathsf{MKTP}[\delta N/\log(N)]$ hardness hypothesis holds, then the $\mathsf{MKTP}[N - \gamma N/\log(N)]$ hardness hypothesis does not admit feasible propositional proofs.*

Proof: The proof is analogous to the proof of Lemma 3, and how it builds on Lemma 2. Hence, rather than giving all details, we explain the structure of the proof, and describe the aspects in which the proof differs from the proofs of Lemma 2 and Lemma 3.

Given $\gamma > 0$, we choose $\delta > 0$ to be any constant less than γ.

As in the proof of Lemma 2, the hardness hypothesis is used to construct a collection of hitting tautologies. For each $\epsilon > 0$ and each positive integer k, we will get a collection W of R-easy ϵ-hitting tautologies against nondeterministic size N^k for some pps R, as follows.

Using the $\mathsf{MKTP}[\delta N/\log(N)]$ hardness hypothesis in Proposition 6, for each $\epsilon > 0$ and positive integer $k > 0$, we get a sequence $\{H_N\}$ of ϵ-hitting sets supported on N-bit strings of KT-complexity at most $\delta N/\log(N)$ against nondeterministic size N^k.

We define a sequence $\{z_N\}$ of N-bit strings as follows: z_N is the lexicographically first N-bit string such that $KT(z_N) \geq N - 1$. The existence of z_N for every N follows from Proposition 4. Now we define a new sequence $\{H'_N\}$ of ϵ-hitting sets against nondeterministic size N^k by including y of length N in H'_N iff $y \oplus z_N$ is in H_N. The argument that H'_N is a hitting set is exactly the same as in the proof of Lemma 2.

Define the poly-time computable function $f : \{0,1\}^* \to \{0,1\}^*$ by $f(x) = \mathsf{random}(x, N - \gamma N/\log(N)$. Let $W = \bigcup_N f(H'_N)$. W is our candidate set of ϵ-hitting tautologies against nondeterministic size N^k.

Condition (i) in Definition 11 is satisfied trivially. Condition (ii) follows from the fact that for strings x and x', $\mathsf{KT}(x \oplus x') \leq \mathsf{KT}(x) + \mathsf{KT}(x') + O(\log(|x|))$. Condition (iv) follows from the second item of Proposition 4.

We next define a pps R with polynomial advice such that W is R-easy. R is defined exactly as in the proof of Lemma 2, except that linear advice is needed to encode z_N. The proof that R is a pps with advice is along the same lines of before.

To argue condition (iii) in Definition 11, note that any tautology in W of the form $\mathsf{random}(y \oplus z_N, N - \gamma N/\log(N)$ for $y \in H_N$ has a short proof given by the program witnessing that y has KT-complexity at most $\delta N/\log(N)$. Regarding condition (v), this follows from the $\mathsf{MKTP}[N - \gamma N/\log(N)]$ Hypothesis exactly as condition (v) in the proof of Lemma 2 follows from Rudich's Conjecture. Now note that the $\mathsf{MKTP}[\delta N]/\log(N)$ Hypothesis implies the $\mathsf{MKTP}[N - \gamma N/\log(N)]$ Hypothesis for any constants δ and γ.

The way that the set W of ϵ-hitting tautologies is used to argue that there are no feasible proofs of $\mathsf{MKTP}[N - \gamma N/\log(N)]$ Hardness Hypothesis is exactly analogous to the proof of Lemma 3. ■

The following is a restatement of Theorem 2.

Corollary 3. *Under Rudich's Conjecture, Nondeterministic Feige's Hypothesis does not have feasible propositional proofs.*

Proof: Suppose, for the sake of contradiction, that Nondeterministic Feige's Hypothesis has feasible propositional proofs. By Lemma 5, there is a constant $\gamma > 0$ such that the $\mathsf{MKTP}[N-\gamma N/\log(N)]$ Hardness Hypothesis has feasible propositional proofs. By Proposition 5, Rudich's Conjecture implies the $\mathsf{MKTP}[\delta N/\log(N)]$ Hardness Hypothesis for every $\delta > 0$. Now by applying Lemma 5 for the $\delta > 0$ in the statement of the Lemma, we have that the $\mathsf{MKTP}[N-\gamma N/\log(N)]$ Hardness Hypothesis does not have feasible propositional proofs. Contradiction. ∎

