The Computational Complexity Column

by

Lance FORTNOW

Department of Computer Science
University of Chicago
1100 East 58th St., Chicago, IL 60637 USA
fortnow@cs.uchicago.edu
http://www.cs.uchicago.edu/~fortnow/beatcs

I have moved back to the University of Chicago and so has the web page for this column. See above for new URL and contact informaion.

This issue Scott Aaronson writes quite an interesting (and opinionated) column on whether the P = NP question is independent of the usual axiom systems. Enjoy!

Is P Versus NP Formally Independent?

Scott Aaronson[1]
University of California
Berkeley, California USA
aaronson@cs.berkeley.edu

Abstract

This is a survey about the title question, written for people who (like the author) see logic as forbidding, esoteric, and remote from their usual concerns. Beginning with a crash course on Zermelo-Fraenkel set theory, it discusses oracle independence; natural proofs; independence results of Razborov, Raz, DeMillo-Lipton, Sazanov, and others; and obstacles to proving P vs. NP independent of strong logical theories. It ends with some philosophical musings on when one should expect a mathematical question to have a definite answer.

1 Introduction

The P vs. NP problem has been called "one of the most important problems in contemporary mathematics and theoretical computer science" [60]. That is an understatement. Not only is P vs. NP the defining question of our field; it's one of the deepest questions ever asked for which we'd know how to recognize an answer.[2] (In other words, one of the deepest questions in NP.) If you doubt this, read the Clay Math Institute's list of million-dollar prize problems [12], and notice how P vs. NP stands out, not merely as the only problem of the seven relevant practically, but

[1] Supported by an NSF Graduate Fellowship, by NSF ITR Grant CCR-0121555, and by the Defense Advanced Research Projects Agency (DARPA).

[2] We are taking P vs. NP here as representative of a whole constellation of related questions, which are arguably just as important: NP vs. P/poly, P vs. PSPACE, NP vs. coNP, BPP vs. BQP, the existence of public-key cryptography, and so on. For simplicity, however, we focus on P vs. NP.

as the only one pregnant philosophically. Does the ability to recognize an answer to the other six questions—or to P vs. NP, or to any question—entail the ability to *find* an answer? We are after not projective algebraic varieties or zeros of the Riemann zeta function, but the nature of mathematical thought itself.

And therein lies the subject of this survey. If P vs. NP asks something so ineluctably metamathematical, then should we be surprised if the tools of mathematics don't suffice to answer it? In other words, could P vs. NP be independent of the standard axiom systems, such as Zermelo-Fraenkel set theory? Whether one judges this plausible seems related to one's view on another question: is P vs. NP "really" a question about logic, or about combinatorics?

Attitudes toward the last question have shifted over the past fifty years. In Gödel's famous 1956 letter to von Neumann (translated in [60]), where P vs. NP was first posed, Gödel apparently saw the problem as a finitary analogue of the Hilbert Entscheidungsproblem:[3]

> It is evident that one can easily construct a Turing machine which, for each formula F of the predicate calculus and for every natural number n, will allow one to decide if F has a proof of length n. Let $\Psi(F,n)$ be the number of steps that the machine requires for that and let $\varphi(n) = \max_F \Psi(F,n)$. The question is, how fast does $\varphi(n)$ grow for an optimal machine. One can show that $\varphi(n) \geq Kn$. If there actually were a machine with $\varphi(n) \sim Kn$ (or even only with $\sim Kn^2$), this would have consequences of the greatest magnitude. That is to say, it would clearly indicate that, despite the unsolvability of the Entscheidungsproblem, the mental effort of the mathematician in the case of yes-or-no questions could be completely (footnote: apart from the postulation of axioms) replaced by machines. One would indeed have to simply select an n so large that, if the machine yields no result, there would then also be no reason to think further about the problem.

In the seventies, when serious work on P vs. NP began following the discovery of NP-completeness by Cook [14], Karp [30], and Levin [36], the problem continued to be seen in logical terms. People tried to separate P and NP using the same ideas that worked in recursion theory; this is the context of the Baker-Gill-Solovay [4] result that P vs. NP admits contradictory relativizations. In Section 3 we'll see other logic-oriented work from this period, which explicitly raised the possibility that $\mathsf{P} \neq \mathsf{NP}$ might go the way of the Continuum Hypothesis—its truth forever unknowable, or dependent upon which axioms we choose.

In the eighties, the combinatorial view became dominant, in the wake of the circuit lower bounds of Furst-Saxe-Sipser [18], Ajtai [2], Razborov [50, 52], and Smolensky [61]. On this view, P vs. NP is no more likely to be formally independent than Goldbach's Conjecture, the Riemann Hypothesis, or any other 'natural' mathematical problem. Proving $\mathsf{P} \neq \mathsf{NP}$ (and indeed the stronger statement $\mathsf{NP} \not\subseteq \mathsf{P/poly}$) is really just a matter of rolling up our sleeves and analyzing increasingly powerful kinds of circuits.

The nineties saw a return to the logicism and brooding pessimism of the seventies. As discussed in Section 4, Razborov and Rudich [56] pointed out a basic limitation of the circuit lower bound techniques from the eighties: these techniques would (almost) argue against themselves, yielding

[3] The Entscheidungsproblem asks for a procedure that, given a mathematical statement, either finds a proof or tells us that none exists. $\mathsf{P} = \mathsf{NP}$ asks for an *efficient* procedure that finds a *short* proof. In one sense, however, P vs. NP is *not* a finitary analogue of the Entscheidungsproblem. For once we require all branches of a computation to halt eventually, the 'nondeterministically recursive' languages are just the recursive languages themselves, not the recursively enumerable languages. So the distinction between finding and verifying a bounded-size object is one that doesn't exist in the recursion-theory world.

efficient ways to solve the very problems they were supposed to prove intractable! Subsequently, as we'll see in Section 4.1, Razborov [55] used this idea to prove, under cryptographic assumptions, that P vs. NP is independent of certain theories of 'bounded arithmetic.'

I don't know if it's possible any longer to discern a unifying trend. Mulmuley and Sohoni [39, 57] have proposed to tackle P vs. NP using algebraic geometry—an approach that, they think, might escape the jaws of Razborov and Rudich. Using the deep result that $\mathsf{MIP} = \mathsf{NEXP}$ [3], Buhrman, Fortnow, and Thierauf [8] showed that $\mathsf{MA}_{\mathsf{EXP}}$, the exponential-time analogue of MA, doesn't have polynomial-size circuits. This result is notable because it avoids both the Baker-Gill-Solovay and the Razborov-Rudich limitations. So perhaps there's hope after all.

In any event, however, it seems likely that future work on P vs. NP will be inseparable from work on the *logical status* of the question. Again and again in this survey, we'll encounter 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. For this reason, I think that anyone interested in complexity—even those who, like me, have no logic background whatsoever—ought to learn something about what it would mean for P vs. NP to be formally independent, and what can be said for and against that possibility. Hence this survey. Although it does contain the recommended daily allowance of theorems (for an average-weight theoretician), its central concern is philosophical:

> *How could such a clear mathematical question—a question about finite machines and problems, a question of great importance for science and industry—be forever unanswerable?*

Thus I'll start, as I'll end, by talking not about P vs. NP, but about logic, provability, and truth.

2 Logic Primer

Have you ever lay awake at night, terrified of what would happen were the Zermelo-Fraenkel axioms found to be inconsistent the next morning? Would bridges collapse? Would every paper ever published in STOC and FOCS have to be withdrawn? ("The theorems are still true, but so are their negations.") Merely contemplating such possibilities is enough to cause feelings of vertigo. How could a theorem be invalidated because of axioms that seem to have nothing to do with it? It would be as if you added up a restaurant bill, and then a physicist explained that your total was wrong because of the expansion of the universe.

This is the sort of issue philosophers like Wittgenstein talked about a lot. I won't try to summarize what they said, for the same reason that Turing, after sitting in on Wittgenstein's seminar, became exasperated and dropped out [38]. My concern is more practical: should we, who dwell in the tower of mathematics, worry about a collapse of the foundations?

In my opinion, no. If you don't believe Euclid's proof that there are infinitely many primes, then I don't know why seeing a formal proof in ZF set theory should quell your doubts. You could deny Modus Ponens, you could demand a formal proof for the assertion "The infinitude of primes is a theorem of ZF" itself (ad infinitum), you could refuse to identify the formal notions of 'prime' or 'infinity' with the informal ones, and so on. Therefore *the informal proof already yields as much certainty about the infinitude of primes as it's humanly possible to have.* If we try to place the proof on a "more secure foundation," then the foundation is as open to doubt as the positive integers are.

If you accept this, I think it becomes easier to understand what a formal system is. It's just a way to codify a form of reasoning—which presumably we've already accepted as valid—so that

(1) we can program a computer to reason for us, and (2) we can explore the ultimate limitations of that form of reasoning.

2.1 The First-Order Sentences Game

As an example, consider the definition of a group. We have a ternary predicate $M(x, y, z)$ (intuitively $x \cdot y = z$) that satisfies a set of three axioms, called a *theory*:

(1) **Uniqueness.** $\forall x, y\ \exists z\ \ (M(x, y, z) \wedge \forall w\ \ (M(x, y, w) \Longrightarrow (w = z)))$

(2) **Associativity.** $\forall x, y, z, w\ \ (\exists v\ M(x, y, v) \wedge M(v, z, w)) \Longrightarrow (\exists v\ M(y, z, v) \wedge M(x, v, w))$

(3) **Identity and Inverse.** $\exists x\ \forall y\ \ (M(x, y, y) \wedge M(y, x, y) \wedge \exists z\ \ (M(y, z, x) \wedge M(z, y, x)))$

Then a *model* for the theory is just a set U of objects (called a *universe*), together with an assignment of 'true' or 'false' to $M(x, y, z)$ for every x, y, z in U, such that the axioms hold when the quantifiers range over U. In general, an axiom can be any first-order sentence—meaning that it can have variable, constant, and predicate symbols; Boolean connectives ($\wedge$, $\vee$, $\urcorner$, $\Longrightarrow$, etc.); quantifiers ($\forall$, $\exists$); and equal signs ($=$). The equal sign is not shorthand for a binary predicate $E(x, y)$; it really means that two objects are the same. Crucially, the quantifiers can range only over objects in U, not *sets* of objects. That's what 'first-order' means.

