Electronic Colloquium on Computational Complexity, Report No. 47 (2023)

Ruling Out Short Proofs of Unprovable Sentences is Hard

Hunter Monroe

April 2, 2023

Abstract

If no optimal propositional proof system exists, we (and independently Pudlák) prove that ruling out length t proofs of any unprovable sentence is hard. This mapping from unprovable to hard-to-prove sentences powerfully translates facts about noncomputability into complexity theory. For instance, because proving string x is Kolmogorov random ($x{\in}R$) is typically impossible, it is typically hard to prove "no length t proof shows $x{\in}R$", or tautologies encoding this. Therefore, a proof system with one family of hard tautologies has these densely in an enumeration of families. The assumption also implies that a natural language is **NP**-intermediate: with R redefined to have a sparse complement, the complement of the language $\{\langle x, 1^t\rangle|$ no length t proof exists of $x{\in}R\}$ is also sparse.

Efficiently ruling out length t proofs of $x{\in}R$ might violate the constraint on using the fact of $x{\in}R$'s unprovability. We conjecture: any computable predicate on R that might be used in if-then statements (or case-based proofs) does no better than branching at random, because R appears random by any effective test. This constraint could also inhibit the usefulness in circuits and propositional proofs of NOT gates and cancellation—needed to encode if-then statements. If R defeats if-then logic, exhaustive search is necessary.

1 Introduction

We prove a deep linkage between noncomputability and complexity under a widely believed conjecture—that there is no optimal propositional proof

ISSN 1433-8092

system for tautologies.[1] That conjecture originated as an assertion that a noncomputability result also holds with a resource bound. Gödel's Second Incompleteness Theorem states that no consistent sufficiently powerful theory can prove its own consistency. Pudlák[19] and Friedman independently formulated a feasible consistency conjecture: it is hard to rule out any length t proof in a theory of its own inconsistency.[2] Krajíček and Pudlák[12] proved the lack of efficient proofs (in a weaker theory) of inconsistency is equivalent to the nonexistence of an optimal proof system, which remains a key conjecture in proof complexity theory.[3]

We show: if it is possible to efficiently rule out length t proofs of some unprovable sentence ϕ, it is also possible to efficiently rule out a slightly shorter proof of inconsistency, which could be used in a length t proof of ϕ by contradiction. This implies a powerful generalization—if it is hard to rule out length t proofs of inconsistency, it is hard to rule of length t proofs of any unprovable sentence. This in turn implies that facts about unprovability and noncomputability, which are well understood, can be imported into complexity theory. This has wide ramifications—diverse types of unprovable sentences translate into assertions that open questions in complexity theory have the expected answers. For instance, unprovable sentences of the form $x{\in}R$ are dense, so hard families of tautologies encoding "no length t proof shows $x{\in}R$" are also dense. With R redefined to have a sparse complement—a string is in R unless exponentially compressible—the complement of the

[1] This paper was prepared in honor of past and present faculty of Davidson College, including Hansford Epes, L. Richardson King, Benjamin Klein, and Clark Ross. Comments are appreciated from Pavel Pudlák and Bill Gasarch. The ideas in this paper and earlier versions have benefited from discussions with the following: Scott Aaronson, Eric Allender, Olaf Beyersdorff, Ilario Bonacina, Maria Luisa Bonet, Cristian Calude, Marco Carmosino, Yuval Filmus, Vijay Ganesh, Bill Gasarch, Valentina Harizonov, Pavel Hrubeš, Rahul Ilango, Russell Impagliazzo, Valentine Kabanets, Mehmet Kayaalp, Yanyi Liu, Ian Mertz, Daniel Monroe, Igor Oliveira, Toniann Pitassi, Hanlin Ren, Rahul Santhanam, Till Tantau, Neil Thapen, Luca Trevisan, Avi Wigderson, Ryan Williams, Marius Zimand, and other participants in seminars at George Washington University and Davidson College, the Simons Institute 2023 Meta-Complexity Program, the Computational Complexity Conference 2022, the Workshop on Proof Complexity 2022, and the Conference on Complexity with a Human Face 2022. Remaining errors are my own.

[2] See Pudlák[21] Section 6.4 and [22]. Pudlák[19] shows the initial conjecture was incorrect—a theory $\mathcal{T}$ can efficiently prove that $\mathcal{T}$ lacks a length t proof of '0=1'. The 1989 reformulation refers to the lack of efficient proofs in a weaker theory. See also Theorem 59 of Pudlák[21].

[3] See also Krajíček[11] Section 21.3.

language $\{\langle x, 1^t\rangle |$ no length t proof exists of $x \in R\}$ is neither in **P** nor **NP**-complete, but is **NP**-intermediate.

The hardness of ruling out length t proofs of any unprovable sentence implies a deep linkage between noncomputability and complexity. We show that the implicit mapping from unprovable sentences to families of hard-to-prove sentences in a theory is an isomorphism. This would be a significant previously unnoticed structural feature of theories such as ZFC.

Formalizing the intuition "ruling out length t proofs is hard" requires specifying which theory lacks length t proofs and which theory has difficulty ruling them out. These theories must be different, as a theory that proves it lacks short proofs of some ϕ would prove its own consistency. Our main result is:

Theorem 1.1 *The following are equivalent:*[4]

(i) No optimal propositional proof system exists.

(ii) For consistent theory $\mathcal{S}$, for some stronger theory $\mathcal{T}$, $\mathcal{S}$ cannot efficiently rule out length t proofs in $\mathcal{T}$ of $0{=}1$ *(that is, $\mathcal{S}\overset{t^{\mathcal{O}(1)}}{\not\vdash}\mathcal{T}\overset{t}{\not\vdash}$'$0{=}1$').*

(iii) For the $\mathcal{S}$ and $\mathcal{T}$ in (ii) and for any ϕ unprovable in $\mathcal{T}$, $\mathcal{S}$ cannot efficiently rule out length t proofs in $\mathcal{T}$ of ϕ (that is, $\mathcal{T}\not\vdash\phi$ implies $\mathcal{S}\overset{t^{\mathcal{O}(1)}}{\not\vdash}\mathcal{T}\overset{t}{\not\vdash}\phi$).[5]

Proof: (i) and (ii) are equivalent by Krajíček and Pudlák[12].

(ii)→(iii) Suppose (ii) holds. If $\mathcal{T}$ lacks a length t proof of ϕ, there can be no proof of 0=1 slightly shorter than t, as that would yield a length t proof by contradiction of ϕ. Therefore, if $\mathcal{S}$ efficiently proves that $\mathcal{T}$ lacks a length t proof of ϕ, contrary to (iii), it also efficiently proves that there can be no proof of 0=1 slightly shorter than t. This contradicts (ii). Therefore, (ii) implies (iii). If ϕ is provable in $\mathcal{T}$, it is provable within some length t, so (iii) cannot hold, as $\mathcal{S}$ is consistent and cannot show $\mathcal{T}$ lacks a proof of that length.

(iii)→(i) Chen et al[4]. ∎

[4] Monroe[17] shows another equivalent condition: For any M accepting `coBHP` $= \{\langle N, y, 1^t\rangle |$ there is no accepting path of nondeterministic TM (NTM) N on input y with t or fewer steps$\}$, there exists some $\langle N', y'\rangle$ where N' does not halt on y' such that $\langle N', y', 1^t\rangle$ is a hard family of inputs.

[5] This conjecture was formulated by the author and proved independently by Pudlák in general and by the author for sentences $x \in R$.

In the notation above in parentheses, write $\mathcal{T}\vdash\phi$ or $\mathcal{T}\nvdash\phi$ respectively if $\mathcal{T}$ does or does not have a proof of ϕ of any length respectively. Write $\mathcal{T}\overset{t}{\vdash}\phi$ if theory $\mathcal{T}$ has a length t (or shorter) proof of sentence ϕ and $\mathcal{T}\overset{t}{\nvdash}\phi$ if not, where proof length is the number of symbols in the binary string representing the proof.[6] Likewise, $\mathcal{T}\overset{t^{\mathcal{O}(1)}}{\nvdash}\phi$ signifies that $\mathcal{T}$ does not have an efficient (polynomially bounded) proof of ϕ. $\mathcal{T}\vdash\phi$ and $\mathcal{T}\overset{\mathcal{O}(1)}{\vdash}\phi$ are equivalent; a provable sentence has a finite proof and is therefore provable within a constant bound.

