Electronic Colloquium on Computational Complexity, Report No. 30 (2023)

A proof complexity conjecture and the Incompleteness theorem

Jan Krajíček

Faculty of Mathematics and Physics
Charles University*

Abstract

Given a sound first-order p-time theory T capable of formalizing syntax of first-order logic we define a p-time function g_T that stretches all inputs by one bit and we use its properties to show that T must be incomplete. We leave it as an open problem whether for some T the range of g_T intersects all infinite NP sets (i.e. whether it is a proof complexity generator hard for all proof systems).

A propositional version of the construction shows that at least one of the following three statements is true:

1. there is no p-optimal propositional proof system (this is equivalent to the non-existence of a time-optimal propositional proof search algorithm),
2. $E \not\subseteq P/poly$,
3. there exists function h that stretches all inputs by one bit, is computable in sub-exponential time and its range $Rng(h)$ intersects all infinite NP sets.

1 Introduction

We investigate the old conjecture from the theory of proof complexity generators[1] that says that there exists of a generator hard for all proof systems. Its rudimentary version can be stated without a reference to notions of the theory as follows:

- *There exists a p-time function $g : \{0,1\}^* \to \{0,1\}^*$ stretching each input by one bit, $|g(u)| = |u| + 1$, such that the range $Rng(g)$ of g intersects all infinite NP-sets.*

*Sokolovská 83, Prague, 186 75, The Czech Republic, krajicek@karlin.mff.cuni.cz
ORCID: 0000-0003-0670-3957

[1]We are not going to use anything from this theory but the interested reader may start with the introduction to [6] or with [4, 19.4].

ISSN 1433-8092

We present a construction of a function g_T (p-time and stretching) based on provability in a first-order theory T that is able to formalize syntax of first-order logic. Function g_T has the property, assuming that T is sound and complete, that it intersects all infinite definable subsets of $\{0,1\}^*$. As that is clearly absurd (since $\{0,1\}^* \setminus Rng(G)$ is infinite and definable) this offers a proof of Gödel's First Incompleteness theorem. We leave it as an open problem (Problem 2.4) whether g_T for some T satisfies the conjecture above.

We then give a propositional version of the construction and use it to show that at least one of the following three statements has to be true:

1. there is no p-optimal propositional proof system,
2. $E \not\subseteq P/poly$,
3. there exists function h that stretches all inputs by one bit, is computable in sub-exponential time $2^{O((\log n)^{\log \log n})}$ and its range $Rng(h)$ intersects all infinite NP sets.

We assume that the reader is familiar with basic notions of logic and of computational and proof complexity (all can be found in [4]).

2 The construction

We take as our basic theory S^1_2 of Buss [1] (cf. [4, 9.3]), denoting its language simply L. The language has a canonical interpretation in the standard model $\mathbf{N}$. The theory is finitely axiomatizable and formalizes smoothly syntax of first-order logic. Language L allows to define a natural syntactic hierarchy Σ^b_i of bounded formulas that define in $\mathbf{N}$ exactly corresponding levels Σ^p_i, for $i \geq 1$, of the polynomial time hierarchy.

An L-formula Ψ will be identified with the binary string naturally encoding it and $|\Psi|$ is the length of such a string. An L-theory T is thus a subset of $\{0,1\}^*$, a set of L-sentences, and it makes sense to say that it is p-time. It is well-known (and easy) that each r.e. theory has a p-time axiomatization (Craig's trick).

If u, v are two binary strings we denote by $u \subseteq_e v$ the fact that u is an initial subword of v. The concatenation of u and v will be denoted simply by uv. Both these relation and function are definable in S^1_2 by both Σ^b_1 and Π^b_1 formulas that are (provably in S^1_2) equivalent. We shall assume that no formula is a proper prefix of another formula.

Let $T \supseteq S^1_2$ be a first-order theory in language L that is sound (i.e. true in $\mathbf{N}$) and p-time. Define function g_T as follows:

1. Given input u, $|u| = n$, find an L formula $\Phi \subseteq_e u$ with one free variable x such that $|\Phi| \leq \log n$. (It is unique if it exists.)
   - If no such Φ exists, output $g_T(u) := \overline{0} \in \{0,1\}^{n+1}$.

- Otherwise go to 2.

2. Put $c := |\Phi| + 1$. Going through all $w \in \{0,1\}^{c+1}$ in lexicographic order, search for a T-proof of size $\leq \log n$ of the following sentence Φ^w:

$$\exists y \forall x > y \ \Phi(x) \to \neg(w \subseteq_e x) \ . \tag{1}$$

   - If a proof is found for all w output $g_T(u) := \overline{0} \in \{0,1\}^{n+1}$.
   - Otherwise let $w_0 \in \{0,1\}^{c+1}$ be the first such w such that no proof is found. Go to 3.

3. Output $g_T(u) := w_0 u_0 \in \{0,1\}^{n+1}$, where $u = \Phi u_0$.

Lemma 2.1 *Function g_T is p-time, stretches each input by one bit, and the complement of its range is infinite.*

Theorem 2.2 *Let $A \subseteq \{0,1\}^*$ be an infinite L-definable set and assume that for some definition Φ of A theory T proves all true sentences Φ^w as in (1), for $w \in \{0,1\}^{c+1}$ where $c = |\Phi|$. Then the range of function g_T intersects A.*

Proof :

Assume A and Φ satisfy the hypothesis of the theorem. As A is infinite some prefix w has to appear infinitely many times as a prefix of words in A and hence some sentence Φ^w is false. By the soundness of T it cannot be provable in the theory.

Assuming that T proves all true sentences Φ^w let ℓ be a common upper bound to the size of some T-proofs of these true sentences. Then the algorithm computing $g_T(u)$ finds all of them if $n \geq 2^\ell$.

Putting this together, for $n \geq 2^\ell$ the algorithm finds always the same w_0 and this w_0 does indeed show up infinitely many times in A. In particular, if $v \in \{0,1\}^{n+1} \cap A$ is of the form $v = w_0 u_0$ and $n \geq 2^\ell$, then $v = g_T(\Phi u_0)$.

q.e.d.

Applying the theorem to $A := \{0,1\}^* \setminus Rng(g)$ (and using Lemma 2.1) yields the following version of Gödel's First Incompleteness theorem.

Corollary 2.3 *No sound, p-time $T \supseteq S^1_2$ is complete.*

Note that the argument shows that for *each* formula Φ defining the complement, some true sentence Φ^w as in (1) is unprovable in T. The complement of $Rng(g_T)$ is in coNP and that leaves room for the following problem.

Problem 2.4 *For some T as above, can each infinite NP set be defined by* some *L-formula Φ such that all true sentences Φ^w as in (1) are provable in T?*

The affirmative answer would imply by Theorem 2.2 that $Rng(g_T)$ intersects all infinite NP sets and hence g_T solves the proof complexity conjecture mentioned at the beginning of the paper, and thus NP $\neq$ coNP. Note that, for each T, it is easy to define even as simple set as

$$\{1u \mid u \in \{0,1\}^*\}$$

by a formula Φ such that T does not prove that no string in it starts with 0. But in the problem we do not ask if there is *one definition* leading to unprovability but whether *all definitions* of the set lead to it.

3 Down to propositional logic

The reason why the algorithm computing g_T searches for T-proofs of formulas Φ^w rather than of $\neg\Phi^w$ which may seem more natural is that NP sets can be defined by Σ^b_1-formulas Φ and for those, after substituting a witness for y, Φ^w becomes a Π^b_1-formula. Hence one can apply propositional translation (cf. [2] or [4, 12.3]) and hope to take the whole argument down to propositional logic, replacing the incompleteness by lengths-of-proofs lower bounds. There are technical complications along this ideal route but we are at least able to combine the general idea with a construction akin to that underlying [3, Thm.2.1][2]

and to prove the following statement.

Theorem 3.1 *At least one of the following three statements is true:*

1. *there is no p-optimal propositional proof system,*
2. *E $\not\subseteq$ P/poly,*
3. *there exists function h that stretches all inputs by one bit, is computable in sub-exponential time $2^{O((\log n)^{\log\log n})}$ and its range $Rng(h)$ intersects all infinite NP sets.*

Note the first statement is by [5, Thm.2.4] equivalent to the non-existence of a time-optimal propositional proof search algorithm.

Before starting the proof we need to recall a fact about propositional translations of Π^b_1-formulas. For $\Phi(x) \in \Sigma^b_1$, $c := |\Phi|$ and $w \in \{0,1\}^{c+1}$, and $n \geq 1$ let $\varphi_{n,w}$ be the canonical propositional formula expressing that

$$(|x| = n + 1 \wedge \Phi(x)) \to \neg w \subseteq_e x \ .$$

We use the qualification *canonical* because the formula can be defined using the canonical propositional translation $||\ldots||^{n+1}$ (cf. [4, 12.3] or [2]) applied to Φ^w after instantiating first y by $1^{(n)}$. Formula $\varphi_{n,w}$ has $n+1$ atoms for bits

[2]That theorem is similar in form to Theorem 3.1 but with 2) replaced by E $\not\subseteq$ $Size(2^{o(n)})$ and 3) replaced by NP $\neq$ coNP.

of x and $n^{O(1)}$ atoms encoding a potential witness to $\Phi(x)$ together with the p-time computation that it is correct. For any fixed Φ the size of $\varphi_{n,w}$ (with $w \in \{0,1\}^{c+1}$) is polynomial in n and, in fact, the formulas are very uniform (cf. [4, [19.1]). We shall need only the following fact.

Lemma 3.2 *There is an algorithm* **transl** *that upon receiving as inputs a Σ_1^b-formula Φ, $w \in \{0,1\}^{c+1}$ where $c := |\Phi|$ and $1^{(n)}$, $n \geq 1$, outputs $\varphi_{n,w}$ such that*

$$(|x| = n + 1 \wedge \Phi(x)) \rightarrow \neg w \subseteq_e x \ .$$

is universally valid iff $\varphi_{n,w}$ is a tautology. In addition, for any fixed Φ the algorithm runs in time polynomial in n, for $n > |\Phi|$.

Proof of Theorem 3.1:

We shall prove the theorem by contradiction: assuming that statements 1) and 2) fail we construct function h satisfying statement 3). Our strategy is akin in part to that of the proof of [3, Thm.2.1].

For a fixed Φ assume that formulas $\varphi_{n,w}$ are valid for $n \geq n_0$. By Lemma 3.2 they are computed by $\mathbf{transl}(\Phi, w, 1^{(n)})$ in p-time. Hence we can consider the pair $1^{(n)}, w$ to be a proof (in an ad hoc defined proof system) of $\varphi_{n,w}$ for $n \geq n_0$

Assuming that statement 1) fails and P is a p-optimal proof system we get a p-time function f that from $1^{(n)}, w$, $n \geq n_0$, computes a P-proof $f(1^{(n)}, w)$ of $\varphi_{n,w}$. Let $|f(1^{(n)}, w)| \leq n^\ell$ where ℓ is a constant (depending on Φ). The function that from n, w, i, with $i \leq n^\ell$, computes the i-th bit of $f(1^{(n)}, w)$ is in the computational class E.

We would like to check the validity of $\varphi_{n,w}$ by checking the P-proof $f(1^{(n)}, w)$ but we (i.e. the algorithm that will compute h) cannot construct f from Φ. Here the assumption that statement 2) fails too, i.e. that $E \subseteq \mathrm{P}/poly$, will help us. By this assumption $f(1^{(n)}, w)$ is the truth-table $\mathbf{tt}(D)$ (i.e. the lexicographically ordered list of values of circuit D on all inputs) of some circuit with $\log n + c + \ell \log n \leq (2+\ell)\log n$ inputs and of size $|D| \leq (\log n)^{O(\ell)}$. In particular, for all ℓ (i.e. for all $\Phi \in \Sigma_1^b$) we have[3] $|D| \leq (\log n)^{\log\log n}$ for $n >> 1$. Hence it is enough to look for P-proofs among $\mathbf{tt}(D)$ for circuits of at most this size.

