A bottom-up approach to foundations of mathematics *

Pavel Pudlák**

Mathematical Institute
Academy of Sciences
Prague, Czech Republic

1. Introduction

There are two basic properties of a logical system: consistency and completeness. These two properties are important for systems used for particular special purposes (say, in artificial intelligence), as well as for systems proposed as foundations of the all of mathematics. Therefore, any systematic study of the foundations of mathematics should address questions of consistency and completeness. Gödel's theorems provide negative information about both: any reasonable sufficiently strong theory is unable to demonstrate its own consistency and any such system is incomplete.

There has been impressive success in proving independence results for set theory. Cohen's forcing method and its boolean valued version was applied to solve the continuum problem and to prove a lot of other sentences to be independent of set theory. This may give the impression that logic is doing very well in studying the independence phenomenon which is only partly true. The known independent sentences in set theory express statements about infinite sets. The type of the infiniteness in these independence results is, in a sense, of higher order than in classical mathematics. For instance, the continuum hypothesis talks about the cardinality of the set of real numbers, which is in any case uncountable. While most of classical mathematics is about real numbers, almost everything there can be encoded using only a countable number of elements, hence can be expressed as a statement about natural numbers. As an example, consider a continuous real function. Clearly, such a function is described by its values on rationals, thus statements about continuous functions can be written as arithmetical formulas. For arithmetical formulas, current methods of logic almost completely fail. An exception are Paris-Harrington type independence results [28]. Still it is true that no problem from classical mathematics (i.e. number theory, algebra, calculus) was proved independent from a logical theory before it had actually been solved. Furthermore, it seems that even the most difficult classical results in number

* This paper is in its final form and no similar paper has been or is being submitted elsewhere.

** Partially supported by grant No. A1019602 of AVČR and US-Czechoslovak Science and Technology Program grant No. 93025

theory and finite combinatorics can be proved in Peano arithmetic, a theory which is much weaker than Zermelo-Fraenkel set theory.

Therefore one of the principal goals of mathematical logic should be development of new techniques for proving independence. Naturally we should start with weak theories and extend our techniques to stronger ones. This is what I referred to in the title as *the bottom-up approach.* We can view the independence result of Paris and Harrington for Peano arithmetic as the first result in this direction. Later most of the research concentrated on weaker systems called bounded arithmetic. The reason for studying bounded arithmetic is not only the fact that it is much weaker. Bounded arithmetic is interesting mainly because of its connection to computational complexity. There seems to be a correspondence between various theories from the class of bounded arithmetic and complexity classes. This correspondence manifests itself in many ways; e.g. to separate two theories looks to be almost as hard as to separate the corresponding complexity classes.

It turned out that bounded arithmetic is not the very bottom level from which one should start. Independence and separation problems for bounded arithmetic can be further reduced to combinatorial problems about propositional calculus, namely, questions about the *lengths* of proofs. Here we come even closer to computational complexity, as proofs in propositional calculus are essentially nondeterministic computations.

Concerning the consistency question there has been practically no progress whatsoever. Maybe, it is because this is not a mathematical question and we should discard it for this reason. However the sentences expressing consistency of a theory are a very important tool in proof theory. They can be used to separate a weak theory from a strong one, but not in the case of bounded arithmetic, as we shall see below.

There is a lot of activity around bounded arithmetic, the complexity of propositional calculus and many questions studied in theoretical computer science are related to it too. The purpose of this paper is to survey some results which should give an idea to an outsider of what is going on in this field and explain motivations for the studied problems. We recommend [3, 5, 15, 11, 34] to those who want to learn more about this subject.

2. Basic concepts

2.1

The basic theory used in logic for studying natural numbers is the so called *Robinson's arithmetic,* or Q, which is axiomatized by the following axioms:

$$S(x) \neq 0,\ S(x) = S(y) \to x = y,\ x \neq 0 \to \exists y(x = S(y)),$$

$$x + 0 = x,\ x + S(y) = S(x + y),\ x \cdot 0 = 0,\ x \cdot S(y) = x \cdot y + x,$$

$$x \leq y \equiv \exists z(z + x = y).$$

These axioms express the inductive properties of the operations of the successor S, addition and multiplication, the last axiom is just a definition of $\leq$. The theory is useless for practical purposes as it does not prove even such basic statements as is the associativity law for addition.

Another basic system, called *Peano arithmetic* or PA, consists of Robinson's arithmetic and induction axioms

$$(\varphi(0) \wedge \forall x(\varphi(x) \rightarrow \varphi(S(x)))) \rightarrow \forall x \varphi(x), \tag{2.1}$$

for all formulas in the language $\{0, S, +, \cdot, \leq\}$.

A bounded quantifier is a quantifier of the form $\forall x \leq \tau$ or $\exists x \leq \tau$ where τ is a term not containing the variable x. These constructs are, of course, not present in the usual first order language, so we treat them as abbreviations. (Alternatively, they can be used as part of the language, if we add appropriate logical rules.) A formula where all quantifiers are bounded is called *a bounded formula* or Δ_0 or Σ_0. Formulas of the form $\forall x_1 \dots \forall x_n \varphi$, $\exists x_1 \dots \exists x_n \varphi$, $\forall x_1 \dots \forall x_n \exists y_1 \dots \exists y_m \varphi$, $\exists x_1 \dots \exists x_n \forall y_1 \dots \forall y_m \varphi$, etc. where φ is a bounded formula are called, respectively, Π_1, Σ_1, Π_2, Σ_2 etc. formulas.

In this hierarchy the simplest sentences which are unprovable in arithmetical theories are Π_1. This is because a true Σ_1 sentence is already provable in Q (take the numbers which witness the existential quantifiers and check the Δ_0 part using the axioms of Q). Also because of their simplicity Π_1 sentences are the most interesting ones from the point of view of provability. Many famous problems can be stated as Π_1 sentences: Fermat's last theorem, the four color theorem (both being eventually proved). The famous Riemann hypothesis can be also stated as a Π_1 sentence see [22, 24], though it is not so easy to prove. Several conjectures about the distribution of primes are Π_1 sentences.

The most important problem in theoretical computer science is the question if $P = NP$. The statement $P \neq NP$, which most people believe is true, can be expressed as a Π_2 sentence (this follows from the existence of an NP complete set). Thus we may hope that we can prove its independence in the same way as the independence of Paris-Harrington type sentences, but it is very unlikely. Let $n(k)$ be the least n such that satisfiability of formulas of length n cannot be computed by a circuit of size n^k. A Paris-Harrington type independence proof would require to prove that $n(k)$, as a function of k, grows extremely fast. This is the same as to say that the circuit complexity of satisfiability is $n^{f(k)}$ where $f(k)$ grows extremely slowly. Quite on the contrary, the general opinion is that $f(k)$ is probably of the order n^c, for some constant c, and definitely at least, say, $\varepsilon \log n$. The statement that satisfiability does not have circuits of size $n^{\varepsilon \log n}$ is a Π_1 sentence. The same can be said about other conjectures in complexity theory: they are not Π_1 sentences, but we believe that they will follow from stronger Π_1 sentences.

Let us note that we are interested only in theories one of whose models is the *standard model* which is the set of natural numbers with the usual operations (subtheories of the *true arithmetic*, in particular they are *ω-consistent*). For such a theory T, if we prove that a Π_1 sentence φ is consistent with T, then φ is true. The standard model is isomorphic to an initial segment of any model of an arithmetical theory, hence any consistent Π_1 sentence is true. (By an *initial segment* we mean a nonempty subset which contains with each number all smaller numbers.)

The fact that so many important problems are Π_1 sentences is an important reason for looking for techniques by which one can prove independence of such sentences. In fact, as shown above on the example of $P = NP$?, we use sentences of higher quantifier complexity only because we are not quite sure what is the right conjecture. For more information about this subject we recommend to consult Kreisel's paper [22].

2.2

It is necessary to have some background in complexity theory in order to appreciate the importance of the bottom-up approach. We shall assume that the reader knows some basics, in particular knows the definitions of the classes P and NP. For a logician it is easy to define a hierarchy of classes extending NP where $NP = \Sigma_1^p$, Π_1^p are complements of the NP classes (also denoted by $coNP$), Σ_2^p are classes defined by polynomially bounded existential quantifiers followed by universal polynomially bounded quantifiers bounding variables in a predicate from P etc. The union of these classes is called *the Polynomial Hierarchy* and denoted by PH. If we use *linearly bounded computations* instead of polynomially bounded ones, we obtain *the Linear Hierarchy.* The Linear Hierarchy when restricted to sets of natural numbers coincides with sets definable by Δ_0 formulas.

3. Independent sentences

In this section we shall explain why we are not satisfied with the current methods of proving independence results. The main reason is that, except for Gödel's theorem which gives only some special formulas, no general method is known for proving independence of Π_1 sentences.

Cohen's forcing method can be roughly described using model theory as follows. We start with a model M of ZF and extend it to M' by adding new sets. When doing that we must follow some rules in order to get a model of ZF again. The details are not important here, what we need is only the fact that the ordinal numbers are used as a sort of skeleton for building the new universe. As a result the ordinal numbers of the new model M' are the same as the ones of M. Furthermore, the concept of being finite is also preserved by the extension. The natural numbers are the finite ordinals hence, in particular, they are the same in both models. Consequently the arithmetical sentences which are true in M' are the same as those in M. Therefore this method does not give any independent arithmetical sentence.

Modifications of the forcing method have been used in various situations, so it is not completely excluded that some variant can give arithmetical independent sentences. However in order to get something interesting one would have to come up with a substantial change. To see what kind of problems it is necessary to overcome let us mention at least one. Suppose M is just a model of the natural numbers, say a model of Peano arithmetic. Trying to follow the original idea of forcing we attempt to add new numbers *between* the old ones, i.e. for some $a \in M$ there will be some new $b \in M'$ $b < a$. However there is a number $c \in M$ which codes (say using Gödel β-function) all the numbers of M below a. This c will have the same property in M', so we cannot add such a b!

Let us turn now to Paris-Harrington type independence results. Again we shall not talk about details and concentrate only on the overall structure of the argument. Let M be a nonstandard model of PA, say, a countable one. Instead of enlarging the model we look for an initial part M' of M which is also a model of PA. It turns out that M' is a model of PA if it is closed with respect to certain functions definable in M. The point of the argument is that it is possible to characterize, using some natural combinatorial sentences, where such segments can be found. Various combinatorial characterizations give various independent combinatorial sentences. The fact that we use initial segments of a given model is a serious limitation of the method. This method can prove independence of sentences which are Π_2 and, moreover, directly or indirectly refer to fast growing functions. Also it is known that these sentences are equivalent in PA to the Σ_1 reflection principle. This is not necessarily a drawback, as we consider equivalence with respect to a strong theory, but still it means that the class of such sentences is restricted.

It is obvious that we cannot prove independence of a Π_1 sentence in this way: any such sentence is preserved on initial segments. The only way that we know of to prove independence of a Π_1 sentence is Gödel's theorem. Gödel showed that a suitable diagonal sentence (*"I am not provable in PA"*) is not provable in PA. Then he proved that the sentence is equivalent to a sentence Con_{PA} asserting the consistency of PA. The same argument works for any sufficiently strong theory; with some modification it can be applied even to Robinson's Q. The above remark that the Paris-Harrington sentence is equivalent to the Σ_1 reflection principle suggests that one might prove the independence of a concrete combinatorial statement by showing that it implies the consistency of PA or another theory, but so far there are no such results.

By "concrete combinatorial" we mean a sentence expressing some principle from finite combinatorics or number theory. If one takes into account infinite principles, then such reductions to consistency statements are quite common. In this way one proves e.g. that some large cardinal axiom is stronger than another one.

4. Bounded arithmetic

4.1

By these words we denote a class of arithmetical theories which are axiomatized by induction axioms restricted to bounded formulas in various languages and with various modifications, restrictions, etc. We should stress again that the reason for restricting to weak theories is not that we accept some philosophy of "feasible mathematics" where only certain constructive reasoning is allowed. We want to study metamathematical properties of such theories *using any mathematical means, however nonconstructive.* The true reason why we are interested in these systems is that they seem to be more amenable to logical analysis and for their close relation to complexity theory.

The oldest considered system is $I\Delta_0$ which is Q plus induction axioms (2.1) for Δ_0 formulas, introduced in [27]. If M is any model of $I\Delta_0$, then any infinite initial segment closed under multiplication is a model of $I\Delta_0$ too. This implies that if $I\Delta_0$ proves that a function $f(x)$ is defined for all numbers, then the function is bounded by a polynomial in x. In particular it is not provable in $I\Delta_0$ that $x^{\lfloor \log_2(x+1) \rfloor}$, 2^x etc. are defined for all numbers.

We can view a number x as encoding a binary string of length $\approx \log x$. The strings of quadratic length, i.e. of length $(\log x)^2$, can be encoded by numbers of size $\approx x^{\lfloor \log_2(x+1) \rfloor}$, the length $(\log x)^4$ requires the function $x^{\lfloor \log_2(x+1) \rfloor}$ iterated twice and so on. Hence the function $x^{\lfloor \log_2(x+1) \rfloor}$ gives us the right growth rate if we want to extend the lengths of strings polynomially, e.g. using polynomial time computations. This is the reason for extending $I\Delta_0$ by an axiom saying that is a total function. This theory is usually denoted by $I\Delta_0 + \forall x \exists y\ (y = x^{\lfloor \log_2(x+1) \rfloor})$ or shortly $I\Delta_0 + \Omega_1$. (The last formula is just an abbreviation of a rather complicated formula, since we do not have x^y in our language.) This theory is capable of formalizing properly polynomial time computations, not only deterministic, but in fact with finitely many alternations – this is what is needed for machine based definitions of the Polynomial Hierarchy.

4.2

Having these two theories we can give an example of a basic problem in bounded arithmetic which seems to be closely connected with a problem in complexity theory.

Problem 4.1. Is every Π_1 sentence provable in $I\Delta_0 + \Omega_1$ provable in $I\Delta_0$?

The corresponding problem in complexity theory is:

Problem 4.2. Is the Polynomial time hierarchy equal to the Linear time hierarchy?

The connection between the two problems is: if we can *prove in* $I\Delta_0$ that the polynomial time hierarchy equals to the linear time hierarchy, then every Π_1 sentence provable in $I\Delta_0 + \Omega_1$ is provable in $I\Delta_0$. Note that this does not exclude that the answer to the second problem is YES while the answer to the first one is NO. We conjecture that the answer to both problems is NO and hope that the first problem is somewhat easier. A negative answer to Problem 1 would give us at least some supporting evidence that the answer to Problem 2 is NO, namely that the equality of the two complexity classes is not provable in $I\Delta_0$.[1]

4.3

The natural approach is to try to show that the consistency of $I\Delta_0$ is provable in $I\Delta_0 + \Omega_1$, while by Gödel's theorem it is not provable in $I\Delta_0$. This approach is, however, quite hopeless: even the consistency of Q is not provable in $I\Delta_0 + \Omega_1$. This follows from:

Theorem 4.1. (Wilkie [39]) *There is an interpretation of* $I\Delta_0 + \Omega_1$ *in* Q.

It follows that $I\Delta_0 + \Omega_1 \vdash Con_{I\Delta_0+\Omega_1} \equiv Con_Q$, whence $I\Delta_0 + \Omega_1 \nvdash Con_Q$.

Some weaker versions of consistency statements have been proposed, but the general feeling is that there is some deeper reason why this cannot work. What we suspect is that such results cannot be proved by *diagonalization.* This term refers to a method which was used for the fist time by Cantor to prove that the cardinality of the power set is larger than the cardinality of the given set. A similar idea was used in recursion theory and later in complexity theory to separate some complexity classes. It seems that the use of *self-referential sentences* to separate theories in logic is just another facet of this method. As all attempts to separate pairs of complexity classes such as P and NP have failed and because of the similarity of self-reference to diagonalization, there is little hope that we can solve such problems as Problem 1 using a version of Gödel's theorem.

4.4

In order to get connections to more interesting problems in complexity theory than Problem 2, we have to introduce fragments of $I\Delta_0 + \Omega_1$. The idea

[1] We cannot speak about classes in $I\Delta_0$ directly, instead we talk about definability by formulas of certain complexity. Another option is to consider second order bounded arithmetic.

is to restrict the induction schema further to subsets of bounded formulas. From the point of view of logic, a natural hierarchy is obtained by restricting the number of alternations of bounded quantifiers. Unfortunately, if we use the usual language of arithmetic, we do not get classes which correspond to natural Turing machine definable classes. Therefore, the language must be extended further by some simple functions; in particular $\lfloor \log_2(x+1) \rfloor$ and a function whose growth rate is similar to $x^{\lfloor \log_2(x+1) \rfloor}$ are added as primitives. Furthermore, when defining the hierarchy of the formulas, the bounded quantifiers in which the outer function is log, are not counted. The particular details are not important, we only need that the classes of formulas Σ_i^b define exactly the sets in the complexity classes Σ_i^p and similarly for the Π classes.

We also have to modify the induction axioms. Instead of (2.1) we take

$$(\varphi(0) \wedge \forall x(\varphi(x) \rightarrow (\varphi(2x) \wedge \varphi(S(2x)))) \rightarrow \forall x \varphi(x). \qquad (4.1)$$

This is induction on the length of the binary representation of numbers. The intuitive explanation of why we take this weaker form of induction is that to verify an instance of $\varphi(n)$ we need to unwind the premise only $\log n$ times and $\log n$ is the size of the input when we compute with n.

The theory axiomatized by some basic open axioms and the schema (4.1) is denoted by S_2; it is a conservative extension of $I\Delta_0 + \Omega_1$. The fragments where (4.1) is restricted to Σ_i^b are denoted by S_2^i. So we can think of S_2^i as a theory for Σ_i^p. The theory S_2^1 deserves a particular attention as it is a theory for NP. The definition of these theories is due to Buss [3]; the idea of (4.1) goes back to Cook [6] where he introduced an equational theory PV.

Now you may ask, what is a theory for P? Well, one can give some suggestions, e.g. PV, but I have to warn you that the correspondence between the theories and the complexity classes is very loose and should not be taken literally. For instance, S_2^1 proves all instances of (4.1) for Π_1^b formulas, so the theories for NP and for $coNP$ coincide. We still think that this is not evidence for $NP = coNP$. In a moment we will see that there is also a good reason to associate S_2^1 with P rather than with NP.

5. Relations to computational complexity

There are not only "morphological" similarities between theories and complexity classes, but also several *provable* relations. We shall mention at least two basic results.

5.1

The first result concerns the question, how difficult is it to witness the existential quantifier in $\forall\exists$ sentences provable in a given theory. The paradigm is the classical result on the fragment of Peano arithmetic $I\Sigma_1$ (induction restricted to Σ_1 arithmetical formulas) due to Mints and Takeuti. This is the result that $I\Sigma_1 \vdash \forall x \exists y \varphi(x,y)$, for φ a Σ_1 formula, implies that there exists a primitive recursive function f such that $\varphi(n, f(n))$ is true for all n.

The most important of such "witnessing" theorems is the following one:

Theorem 5.1. (Buss [3]) *Suppose $S_2^1 \vdash \forall x \exists y \varphi(x,y)$, for φ a Σ_1^b formula. Then there exists a polynomial time computable function f such that $\varphi(n, f(n))$ is true for all n.*

This theorem has a natural extension to theories S_2^i and there are several witnessing theorems for other theories and other classes of formulas. Theorem 5.1 has an easy corollary which can be interpreted as an independence result.

Corollary 5.1. *If it is provable in S_2^1 for a class X that $X \in NP \cap coNP$, then $X \in P$, so $NP \cap coNP \neq P$ is not provable in S_2^1.*

It has also been shown that there exists a *model* of PV in which $NP \cap coNP \neq P$ [15]. We cannot deduce much for complexity theory from this result, since as soon as we make the theory only slightly stronger, the proof breaks down. Still it is a nice result and we would like to get more results like this.

5.2

The second result concerns the problem of the hierarchy of the theories S_2^i. For all we know they may be the same (the sets of their theorems may coincide).

Problem 5.1. Are there infinitely many i for which S_2^{i+1} properly extends S_2^i?

It is well-known that this is equivalent to the statement that S_2 is not finitely axiomatized, and to the statement that $I\Delta_0 + \Omega_1$ is not finitely axiomatized; it also implies that $I\Delta_0$ is not finitely axiomatized. The corresponding problem about complexity classes is:

Problem 5.2. Are there infinitely many i for which $\Pi_i^p \neq \Pi_{i+1}^p$?

It has been conjectured that the answer to this problem is positive. This conjecture belongs to one of the strongest considered in complexity theory; in particular it implies $P \neq NP$ and $NP \neq coNP$. Unlike with Problems 1 and 2, one can prove:

Theorem 5.2. (Krajíček et al. [20]) *A positive answer to Problem 5.2 implies a positive answer to Problem 5.1.*

This theorem is proven using a special kind of witnessing theorems for theories S_2^i. Note that contrary to what one may expect, we need $\Pi_{i+2}^p \neq \Sigma_{i+2}^p$ for proving $S_2^i \not\equiv S_2^{i+1}$. There are several extension and variation of this result. In particular, we know that $S_2^i = S_2^{i+1} \Rightarrow S_2^i = S_2$ [4]; the corresponding fact for the polynomial time hierarchy is trivial.

To prove unconditional separation of S_2^{i+1} from S_2^i, we have to find concrete sentences which are provable in S_2^{i+1} but not in S_2^i. We do have some candidates for such sentences but we have no idea how to prove their unprovability. There is however something which is halfway between this and using conjectures from complexity theory. We can extend the language by adding an uninterpreted predicate, say α and consider variants of the theories in this language. I.e. we do not add any basic axioms about α, but we extend the induction axioms to all formulas of appropriate complexity containing α. We denote such an extension of a theory T by $T(\alpha)$. In several cases we can solve separation problems for such extensions; in particular we know that $S_2^i(\alpha) \not\equiv S_2^{i+1}(\alpha)$. For those who know basics of complexity theory it is not that surprising, since there is a similar concept for complexity classes. This is the so called *relativization*, which means that we augment Turing machines with an *oracle*, which is essentially free access to some possibly complex set. Then one can prove a lot of separations. The symmetry between theories and complexity classes is, however, again not complete: using relativizations we can make classes both unequal *and equal* (e.g. $P^A \neq NP^A$ for some A and $P^B = NP^B$ for another B), while we can only separate theories by adding the uninterpreted predicate (unless we also add additional axioms).

6. Propositional calculus

We have considered theories and complexity classes. Now we shall talk about a third kind of system related to the previous two: *proof systems for propositional calculus*. Thus we sink to the bottom of the universe of formal systems.

Most of the research into propositional calculus deals with semantical modifications (extensions of the classical propositional calculus by modalities, weakenings, such as intuitionistic logic etc). What we are interested in is something different. We use only the classical propositional calculus and we study possible ways in which the concept of the proof can be formalized. Again this is related to complexity theory, if not just a branch of it.

As the set of tautologies is fixed once for ever, we cannot classify the proof systems by what they prove. Instead we shall distinguish them by what they prove *using short proofs*. By "short" we mean, of course, polynomial in the size of the formula. Furthermore we can (quasi)order the systems by defining $P \leq Q$ for two proof systems, if for any tautology τ its shortest proof in Q is at most polynomially longer than its shortest proof in P. In all concrete cases that we have so far encountered, if $P \leq Q$, then there exists a polynomial algorithm which any proof of τ in P transforms in a proof (at most polynomially longer) of τ in Q. Then we say that Q *polynomially simulates* P.

The most common proof systems for propositional calculus are the systems based on finitely many axiom schemas and finitely many rules of the type of *Modus Ponens*; quite often *Modus Ponens* is the only rule of the system. The technical term for such systems is *Frege system.* It has been proved [7] that all Frege systems have essentially the same power, namely they polynomially simulate each other. (This is very easy to prove if the two systems use the same connectives.)

To make the system stronger we add a rule that allows us to abbreviate (long) formulas by a single variable. Formally this means that we can introduce for every formula φ the equivalence $\varphi \equiv p$, where p is a propositional variable not used in the previous part of the proof nor in the proved formula. Frege systems with this rule added are called *Extended Frege systems.* Note that in this way we may reduce the total size of a proof, but we cannot save on the number of steps.

To get substantially stronger systems we have to abandon the idea that the proof consists of propositional formulas.[2] For instance, the next natural system after Extended Frege systems is the *Quantified propositional calculus* [8]. Such a system is obtained from a Frege system by adding rules for quantifiers. A proof is a sequence of quantified propositional formulas derived according to the rules. We may use this system to derive quantified propositional formulas. If we want to use it as a proof system for propositional calculus, we think of quantified propositional formulas as auxiliary means to eventually derive a quantifier free propositional formula.

In general we require only that one can effectively check the proofs of the system in question. More precisely, there must exist a polynomial time algorithm to decide for a given sequence if it is a proof in the system. Thus the proofs need not even be structured into steps as in usual proofs.

To give an example of a very strong proof system for propositional calculus, define d to be a proof of τ, if d is a proof in ZF of the statement $Taut(\tau)$ expressing that τ is a tautology. To see why this system is strong just realize that ZF proves that Frege, Extended Frege, Quantified propositional calculus and lot of others are sound systems. Thus given e.g. an Extended Frege proof of τ we only need to check in ZF that it is an Extended Frege proof of τ and then immediately we get that τ is a tautology (provably in ZF). Hence this proof system polynomially simulates all the above systems.