V. A LOSE-LOSE THEOREM FOR EXTENDED FREGE

Lemma 6. *Assume there is an s-size EF-proof of $\mathsf{tt}(f_n, 3n^k)$ for some k and sufficiently large n. Further, let g_n be a function computable by a circuit of size n^k. Then there is a $2s + K2^n$-size EF-proof of $\mathsf{tt}(f_n \oplus g_n, n^k)$ where K is an absolute constant.*

Proof: We will use the Extended Resolution proof system to argue the lemma instead, taking advantage of the standard fact that Extended Resolution and Extended Frege simulate each other [18].

Suppose a set of clauses S_0 expresses $C_0 = f_n \oplus g_n$ for a circuit C_0 of size n^k, i.e. S_0 is a negation of $\mathsf{tt}(f_n \oplus g_n, n^k)$, and a set of clauses S_1 expresses $C_1 = g_n$ where C_1 is a circuit of size n^k coded by variables not occurring in S_0. Let u_i^a be the variable encoding the output of C_i on input a. Further, for each n-bit string a let y^a be a new variable, and let Y be a set of clauses expressing that

$$y^a = u_0^a \oplus u_1^a.$$

The set Y can contain extra variables that help to encode $u_0^a \oplus u_1^a$.

From $S_0 \cup S_1 \cup Y$ we can derive a set of clauses $S_\oplus$ expressing $C_0 \oplus C_1 = f_n$ where $C_0 \oplus C_1$ is a circuit of size $3n^k$ outputting y^a on input a. This derivation takes $K2^n$ steps, for an absolute constant K. Specifically, for each a, it derives $f_n(a) = u_0^a \oplus u_1^a = y^a$ from $u_0^a = f_n(a) \oplus g_n(a)$ and $u_1^a = g_n(a)$. Since we assume that we can refute $\mathsf{tt}(f_n, 3n^k)$ in size s and substituting some variables for constants does not increase the refutation size, we can refute $S_\oplus$ in size s, and $S_0 \cup S_1 \cup Y$ in size $s + K2^n$. We can now substitute the n^k-size circuit computing g_n in S_1 and refute in size $s+K2^n$ the set of clauses $S_0 \cup Y'$ where Y' is obtained from Y by substituting the circuit for g_n. Note that the clauses in Y' say just that $y^a = u_0^a$ or $y^a = \neg u_0^a$. We can thus get rid of them with a substitution of literals which at most doubles the resulting refutation size. This yields a $2s + 2K2^n$-size refutation of S_0. ∎

In fact, Lemma 6 holds even with 'Extended Frege' replaced by 'Resolution', using essentially the same proof.

Theorem 7. *Assuming Rudich's Conjecture, at least one of the following is true:*

1) *There is no sequence of Boolean functions $\{f_n\}, f_n \in \mathcal{F}_n$, such that $\mathsf{tt}(f_n, n^k)$ has polynomial-size EF-proofs for every $k > 0$*
2) *There is no pps Q such that there are polynomial-size Q-proofs of $\mathsf{lb}_{\mathsf{EF}}(\mathsf{tt}(f_n, n^k), m^k)$ (where $m = |\mathsf{tt}(f_n, n^k)|$) for a $1/2^{O(n)}$ fraction of Boolean functions f_n for all constants k and all large enough n.*

Proof: Suppose that the first item is false, and that there is a sequence of Boolean functions $\{f_n\}, f_n \in \mathcal{F}_n$, such that $\mathsf{tt}(f_n, n^k)$ has polynomial-size EF proofs for every $k > 0$. Assume Rudich's Conjecture. Then, using the falsehood of the first item and Lemma 6, the pps R in the statement of Lemma 2 can be taken to be EF. It then follows that Corollary 2 holds with $R = \mathsf{EF}$, and this implies the second item. ∎

The following is a more rigorous formulation of Theorem 3.

Theorem 8. *Assume that Rudich's Conjecture holds, and that there is an $\epsilon > 0$ such that $\mathsf{E} \not\subseteq \mathsf{io}-\mathsf{NSIZE}(2^{\epsilon n})$. Then at least one of the following is true:*

1) *There is no sequence of Boolean functions $\{f_n\}, f_n \in \mathcal{F}_n$, such that $\mathsf{tt}(f_n, n^k)$ has polynomial-size EF-proofs for every $k > 0$*
2) *For each $k > 0$ and each pps Q, there is a polynomial-time algorithm that on input 1^N (where $N = 2^n$), outputs a set S_N of truth tables of Boolean functions on n bits such that for each large enough N, there is $F \in S_N$ such that $\mathsf{tt}(\mathrm{fn}(F), n^k)$ is a tautology, there are no N^k-size EF-proofs of $\mathsf{tt}(\mathrm{fn}(F), n^k)$, and moreover $\mathsf{lb}_{\mathsf{EF}}(\mathsf{tt}(\mathrm{fn}(F), n^k), N^k)$ does not have Q-proofs of size N^{k^2}.*

Proof: The basic idea is to derandomize the hard tautologies in Theorem 7 by using the assumption that $\mathsf{E} \not\subseteq \mathsf{io}-\mathsf{NSIZE}(2^{\epsilon n})$ in Theorem 4. Assume the first item is false, and let d be a constant such that the pps $Q(x, y)$ is verifiable in time $(|x + y|)^d$.

By a standard counting argument, we have that for most F of size N, when N is large enough, $\mathsf{tt}(\mathrm{fn}(F), n^k)$ is a tautology. Since Rudich's Conjecture is true, we have that for most F of size N, when N is large enough, there are no N^k-size EF proofs of $\mathsf{tt}(\mathrm{fn}(F), n^k)$. By the falsity of the first item, and applying Theorem 7, we have that for most F of size N, there are no polynomial-size Q-proofs of $\mathsf{lb}_{\mathsf{EF}}(\mathsf{tt}(f_n, n^k), m^k)$ (where $m = |\mathsf{tt}(f_n, n^k)|$), when N is large enough.

We use the PRG given by Theorem 4 to simultaneously derandomize all three of these conditions, using the fact that each of them can be tested, on input F, by a fixed polynomial-size co-nondeterministic circuit. Indeed, the first condition can be tested by a co-nondeterministic circuit of size $\tilde{O}(Nn^k)$ which simply guesses a circuit of size n^k and verifies that it does not compute F correctly. The second condition can be tested by a co-nondeterministic circuit that uses the verification algorithm for EF and has size $\tilde{O}(N^k)$.

The third condition can be tested by a co-nondeterministic circuit that uses the verification algorithm for Q and has size $\tilde{O}(N^{dk^2})$.

Applying Theorem 4, there is a complexity-theoretic $1/N^{2dk^2}$-PRG with seed length $O(\log(N))$ against nondeterministic size N^{2dk^2}. Let this PRG be $\{G_N\}$ and let S_N be the range of $G_{N^{2dk^2}}$. Clearly S_N has size $\mathrm{poly}(N)$, and by a union bound, most F in S_N satisfy all three conditions above, which proves the theorem. ■

VI. THE FEASIBLE COOK-RECKHOW PROGRAM AND OPTIMALITY

We observe that if we reason inside a proof system P, it is impossible to prove a lower bound on a stronger proof system Q. This is a straightforward consequence of a seminal work of Krajíček and Pudlák [34].

Theorem 9 (es. Krajíček and Pudlák [34]). *For any propositional proof systems P, Q simulating EF, P does not admit p-size proofs of $\mathsf{lb}_Q(\phi_n, n^{\log n})$ for any sequence of propositional formulas ϕ_n of length n unless every sequence of tautologies with p-size Q-proofs admits $n^{O(\log n)}$-size P-proofs.*

Proof sketch: Let ψ_n be a sequence of tautologies with p-size Q-proofs. If P proves efficiently $\mathsf{lb}_Q(\phi_n, n^{\log n})$ for some ϕ_n , P proves ψ_n in size $n^{O(\log n)}$. The P-proof proceeds as follows. If $\neg\psi_n(a)$ held for some a, it could be combined with a p-size Q-proof of ψ_n to get a contradiction and a p-size Q-proof of ϕ_n. ■

We consider consequences of the failure of the Feasible Cook-Reckhow Program in a strong sense. This can be linked to the existence of a certain variant of optimal proof systems.