Below, we will show that the nonexistence of an optimal proof system implies various complexity theory conjectures, by identifying some set of unprovable sentences and invoking Theorem 1.1(iii). In many cases, we choose unprovable sentences stating that a string x is Kolmogorov random (written $x{\in}R$), that is, x is incompressible by half, with no short description in the form of a program that prints x.[7] Chaitin's Incompleteness Theorem states that proving $x{\in}R$ is typically impossible in a theory with a computably enumerable (c.e.) set of theorems. Otherwise, "the first length n string that provably has no short description" would itself be a short description of some string, which is a contradiction. Here, $x{\in}R$ is an arithmetic sentence encoding that a string x (represented as a binary number) lacks a short description. Because, R is dense and noncomputable, the set of $x{\in}R$ provides a dense nonconstructive pool of unprovable sentences.[8]

More formally, define the set of Kolmogorov random strings as $R{=}\{x|\forall p$: if $|p|{\leq}|x|/2$, then $p{\nearrow}$ or $p{\downarrow}$ with $U(p){\neq}x\}$, with U a deterministic universal TM with no limit on its running time (not necessarily prefix free), x and p binary strings with $|x|$ denoting x's length, $p{\downarrow}$ and $p{\nearrow}$ signifying program p does or does not halt, and '$x{\in}R$' is an arithmetic sentence encoding $x{\in}R$. Single and double quotes signify a sentence, a sequence of symbols, encoding a mathematical statement.

If there is no optimal proof system, there are strong implications by Theorem 1.1:

- Ruling out length t proofs is hard on with positive density, because unprovable sentences $x{\in}R$ have positive density. Equivalently, proving

[6] See Pudlák[20]'s survey on proof length.

[7] The definition in terms of incompressibility by half is arbitrary, except for Theorem 5.1 which requires logarithmic incompressibility.

[8] For an overview of Kolmogorov complexity, see Li and Vitanyi[14]. There is a rapidly growing recent literature on meta-complexity; see Santhanam[25].

tautologies encoding "there is no length t proof of $x \in R$" is hard with positive density. There is no optimal proof system for tautologies, with dense set of hard **P**-uniform families witnessing the nonoptimality.

- A natural language is **NP**-intermediate: the sparse complement of the language "$x \in R$ lacks a length t proof" (where R is redefined, by requiring logarithmic incompressibility, to have a sparse complement). This language is not in **P** but has **P/poly** circuits.
- The implicit mapping from unprovable to hard-to-prove sentences is an isomorphism. However, it is incomplete—for instance, stronger conjectures are required to imply that the polynomial hierarchy (**PH**) does not collapse—and substantial work may be needed to identify conjectures related to other open complexity questions and the associated isomorphisms.

The paper is organized as follows. Section 2 provides preliminaries. Section 3 shows that unprovable sentences '$x \in R$' are dense among length n sentences. Section 4 discusses implications for tautologies and proof systems. Section 5 shows that a natural language is **NP**-intermediate. Section 6 shows that the mapping from unprovable to hard-to-prove sentences is an isomorphism and discusses open questions. Section 7 concludes.

2 Preliminaries

Strings: With a binary alphabet $\{0,1\}$, let S^n be the set of length n strings, which are ordered n-tuples. Let $|x|$ be the length of a string and $|S|$ be the cardinality of set S. A language L is a subset of $\cup_{n \geq 0} S^n$.

Density: Say the share of length n strings in L is bounded above zero if there exists $c > 0$ such that $|L \cap S^n|/n \geq c$ for sufficiently large n. This implies the weaker condition that L has positive upper density, i.e., that $\limsup_{n\to\infty} \dfrac{|L \cap \{1,2,\ldots,n\}|}{n} > 0$. If an event depending on n occurs with probability that tends to one as n tends to infinity, such as $x \in R$ where $|x|=n$, say that it occurs with high probability (w.h.p.).

Theories: Theories are assumed to be the Peano arithmetic (PA) or an extension of PA.[9] To allow for average-case analysis, the standard definition

[9] The conjecture could coherently refer to a weaker theory such as Robinson's Q without induction or unbounded quantifiers, which can still prove '$p\downarrow$' if in fact $p\downarrow$, by verifying the transcript of a halting computation.

of PA is modified so binary strings are encoded in arithmetic sentences as natural numbers, in binary not unary, adding a leading “1” to avoid losing leading zeros.