How much can we say using these first-order sentences? To get some intuition, let's ask how well we can control the *cardinality* of the universe U. Groups can be finite, countable, or uncountable, so of course the axioms for a group have finite, countable, and uncountable models. If we want, though, we can force $|U|$ to be at most 3:

- $\exists x, y, z\ \forall w\ \ (w = x \vee w = y \vee w = z)$

We can also force $|U|$ to be infinite. Peano Arithmetic uses a binary predicate $S(x, y)$ (intuitively $y = x + 1$) that satisfies these two axioms:

(1) **Zero.** $\exists y\ \forall x\ \urcorner S(x, y)$

(2) **Unique Successor.** $\forall x\ \exists y\ \ (S(x, y) \wedge \urcorner (y = x) \wedge \forall z\ \ (S(x, z) \Longrightarrow (z = y)))$

Can we force $|U|$ to be uncountable? We might try to by writing down axioms that encompass all of mathematics. That sounds like a tall order, but as is well known, it's enough to describe a universe whose objects are *sets*.[4] What do these sets contain? Other sets! Zermelo-Fraenkel (ZF) set theory involves a binary predicate $S(x, y)$ (intuitively $x \in y$), that satisfies the following axioms. (I'll write the axioms in English, since it should be obvious by now how to convert them to first-order notation.)

(1) **Empty Set.** There exists a set (denoted $\varnothing$) that does not contain any members.

(2) **Extensionality.** If two sets contain the same members then they are equal.

(3) **Pairing.** For all sets x and y there exists a set whose members are x and y.

[4] How can we use axioms to define sets, if we need the concept of set even to say what we mean by a *model* for axioms? This is the kind of paradox that I think vanishes once we give up the idea of putting mathematics on a 'more secure foundation,' and see formal systems as just codifying forms of reasoning we've already accepted as valid.

(4) **Union.** For all sets x and y there exists a set (denoted $x \cup y$) that contains z if and only if $z \in x$ or $z \in y$.

(5) **Infinity.** There exists a set x that contains $\varnothing$ and that contains $y \cup \{y\}$ for every $y \in x$.

(6) **Power Set.** For all sets x there exists a set (denoted 2^x) that contains y if and only if $y \subseteq x$.

(7) **Replacement for Predicate A.** For all sets u, if for all $x \in u$ there exists a unique y such that $A(x, y)$, then there exists a z such that for all $x \in u$, there exists a $y \in z$ such that $A(x, y)$.

(8) **Foundation.** All nonempty sets x contain a member y such that for all z, either $z \notin x$ or $z \notin y$.

There are actually infinitely many axioms above, since we have an Axiom of Replacement for every binary predicate $A(x, y)$. This doesn't cause problems (though if we don't like it, it turns out that there's an equivalent, more complicated system with finitely many axioms, called Gödel-Bernays set theory).

So, do all models of ZF have uncountably many sets? One would think so—since after all, the Infinity and Power Set Axioms imply the *existence* of uncountable sets! To get a better handle on the question, though, let's examine the inference rules of first-order logic, by which we prove theorems. We say a first-order sentence is *valid* if can be obtained by the following rules:

(1) **Propositional Rule.** Any propositional tautology is valid.

(2) **Modus Ponens.** If A and $A \Rightarrow B$ are valid then B is valid.

(3) **Equality Rules.** The following are valid: (a) $x = x$, (b) $x = y \Longrightarrow y = x$, (c) $x = y \wedge y = z \Rightarrow x = z$, (4) $x = y \Longrightarrow (A(x) \Longrightarrow A(y))$.

(4) **Change of Variables.** Changing variable names leaves a statement valid.

(5) **Quantifier Elimination.** If $\forall x A(x)$ is valid then $A(y)$ is valid.

(6) **Quantifier Addition.** If $A(y)$ is valid where y is an unrestricted constant then $\forall x A(x)$ is valid.

(7) **Quantifier Rules.** The following are valid: (a) $\urcorner \forall x A(x) \Longleftrightarrow \exists x \urcorner A(x)$, (b) $(B \wedge \forall x A(x)) \Longleftrightarrow \forall x (B \wedge A(x))$, (c) $(B \wedge \exists x A(x)) \Longleftrightarrow \exists x (B \wedge A(x))$.

Obviously, if a set of axioms has a model, then applying the inference rules above can never lead to a contradiction. Gödel's *Completeness* Theorem says the converse: we didn't accidentally leave any rules out of the list, in the sense that if you can't get a contradiction by applying the rules, then the axiom set has a model. Equivalently, any sentence that's true in all models is provable. The Completeness Theorem is confusing for two reasons: on the one hand, it sounds like a tautology ("that which is consistent, is consistent")—what could it possibly mean to prove such a thing? And on the other hand, it seems to contradict the *Incompleteness* Theorem.

We're going to clear up this mess, and as a bonus, answer our question about whether all models of ZF are uncountable. The best way to understand the Completeness Theorem is to make up a consistent axiom set that you'd guess *doesn't* have a model. Given a theory T, let Con (T) be the assertion that T is consistent. We know from Gödel's Incompleteness Theorem that Con (ZF) can be expressed in ZF, and also that Con (ZF) can't be *proved* in ZF, assuming ZF is consistent.[5] It follows that assuming ZF is consistent, the "self-hating theory" $\mathrm{ZF} + \neg\,\mathrm{Con}\,(\mathrm{ZF})$, or ZF plus the assertion of its own inconsistency, must also be consistent. So by the Completeness Theorem, $\mathrm{ZF} + \neg\,\mathrm{Con}\,(\mathrm{ZF})$ has a model. What on earth could it be? We'll answer this question via a fictional dialogue between you and the axioms of $\mathrm{ZF} + \neg\,\mathrm{Con}\,(\mathrm{ZF})$.

You: Look, you say ZF is inconsistent, from which it follows that there's a proof in ZF that $1 + 1 = 3$. May I see that proof?

Axioms of $\mathrm{ZF} + \neg\,\mathrm{Con}\,(\mathrm{ZF})$**:** I prefer to talk about integers that encode proofs. (Actually *sets* that encode integers that encode proofs. But I'll cut you a break—you're only human, after all.)

You: Then show me the integer.

Axioms: OK, here it is: X.

You: What the hell is X?

Axioms: It's just X, the integer encoded by a set in the universe that I describe.

You: But what is X, as an *ordinary integer*?

Axioms: No, no, no! Talk to the axioms.

You: Alright, let me ask you about X. Is greater or smaller than a billion?

Axioms: Greater.

You: The $10^{10^{1,000,000,000}}$th Ackermann number?

Axioms: Greater than that too.

You: What's $X^2 + 100$?

Axioms: Hmm, let me see... Y.

You: Why can't I just add an axiom to rule out these weird 'nonstandard integers?' Let me try: for all integers X, X belongs to the set obtained by starting from 0 and...

Axioms: Ha ha! This is first-order logic. You're not allowed to talk about sets of objects—even if the objects are *themselves* sets.

You: Argh! I know you're lying about this proof that $1 + 1 = 3$, but I'll never catch you.

Axioms: That right! What Gödel showed is that we can keep playing this game forever. What's more, the infinite sequence of bizarre entities you'd force me to make up—X, Y, and so on—would then constitute a *model* for the preposterous theory $\mathrm{ZF} + \neg\,\mathrm{Con}\,(\mathrm{ZF})$.

You: But how do you know I'll never trap you in an inconsistency?

Axioms: Because if you did, the Completeness Theorem says that we could convert that into an inconsistency in the original axioms, which contradicts the obvious fact that ZF is consis—no, wait! I'm not supposed to know that! Aaahh! *[The axioms melt in a puddle of inconsistency.]*

As a corollary of Gödel's Completeness Theorem, we get the *Löwenheim-Skolem Theorem*, which actually predates Gödel: if a theory T has a model, then it has a model of at most countable cardinality. Why? Because the game above—where we keep challenging T to name the objects it says exist, and T responds by 'cooking new objects to order'—lasts at most countably many steps, since each 'challenge' can be expressed as a finite string. And the Completeness Theorem guarantees that the final result will be a model for T, assuming T was consistent.

In fact, it's known that any theory that has arbitrarily large finite models has an infinite model, and any theory that has an infinite model has models of whatever infinite cardinality we want. This is already a tip-off that the first-order sentences game can't answer any question we might ask. It can't even tell us how many objects are in the universe!

[5] For suppose Con (ZF) could be proved in ZF. It's easy to prove in ZF that Con (ZF) implies G (ZF), where G (ZF) is the famous Gödel sentence that says "I am not provable in ZF." But then we would have a proof in ZF of G (ZF), so ZF would be inconsistent. (Indeed, Con (ZF) and G (ZF) are easily seen to be equivalent.)