Definition 16 (i.o. optimal proof system). *A proof system P is i.o. optimal if every proof system Q is i.o. simulated by P, i.e. there is a polynomial p such that for every sequence of tautologies ϕ_n with Q-proofs of length s_n, infinitely many ϕ_ns admit P-proofs of length $p(s_n)$.*

Failure of Feasible Cook-Reckhow Program. *There exists a proof system P such that for every proof system Q, every sequence of tautologies ϕ_n there exists a constant k such that Q does not admit p-size proofs of $\mathsf{lb}_P(\phi_n, n^k)$ for every sufficiently big n.*

Theorem 10. *Failure of the Feasible Cook-Reckhow Program implies the existence of an i.o. optimal proof system.*

Proof: Assume there is a proof system P that witnesses the failure of the Feasible Cook-Reckhow Program. We will show that P i.o. simulates every proof system. W.l.o.g. P simulates EF since otherwise we could consider an extension of P by EF. It is known that every proof system Q can be simulated by EF extended with axioms Ref_Q encoding the so called reflection principle for Q, cf. [31, Theorem 8.4.3.]. Specifically, $Ref_Q(\pi, \phi, y)$ is a tautology with free variables encoding π, ϕ, y which states that if π is a Q-proof of a formula ϕ, then ϕ is satisfied by y.

Therefore, it suffices to show that for every system Q, proof system P admits p-size proofs of $Ref_Q(\pi, \phi_m, y)$ for infinitely many m. If this was not the case, then for every k and for all sufficiently big n denoting the length of Ref_Q, formula $\mathsf{lb}_P(Ref_Q, n^k)$ would be a tautology. Moreover, there would be a p-time algorithm A which given n outputs the tautology. This yields a contradiction with the Failure of Feasible Cook-Reckhow Program because A could be used to define an extension of EF admitting p-size proofs of $\mathsf{lb}_P(Ref_Q, n^k)$ for every sufficiently big n. ■

We also explore consequences of switching the order of quantifiers in the hypothesis above about failure of the Feasible Cook-Reckhow Program.

Definition 17 (Optimal proof system). *A proof system P is optimal if every proof system Q is simulated by P, i.e. there is a polynomial p such that for every sequence of tautologies ϕ_n with Q-proofs of length s_n, each ϕ_ns has P-proofs of length $p(s_n)$.*

Theorem 11. *If optimal proof systems do not exist, then for every proof system Q, there exists a proof system P such that for every sequence of tautologies ϕ_n there exists a constant k such that Q does not admit p-size proofs of $\mathsf{lb}_P(\phi_n, n^k)$ for every sufficiently big n.*

Proof: Let Q be an arbitrary proof system and Q' be a proof system simulating both Q and EF. Since Q' is not optimal, there is a proof system P such that Q' does not have p-size proofs of Ref_P. By an argument similar to the proof of Theorem 9, this implies that for every sequence of tautologies ϕ_n there is a constant k such that Q' (and hence also Q) does not have p-size proofs of formulas $\mathsf{lb}_P(\phi_n, n^k)$. ■

ACKNOWLEDGMENTS

We thank Udi Hrushovski, Igor Oliveira and Iddo Tzameret for useful conversations. Anonymous reviewers provided helpful feedback on an earlier version of this manuscript.

Both authors were supported in part by the European Research Council under the European Union's Seventh Framework Programme (FP7/ 2007-2014)/ERC Grant Agreement No. 615075. The second author was supported in part by Grant 19-05497S of GA ČR.

REFERENCES

[1] Aaronson S.; *Is P Versus NP Formally Independent*, Bulletin of the European Association of Theoretical Computer Science (EATCS), 81, pp. 109-136, 2003.

[2] Ajtai M.; Σ_1^1 *Formulae on Finite Structures*, Annals of Pure and Applied Logic, 24(1), pp. 1-48, 1983.