Proof Systems: A propositional proof system is a polynomial time function $h \in \mathbf{FP}$ with range $\texttt{TAUT}$ (Cook and Reckhow[6]). For tautology τ, any string w such that $h(w) = \tau$ is a proof of τ. The proof system h is *optimal* if there exists $c \geq 1$ such that the length of minimal f proofs of x are polynomially bounded in $|x|$ with exponent c by minimal h proofs (Krajíček and Pudlák[12]). A proof system is not optimal if and only if there is a $\mathbf{P}$-uniform family of tautologies for which it requires superpolynomial proof length.

3 Density of Unprovable Sentences

Calude and Jürgensen[3] show that the share of length n arithmetic sentences that are true and unprovable is bounded above zero. The result relies on two facts: ‘$x{\in}R$’ is typically unprovable, and length n strings are in R w.h.p.[10] With that context, Theorem 1.1 implies that a similar result holds for $\texttt{coTHEOREMS}_{\leq t}= \{\langle\phi, 1^t\rangle|\mathcal{T}\overset{t}{\not\vdash}\phi\}$.

Chaitin’s Incompleteness Theorem states:

Theorem 3.1 *For every consistent, arithmetically sound theory $\mathcal{T}$ with a c.e. set of theorems, $\exists k\forall x{:}|x|{>}k$, $\mathcal{T}\not\vdash$‘$x{\in}R$’.*

Proof: Otherwise, a string x could be concisely described as “the first string x of length n such that $\mathcal{T}$ proves ‘$x{\in}R$”’, contrary to the definition of R. A TM with input n in binary (of length $\log n$) could enumerate the theorems of $\mathcal{T}$, printing the first string x such that $\mathcal{T}$ proves ‘$x{\in}R$’. Then, k is determined by the length of the description of this TM, which would need to be doubled as R consists of strings not compressible by half. See Li and Vitanyi[14] Corollary 2.7.2 for a formal treatment. ▮

Lemma 3.2 *$x{\in}R$ w.h.p.*

Proof: By a counting argument, the number of possible short descriptions is small. The number of length n strings is 2^n. The number of programs p with $|p|{\leq}n/2$ is $2^{n/2+1}{-}1$, which is an upper bound on the number of length n

[10] See the proof of Theorem 5.2 in [3].

strings not in R. Therefore, R's share of length n strings is at least $1-2^{-n/2}$, so $x{\in}R$ w.h.p. ▮

Calude and Jürgensen's result implies:

Theorem 3.3 *For every theory $\mathcal{T}$, the share of sentences $\{$‘$x{\in}R$’$\mid x{\in}R$ and $\mathcal{T}\not\vdash$‘$x{\in}R$’$\}$ in length n arithmetic sentences is bounded above zero, for n sufficiently large. In an enumeration of sentences, for instance in lexicographic order, unprovable sentences have positive upper density.*

Proof: Theory $\mathcal{T}$ cannot typically prove sentences ‘$x{\in}R$’ where $x{\in}R$, by Theorem 3.1. The sentences ‘$x{\in}R$’ satisfy $|$‘$x{\in}R$’$| = |x| + c$, where c is a constant not depending on $|x|$, giving the overhead of encoding ‘$x{\in}R$’ net of $|x|$. The share of length n sentences of form ‘$x{\in}R$’ is exactly 2^{-c} and these satisfy $x{\in}R$ w.h.p. Therefore, for $\epsilon{>}0$, this share is bounded below by $2^{-c}{-}\epsilon$ for n sufficiently large. Therefore, in an enumeration of sentences, unprovable sentences have positive upper density. ▮

The fact that a sentence ‘$x{\in}R$’ needs only a constant c bits of overhead, net of $|x|$, to encode $x{\in}R$ is needed in the next section.

4 Tautologies and Proof Systems

A tautology can encode the sentence $\mathcal{T}\not\vdash^{t}$‘$x{\in}R$’ as follows. For a given x, $\mathcal{T}\not\vdash^{t}$‘$x{\in}R$’ is equivalent to $\langle$‘$x{\in}R$’$, 1^t\rangle{\in}\texttt{coTHEOREMS}_{\leq t}$. $\texttt{coTHEOREMS}_{\leq t}$ and $\texttt{TAUT}$ are both **coNP**-complete languages, so some polynomial-time reduction r from $\texttt{coTHEOREMS}_{\leq t}$ to $\texttt{TAUT}$ maps $\langle\phi, 1^t\rangle$ to tautology $r(\langle\phi, 1^t\rangle)$.