2.2 Proving Independence

Moving closer to the subject of the survey, how does one prove a statement independent of a first-order theory? There are actually two methods: *consistency strength*, and *relative consistency.*

Let's start with consistency strength. If we remove the Axiom of Infinity from ZF, we get a theory equivalent to Peano Arithmetic (PA). Now, it's not hard to see that $\mathrm{ZF} \vDash \mathrm{Con}(\mathrm{PA})$; that is, $\mathrm{Con}(\mathrm{PA})$ is a theorem of ZF. The reason is that in ZF, there exist infinite sets—for example, the set of all finite sets—that we can take as models for PA. On the other hand, Gödel tells us that $\mathrm{ZF} \nvDash \mathrm{Con}(\mathrm{ZF})$. If we want to prove $\mathrm{Con}(\mathrm{ZF})$, we have to postulate an infinity bigger than anything definable in ZF—a so-called 'large cardinal.' (When set theorists say large, they *mean* large.) If LC asserts the existence of such a cardinal, then $\mathrm{ZF} + \mathrm{LC} \vDash \mathrm{Con}(\mathrm{ZF})$. So we can rank the theories in a 'consistency strength hierarchy,' from PA to ZF to $\mathrm{ZF} + \mathrm{LC}$. Notice that in PA, we can't even prove that $\mathrm{Con}(\mathrm{PA}) \Longrightarrow \mathrm{Con}(\mathrm{ZF})$. For then we could also prove that in ZF, and since $\mathrm{ZF} \vDash \mathrm{Con}(\mathrm{PA})$, we'd have $\mathrm{ZF} \vDash \mathrm{Con}(\mathrm{ZF})$, contradiction.

In general, to show that a statement is unprovable in some theory, we argue that if it wasn't, then we could collapse two levels of the consistency strength hierarchy—and thereby violate the Incompleteness Theorem. As another application, you might wonder: suppose $\mathsf{P} \neq \mathsf{NP}$ were unprovable in a formal system such as ZF. Maybe we could still prove $\mathsf{P} \neq \mathsf{NP}$ by *assuming* from the outset (that is, adding as an axiom) that ZF does prove $\mathsf{P} \neq \mathsf{NP}$? A result called Löb's Theorem uses consistency strength to rule out this strange possibility. (I prefer to call it the "You-Had-The-Mojo-All-Along Theorem.")

Theorem 1 (Löb) *If sentence A can be proven in* $\mathrm{ZF} + (\mathrm{ZF} \vDash A)$*, then it can also be proven in* ZF *alone.*

Proof. If ZF is inconsistent then it can prove anything, so assume $\mathrm{Con}(\mathrm{ZF})$. Suppose ZF proves $(\mathrm{ZF} \vDash A) \Longrightarrow A$. Then $\mathrm{ZF} + \urcorner A \vDash (\mathrm{ZF} \nvDash A)$. Suppose also that we *need* the axiom $(\mathrm{ZF} \vDash A)$ to prove A—that is, that $\mathrm{ZF} \nvDash A$. Then since ZF is consistent, $\mathrm{ZF} + \urcorner A$ is also consistent. But $\mathrm{ZF} + \urcorner A$ proves $\mathrm{ZF} \nvDash A$, which is equivalent to $\mathrm{Con}(\mathrm{ZF} + \urcorner A)$. So $\mathrm{ZF} + \urcorner A \vDash \mathrm{Con}(\mathrm{ZF} + \urcorner A)$, which contradicts the consistency of $\mathrm{ZF} + \urcorner A$. ■

The other method for proving independence is *relative consistency.* As Shelah [59] put it, consistency strength makes the universe of logical theories 'taller,' while relative consistency makes it 'fatter.' Relative consistency was what Gödel [20] and Cohen [13] used to prove the independence of the Axiom of Choice (AC) and Continuum Hypothesis (CH) from ZF. Recall that AC is the assertion that, given a set x of nonempty, pairwise disjoint sets, there exists a set that shares exactly one element with each set in x. CH is the assertion that there's no set of intermediate cardinality between the integers and the sets of integers. Then Gödel proved that

$$\mathrm{Con}(\mathrm{ZF}) \Longrightarrow (\mathrm{Con}(\mathrm{ZF} + \mathrm{AC}) \wedge \mathrm{Con}(\mathrm{ZF} + \mathrm{CH})),$$

while Cohen proved that

$$\mathrm{Con}(\mathrm{ZF}) \Longrightarrow (\mathrm{Con}(\mathrm{ZF} + \urcorner \mathrm{AC}) \wedge \mathrm{Con}(\mathrm{ZF} + \urcorner \mathrm{CH})).^{6}$$

In other words, by *starting* with a model for ZF, we get another model for ZF with specific properties we want—for example, that CH is true, or that AC is false. For this reason, it's clear that $\mathrm{Con}(\mathrm{ZF})$ is not a theorem of, say, $\mathrm{ZF} + \mathrm{CH}$—for if it were, then we'd have

$$\mathrm{ZF} + \mathrm{CH} \vDash \mathrm{Con}(\mathrm{ZF} + \mathrm{CH}),$$

[6] Amusingly, both of these implications are actually theorems of PA—since to prove them, one ultimately just talks about axioms, which are finite strings of symbols.

therefore $\mathrm{ZF}+\mathrm{CH}$ would be inconsistent, therefore ZF itself would be inconsistent. So unlike with the method of consistency strength, adding CH doesn't lead to a 'stronger' theory—just a different one.

In summary, then, if we wanted to prove that $\mathsf{P} \neq \mathsf{NP}$ (or $\mathsf{P} = \mathsf{NP}$) is unprovable in some theory, there are two ways we might go about it: consistency strength or relative consistency.

3 Early Results

With the logic background out of the way, it's finally time to say something about P vs. NP. We'll start with some results from the seventies and eighties: in Section 3.1, oracle independence theorems due to Hartmanis and Hopcroft [26] and Hartmanis [25]; and in Section 3.2, non-oracle independence theorems (concerning weak logical theories) due to DeMillo and Lipton [15] and Sazanov [58].

3.1 Oracles

Given a Turing machine M, let $L(M)$ be the language accepted by M. The following was shown in [24, 26].

Theorem 2 (Hartmanis-Hopcroft) *There exists a Turing machine M that halts on every input, such that relative to the oracle $L(M)$, neither $\mathsf{P} = \mathsf{NP}$ nor $\mathsf{P} \neq \mathsf{NP}$ is provable in* ZF*, assuming* ZF *is consistent.*

Proof. The language $L(M)$ will turn out to be the empty set (that is, M always rejects). The catch is that its emptiness can't be *proven* in ZF. Here's how it works: by Baker, Gill, and Solovay [4], there exists an oracle A relative to which $\mathsf{P} = \mathsf{NP}$, and another oracle B relative to which $\mathsf{P} \neq \mathsf{NP}$.[7] Both of these oracles are computable. Let $M_1, M_2, \ldots$ be a standard enumeration of Turing machines, and let $P_1, P_2, \ldots$ be a standard enumeration of ZF proofs. Then we define M as follows: given an integer x as input, M accepts if either

(1) There's a proof that $\mathsf{P}^{L(M)} = \mathsf{NP}^{L(M)}$ among $P_1, \ldots, P_x$, and $x \in B$; or

(2) There's a proof that $\mathsf{P}^{L(M)} \neq \mathsf{NP}^{L(M)}$ among $P_1, \ldots, P_x$, and $x \in A$.

You might complain that we used M in the definition of M. What justifies this, though, is the Recursion Theorem, which says that we can always assume without loss of generality that a program has access to its own code. The idea is a generalization of the famous self-printing program:

```
Print the following twice, the second time in quotes.
``Print the following twice, the second time in quotes.''
```

Now suppose there exists a ZF proof that $\mathsf{P}^{L(M)} = \mathsf{NP}^{L(M)}$. Then beyond some finite point, the oracle $L(M)$ equals B, and thus $\mathsf{P}^{L(M)} \neq \mathsf{NP}^{L(M)}$. Similarly, if there exists a ZF proof that $\mathsf{P}^{L(M)} \neq \mathsf{NP}^{L(M)}$, then beyond some finite point $L(M)$ equals A, and thus $\mathsf{P}^{L(M)} = \mathsf{NP}^{L(M)}$. We conclude that assuming ZF is consistent, there is no ZF proof of either statement. ■

The Hartmanis-Hopcroft theorem is called 'representation-dependent,' since the independence really relies on the Turing machine M by which we're *told* about the oracle, not the oracle itself.

[7] Basically, A is just a PSPACE-complete language, which collapses both P and NP to PSPACE; while B is an infinite sequence of hard search problems, constructed by diagonalizing over all polynomial-time Turing machines.

If someone told us that $L(M)$ was empty, then we'd be back to the standard (unrelativized) P vs. NP question, which of course isn't known to be independent of ZF. So you might wonder: can we construct a computable oracle $\mathcal{O}$, such that $\mathsf{P}^{\mathcal{O}}$ vs. $\mathsf{NP}^{\mathcal{O}}$ is independent of ZF, no matter which Turing machine is used to specify $\mathcal{O}$?

The answer turns out be yes, as shown by Hartmanis [25] and also by Kurtz, O'Donnell, and Royer [33]. Skipping technicalities, the intuition is as follows. We construct $\mathcal{O}$ so that for almost all input lengths, $\mathcal{O}$ collapses P and NP. But for a few, widely-separated input lengths, call them $f(1), f(2), f(3), \ldots$, $\mathcal{O}$ separates P and NP, and that is enough to guarantee that $\mathsf{P} \neq \mathsf{NP}$ relative to $\mathcal{O}$, since there are infinitely many such lengths. The catch now is that the function f, though computable, grows *so* quickly that one can't prove in ZF that f is total, or even that f is defined for infinitely many values of n. (We'll have more to say about such fast-growing computable functions in Section 5.) If f were defined for only finitely many n, then $\mathcal{O}$ would differ only finitely from an oracle that collapses P and NP, so of course we'd have $\mathsf{P}^{\mathcal{O}} = \mathsf{NP}^{\mathcal{O}}$. That's what makes $\mathsf{P}^{\mathcal{O}}$ vs. $\mathsf{NP}^{\mathcal{O}}$ independent of ZF.