[3] Ajtai M.; *The Complexity of the Pigeonhole Principle*, Combinatorica, 14(4), pp. 417–433, 1994.

[4] Allender E.; *When Worlds Collide: Derandomization, Lower Bounds, and Kolmogorov Complexity*, Proceedings of 21st Conference on Foundations of Software Technology and Theoretical Computer Science, pp 1-15, 2001.

[5] Allender E., Buhrman H., Koucky M., van Melkebeek D., Ronneburger D.; *Power from Random Strings*, SIAM Journal on Computing, 35(6):1467-1493, 2006.

[6] Alekhnovich M., Ben-Sasson E., Razborov A.A., Wigderson A.; *Pseudorandom generators in propositional proof complexity*, SIAM Journal on Computing, 34(1):67-88, 2004.

[7] Baker T., Gill J., Solovay R.; *Relativizations of the* $\mathsf{P} = \mathsf{NP}$ *Question*, SIAM Journal on Computing, 4(4), pp. 431-442, 1975.

[8] Barak B.; *Truth vs. Proof in Computational Complexity*, Bulletin of the EATCS, 108, pp 130-142, 2002.

[9] Beame P., Impagliazzo R., Krajicek J., Pitassi T., Pudlak P., Woods A.; *Exponential Lower Bounds for the Pigeonhole Principle*, Proceedings of the 24th Annual ACM Symposium on Theory of Computing, pp. 200-220, 1992.

[10] Beame P., Pitassi T.; *Propositional Proof Complexity: Past, Present and Future*, Bulletin of the European Association of Theoretical Computer Science (EATCS), 65, pp. 66-89, 1998.

[11] Bellantoni S., Pitassi T., Urquhart A.; *Approximation and Small Depth Frege Proofs*, Proceedings of the Sixth Annual Structure in Complexity Theory Conference, pp. 367-390, 1991.

[12] Bonet M.L., Pitassi T., Raz R.; *On Interpolation and Automatization for Frege Systems*, SIAM Journal on Computing, 29(6), pp. 1939-1967, 2000.

[13] Buss S.; *On Model Theory for Intuitionistic Bounded Arithmetic with Applications to Independence Results*, Feasible Mathematics: A Mathematical Sciences Institute Workshop held in Ithaca, New York, 1989.

[14] Cook S.A.; *The Complexity of Theorem-Proving Procedures*, Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, pp. 151-158, 1971.

[15] Cook S.A.; *Feasibly constructive proofs and the propositional calculus*, Proceedings of the 7th Annual ACM Symposium on Theory of Computing, ACM Press, pp. 83-97, 1975.

[16] Cook S., Krajíček J.; *Consequences of the Provability of* $\mathsf{NP} \subseteq \mathsf{P/poly}$, Journal of Symbolic Logic, 72(4), 2007.

[17] Cook S., Pitassi T.; *A Feasibly Constructive Lower Bound for Resolution Proods*, Information Processing Letters, 34(2), pp. 81-85, 1990.

[18] Cook S.A., Reckhow R.A.; *The relative eciency of propositional proof systems*, Journal of Symbolic Logic, 44(1), pp.36-50, 1979.

[19] Cook S.A., Urquhart; *Functional interpretations of feasibly constructive arithmetic*, Annals of Pure and Applied Logic, 63:103-200, 1993.

[20] Feige U., *Relations between average case complexity and approximation complexity*, Proceedings of the 34th Annual ACM Symposium on Theory of Computing, pp. 534-543, 2002.

[21] Furst M., Saxe, J., Sipser M.; *Parity, Circuits and the Polynomial-Time Hierarchy*, Mathematical Systems Theory, 17(1), pp. 13-27, 1984.

[22] Grochow J., Pitassi T.; *Circuit Complexity, Proof Complexity, and Polynomial Identity Testing: The Ideal Proof System*, Journal of the ACM, 65(6), 37:1-37:59, 2018.

[23] Haken, A.; *The Intractability of Resolution*, Theoretical Computer Science, 39, pp. 297-308, 1985.

[24] Hastad J.; *Almost Optimal Lower Bounds for Small Depth Circuits*, Proceedings of the 18th Annual ACM Symposium on Theory of Computing, pp. 6-20, 1986.