Tautologies produced by the reduction r confirm that every possible proof of $\mathcal{T}\not\vdash^{t}$‘$x{\in}R$’ is not a valid proof. The reduction r translates a family of sentences stating that no length t proof exists to a family of tautologies. It should not be confused with propositional translations, which translate sentences with a single universal bounded quantifier that are easy to prove in a weak fragment of arithmetic into easy-to-prove tautologies.[11]

With this encoding, two implications immediately follow: families of tautologies that are hard to prove have positive upper density in an enumeration of families, and there are dense witnesses to the nonoptimality of proof systems.

[11] See Krajíček[11] and Cook and Nguyen[5].

4.1 Proving Tautologies is Hard with Positive Density

R's density immediately implies families of tautologies hard to prove have positive upper density in an enumeration of such families. Consider an enumeration of families of Boolean formulas encoding "no length t proof of ϕ exists", with each family for ϕ indexed by t, with families enumerated in lexicographic order by ϕ. Some formulas will not be tautologies, when ϕ is provable within length t. In this enumeration, families with ϕ of the form '$x{\in}R$' where $x{\in}R$ have positive upper density, and these families are typically hard-to-prove tautologies.

This definition does not necessarily imply that length n elements of `TAUT` are average-case hard to accept. For instance, an algorithm allowed to make errors with small probability can accept for any ϕ of the form '$x{\in}R$' and be correct w.h.p. An error-free probabilistic polynomial time algorithm would necessarily fail with non-zero probability.

4.2 Dense Witnesses to Nonoptimality

If there is no optimal proof system, then for any proof system P, there is a dense set of hard families of tautologies $r(\langle\text{'}x{\in}R\text{'}, 1^t\rangle)$ letting x range over all $x{\in}R$. A probabilistic, polynomial-time computable procedure to produce such a family w.h.p. is to choose a sufficiently long random string x. Then, $x{\in}R$ w.h.p. by Lemma 3.2, so tautologies $r(\langle\text{'}x{\in}R\text{'}, 1^t\rangle)$ are hard for P w.h.p. Tautologies that are hard for ZFC to prove are also hard for any other known proof system, as their soundness is proved by ZFC. "Sufficiently long" is the same as k in Chaitin's theorem, based on the length of the description of a TM that enumerates the theorems of a theory.

5 From Turing Intermediate to NP Intermediate

The set R is Turing intermediate—it is not computable, and its complement is c.e. but not complete under many-one computable reductions (Rogers[10] Theorem 8.I(a) and (c)). This raises the question whether Theorem 1.1 implies that some related language is **NP**-intermediate—that is, in **NP**, not in **P**, and not **NP**-complete under polynomial time many-one reductions. The final paragraph provides context on **NP**-intermediate languages.

We show that deciding the language “has no proof of ‘$x\in R$’ within length t” is **NP**-intermediate relaxing R’s definition to make its complement sparse. This relaxed definition counts strings as random unless they can be compressed exponentially, not just by half. This makes the set of possible short descriptions sparse, growing polynomially in $|x|$, so the the set of non-random strings is also sparse. Define this sparse version of R as $R^{sp}=\{x|\forall p$: if $|p|\leq \log|x|$, then $p\nearrow$ or $p\downarrow$ with $U(p)\neq x\}$. R^{sp}, like R, is noncomputable. Chaitin’s Theorem still holds, but the parameter k is exponentially larger. Fix $\mathcal{S}$ and $\mathcal{T}$ per Theorem 1.1. $\mathcal{T}\overset{t}{\nvdash}$‘$x\in R^{sp}$’ iff $\langle$‘$x\in R^{sp}$’$, 1^t\rangle\in\texttt{coTHEOREMS}_{\leq t}$, by definition. Let $R^{sp}_t=\{\langle$‘$x\in R^{sp}$’$, 1^t\rangle|\mathcal{T}\overset{t}{\nvdash}$‘$x\in R^{sp}$’$\}$, so $R^{sp}_t\in\texttt{coTHEOREMS}_{\leq t}$. Define $\overline{R^{sp}_t}=\{\langle$‘$x\in R^{sp}$’$, 1^t\rangle|\neg\mathcal{T}\overset{t}{\nvdash}$‘$x\in R^{sp}$’$\}$. Based on x, $\overline{R^{sp}_t}$ can be divided into $x\notin R^{sp}$ where $\langle$‘$x\in R^{sp}$’$, 1^t\rangle\in\overline{R^{sp}_t}$ for all t, and $x\in R^{sp}$ where $\langle$‘$x\in R^{sp}$’$, 1^t\rangle\in\overline{R^{sp}_t}$ for sufficiently large t. Then:

Theorem 5.1 *If there is no optimal proof system, then: (i) $\overline{R^{sp}_t}$ is **NP**-intermediate; and (ii) $\overline{R^{sp}_t}$ and therefore R^{sp}_t have minimal circuits in **P/poly** which are not **P**-uniform.*

Proof: (i) $R^{sp}\notin\mathbf{P}$ by assumption and Theorem 1.1. $\overline{R^{sp}_t}$ is sparse, as R^{sp} was defined to ensure this. A sparse language is not **NP**-complete under many-one reductions unless **P**=**NP**, which the assumption rules out (Mahaney[16]).

(ii) $\overline{R^{sp}_t}$ is sparse, so it has minimal circuits in **P/poly**. These are not **P**-uniform, which would imply $R^{sp}\in\mathbf{P}$, which does not hold by assumption. ▮

Ladner[13] constructed artificial **NP**-intermediate languages, assuming **P**≠**NP**. Mahaney showed that a sparse language is not **NP**-complete under many-one reductions unless **P**=**NP**, and under Turing reductions unless **PH** collapses at the second level. Ogiwara and Watanabe[18] provide a result employing bounded truth table reductions. Homer and Longpré[9] provide additional results and alternative proofs.

Allender and Hirahara[1] also provide examples of natural languages that are conditionally **NP**-intermediate. They show that if one-way functions exist, then approximating minimum circuit size and time-bounded Kolmogorov complexity are **NP**-intermediate. Determining whether these problems without approximation are **NP**-hard or not is an area of active research; see for instance Hirahara[8].

If no optimal proof system exists, then **NEXP**$\neq$**coNEXP** (Krajíček and Pudlák[12]), and therefore there are sparse languages in **NP** but not in **P** (Hartmanis et al[7]). Our example differs by providing an explicit natural language.

6 Isomorphisms and Open Questions

If there is no optimal proof system, there is an implicit mapping from unprovable sentences ϕ to families of hard-to-prove sentences "no length t proof exists of ϕ". This mapping can be extended to map provable sentences to families of sentences with a length t proof. If this mapping were onto, it would be an isomorphism. This is an elegant picture—an unnoticed symmetry within mathematics. However, there are several loose ends.

First, the mapping is not onto within the set of all families of hard-to-prove sentences. Suppose theory $\mathcal{S}$ cannot efficiently prove some family of sentences not of the form "no length t proof of ϕ in $\mathcal{T}$ exists" and that this family is **P**-uniform. We can make the mapping onto as follows. For each such family hard for $\mathcal{S}$ not in the range of the mapping, there is a sentence unprovable in $\mathcal{S}$ which states "$\mathcal{S}$ cannot efficiently prove the family". This is unprovable since $\mathcal{S}$ is consistent by assumption, and $\mathcal{S}$ cannot prove that it has a hard family, as it would prove its own consistency. Therefore, map this unprovable sentence onto the hard family. This extended mapping is onto. A similar solution can address the fact that a mapping from unprovable sentences to families of tautologies encoding "no length t proof exists" is not onto.[12] A curious interpretation is that the role of hard families of tautologies in proof complexity, with a powerful theory such as ZFC as a proof system, can be fully understood by focusing solely on the role of unprovable sentences in ZFC. Thus, one can understand proof complexity without reference to tautologies.

Second, additional conjectures are needed to extend this question to other open questions. For instance, the conjecture "no optimal proof system exists `TAUT`", a Π_1^p-complete language, and is not strong enough to imply that **PH** does not collapse. The stronger conjecture "no optimal proof system exists for a Π_2^p-complete language, even for a proof system with an oracle for `TAUT`"

[12] Suppose the **P**-uniform family of tautologies τ_n is hard for proof system P proven sound by theory $\mathcal{S}$ such that the family τ_n is also hard for $\mathcal{S}$. Then there unprovable sentences in $\mathcal{S}$: "$\mathcal{S}$ cannot efficiently prove τ_n" and "P cannot efficiently prove τ_n".

implies that $\Pi_2^p \neq \Pi_1^p$.[13] A version of Theorem 1.1(iii) would hold for $\mathcal{S}$ with a predicate for membership in Π_1 in the arithmetic hierarchy (**AH**), setting up an isomorphism for sentences with a higher degree of unsolvability.[14] A set of such conjectures for each level of **PH** would assert: **PH** does not collapse due to the existence of unprovable sentences at each level of **AH**. These would assert, elegantly, that **PH** does not collapse because **AH** does not collapse.

This suggests a research program could identify a conjecture and implied isomorphism associated with each open question in complexity theory, or identify obstacles to doing so. For instance, the recent flurry of results by Liu and Pass[15] and others suggest that asserting the hardness of showing $\mathcal{T} \overset{t}{\nvdash} `x \in R^t'$, where R^t is defined with respect to time-bounded Kolmogorov complexity, would imply the existence of one-way functions by asserting the average-case hardness of time-bounded Kolmogorov complexity.

To the extent each of these conjectures has the same structure, they can be rolled up into a single overarching conjecture, potentially providing insight into multiple open questions. A very strong conjecture is that some condition of the form in Theorem 1.1(iii) asserts the resolution of most open questions in complexity theory.

7 Conclusion

The conditions in Theorem 1.1 have such strong implications for complexity theory, determining whether they are true and even provable would be desirable. An informal argument is: ruling out length t proofs of an unprovable sentence ϕ is hard because the crucial fact is inaccessible that no proof exists of any length. This informal argument seems strongest for sentences $x \in R$, which are dense, nonconstructive, and typically impossible to prove. To state this in the most extreme form, suppose no other effectively computable fact about $x \in R$ may be useful at all in ruling out length t proofs. In any program ruling out length t proofs of $x \in R$, an if-then statement would need to compute a predicate on R to determine which branch to take. Likewise, in any proof doing the same, any case-based reasoning would need to compute a predicate on R. However, predicates on R are constrained by the fact

[13] Chen et al[4] show that a Π_2^p-complete language does not have an optimal proof system if and only if `TAUT` does not have an optimal proof system, so the reference to an oracle is necessary to separate Π_2^p and Π_1^p.

[14] See Pudlák[21] p. 569 for the construction for `TAUT`.

that R passes all known and conceivable effective tests of randomness (Li and Vitanyi[14] Section 2.4). It is possible that if-then statements and case-based proofs might appear to behave in a purely random manner in ruling out length t proofs of $x{\in}R$. If so, a program or proof can do no better than loops that exhaustively check all cases.

This constraint might also bind non-uniformly. Boolean circuits and propositional proofs require NOT gates and cancellation to implement conditional logic, such as encoding if-then statements and case-based reasoning. Such circuits and proofs may therefore gain limited benefit their use of NOT gates and cancellation, in line with an old conjecture. It is known that for some monotone Boolean functions, the gap between their non-monotone and monotone circuit complexity (the number of gates in minimal circuits with and without NOT gates respectively) is exponential (Razborov[23], Tardos[27]), and hoped that it is small for some other monotone Boolean functions such as CLIQUE (Razborov[24], Alon and Boppana[2]). This conjecture generalized to non-monotone Boolean functions is that for certain functions, the gap is small between their cancellative and non-cancellative circuit complexity is small, where a non-cancellative circuit has a formal polynomial in which no monomial includes both a literal and its negation (Sengupta and Venkateswaran[26]).[15] This argument might support a claim that computational tasks such as decryption of small messages are hard in practice and not just asymptotically.

References

[1] Eric Allender and Shuichi Hirahara, *New insights on the (non-)hardness of circuit minimization and related problems*, ACM Trans. Comput. Theory **11** (2019), no. 4, 27:1–27:27.

[2] Noga Alon and Ravi Boppana, *The monotone circuit complexity of Boolean functions*, Combinatorica **7** (1987), 1–22.

[3] Cristian S. Calude and Helmut Jürgensen, *Is complexity a source of incompleteness?*, Advances in Applied Mathematics **35** (2005), no. 1, 1–15.

