ON THE CORRESPONDENCE BETWEEN ARITHMETIC THEORIES AND PROPOSITIONAL PROOF SYSTEMS

OLAF BEYERSDORFF

Abstract. Bounded arithmetic is closely related to propositional proof systems, and this relation has found many fruitful applications. The aim of this paper is to explain and develop the general correspondence between propositional proof systems and arithmetic theories, as introduced by Krajíček and Pudlák [41]. Instead of focusing on the relation between particular proof systems and theories, we favour a general axiomatic approach to this correspondence. In the course of the development we particularly highlight the role played by logical closure properties of propositional proof systems, thereby obtaining a characterization of extensions of *EF* in terms of a simple combination of these closure properties.

Using logical methods has a rich tradition in complexity theory. In particular, there are very close relations between computational complexity, propositional proof complexity, and bounded arithmetic, and the central tasks in these areas, i.e., separating complexity classes, proving lower bounds to the length of propositional proofs, and separating arithmetic theories, can be understood as different approaches towards the same problem. While each of these fields supplies its own techniques to address these problems, many exciting results have been obtained that decisively use the interplay of combinatorial and logical methods (e.g. [1, **??**, 48, 49]), and it is expected that this exchange of ideas will continue to exert substantial influence on the development of theoretical computer science in general.[1]

Nevertheless, complexity theorists and logicians quite often seem to have different traditions, regarding notation and prerequisites that can be assumed without explanation, and these "cultural" differences sometimes make logic-oriented research difficult to access for a wider complexity-theoretic audience. These observations particularly apply, in my opinion, to the field of propositional proof complexity, that can be addressed both from a completely combinatorial perspective as well as by utilizing the correspondence to bounded arithmetic.[2]

Supported by DFG grant KO 1053/5-1

THIS IS THE PRE-PEER REVIEWED VERSION OF THE FOLLOWING ARTICLE: FULL CITE

[1] In [34] Jan Krajíček formulates "It is to be expected that a nontrivial combinatorial or algebraic argument will be required for the solution of the P versus NP problem. However, I believe that the close relations of this problem to bounded arithmetic and propositional logic indicate that such a solution should also require a nontrivial insight into logic."

[2] This opinion has been confirmed in many conversations and was also reiterated by some of the referees, when I used bounded arithmetic in a complexity-theoretic context (e.g. in [9]).

This relation works for a number of diverse pairs of proof systems and corresponding arithmetic theories, each of which presents a number of specific nontrivial technical problems. A unifying approach for a general correspondence was suggested by Krajíček and Pudlák [41]. It is the aim of this paper to explain and develop this general correspondence in sufficient detail, which is necessary in order to convey this material to a broad complexity-oriented audience. This task seems desirable, as the correspondence has found many applications (e.g. [1, 31, 37, 38, 42]). To my knowledge, however, there is no account that develops the general correspondence between bounded arithmetic and propositional proof systems in full detail. The original source [41] introduces this correspondence in a very condensed way, and it is unfortunately left out from the standard reference [34]. There is, however, a number of beautiful introductory expositions, most notably [47] and [36]. A somewhat different approach, using two-sorted theories, is presented in [23, 24].

In this exposition we emphasize the role of logical closure properties of propositional proof systems, thereby obtaining a characterization of schematic extensions of the extended Frege system *EF* in terms of a simple combination of these closure properties. While these results certainly do not come unexpected, they might still shed some light on the subject from a new perspective.

Before we start to develop this material, we will try to sketch the overall picture of the relations between computational complexity, bounded arithmetic, and propositional proof systems, of which the correspondence between arithmetic theories and proof systems is just one, albeit important, facet.

§1. Three Approaches to One Problem: Computational Complexity, Bounded Arithmetic, and Propositional Logic. Computational complexity studies the amount of resources which is required for the solution of computational tasks. A major open problem in the field is the precise comparison between deterministic and nondeterministic computations, leading for polynomial-time computations to the famous P/NP-problem formulated already more than 30 years ago by Cook [21] and Karp [32]. The solution of the P/NP-problem has far reaching implications, mainly because, starting with Cook's completeness result, a vast number of problems with immense practical relevance have been shown to be NP-complete. Despite enormous efforts the separation of complexity classes remains elusive today. Current techniques such as diagonalization and circuit lower bounds are all ineffectual, with even theoretical evidence supporting the failure of these approaches [4, 52].

A different, logic-oriented way of studying complexity classes is through weak fragments of arithmetic, usually referred to as theories of bounded arithmetic. These fragments have the right strength to formalize and reason about efficient computations. More formally, definable functions and predicates in these theories can be used to characterize functions and languages from standard complexity classes, the most prominent example being the hierarchy of theories S^i_2 and T^i_2 defined by Buss [12], which correspond to the computational strength of the levels of the polynomial hierarchy. These strong relations between the theories S^i_2 and PH were established by a series of witnessing theorems due to Buss [12, 14] and Krajíček, Pudlák, and Takeuti [43]. In particular, Krajíček, