Complexity of Propositional Proofs

Alexander A. Razborov *

1. Introduction

Propositional proof complexity is an area of study that has seen a rapid development over a couple of last decades. It plays as important a role in the theory of feasible proofs as the role played by the complexity of Boolean circuits in the theory of efficient computations. Propositional proof complexity is in a sense complementary to the (non-uniform) computational complexity; moreover, there exist extremely rich and productive relations between the two areas.

Besides the theory of (first-order) feasible proofs and computational complexity, propositional proof complexity is tightly connected in many ways with other areas like automated theorem proving and cryptography, and these connections are numerous both at the level of motivations and when it comes to proof techniques. Given this diversity, as well as quite a substantial amount of material gathered during recent years, it is fairly impossible even to touch all aspects of propositional proof complexity in one lecture. I will not even try that; instead let me refer to the sources [Urq95, Kra95, Raz96, BP98, Pud98] that serve various tastes and, taken together, cover virtually everything which is currently known about the complexity of propositional proofs (see, however, the remark at the end of Section 4).

2. Definitions

The framework underlying propositional proof complexity was developed in the seminal paper [CR79].

*Steklov Mathematical Institute, Moscow, Russia and Institute for Advanced Study, Princeton, US, razborov@mi.ras.ru.

Definition. Let TAUT be the set of all propositional tautologies. A *propositional proof system* is any poly-time computable function

$$P : \{0,1\}^* \xrightarrow{\text{onto}} \mathrm{TAUT}.$$

Intuitively, binary words $w \in \{0,1\}^*$ are viewed as (encodings of) P-proofs, and $P(w)$ is the tautology proved by the proof w. P must be *sound* (that is, $P(w) \in \mathrm{TAUT}$ for all proofs w) and *complete* (P is onto).

Let $\mu(w)$ be any integer-valued complexity measure on P-proofs. The *proof complexity* $\mu_P(\phi)$ *of a tautology* $\phi \in \mathrm{TAUT}$ (w.r.t. the measure μ) is naturally defined as $\mu_P(\phi) \stackrel{\text{def}}{=} \min_{P(w)=\phi} \mu(w)$. The basic task of propositional proof complexity can be now formulated very simply: learn as much as possible about the behaviour of $\mu_P(\phi)$ for "interesting" proof systems P and "interesting" tautologies ϕ.

It is important to stress that if we universally quantify over all conceivable proof systems P, this problem will become equivalent to the most basic (and widely open) questions from computational complexity. Let, for example, $\mu(w)$ be simply the bit length $|w|$ of the proof w (incidentally, this is by far the most important complexity measure), and denote by $S_P(\phi)$ the corresponding complexity measure ("S" stands for "size"). Let us call a proof system P *p-bounded* if $S_P(\phi) \leq |\phi|^{O(1)}$, i.e. every tautology has a proof whose length is comparable with its own length. Then, as observed already in [CR79], there exists *some* p-bounded propositional proof system *if and only if* $\mathbf{NP} = co - \mathbf{NP}$.

The typical question of propositional proof complexity, however, is not whether there exists any p-bounded system, but whether concrete important systems are so. In this talk we will be in particular interested in such proof systems (the definitions will be provided) as *Resolution*, *Polynomial Calculus*, *bounded-depth Frege*, *Frege* and *Extended Frege*. All these proof systems are in the focus of attention today.

3. Motivation

We will be considering in detail only one motivation for studying the complexity of propositional proofs rooted in the classical proof theory. Suppose that we are interested in provability of first-order formulas A from a certain class in a theory T of Bounded Arithmetic. If A is a Σ_i^b-formula[1] for $i \geq 1$ then there exists a rich and powerful *witnessing*

[1] the superscript "b" stands for "bounded" and signifies that the hierarchy is built up according to the number of alterations of bounded quantifiers $(\exists x \leq t)$, $(\forall x \leq t)$

technique which very tightly connects the provability of A in T with the fact that Skolem functions for A can be constructed in a certain complexity class $\mathcal{C}_T$ associated with the theory T.

If, however, $A \in \Sigma_0^b$ (and this case for certain reasons is also very interesting), then this witnessing technique faces very serious problems, just because there are no more Skolem functions to witness. And what comes as replacement in this case is exactly the complexity of propositional proofs. Namely, with the theory T we usually can associate a propositional proof system P_T, and with any Σ_0^b-formula A we can associate a sequence of propositional tautologies $\{A_n\}$ (A_n asserting the validity of A on strings of binary length n) such that the provability of A in T is extremely closely related to the existence of *efficient* P_T-proofs for the tautologies $\{A_n\}$. Moreover, the system P_T naturally corresponds to the non-uniform analogue of the complexity class $\mathcal{C}_T$ in the sense that lines in P_T-proofs essentially represent Boolean circuits underlying this complexity class.

Thus, if desired, a substantial portion of propositional proof complexity can be viewed as understanding the provability of Σ_0^b-formulas in first-order theories of Bounded Arithmetic. The main advantage of the "propositional" approach is that this model is usually much simpler to formulate and much cleaner combinatorially than its first-order analogue.

4. Weak systems: methods

Understanding the behaviour of $\mu_P(\phi)$ (see Section 2) should involve proving upper and lower bounds on its value. Upper bounds are often deep and elegant but there is little mystery about them: we simply exhibit something (in our case, efficient proofs for tautologies of interest). Like in computational complexity, the real question is how to prove lower bounds.

Let $\mathcal{C}$ be the non-uniform (= circuit) complexity class that consists of functions computable by lines in efficient (which in most cases stands for "polynomial size") P-proofs (cf. Section 3). Somewhat conditionally, let us call P *weak* if we know how to prove lower bounds for the circuit class $\mathcal{C}$ and *strong* otherwise. In this classification, Resolution, Polynomial Calculus and bounded-depth Frege are weak, whereas Frege and Extended Frege are strong.

For weak proof systems P one might expect that the existing methods of proving lower bounds for the accompanying circuit class $\mathcal{C}$ should transfer to the methods of proving lower bounds on (say) $S_P(\phi)$. Often (but not always) this is indeed the case, and quite a number of beautiful lower bounds and structural results have been proved along these lines.

It would be somewhat too technical to try to assess all these results in one lecture. Instead, I will chose a different approach and concentrate on lower and upper bounds describing the complexity of particular tautologies that express various forms of the so-called *Pigeonhole principle.* This principle (asserting that there is no injective mapping from m pigeons to n holes when $m > n$) is probably the most extensively studied combinatorial principle in proof complexity. It is amazingly simple and at the same time captures one of the most basic primitives in mathematics and Theoretical Computer Science (counting).

Surprisingly, the proof complexity of the pigeonhole principle essentially depends on the number of pigeons m (as the function of the number of holes n) and on subtle details of its representation as a propositional tautology. This leads to a rich structural picture, and results making the skeleton of this picture are amongst the most beautiful and important in the whole theory. Moreover, for a long time they have been determining its methods, machinery and ideology. Therefore, from our survey of these particular results one can (hopefully) get at least a general impression of what can be done for weak proof systems using the machinery common for circuit complexity.

In the last couple of years several more important touches have been added to this picture, and these results do not seem to have been surveyed in the literature yet. For this reason the final version of the paper will be entirely devoted to the pigeonhole principle.

5. Strong systems: approaches

Although there are no rigorous results along these lines, the empirical evidence suggests that it should be even harder to show that something does not have efficient P-proofs than to show that some explicit problem can not be computed by efficient circuits from the accompanying circuit class $\mathcal{C}$. Since for strong proof systems P the latter task is essentially equivalent to fundamental open problems of computational complexity like $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ or $\mathbf{NC}^1 \stackrel{?}{=} \mathbf{P}$, it is expected to be very hard. Can we say at least something intelligent about the complexity of proving tautologies in strong proof systems without actually solving these fundamental problems?

We need to return at this point to the discussion from Section 3. Recall that studying the complexity of propositional proofs is essentially equivalent to studying the provability of Σ_0^b-formulas in appropriate systems of Bounded Arithmetic. Also, many intelligent things can be said (via the witnessing technique) about the provability of Σ_i^b-formulas for $i \geq 1$.

We also mentioned in Section 3 that this witnessing technique fails for Σ_0^b-formulas since there are no more quantifiers to witness. But this remark applies only to straightforward attempts of finding computational witnesses for efficient propositional proofs (or first-order proofs for Σ_0^b-formulas). It seems that currently there are no inherent reasons which would prevent more sophisticated methods from extracting a non-trivial computational information from any such proof.

There are indeed nice results and approaches in this direction but at the moment they look somewhat discouraging. Namely, they have been successful only for weak systems. On the other hand, even these limited results (the most important of them is the so-called *Efficient Interpolation Theorem*) do provide new insights into the core of propositional proof complexity as well as reveal new interesting connections with classical proof theory and cryptography. This is why I have nonetheless chosen to spend some time on this direction as well, even if our current success here is rather limited.

References

[BP98] P. Beame and T. Pitassi. Propositional proof complexity: Past, present and future. Technical Report TR98-067, Electronic Colloquium on Computational Complexity, 1998.

[Kra95] J. Krajíček. *Bounded arithmetic, propositional logic and complexity theory.* Cambridge University Press, 1995.

[CR79] S. A. Cook and A. R. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44(1):36–50, 1979.

[Pud98] P. Pudlák. The lengths of proofs. In S. Buss, editor, *Handbook of Proof Theory*, pages 547–637. Elsevier, 1998. To appear in *Handbook of Proof Theory.*

[Raz96] A. Razborov. Lower bounds for propositional proofs and independence results in Bounded Arithmetic. In F. Meyer auf der Heide and B. Monien, editors, *Proceedings of the 23rd ICALP,* *Lecture Notes in Computer Science,* 1099, pages 48–62, New York/Berlin, 1996. Springer-Verlag.

[Urq95] A. Urquhart. The complexity of propositional proofs. *Bulletin of Symbolic Logic*, 1:425–467, 1995.