[15] Shannon's counting argument shows that most Boolean functions require $2^n/n$ gates, the gap between cancellative and non-cancellative circuits for a random Boolean functions cannot be so large as to reduce circuits to polynomial size, as with Tardos' example.

[4] Yijia Chen, Jörg Flum, and Moritz Müller, *Hard instances of algorithms and proof systems*, How the World Computes (Berlin, Heidelberg) (S. Barry Cooper, Anuj Dawar, and Benedikt Löwe, eds.), Springer Berlin Heidelberg, 2012, pp. 118–128.

[5] Stephen Cook and Phuong Nguyen, *Foundations of proof complexity: Bounded arithmetic and propositional translations*, Cambridge University Press, 2014.

[6] Stephen Cook and Robert Reckhow, *The relative efficiency of propositional proof systems*, J. Symb. Log. **44** (1979), 36–50.

[7] Juris Hartmanis, Neil Immerman, and Vivian Sewelson, *Sparse sets in NP-P: EXPTIME versus NEXPTIME*, Inf. Control. **65** (1985), no. 2/3, 158–181.

[8] Shuichi Hirahara, *NP-hardness of learning programs and partial MCSP*, Electron. Colloquium Comput. Complex. **TR22-119** (2022).

[9] Steven Homer and Luc Longpré, *On reductions of NP sets to sparse sets*, J. Comput. Syst. Sci. **48** (1994), no. 2, 324–336.

[10] Hartley Rogers Jr., *Theory of recursive functions and effective computability*, MIT Press, Cambridge, MA, 1987.

[11] Jan Krajíček, *Proof complexity*, Cambridge University Press, New York, NY, 2019.

[12] Jan Krajíček and Pavel Pudlák, *Propositional proof systems, the consistency of first order theories and the complexity of computations*, J. Symb. Log. **54** (1989), 1063–79.

[13] Richard E. Ladner, *On the structure of polynomial time reducibility*, J. ACM **22** (1975), no. 1, 155–171.

[14] Ming Li and Paul M. B. Vitányi, *An introduction to Kolmogorov complexity and its applications*, Texts in Computer Science, Springer, 2008.

[15] Yanyi Liu and Rafael Pass, *On one-way functions and Kolmogorov complexity*, 2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS), 2020, pp. 1243–1254.

[16] Stephen R. Mahaney, *Sparse complete sets of NP: solution of a conjecture of Berman and Hartmanis*, J. Comput. Syst. Sci. **25** (1982), no. 2, 130–143.

[17] Hunter Monroe, *Speedup for natural problems and noncomputability*, Theoretical Computer Science **412** (2011), no. 4-5, 478–481.

[18] Mitsunori Ogiwara and Osamu Watanabe, *On polynomial-time bounded truth-table reducibility of NP sets to sparse sets*, SIAM J. Comput. **20** (1991), no. 3, 471–483.

[19] Pavel Pudlák, *On the length of proofs of finitistic consistency statements in first order theories*, Studies in Logic and the Foundations of Mathematics, vol. 120, Elsevier, 1986, pp. 165–196.

[20] ______, *The lengths of proofs*, Handbook of Proof Theory (Samuel R. Buss, ed.), Elsevier, 1998.

[21] ______, *Logical foundations of mathematics and computational complexity: A gentle introduction*, Springer, 2013.

[22] ______, *Incompleteness in the finite domain*, Bull. Symb. Log. **23** (2017), no. 4, 405–441.

[23] Alexander A. Razborov, *A lower bound on the monotone network complexity of the logical permanent*, Mathematical Notes of the Academy of Sciences of the USSR **37** (1985), 485–493.

[24] ______, *Lower bounds on the monotone complexity of some Boolean functions*, Doklady Akademii Nauk SSSR **281** (1985), 798–801, In Russian. English translation in *Soviet Mathematics Doklady*, 31:354–57, 1985.

[25] Rahul Santhanam, *Introduction to meta-complexity*, Meta-Complexity Boot Camp, Simons Institute for the Theory of Computing, January 2023.

[26] Rimli Sengupta and H. Venkateswaran, *Non-cancellative Boolean circuits: a generalization of monotone Boolean circuits*, Theor. Comput. Sci. **237** (2000), 197–212.

[27] Éva Tardos, *The gap between monotone and non-monotone circuit complexity is exponential*, Combinatorica **8** (1988), 141–42.