We can use the same construction for any theory in which the concept $Taut$ can be reasonably formalized. So we define for a such a theory T the propositional proof system P_T to be the system where a proof of τ is a proof of $Taut(\tau)$ in T. The P_T is not only an interesting construction, but it could

[2] Of course it is always possible to use infinitely many axiom schemas, but then we have to talk about how these schemas are defined.

be useful for showing unprovability of certain Π_1 sentences, namely universal closures of Π_1^b formulas. We denote this class of sentences by $\forall\Pi_1^b$.

Let φ be a Π_1^b sentence, i.e., φ has no free variables. Then we can express φ using a propositional formula by replacing *log* bounded quantifiers by several disjunctions or conjunctions and coding the variables bounded by universal bounded quantifiers by propositional variables. Instead of going into details of this transformation, let us only note that a propositional formula is true iff it is satisfied for all possible values of the propositional variables. Thus we implicitly interpret it as if there were universal quantifiers (which we can actually add in the quantified propositional calculus). The range of this quantification is exponential in the size of the formula in the same way as it is in Π_1^b sentences.

Let $\varphi(x)$ be a Π_1^b formula. Then we get a sequence of propositional formulas τ_n from the closed instances $\varphi(n)$, $n = 0, 1, 2, \ldots$. Since we encode n in binary, the length of τ_n is polynomial in $\log n$. If $\forall x\varphi(x)$ is provable in T, then we get important information on the lengths of proofs of these propositional formulas in proof system P_T:

Theorem 6.1. *Let T be a sufficiently strong arithmetical theory and $\varphi(x)$ be a Π_1^b formula. Suppose that $\forall x\varphi(x)$ is provable in T. Then the propositional translations of sentences $\varphi(n)$ have proofs in P_T whose lengths are polynomial in the lengths of these propositions.*

Proof-sketch. Let τ_n be the translation of $\varphi(n)$. If T is sufficiently strong, then it proves $\forall x(\varphi(x) \to Taut(\tau_x))$. Hence if T proves $\forall x\varphi(x)$, then all instances τ_n have polynomial size proofs. □

To show that a given Π_1 sentence is not provable, we have to prove a superpolynomial lower bound on the lengths of proofs for the corresponding tautologies. But even if we only show that there are some tautologies which do not have polynomial size proofs in the proof system P_T, we get a very interesting independence result:

Theorem 6.2. *Suppose that the proof system P_T of a sufficiently strong theory T is not polynomially bounded, i.e., there is no polynomial upper bound on the length of the shortest P_T proofs of the propositional tautologies. Then T does not prove $NP = coNP$.*

Proof-sketch. Suppose that T does prove $NP = coNP$. Then in T the $coNP$ predicate $Taut(x)$ is equivalent to an NP predicate $\alpha(x)$. To prove some τ in P_T we need to show $Taut(\tau)$; the proof of this sentence can be longer than the shortest proof of $\alpha(\tau)$ only by a constant factor. Since α is NP we only have to take a polynomial size witness for the truth of $\alpha(\tau)$ and check it. This gives a polynomial size proof of τ. □

A possible approach for proving $NP \neq coNP$ is to prove gradually for stronger and stronger propositional proof systems that they are not polynomially bounded hoping that eventually we develop a technique allowing us to prove that no propositional proof system is polynomially bounded (which is equivalent to $NP \neq coNP$ because of the general definition of the concept of a propositional proof system). The theorem above shows that this is essentially the same as showing unprovability of $NP \neq coNP$ for stronger and stronger theories.

A successful application of this reduction clearly depends very much on how much hold we can get on the proof system P_T. This system seems to be very strong even for weak theories. There is another way of associating a propositional proof system to a theory which produces weaker systems, but it is not as universal as the construction of P_T. The system is called *the associated proof system* of T and it is defined, roughly speaking, by requiring the simulation of Theorem 6.1 to hold and that T proves its soundness (for a precise definition see [34]). The latter condition is not satisfied by P_T. Associated propositional proof systems have some nice properties, in particular Theorem 6.2 holds for them too.

For strong theories it seems hopeless to give a comprehensible combinatorial description even of the associated proof system. Fortunately, at least for some systems of bounded arithmetic we get natural associated proof systems. The most interesting particular case is the theory S^1_2 whose associated propositional proof system is, up to polynomial simulation, an Extended Frege system [6, 3]. For S^i_2 in general we can take fragments of the quantified propositional calculus obtained by an appropriate restriction on the quantifier complexity of formulas [18].

A large part of the activity is concentrated on proving lower bounds on the lengths of propositional proofs and we can report steady progress [1, 12, 13, 21, 31, 32]. Unfortunately the proof systems for which one can prove that they are not polynomially bounded are still much weaker than Extended Frege; even proving a superpolynomial lower bound for Frege systems would be a breakthrough. Thus we do not expect that concrete independent Π_1 sentences will be found for strong theories in the near future. Still we cannot exclude that somebody finds a completely new powerful method for proving independence. In particular, Gödel's theorem does not quite fit into the picture drawn above, where provability is thought of as polynomial length proofs in the associated propositional proof system. We know that consistency statements are not provable and can be formalized as $\forall\Pi^b_1$, but we do not have superpolynomial lower bounds on the lengths of proofs of its propositional translations in the associated proof system.[3]

[3] For P_T of a sufficiently strong theory T we have polynomial size *upper* bounds on the lengths of proofs of its propositional translations. Let a sufficiently strong theory T be given. The translations of Con_T are some tautologies τ_n expressing that "no $x \leq n$ is a proof of contradiction in T". Let σ_m be tautologies expressing that "there is no proof of contradiction in T of length $\leq m$"; the lengths of of σ_m are polynomial in m. If m is the length of n, then $\sigma_m \rightarrow \tau_n$ has a proof polynomial in m, i.e., in the length of τ_n. In [33] we proved that first order sentences $Con_T(m)$ expressing that there is no proof of contradiction in

7. Model theory of weak arithmetical theories

The natural numbers are a basic algebraic structure, thus we can also use the machinery of algebra and model theory. Then it is more convenient to axiomatize the integers instead of just the positive ones. The basic theory is the theory of discretely ordered commutative rings. The next stronger system which has been studied is obtained by adding induction for open formulas; it is denoted by *IOpen*. For this theory it is possible to prove independence by constructing explicitly models. Thus it has been proved that it is consistent with *IOpen* that $x^3+y^3=z^3$ has a nontrivial solution and that $\sqrt{2}$ is rational [36]. It is possible to get in such a way independence for slightly stronger theories (e.g. postulating that the ring is integrally closed in its fraction field, which, in particular, implies that $\sqrt{2}$ is irrational [23]), but there seems to be a serious obstacle to do it for theories which contain $I\Delta_0$ and S^1_2. It is well-known that these theories do not have nonstandard recursive models, but it is even worse: any nonstandard model contains an initial segment which is a nonstandard model of PA. Thus, in spite of using weak theories we get the whole complexity of nonstandard models of PA [25]. Even if this research does not lead directly to independence results for stronger theories, there are interesting problems in this area from the point of view of both logic and number theory [9, 26, 38].

Instead of constructing models directly one can try to modify a given nonstandard model. We mentioned in Section 3. that we can extend a model of PA only by adding elements which are larger than all the old elements. That argument is not valid for models of bounded arithmetic, there it is possible to add small elements (if the model is "short"). Several partial results have been obtained in this way [1, 19, 35]. Another possibility is to choose a submodel. In this way one can prove Theorem 5.1 [40] (for an exposition see [11][Chap. V, Sec. 4]). For the most recent applications of model theory in bounded arithmetic see [16, 37]

8. Conclusions

The reader may be a little disappointed now, because we promised to talk on foundations of mathematics, but instead most of the time we talked about computational complexity. That is not a mistake. Any reasonable definition of a formal system presupposes the concept of computability. We have to use some formalism, as pure reliance on intuition cannot be considered a foundation. When working with weak theories, natural connections with computational complexity are almost ubiquitous. Our feeling is then that we

T of length $\leq m$ have proofs in T of length polynomial in m. However, if T is sufficiently strong, then it proves $Con_T(m) \equiv Taut(\sigma_m)$ using polynomial size proofs. Thus also $Taut(\tau_n)$ have polynomial size proofs in T, which means that τ_n have polynomial size proofs in P_T.

cannot solve problems of foundations of mathematics without solving or at least understanding more deeply problems in complexity theory. But maybe also in order to solve fundamental problems in complexity theory we need to understand more about the foundations of mathematics.

Acknowledgement. I would like to thank to Sam Buss, Armin Haken, Jan Krajíček and Jiří Sgall for suggesting several corrections and improvements. I am also grateful to Claus Lange and Yurij Matiyasevich for informing me that Riemann's Hypothesis is a Π_1 sentence.

References

1. M. Ajtai. The complexity of the pigeonhole principle, in: *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society, Piscataway, New Jersey, 1988, pp. 346–355.
2. P. Beame, R. Impagliazzo, J. Krajíček, T. Pitassi and P. Pudlák. Lower bounds on Hilbert's Nullstellensatz and propositional proofs, Proc. London Math. Soc., to appear.
3. S. R. Buss. *Bounded Arithmetic*, Bibliopolis 1986, Napoli. Revision of 1985 Princeton University Ph.D. thesis.
4. S. R. Buss. Relating the bounded arithmetic and polynomial time hierarchies, *Annals of Pure and Applied Logic* 75 (1995), pp. 67–77.
5. S. R. Buss. First order proof theory of arithmetic, in *Handbook of Proof Theory*, S. R. Buss ed., North Holland, to appear.
6. S. A. Cook. Feasibly constructive proofs and the propositional calculus, in: *Proceedings of the Seventh Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery 1975, New York, pp. 83–97.
7. S. A. Cook and R. A. Reckhow. The relative efficiency of propositional proof systems, *Journal of Symbolic Logic* 44, (1979) pp. 36–50.
8. M. Dowd. *Propositional Representation of Arithmetic Proofs*, PhD thesis, University of Toronto 1979.
9. Van den Dries. Which curves over **Z** have points with coordinates in a discretely ordered ring? *Transactions AMS* 264 (1990) pp. 33-56.
10. F. Ferreira. Binary models generated by their tally part, Archive for Math. Logic, to appear.
11. P. Hájek and P. Pudlák. *Metamathematics of First-order Arithmetic*, Springer-Verlag 1993, Berlin.
12. A. Haken. The intractability of resolution, *Theoretical Computer Science*, 39, (1985) pp. 297–308.
13. J. Krajíček. Lower bounds to the size of constant-depth propositional proofs, *Journal of Symbolic Logic*, 59, (1994) pp. 73–86.
14. J. Krajíček. On Frege and extended Frege proof systems, in: *Feasible Mathematics II*, J. Krajíček and J. Remmel, eds., Birkhäuser, Boston, 1994, pp. 284–319.
15. J. Krajíček. *Bounded Arithmetic, Propositional Logic and Complexity Theory*, Cambridge University Press 1995.
16. J. Krajíček. Extensions of models of PV, manuscript, 1996.
17. J. Krajíček and P. Pudlák. Propositional proof systems, the consistency of first-order theories and the complexity of computations, *Journal of Symbolic Logic*, 54, (1989) pp. 1063–1079.