3.2 Weak Independence

If we want to study the provability of P vs. NP, a natural idea is to show the problem independent of extremely *weak* logical theories. If nothing else, that would at least rule out any approach to P vs. NP that can be formalized within those theories. Already in the seventies and early eighties, there was a good deal of work in this direction [15, 28, 58]; we'll look at two examples.

DeMillo and Lipton [15] show $\mathsf{P} \neq \mathsf{NP}$ unprovable in a fragment of number theory they call ET. The objects of ET are integers, while the language consists of the functions $x+y$, $x-y$, $x \cdot y$, $\min\{x,y\}$, $\max\{x,y\}$, and c^x (where c is a constant); as well as all polynomial-time computable *predicates* (but not functions). For example, given integers x, y, z, we can write a predicate that tests whether $2^x + 3^y = 5^z$. We can do this even though $|2^x| + |3^y| + |5^z|$ is exponentially larger than $|x| + |y| + |z|$—since it's only the predicates (in this case equality testing), not the arithmetic operations, that need to be polynomial-time. Finally, the axioms of ET are all true sentences of the form $\forall x \ A(x)$, where A is a quantifier-free predicate. (Here 'true' means true for the ordinary integers.) In one sense the axioms are absurdly powerful—among other things, they give us Fermat's Last Theorem for free (for each fixed value of n)—but in another sense the axioms are weak, since they have only one universal quantifier. The proof that $\mathsf{P} \neq \mathsf{NP}$ is unprovable in ET would take us too far afield. I'll just say that a key part is to bound the possible growth rates of functions expressible with the allowed arithmetic operations.[8]

Sazanov [58] shows exponential lower bounds to be unprovable in a quite different theory, by using the method of consistency strength discussed in Section 2.2. He defines a theory T in which the objects are finite binary strings, and we have available all polynomial-time computable functions and predicates. Call a sentence 'true' if it's true in the standard model. Then the axioms consist of all true first-order sentences, with all quantifiers *bounded* (except for an initial universal quantifier). A bounded quantifier has the form $\forall x \leq a$ or $\exists x \leq a$, where a doesn't depend on x and $\leq$ denotes lexicographic ordering of strings. This is a richer axiom set than that considered by DeMillo and Lipton; on the other hand, the arithmetic operations—and in particular the exponential function—aren't available.

[8] As a further result, DeMillo and Lipton show that assuming $\mathsf{P} \neq \mathsf{NP}$, the statement $\mathsf{NP} \neq \mathsf{coNP}$ is consistent with another fragment of number theory called PT. This represents a direction on which there hasn't been much work to my knowledge: providing "evidence" for widely-believed complexity assumptions (such as $\mathsf{NP} \neq \mathsf{coNP}$, or $\mathsf{P} \neq \mathsf{NP}$), by showing that the assumptions are *consistent* with weak logical theories (or equivalently, that their negations are unprovable in those theories).

Given strings x and y, we can define a predicate $\mathrm{EXP}(x,y)$, that tests whether $2^{|x|} \leq |y|$ (where $|x|$ is the length of x). However, we don't have an axiom E saying that $\forall x \exists y \ \mathrm{EXP}(x,y)$. In other words, if our quantifiers range over strings of length n, then strings of length 2^n are outside our "universe of discourse." Indeed Sazanov shows that $\mathrm{T} + \mathrm{E} \vDash \mathrm{Con}(\mathrm{T})$. Intuitively, once we have an exponentially-long string y, we can construct the set of *all* polynomial-size strings, and thereby obtain a model for T. This is analogous to the fact that $\mathrm{ZF} \vDash \mathrm{Con}(\mathrm{PA})$—the only difference being that in Sazanov's case, the Axiom of Infinity is scaled down to the "Axiom of Exponentiality" E. Continuing the analogy, we can conclude that $\mathrm{T} \nvDash \mathrm{E}$—since otherwise we'd have $\mathrm{T} \vDash \mathrm{Con}(\mathrm{T})$, contradicting the Incompleteness Theorem. Indeed, letting ACCEPT be the assertion that there exists an algorithm for SAT, Sazanov shows further that $\mathrm{T} + \mathrm{ACCEPT} \nvDash \mathrm{E}$. Now let ELB (Exponential Lower Bound) be the assertion that any algorithm for SAT requires exponential time. Then

$$\mathrm{T} + \mathrm{ACCEPT} + \mathrm{ELB} \vDash \mathrm{E}.$$

For ACCEPT tells us that for all SAT instances x, there exists a y that is the tableau of a computation of $\mathrm{SAT}(x)$ by some fixed algorithm. Meanwhile ELB tells us that $|y|$ is exponentially larger than $|x|$ infinitely often. We conclude that $\mathrm{T} \nvDash \mathrm{ELB}$; that is, T is unable to prove exponential lower bounds for SAT. It's interesting that this proof rules out only lower bounds of the form $2^{n^{\varepsilon}}$ for $\varepsilon > 0$, since we need an exponentially long string to get a model for T. For all we know, T *does* prove that SAT requires $n^{\log n}$ time.

All in all, we depart these early results feeling we learned a bit about logic, independence, and relativization, but less sure that we learned anything about P vs. NP.

4 Natural Proofs

In 1993, Razborov and Rudich [56] wrote a paper that contains, arguably, the most important insight anyone has thus far had into the P vs. NP question. Loosely speaking, they said that the reason $\mathsf{P} \neq \mathsf{NP}$ is so difficult to prove is that $\mathsf{P} \neq \mathsf{NP}$![9] I'll first review their argument; then, in Section 4.1, I'll discuss how it can be used to show the unprovability of circuit lower bounds in certain logical theories.

As mentioned in Section 1, in the 1980's people developed many 'combinatorial' techniques for proving circuit lower bounds [2, 18, 52, 61]. These techniques showed, for example, that the PARITY function requires constant-depth circuits of exponential size, provided the only gates allowed are $\vee, \wedge, \neg$ (with unbounded fanin). What Razborov and Rudich noticed was that these techniques all follow the same basic strategy. To show that a Boolean function $f : \{0,1\}^n \rightarrow \{0,1\}$ (say, PARITY) is not in complexity class $\mathcal{C}$, we do the following.

(1) Invent a 'complexity measure' μ for Boolean functions. (In [61], for example, μ is the minimum degree of an approximating polynomial over a finite field.)

(2) Show that for any function $f^* \in \mathcal{C}$, $\mu(f^*)$ is small. (Say, because each gate in a circuit can only increase μ by a small amount.)

(3) Show that $\mu(f)$ is large—from which it follows that $f \notin \mathcal{C}$.

[9] Of course, assuming $\mathsf{P} \neq \mathsf{NP}$, one could cite that as the reason why *anything* is hard to prove! But Razborov and Rudich connected the hardness of NP problems to the hardness of proving their hardness in a more interesting way.

The point is that, if the strategy above works, then it should work not only for the function f we're interested in, but for many other functions as well. For under any 'reasonable' measure of complexity, a random function is extremely complex. Intuitively, if 'simple' means having a short description of some form, then most functions can't be simple just by a counting argument! Indeed, this is how one shows that almost all Boolean functions have exponential circuit complexity, and that almost all strings have large Kolmogorov complexity. Furthermore, assuming we could analyze $\mu(f)$ well enough to carry out step (3), we presumably have an *efficient procedure* to estimate μ—where 'efficient' means using time polynomial in 2^n, the size of the truth-table of f.

Assuming our proof is 'natural' in the above senses, Razborov and Rudich show that we could turn the proof on its head, and use it to break pseudorandom functions computable by the class $\mathcal{C}$. The idea is simple: if $f^* \in \mathcal{C}$, then $\mu(f^*)$ must be small, while if f is a genuinely random function, then $\mu(f)$ is large with non-negligible probability. A subtlety is that the procedure to estimate μ took time polynomial in 2^n, not in n. To deal with this, we choose the seed for the pseudorandom function $f^* \in \mathcal{C}$ to have length n^c, where c is a large constant. Then distinguishing f^* from random by testing every seed exhaustively would take time roughly 2^{n^c}. We succeeded instead in time $2^{O(n)}$, or $2^{N^{1/c}}$ in terms of the seed length $N = n^c$.

Why is this considered implausible? In the case $\mathcal{C} = \mathsf{P/poly}$ (that is, the lower bound proof works against polynomial-size circuits), the evidence comes cryptographic reductions. Håstad et al. [27] showed that given any one-way function, we can construct a pseudorandom generator that's roughly as hard to break. Earlier, Goldreich, Goldwasser, and Micali [21] had shown that given any pseudorandom generator, we can construct a pseudorandom *function* that's roughly as hard to break. Putting these together, we find that our lower bound proof would give us a way to break any one-way function (including factoring, discrete logarithm, etc.), in time $2^{O(n^{\varepsilon})}$ for any $\varepsilon > 0$. By comparison, the best known factoring algorithm is conjectured to run in time roughly $2^{n^{1/3}}$ [35].

If $\mathcal{C}$ is a smaller class, for instance the class TC^0 of constant-depth polynomial-size threshold circuits, then the reductions of [27] and [21] can't be carried out. However, Naor and Reingold [40] gave a direct construction of a pseudorandom function in TC^0, which is provably as hard as factoring and discrete logarithm. It follows that any natural proof that a function isn't in TC^0 would yield an algorithm for these problems much faster than any known today.