We can now define function h_P in a way analogous to the definition of function g_T. Namely:

1. Given input u, $|u| = n$, find a Σ_1^b-formula $\Phi \subseteq_e u$ with one free variable x such that $|\Phi| \leq \log n$. (It is unique if it exists.)
    - If no such Φ exists, output $h_P(u) := \overline{0} \in \{0,1\}^{n+1}$.
    - Otherwise go to 2.
2. Put $c := |\Phi| + 1$. Going through all $w \in \{0,1\}^{c+1}$ in lexicographic order, do the following.

[3]Note that the function $\log\log n$ bounding ℓ can be replaced by any $\omega(1)$ time-constructible function, making the time needed to compute function h closer to quasi-polynomial.

Using **transl** compute formula $\varphi_{n,w}$. If the computation does not halt in time $\leq n^{\log n}$ stop and output $h_P(u) = \overline{0} \in \{0,1\}^{n+1}$. Otherwise search for a P-proof of formula $\varphi_{n,w}$ by going systematically through all circuits D with $\leq \log n \cdot \log\log n$ inputs and of size $\leq (\log n)^{\log\log n}$ until some $\mathbf{tt}(D)$ is a P-proof of $\varphi_{n,w}$.

- If a proof is found for all $w \in \{0,1\}^{c+1}$ output $h_P(u) := \overline{0} \in \{0,1\}^{n+1}$.
- Otherwise let $w_0 \in \{0,1\}^{c+1}$ be the first such w such that no P-proof is found. Go to 3.

3. Output $h_P(u) := w_0 u_0 \in \{0,1\}^{n+1}$, where $u = \Phi u_0$.

It is clear from the construction that function h_P stretches each input by one bit (and hence the complement of its range is infinite) and that

$$Rng(h_P) \cap \{x \in \{0,1\}^{n+1} \mid \Phi(x)\} \neq \emptyset$$

for any $\Phi(x) \in \Sigma^b_1$ and $n >> 1$.

The time needed for the computation of $h_P(u)$ is $O(n)$ for step 1 and for step 2 it is bounded above by

$$2^{c+1} \cdot n^{\log n} \cdot 2^{(\log n)^{\log\log n}} \cdot 2^{O((\log n)^{\log\log n})} \leq 2^{O((\log n)^{\log\log n})} \ .$$

The first factor bounds the number of w, the second bounds the time needed to compute $\varphi_{n,w}$, the third bounds the number of circuits D and the fourth one bounds the time needed to compute $\mathbf{tt}(D)$ and to check wether it is a P-proof of $\varphi_{n,w}$ (this is p-time in $|\mathbf{tt}(D)|$).

q.e.d.

Acknowledgments: Section 3 owns its existence to J.Pich (Oxford) who suggested I include some propositional version of the construction.

References

[1] S. R. Buss, *Bounded Arithmetic.* Naples, Bibliopolis, (1986).

[2] S. A. Cook, Feasibly constructive proofs and the propositional calculus, in: *Proc. 7^{th} Annual ACM Symp. on Theory of Computing* (STOC), (1975), pp. 83-97. ACM Press.

[3] J. Krajíček, Diagonalization in proof complexity, *Fundamenta Mathematicae*, **182**, (2004), pp.181-192.

[4] J. Krajíček, *Proof complexity*, Encyclopedia of Mathematics and Its Applications, Vol. **170**, Cambridge University Press, (2019).

[5] J. Krajíček, Information in propositional proofs and proof search, *J. Symbolic Logic*, **87(2)**, (2022), pp.852-869.

[6] J. Krajíček, On the existence of strong proof complexity generators, preprint 2022. DOI: https://doi.org/10.48550/arXiv.2208.11642

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il