18. J. Krajíček and P. Pudlák. Quantified propositional calculi and fragments of bounded arithmetic, *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 36, pp. 29–46.
19. J. Krajíček and P. Pudlák. Propositional provability in models of weak arithmetic, in: *Computer Science Logic'89*, E. Boerger et al. eds., Springer-Verlag LNCS 440, (1990), pp. 193-210.
20. J. Krajíček, P. Pudlák and G. Takeuti. Bounded arithmetic and the polynomial hierarchy, *Annals of Pure and Applied Logic* 52, (1991) pp. 143-153.
21. J. Krajíček, P. Pudlák, and A. Woods. An exponential lower bound to the size of bounded depth Frege proofs of the pigeonhole principle, *Random structures and Algorithms* 7/1 (1995), pp. 15–39.
22. G. Kreisel. Mathematical significance of consistency proofs. *Journal of Symbolic Logic*, 23 (1958), pp. 155-182.
23. D. Marker and A. Macintyre. Primes and their residue rings in models of open induction, *Annals of Pure and Applied Logic* 43, (1989) pp. 57-77.
24. Yu. Matiyasevich. The Riemann hypothesis from a logician's point of view. *Proc. First Conference of the Canadian Number Theory Association*, Ed. R.Mollin. Walter de Gruyter, 1990, pp. 387-400.
25. K. McAloon. On the complexity of models of arithmetic, *Journal of Symbolic Logic 47,* (1982), pp. 403-415.
26. M. Otero. Quadratic forms in normal open induction, *Journal of Symbolic Logic*, to appear.
27. R. Parikh. Existence and feasibility in arithmetic, *Journal of Symbolic Logic*, 36, (1971) pp. 494–508.
28. J. B. Paris and L. Harrington. A mathematical incompleteness in Peano arithmetic, in: *Handbook of Mathematical Logic* North-Holland 1977, pp. 1133-1142.
29. J. B. Paris and A. J. Wilkie. Counting problems in bounded arithmetic, in: *Methods in Mathematical Logic, Proceedings of the 6-th Latin American Symposium, Caracas, Venezuella*, C. A. Di Prisco, ed., Lecture Notes in Mathematics #1130, Springer-Verlag, Berlin, 1985 pp. 317–340.
30. J. B. Paris and A. J. Wilkie. Δ_0 sets and induction, in: *Proc. of the Jadwisin Logic Conf., Poland*, Leeds Univ. Press, pp. 237-248.
31. T. Pitassi, P. Beame, and R. Impagliazzo. Exponential lower bounds for the pigeonhole principle, *Computational Complexity*, 3, (1993) pp. 97–140.
32. P. Pudlák. Lower bounds for resolution and cutting planes proofs and monotone computations, *Journal of Symbolic Logic.* to appear.
33. P. Pudlák. Improved Bounds to the Lengths of Proofs of Finitistic Consistency Statements, in *Logic and Combinatorics,* S. J. Simpson editor, Contemporary Mathematic 65, American Mathematical Society 1987, pp. 309-331.
34. P. Pudlák. The lengths of proofs, in *Handbook of Proof Theory*, S. R. Buss ed., North Holland, to appear.
35. S. Riis. Making infinite structures finite in models of second order bounded arithmetic, in: *Arithmetic, Proof Theory and Computational Complexity*, P. Clote and J. Krajíček eds., Oxford Univ. Press 1993, pp. 289-319.
36. J.C. Shepherdson. A non-standard model of a free variable fragment of number theory, *Bull. Acad. Pol. Sci.*, 12, (1964) pp. 79-86.
37. G. Takeuti and M. Yasumoto. Forcing on bounded arithmetic, this procceedings.
38. A. A. Wilkie. Some results and problems on weak systems of arithmetic, in A. Macintyre et al eds., *Logic Colloquium'77*, North-Holland 1978, pp. 285-296.
39. A. A. Wilkie. On sentences interpretable in systems of arithmetic, in: *Logic Colloquium'84*, North-Holland 1986, pp. 329-342.

40. A. A. Wilkie. A model-theoretic proof of Buss's characterization of the polynomial time computable function, manuscript, 1985.
41. A. A. Wilkie and J. B. Paris. On the schema of induction for bounded arithmetical formulas, *Annals of Pure and Applied Logic* 35, (1987) pp. 261-302.