A final observation is that the stronger the lower bound we obtain for f, the stronger the *upper* bound it implies for breaking pseudorandom functions. For example, if our lower bound puts f outside of $\mathcal{C} = \mathsf{DTIME}(t(n))$ (where t is superpolynomial), then we can take the seed for the pseudorandom function to have length about $t(n)$. So in terms of the seed length, we get a roughly $2^{t^{-1}(n)}$ upper bound on the time needed to distinguish the function from random. Razborov and Rudich credit Wigderson with the amusing observation that no natural proof can show a better than half-exponential[10] lower bound for the discrete logarithm function, *unconditionally.* For as we show this function to be harder, we simultaneously show it to be easier!

4.1 Natural Proofs and Independence

In Section 3.2, we saw what might be called the 'first generation' of P vs. NP independence results. These sought to show $\mathsf{P} \neq \mathsf{NP}$ unprovable in weak logical theories, by (say) analyzing the possible growth rates of functions provably total in those theories. Razborov and Rudich opened the way to a 'second generation' of independence results. These focus on a finite statement, such as

[10] For $t(n)$ to be half-exponential means that $t(t(n))$ grows like 2^n. Such functions exist, but are difficult to describe (see Miltersen, Vinodchandran, and Watanabe [37]).

$$\urcorner\,\text{Circuit}_n := \text{ ``SAT}_n \text{ requires circuits of size } n^{\log n}\text{''}$$

where SAT_n is the set of SAT instances of size n. The goal is to show (perhaps under a complexity assumption) that in some proof system, $\urcorner\,\text{Circuit}_n$ requires proofs of length superpolynomial in n. That is, we want to argue against the plausibility of proving lower bounds by proving a lower bound.

The difference between the two generations of independence result highlights an important fact about P vs. NP itself. When we meet a statement—Goldbach's Conjecture, for example—that says something about every natural number n, we usually assume the statement is trivial for any *fixed* n. So the sole problem is to find an argument that works for all n. For $\urcorner\,\text{Circuit}_n$, though, we have no idea how to prove even $\urcorner\,\text{Circuit}_{50}$ or $\urcorner\,\text{Circuit}_{100}$. The resources needed to prove $\urcorner\,\text{Circuit}_n$ grow too quickly as a function of n. Both natural proofs and the second-generation independence results focus exclusively on this latter 'meta-complexity' problem—bypassing the issue of how we'd generalize to all n, even if we *could* prove $\urcorner\,\text{Circuit}_{50}$, $\urcorner\,\text{Circuit}_{100}$, and so on.[11]

To do justice to the second-generation independence results would require another survey altogether, and indeed such surveys exist [10, 46, 49]. Here I'll just sketch the main ideas. Many of the results involve a set of theories called *bounded arithmetic* [9, 10, 11]. In these theories, the objects are natural numbers, and we have available the constants 0 and 1 and the functions $+$, $\times$, $\leq$, $\lfloor x/2 \rfloor$, $|x|$, and $\#$, where $|x| = \lceil \log_2 (x+1) \rceil$ and $x\#y = 2^{|x|\cdot|y|}$. We can use the *bounded quantifiers* $\forall x \leq t$ and $\exists x \leq t$, where t doesn't depend on x. There are standard axioms defining the functions, as well as two types of induction axiom:

$$\text{IND} : [A(0) \wedge \forall x (A(x) \Longrightarrow A(x+1))] \Longrightarrow \forall x A(x)$$
$$\text{PIND} : [A(0) \wedge \forall x (A(\lfloor x/2 \rfloor) \Longrightarrow A(x))] \Longrightarrow \forall x A(x)$$

Here A is a sentence that may contain additional variables. Clearly the IND axioms imply the PIND ones, but the converse is believed to be false. You can think of PIND as 'lazy induction': it forces us, the theorem provers, to do the hard work of showing that $A(\lfloor x/2 \rfloor) \Longrightarrow A(x)$ for all x, so that it can get by with an exponentially 'shallower' induction than IND.

Now, a Σ_1-sentence is any sentence of the form $\exists x P(x)$, where the quantifier $\exists x$ is bounded. Likewise, a Σ_2-sentence has the form $\exists x \forall y P(x,y)$ where $\exists x$ and $\forall y$ are bounded, and so on, in analogy to the polynomial hierarchy PH. Then the theory T_2^i is bounded arithmetic with an IND axiom for every Σ_i-sentence A, and S_2^i is bounded arithmetic with a PIND axiom for every Σ_i-sentence A. The theories can be ordered as follows from less to more powerful:

$$\text{S}_2^1 \subseteq \text{T}_2^1 \subseteq \text{S}_2^2 \subseteq \text{T}_2^2 \subseteq \cdots$$

A major open problem is whether the above inclusions are proper.

The bounded arithmetic theories are weaker than Sazanov's theory T, which we considered in Section 3.2. However, the independence results that can be shown are stronger: they rule out all *superpolynomial* lower bounds, not just exponential lower bounds as in Sazanov's case. Moreover,

[11] So the results really talk about *proof-sequences* $\{P_n\}_{n\geq 1}$, only indirectly about proofs. The notion of a proof-sequence helps to resolve the following puzzle. Suppose any pseudorandom function family in TC^0 could be broken by a quantum computer (not so implausible, given that the only candidate family, due to Naor and Reingold [40], is based on factoring and discrete logarithm). Might we then have a "quantum natural proof" that $\mathsf{TC}^0 \neq \mathsf{P}$? The answer is that at best, we'd get a quantum Turing machine that on input n, verifies in time $2^{O(n)}$ (say) that CVP_n, the Circuit Value Problem of size n, requires TC^0 circuits of size $n^{\log n}$. For a natural proof of $\mathsf{TC}^0 \neq \mathsf{P}$, we'd still need a *classical* algorithm to break pseudorandom functions in TC^0.

as discussed earlier, for each particular n they give a lower bound on the length of a proof of $\urcorner \operatorname{Circuit}_n$.

Razborov [55] showed that if there exists a pseudorandom generator that requires circuits of size $\Omega\left(2^{n^{\varepsilon}}\right)$ to break for some $\varepsilon > 0$ (the same assumption used for natural proofs), then $\mathsf{NP} \not\subseteq \mathsf{P/poly}$ is unprovable in the theory S_2^2. Indeed, no class of functions definable in bounded arithmetic can then be proven to lie outside $\mathsf{P/poly}$, so in particular, $\Sigma_k^{\mathsf{P}} \not\subseteq \mathsf{P/poly}$ is unprovable for all k. Also, Razborov showed that $\mathsf{NP} \not\subseteq \mathsf{P/poly}$ is unprovable in S_2^1, under the weaker assumption that there exists a pseudorandom generator computable in polynomial time that requires bounded fan-in circuits of depth $\Omega\left(n^{\varepsilon}\right)$ to break for some $\varepsilon > 0$.

The key to the above results is what's called the *Efficient Interpolation Property* (EIP). Suppose we have a conjunction $\varphi = A(x,z) \wedge B(y,z)$, where x, y, z are sets of variables (z being a set of 'shared' variables). If φ is unsatisfiable, then for every fixed choice of z, either $A(x,z)$ or $B(y,z)$ must be unsatisfiable. Given z as input, the *interpolation problem* is to return which of A or B is unsatisfiable—or if both are unsatisfiable, then either A or B. We say a proof system $\mathcal{S}$ satisfies the EIP, if whenever φ has a polynomial-size refutation in $\mathcal{S}$, there's a polynomial-time algorithm that solves the interpolation problem.

What Razborov did was to create a φ for which solving the interpolation problem is as hard as breaking a pseudorandom function. Let f be a pseudorandom function mapping n^c bits to 2^n bits, and let s be any explicit function such that

$$\operatorname{Range}(f) \cap (\operatorname{Range}(f) \oplus s) = \varnothing.$$

Then for $z \in \{0,1\}^{2^n}$, define $\varphi = A(z) \wedge B(z)$ where (informally, and suppressing variables besides z)

$$\begin{aligned} A(z) &\Longleftrightarrow z \text{ belongs to } \operatorname{Range}(f), \\ B(z) &\Longleftrightarrow (z \oplus s) \text{ belongs to } \operatorname{Range}(f). \end{aligned}$$

Clearly φ is unsatisfiable. On the other hand, if we could decide which of A or B is unsatisfiable (that is, solve the interpolation problem), then we could distinguish f from a random function in time $2^{O(n)}$. It follows that in any proof system $\mathcal{S}$ satisfying the EIP, φ can't have a polynomial-size refutation assuming there exist $2^{n^{\varepsilon}}$-secure pseudorandom generators. From there it's easy to show that the tautologies $\{\urcorner \operatorname{Circuit}_n\}_{n \geq 1}$ (assuming they are tautologies!) don't have polynomial-size proofs in $\mathcal{S}$. As an example, Pudlák [45] showed that the Cutting Planes proof system satisfies the EIP, and this immediately implies that Cutting Planes can't prove $\mathsf{NP} \not\subseteq \mathsf{P/poly}$, again under the pseudorandomness assumption.

So EIP is a useful property, since it immediately gives us conditional lower bounds on a proof system's ability to prove circuit lower bounds. Thus, many researchers hoped to show the EIP for stronger proof systems, especially the so-called Extended Frege ($\mathcal{EF}$) system. Unfortunately, Krajíček and Pudlák [31] showed that the EIP fails for this system—ironically, under a cryptographic assumption like those used to prove lower bounds on proof size! The idea is simple: they created a disjunction $\varphi = A(z) \wedge B(z)$ where (again informally)

$$\begin{aligned} A(z) &\Longleftrightarrow z \text{ is an RSA encryption of a 0 bit}, \\ B(z) &\Longleftrightarrow z \text{ is an RSA encryption of a 1 bit}. \end{aligned}$$

Clearly φ is unsatisfiable. Moreover, using some number-theoretic facts, Krajíček and Pudlák showed that this unsatisfiability can be *proven* in the theories T_2^3 and $\mathrm{S}_2^1 + \Phi$, where Φ is the axiom that any integer having a Pratt certificate[12] is prime. This implies in particular that φ has a polynomial-size refutation in $\mathcal{EF}$. It follows that, if $\mathcal{EF}$ satisfied the EIP, then there would be a polynomial time algorithm to solve the interpolation problem and thereby break RSA. We conclude that $\mathcal{EF}$ doesn't satisfy the EIP, assuming RSA is secure.

I'll conclude with one example of an *unconditional* 'second-generation' independence result. In the well-known resolution proof system, we're given a set of CNF clauses that we want to prove unsatisfiable. The allowed operation is to "smash together" two previous clauses, of which one contains a literal a and the other contains $\urcorner a$, to derive a new clause. For example, given the clauses $(a \vee \urcorner b)$ and $(\urcorner a \vee c \vee d)$ we can derive $(\urcorner b \vee c \vee d)$. The *length* of a resolution proof is the number of clauses it derives prior to the empty clause. Haken [23] showed the first superpolynomial lower bound on the lengths of resolution refutations. The unsatisfiable formulas he used were based on the Pigeonhole Principle; they encoded that $n + 1$ pigeons are each in one of n holes, with no two pigeons in the same hole. Intuitively, with resolution we're stuck reasoning *locally*: "let's see, if we put this pigeon here, and moved that one there... darn, still doesn't work!" Resolution isn't powerful enough to *count* the pigeons and the holes, and see that the former are more numerous than the latter.

Recently Raz [47] improved Haken's result to show that even formulas based on the so-called 'Weak Onto Pigeonhole Principle'—that $m \gg n$ pigeons can't be assigned to n holes, with exactly one pigeon in every hole—require resolution refutations of size $2^{n^{\varepsilon}}$ for some $\varepsilon > 0$. By a reduction originally due to Razborov [51], a corollary is that the set of formulas $\{\mathrm{Circuit}_n\}_{n \geq 1}$ (under a natural propositional encoding), encoding that SAT has small circuits, don't have polynomial-size resolution refutations. Intuitively, consider a small DNF formula where each clause contains every variable (either negated or non-negated). Though it seems obvious that such a formula can't compute SAT, one does need to show that there are more 'yes' instances of SAT than there are clauses in the DNF—and that requires the Weak Onto Pigeonhole Principle!

Improving on Raz's result, Razborov has shown [54] that $\urcorner \mathrm{Circuit}_n$ has no succinct resolution proofs even when the circuits in the encoding have bounded fan-in. Using a different reduction—based on the Nisan-Wigderson pseudorandom generator [41], rather than the pigeonhole principle—Razborov has also shown [53] that $\urcorner \mathrm{Circuit}_n$ has no succinct proofs in several extensions of resolution, including PCR (Polynomial Calculus and Resolution) and k-DNF resolution, or resolution with k-DNF's instead of clauses, for $k = \varepsilon \log n$.

Taking a step back, do these results tell us anything about P vs. NP—for example, whether the problem is likely to be independent of ZF or some other strong theory? In the appendix of [48], Razborov argues that all known "explicit" techniques for circuit lower bounds can be formalized in S_2^1. Therefore, the inability even of S_2^2 to prove $\mathsf{NP} \not\subseteq \mathsf{P/poly}$ under a pseudorandomness assumption could be taken as strong evidence that those techniques are logically too weak to prove $\mathsf{P} \neq \mathsf{NP}$.

On the other hand, a telling point, mentioned in [48], is that S_2^1 can't even formalize the Shannon counting argument, which shows that almost all Boolean functions require exponential-size circuits. As a consequence, S_2^1 can't formalize Kannan's result [29] that $\mathsf{NP}^{\mathsf{EXP}} \not\subseteq \mathsf{P/poly}$, which uses the Shannon counting argument; or the Buhrman-Fortnow-Thierauf result [8] that $\mathsf{MA}_{\mathsf{EXP}} \not\subseteq \mathsf{P/poly}$, which in turn uses Kannan's result. It's clear, then, that there exist techniques not formalizable in S_2^1, that to us humans are almost the epitome of triviality (unlike many techniques that *are* formalizable in S_2^1!). Whether a theorem can be proven in a weak fragment of arithmetic might not say much about the theorem's conceptual difficulty. In one of the many ironies of P vs. NP,

[12] Pratt [44] showed that PRIMES $\in NP$; that is, every prime has a succinct proof of primality. An obvious question is whether the deterministic primality test of Agrawal, Kayal, and Saxena [1] might eliminate the need for Φ. This seems unlikely, since one would still need to *prove* the correctness of the AKS algorithm in S_2^1.

it's conceivable that the independence results will ultimately contribute to proving $\mathsf{P} \neq \mathsf{NP}$, not because of what they tell us, but because of the lower bound techniques developed to prove the independence results themselves!

5 Proving Independence

Ultimately, of course, we'd want to prove P vs. NP independent of a strong theory, one that encompasses all "ordinary" mathematical reasoning (or at least a large chunk of it). But there's a reason that's probably impossible with current techniques, which was pointed out by Kurtz, O'Donnell, and Royer [33], and (independently) Ben-David and Halevi [6].

Let's define a Π_1-sentence to be any sentence of the form, "For all x, $P(x)$," where P is function that can be proven to be recursive in Peano Arithmetic, PA. (That is, there's a Turing machine M that computes P, together with a proof in PA that M halts on every input.) Similarly, a Π_2-sentence has the form $\forall x \exists y\ P(x, y)$ where P is provably recursive, a Π_3-sentence has the form $\forall x \exists y \forall z\ P(x, y, z)$, and so on.

An amusing exercise is to classify famous unsolved math problems by 'quantifier complexity.' Goldbach's Conjecture is Π_1, since we can write a simple predicate that checks whether an integer is a sum of two primes. The Twin Primes Conjecture is Π_2: for all n, there exists a $p > n$ such that p and $p+2$ are prime. To make this a Π_1-sentence, we'd need to strengthen the conjecture by including an explicit lower bound—say, that there are at least $\log \log \log n$ twin primes less than n. The complexity of the Riemann Hypothesis (RH) is less obvious. But a result of Lagarias [34] shows that RH is equivalent to the assertion that for all positive integers n, the sum of the divisors of n is at most

$$\alpha(n) = H_n + e^{H_n} \ln H_n,$$

where $H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n}$ is the n^{th} harmonic number. This assertion is Π_1, since we can compute increasingly good upper bounds on $\alpha(n)$, dovetailing over all n.

The problem that concerns us, P vs. NP, is of course Π_2: for all Turing machines M and polynomial time bounds p, there exists a SAT instance φ such that M doesn't halt with the correct answer to φ in at most $p(n)$ steps, where n is the size of φ. The NP vs. $\mathsf{P/poly}$ problem is similarly Π_2.

What Kurtz et al. and Ben-David and Halevi noticed is that despite being a Π_2-sentence, P vs. NP can be extremely well "approximated" by Π_1-sentences. For example, let A be the assertion that "SAT does not have circuits of size $n^{\log n}$." Then A is Π_1, since for each n, we can just search exhaustively through all circuits of size $n^{\log n}$ and see whether any of them computes SAT. Moreover, A implies $\mathsf{P} \neq \mathsf{NP}$, and is very likely *true* (we'd imagine) assuming $\mathsf{P} \neq \mathsf{NP}$.

Now assuming PA is consistent, we can't prove a Π_1-sentence independent of PA without proving the sentence—since if the sentence were false, it would have a simple counterexample! (In Apostolos Doxiadis' novel *Uncle Petros and Goldbach's Conjecture* [17], the brilliant Uncle Petros misses this point completely while musing about whether Goldbach's Conjecture is formally independent.)

There's a further point, however. With the exception of Gödel's Incompleteness Theorem, all the techniques we have for proving independence from PA actually prove independence from a stronger theory, PA $+\Pi_1$. This theory consists of PA augmented by the set of all *true* Π_1-sentences. Since the true Π_1-sentences can't even be recursively enumerated, obviously PA $+\Pi_1$ is an unrealistically powerful theory. Despite its power, though, it turns out to be unable to prove even some simple Π_2-sentences.

As an example, consider Goodstein's Theorem [22]. We write a positive integer n (say 40) as a sum of powers of 2, then write the exponents as sums of powers of 2, and so on:

$$40 = 2^5 + 2^3 = 2^{2^2+2^0} + 2^{2+2^0}.$$

(This is called the base-2 *hereditary representation.*) Then we change all the 2's to 3's to obtain $3^{3^3+3^0} + 3^{3+3^0}$, then subtract 1 and re-express in base-3 hereditary representation:

$$3^{3^3+3^0} + 3^{3+3^0} - 1 = 22,876\,,792,455,041 = 3^{3^3+3^0} + 2 \cdot 3^3 + 2 \cdot 3^{2 \cdot 3^0} + 2 \cdot 3^{3^0} + 2 \cdot 3^0.$$

Then we change all the 3's to 4's, subtract 1, re-express in base-4 hereditary representation, and so on, increasing the base by 1 each time. Surprisingly, by an argument involving countable ordinals, Goodstein showed that no matter what n we start with, this process will always converge to 0. Subsequently Paris and Harrington [42] showed that Goodstein's Theorem is independent of PA—meaning that the use of countable ordinals was necessary.

Clearly Goodstein's Theorem is Π_2: for all n, there exists a t such that the process converges in t steps if we start with n. What Paris and Harrington actually showed was that Goodstein's Theorem is independent of PA $+\Pi_1$—in other words, even if we threw in all true Π_1-sentences for free, the theorem would *still* be unprovable in PA. Indeed, Kreisel [32] showed the following (see also [6]).

Lemma 3 (Kreisel) *A function is provably recursive in* PA $+\Pi_1$ *if and only if it's provably recursive in* PA*.*

Now suppose P vs. NP were independent of PA $+\Pi_1$. Then NP would need to have circuits of size $n^{\log n}$—for if it didn't, then we'd have a true Π_1-sentence (and hence theorem of PA $+\Pi_1$) that implies P $\neq$ NP. Indeed, by Kreisel's lemma, NP would need to have circuits of size $n^{\alpha(n)}$, where α is any unbounded function such that α^{-1} is provably recursive in PA. For example, we could take α to be the inverse Ackermann function. One can obtain ridiculous consequences for uniform computation also—although there, to keep the sentences Π_1, one needs to say something like, "there exists in algorithm that decides SAT in time $n^{\alpha(n)}$ for infinitely many long stretches of n's."

6 Conclusion

It's clear that a proof of P $\neq$ NP, if it exists at all, would require utterly new techniques. The results of the previous section twist the knife further by showing that, if P $\neq$ NP is unprovable in a strong theory such as Peano arithmetic, then utterly new techniques would be required to show *that.* These pessimistic results raise a disturbing possibility: could P $\neq$ NP be somehow "neither true nor false?"

Before you dismiss this question as silly, consider the case of the continuum hypothesis CH. Unlike Gödel's Incompleteness Theorem, which tells us that a sentence is true by the very fact of its unprovability, the independence results of Gödel and Cohen don't point us one way or another regarding the "truth" of CH. Cantor thought CH was true. Gödel hoped that new "self-evident" axioms would someday be discovered, which would settle once and for all whether intermediate cardinalities exist. Cohen predicted that CH might come to be seen as *obviously* false—a prediction that's partly come to pass (see, for example, a survey by Woodin [62]). But even if we reject CH, there remains the question: how many intermediate cardinalities are there? It turns out that *any* number would be consistent with ZF. Cohen thought perhaps there are uncountably many; Woodin thinks there's exactly one.

Should we lose sleep over this situation? My own feeling is that we shouldn't—but that perhaps we *should* readjust our intuitions regarding the continuum. You might think you know what an "arbitrary set of reals" means, but do you? Imagine you were a programmer, and your boss ordered you to write a C program that loops through all sets of reals, halting if it finds one of intermediate cardinality. You might be forgiven for protesting, with Kronecker, that "God created the integers; all else is the work of Man."

David Deutsch, of quantum computing fame, has argued that even mathematical statements are ultimately about physical reality—since mathematics is rooted in computation, and the laws of physics determine what is and isn't computable [16].[13] Whether or not you agree with this, it does suggest the following "physical process criterion" for mathematical truth:

> *We should expect a mathematical question to have a definite answer, if and only if we can phrase the question in terms of a physical process we can imagine.*

There might be questions that we can't phrase in terms of a physical process, but that we can nevertheless answer—for example, whether $|S| < \left|2^S\right|$ for every set S. Someone who accepted the physical process criterion would see these cases as fortuitous accidents—and would predict that closely related questions can be found which are unanswerable.

That brings us to P vs. NP. I've argued that, like the Continuum Hypothesis, $\mathsf{P} \neq \mathsf{NP}$ is a statement as much about as of mathematics—one whose truth or falsehood is almost impossible to investigate without also investigating the means of investigation. But in one crucial respect, $\mathsf{P} \neq \mathsf{NP}$ is not at all like CH. It's a Π_2-sentence. It refers, not to a transfinite realm beyond human experience, but to machines, Boolean circuits, 1's and 0's.

Here's a simple argument for why, if you accept the physical process criterion, then you should accept that any Π_α-sentence has a definite truth-value, where α is a countable ordinal. Think of interactive proofs. Naïvely, one might say that whether a language belongs to PSPACE or NEXP is irrelevant, since those classes lie so far beyond what we could compute or even verify. But if we postulate a *prover* who can interact with us—or multiple provers, or a probabilistically-checkable proof—then these questions become relevant. Analogously, although a Π_2-sentence might be beyond our capacity to verify if true or falsify if false in any finite amount of time, that no longer holds if we have an omniscient wizard around. We propose an efficient algorithm, the wizard responds with a SAT instance on which it fails, we propose another algorithm, the wizard responds with another instance, etc. This process will continue forever if and only if $\mathsf{P} \neq \mathsf{NP}$.

The same argument works for a Π_3-sentence, $\forall x \exists y \forall z\ P(x, y, z)$. We just need to dovetail. We propose x_1, the wizard responds with y_1, we check $P(x_1, y_1, z)$ for the first few values of z. Then we propose x_2, the wizard responds with y_2, we check $P(x_2, y_2, z)$ for the first few values of z, as well as $P(x_1, y_1, z)$ for a few more values of z. And so on. It's not hard to show by induction that, for any Π_α-sentence A (where α is a countable ordinal), there exists an "interactive protocol" that continues forever if and only if A is true.

It might be objected that an omniscient wizard doesn't correspond to a physical process we can imagine. But provided we can imagine playing the game against an opponent with an arbitrary fixed strategy, it seems legitimate also—at least to me!—to imagine that the opponent uses an *optimal* strategy.

[13] Most computer scientists would agree that the "extended" (or polynomial-time) Church-Turing thesis depends on physical laws, and indeed is probably falsified by quantum mechanics. Deutsch, however, goes further, and says that under different laws of physics, even the set of computable functions would in general be different.

If you're still reading this, perhaps you've really gotten into the spirit of it and asked: what if the universe is finite? Recent evidence for a positive cosmological constant (e.g. [43]), together with arguments from black hole thermodynamics [7], imply an upper bound of about 10^{122} on the maximum number of bits accessible by any one observer. Intuitively, even if the universe is spatially infinite, most of it recedes too quickly from any one observer to be seen by that observer. Now, what if your conversation with the wizard would terminate eventually in a proof that $\mathsf{P} = \mathsf{NP}$, but only after you'd exchanged strings much longer than 10^{122} bits? Would the conversation then satisfy the physical process criterion or not? I'll leave the answer to this as a matter of taste. I'll say only that my own "Platonic circle"—that is, the set of mathematical entities that I'm willing to regard as physically imaginable—doesn't extend all the way to uncountable infinities, but it *does* extend to integers past 10^{122}, and even $2^{10^{122}}$.

So I'll state, as one of the few definite conclusions of this survey, that $\mathsf{P} \neq \mathsf{NP}$ is either true or false. It's one or the other. But we may not be able to prove which way it goes, and we may not be able to prove that we can't prove it.

7 Further Reading

For the current consensus on P vs. NP, see William Gasarch's "P=?NP poll" [19]—or go to

http://www.ideosphere.com/fx-bin/Claim?claim=P!NP

to bet on whether $\mathsf{P} \neq \mathsf{NP}$ will be proved by 2010. See my Complexity Zoo at

http://www.cs.berkeley.edu/˜aaronson/zoo.html

for more about the complexity classes mentioned in this survey. My favorite reference for the material of Section 2 is the book of Cohen [13]. For a definition of Cutting Planes and other proof systems mentioned in Section 4.1, see the survey of Beame and Pitassi [5]. Finally, there are interesting discussions about the logical status of P vs. NP on the Foundations of Mathematics (FOM) mailing list; see

http://www.cs.nyu.edu/pipermail/fom/2001-August/005045.html.

8 Acknowledgments

I thank Lance Fortnow and Alexander Razborov for their comments (even though they don't agree with everything I say here!).

References

[1] M. Agrawal, N. Kayal, and N. Saxena. Primes is in P, submitted, 2002.
www.cse.iitk.ac.in/users/manindra/primality.ps.

[2] M. Ajtai. Σ_1^1 formulae on finite structures, *Annals of Pure and Applied Logic* 24:1–48, 1983.

[3] L. Babai, L. Fortnow, and C. Lund. Nondeterministic exponential time has two-prover interactive protocols, *Computational Complexity* 1(1):3–40, 1991.
www.neci.nec.com/homepages/fortnow/papers/mip2.ps.

[4] T. P. Baker, J. Gill, and R. Solovay. Relativizations of the P=?NP question, *SIAM Journal on Computing* 4(4):431–442, 1975.

[5] P. Beame and T. Pitassi. Propositional proof complexity: past, present, and future, *Bulletin of the EATCS* 65:66–89, June 1998. external.nj.nec.com/homepages/fortnow/beatcs/column65.ps.

[6] S. Ben-David and S. Halevi. On the independence of P versus NP, Technical Report #699, Technion, 1991. www.cs.technion.ac.il/˜shai/ph.ps.gz.

[7] R. Bousso. Positive vacuum energy and the N-bound, *J. High Energy Phys.* 0011:038, 2000. www.arxiv.org/abs/hep-th/0010252.

[8] H. Buhrman, L. Fortnow, and Thierauf. Nonrelativizing separations, in *Proceedings of IEEE Complexity'98*, pp. 8–12, 1998. www.neci.nec.com/homepages/fortnow/papers/nonrel.ps.

[9] S. Buss. *Bounded Arithmetic*, Bibliopolis, 1986.

[10] S. Buss. Bounded arithmetic and propositional proof complexity, in *Logic of Computation* (H. Schwichtenberg, ed.), pp. 67–122, Springer-Verlag, 1997.
math.ucsd.edu/˜sbuss/ResearchWeb/marktoberdorf95/index.html.