[25] Hirahara S., Santhanam R.; *On the average-case complexity of MCSP and its variants*, Computational Complexity Conference, 2017.

[26] Klivans A., van Melkebeek D.; *Graph Nonisomorphism Has Subexponential Size Proofs Unless the Polynomial-Time Hierarchy Collapses*, SIAM Journal on Computing, 31(5), pp. 1501-1526, 2002.

[27] Krajíček J.; *On the weak pigeonhole principle*, Fundamenta Mathematicae, 170(1-3):123-140, 2001.

[28] Krajíček J.; *Dual weak pigeonhole principle, pseudorsurjective functions, and provability of circuit lower bounds*, Journal of Symbolic Logic, 69(1):265-286, 2004.

[29] Krajíček J.; *On the proof complexity of the Nisan-Wigderson generator based on a hard* $\mathsf{NP} \cap \mathsf{coNP}$ *function*, Journal of Mathematical Logic, 11(1):11-27, 2011.

[30] Krajíček J.; *On the computational complexity of finding hard tautologies*, Bulleting of the London Mathematical Society, 46(1):111-125, 2014.

[31] Krajíček J.; *Proof complexity*, Cambridge University Press, 2019.

[32] Krajíček J., Pudlák P.; *Propositional Provability and Models of Weak Arithmetic*, Computer Science Logic, LNCS 440,193-210, 1990.

[33] Krajíček J., Pudlák P.; *Some Consequences of Cryptographical Conjectures for* S_2^1 *and EF*, Information and Computation, 140(1), pp 82-94, 1998.

[34] Krajíček J., Pudlák P.; *Propositional Proof Systems, the Consistency of First Order Theories and the Complexity of Computations*, Journal of Symbolic Logic, 54(3):1063-1079, 1989.

[35] Lipton R.J., Young N.E.; *Simple strategies for large zero-sum games with applications to complexity theory*, Symposium on Theory of Computing, 1994.

[36] Müller M., Pich J.; *Feasibly constructive proofs of succinct weak circuit lower bounds*, preprint, 2017.

[37] Nisan N., Wigderson A.; *Hardness vs Randomness*, Journal of Computer and System Sciences, 49(2), pp. 149-167, 1994.

[38] O'Donnell, R.; *Personal Communication*, 2016.

[39] Razborov A.A.; *Lower Bounds for the Monotone Complexity of Some Boolean Functions*, Soviet Mathematics Doklady, 31, pp. 354-357, 1985.

[40] Razborov A.A.; *Lower Bounds on the Dimension of Schemes of Bounded Depth in a Complete Basis Containing the Logical Addition Function*, Mat. Zametski, 41, pp. 598–607, 1987.

[41] Razborov A.A.; *On provably disjoint NP-pairs*, Basic Research in Computer Science Center, 1994.

[42] Razborov A.A.; *Unprovability of lower bounds on circuit size in certain fragments of bounded arithmetic*, Izvestiya of the Russian Academy of Science, 59:201-224, 1995.

[43] Razborov A.A.; *Bounded Arithmetic and Lower Bounds in Boolean Complexity*, In Feasible Mathematics II, pp. 344-386, 1995.

[44] Razborov A.A.; *Pseudorandom generators hard for k-DNF resolution and polynomial calculus*, Annals of Mathematics, 181(2):415-472, 2015.

[45] Razborov A.A., Rudich S.; *Natural proofs*, Journal of Computer and System Sciences, 55(1):24-35, 1997.

[46] Rudich S.; *Super-bits, Demi-bits, and NP/qpoly-natural Proofs*, Journal of Computer and System Sciences, 55:204-213, 1997.

[47] Smolensky R.; *Algebraic Methods in the Theory of Lower Bounds for Boolean Circuit Complexity*, Proceedings of the 19th Annual ACM Symposium on Theory of Computing, pp. 77-82, 1987.

[48] Yao, A.C.; *Separating the Polynomial-Time Hierarchy by Oracles*, Proceedings of the 26th Annual Symposium on Foundations of Computer Science, pp. 1-10, 1985.