[11] S. Buss. Bounded arithmetic, cryptography, and complexity, *Theoria* 63:147–167, 1997.
math.ucsd.edu/˜sbuss/ResearchWeb/theoria/index.html.

[12] Clay Math Institute. Millennium Prize Problems. www.claymath.org/Millennium_Prize_Problems.

[13] P. J. Cohen. *Set Theory and the Continuum Hypothesis*, Addison-Wesley, 1966.

[14] S. A. Cook. The complexity of theorem-proving procedures, in *Proceedings of ACM STOC'71*, pp. 151–158, 1971.

[15] R. E. DeMillo and R. J. Lipton. The consistency of P=NP and related problems within fragments of number theory, in *Proceedings of ACM STOC'79*, pp. 153–159, 1979.

[16] D. Deutsch. *The Fabric of Reality*, Penguin, 1997.

[17] A. Doxiadis. *Uncle Petros and Goldbach's Conjecture*, Bloomsbury, 2001.

[18] M. Furst, J. B. Saxe, and M. Sipser. Parity, circuits, and the polynomial time hierarchy, *Math. Systems Theory* 17:13–27, 1984.

[19] W. Gasarch. The P=?NP poll, *SIGACT News* 36, 2002. www.cs.umd.edu/users/gasarch/papers/poll.ps.

[20] K. Gödel. *The Consistency of the Axiom of Choice and of the Generalized Continuum-Hypothesis with the Axioms of Set Theory*, Princeton University Press, 1940.

[21] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions, *J. of the ACM* 33:792–807, 1986.

[22] R. Goodstein. On the restricted ordinal theorem, *J. of Symbolic Logic* 9:33–41, 1944.

[23] A. Haken. The intractability of resolution, *Theoretical Computer Science* 39:297–308, 1985.

[24] J. Hartmanis. *Feasible Computations and Provable Complexity Problems*, SIAM, 1978.

[25] J. Hartmanis. Independence results about context-free languages and lower bounds, *Information Proc. Lett.* 20(5):241–248, 1985.

[26] J. Hartmanis and J. Hopcroft. Independence results in computer science, *SIGACT News* 8(4):13–24, 1976.

[27] J. Håstad, R. Impagliazzo, L. Levin, and M. Luby. A pseudorandom generator from any one-way function, *SIAM J. Comput.* 28, pp. 1364–1396, 1999. www.nada.kth.se/˜johanh/prgfromowf.ps.

[28] D. Joseph and P. Young. Independence results in computer science?, *J. Comp. Sys. Sci.* 23:311–338, 1981.

[29] R. Kannan. Circuit-size lower bounds and non-reducibility to sparse sets, *Information and Control* 55:40–56, 1982.

[30] R. M. Karp. Reducibility among combinatorial problems, in *Complexity of Computer Computations* (R. Miller and J. Thatcher, eds.), Plenum Press, pp. 85–104, 1972.

[31] J. Krajíček and P. Pudlák. Some consequences of cryptographical conjectures for S_1^2 and EF, in *Logic and Computational Complexity* (E. D. Leviant, ed.), Springer LNCS 960, pp. 210–220, 1995. www.math.cas.cz/˜krajicek/j-crypto.ps.gz.

[32] G. Kreisel. On the concepts of completeness and interpretation of formal systems, *Fundamenta Mathematicae* 39:103–127, 1952.

[33] S. Kurtz, M. J. O'Donnell, and S. Royer. How to prove representation-independent independence results, *Information Proc. Lett.* 24, pp. 5–10, January 1987.

[34] J. C. Lagarias. An elementary problem equivalent to the Riemann hypothesis, manuscript, 2000. www.arxiv.org/abs/math.NT/0008177.

[35] A. K. Lenstra and H. W. Lenstra Jr. *The Development of the Number Field Sieve*, Springer-Verlag, 1993.

[36] L. Levin. Universal search problems, *Problemy Peredachi Informatsii* 9(3):265–266 (in Russian), 1973. Partial English translation in B. A. Trakhtenbrot, A survey of Russion approaches to *perebor*, *Annals of the History of Computing* 6(4):384–400, 1984.

[37] P. B. Miltersen, N. V. Vinodchandran, and O. Watanabe. Super-polynomial versus half-exponential circuit size in the exponential hierarchy, in *Proceedings of COCOON'1999*, pp. 210–220, 1999. www.brics.dk/˜bromille/Papers/cocoon.ps.

[38] R. Monk. *Ludwig Wittgenstein: The Duty of Genius*, Vintage, 1991.

[39] K. Mulmuley and M. Sohoni. Geometric complexity theory I: an approach to the P vs. NP and related problems, *SIAM J. Comput.* 31(2):496–526, 2001. http://people.cs.uchicago.edu/˜mulmuley/part1.ps.

[40] M. Naor and O. Reingold. Number-theoretic constructions of efficient pseudorandom functions, in *Proceedings of IEEE FOCS'97*, pp. 458–467, 1997.
www.wisdom.weizmann.ac.il/˜naor/PAPERS/gdh.ps.

[41] N. Nisan and A. Wigderson. Hardness vs. randomness, *J. Comp. Sys. Sci.* 49(2):149–167, 1994.
www.math.ias.edu/~avi/PUBLICATIONS/MYPAPERS/NOAM/HARDNESS/final.ps.

[42] J. Paris and L. Harrington. A mathematical incompleteness in Peano arithmetic, *Handbook of Mathematical Logic* (J. Barwise, ed.), pp. 1133–1142, North-Holland, 1977.

[43] S. Perlmutter and 32 others (Supernova Cosmology Project). Measurements of Ω and Λ from 42 high-redshift supernovae, *Astrophysical Journal* 517(2):565–586, 1999.
www.arxiv.org/abs/astro-ph/9812133.

[44] V. R. Pratt. Every prime has a succinct certificate, *SIAM J. Comput.* 4:214–220, 1975.

[45] P. Pudlák. Lower bounds for resolution and cutting planes proofs and monotone computations, *J. of Symbolic Logic* 62(3):981–998, 1997. www.math.cas.cz/~pudlak/interpol.ps.

[46] R. Raz. $\mathsf{P} \neq \mathsf{NP}$, propositional proof complexity, and resolution lower bounds for the weak pigeonhole principle, in *Proceedings of ICM'2002* (International Congress of Mathematicians), Vol. III, pp. 685–693, 2002. www.wisdom.weizmann.ac.il/~ranraz/publications/Pchina.ps.

[47] R. Raz. Resolution lower bounds for the weak pigeonhole principle, in *Proceedings of ACM STOC'2002*, pp. 553–562, 2002. www.wisdom.weizmann.ac.il/~ranraz/publications/Pwphpres.ps.

[48] A. A. Razborov. Bounded arithmetic and lower bounds in Boolean complexity, in *Feasible Mathematics II* (P. Clote and J. Remmel, eds.), Birkhäuser, pp. 344–386, 1995. genesis.mi.ras.ru/~razborov/bobo.ps.

[49] A. A. Razborov. Lower bounds for propositional proofs and independence results in bounded arithmetic, in *Proceedings of ICALP'1996*, pp. 48–62, 1996. genesis.mi.ras.ru/~razborov/icalp.ps.

[50] A. A. Razborov. Lower bounds for the monotone complexity of some Boolean functions, *Soviet Math. Doklady* 31:354–357, 1985.

[51] A. A. Razborov. Lower bounds for the polynomial calculus, *Computational Complexity* 7(4):291–324, 1988. genesis.mi.ras.ru/~razborov/polynom.ps.

[52] A. A. Razborov. Lower bounds on the size of bounded-depth networks over a complete basis with logical addition, *Math. Notes. of the Acad. Sci. of USSR* 41(4):333–338, 1987.

[53] A. A. Razborov. Pseudorandom generators hard for k-DNF resolution and polynomial calculus resolution, manuscript, 2003. genesis.mi.ras.ru/~razborov/res_k.ps.

[54] A. A. Razborov. Resolution lower bounds for perfect matching principles, to appear in *J. Comp. Sys. Sci.* Preliminary version in *Proceedings of IEEE Complexity'2002*, pp. 29–38, 2002. genesis.mi.ras.ru/~razborov/matching.ps.

[55] A. A. Razborov. Unprovability of lower bounds on circuit size in certain fragments of bounded arithmetic, *Izvestiya Math.* 59(1):205–227, 1995.

[56] A. A. Razborov and S. Rudich. Natural proofs, *J. Comp. Sys. Sci.* 55:24–35, 1997.
www-2.cs.cmu.edu/~rudich/papers/natural.ps.

[57] K. W. Regan. Understanding the Mulmuley-Sohoni approach to P vs. NP, *Bulletin of the EATCS* 78, October 2002. external.nj.nec.com/homepages/fortnow/beatcs/column78.ps.

[58] V. Yu. Sazanov. A logical approach to the problem "P=NP?", in *Mathematical Foundations of Computer Science*, Springer LNCS 88, pp. 562–575, 1980.

[59] S. Shelah. Logical dreams, manuscript, 2002. www.arxiv.org/abs/math.LO/0211398.

[60] M. Sipser. The history and status of the P versus NP question, in *Proceedings of ACM STOC'92*, pp. 603–618, 1992. www.cs.berkeley.edu/˜luca/cs278/papers/sipser.ps.

[61] R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity, in *Proceedings of ACM STOC'87*, pp. 77–82, 1987.

[62] H. Woodin. The continuum hypothesis, part I, *Notices of the AMS* 48(6), June/July 2001. www.ams.org/notices/200106/fea-woodin.pdf.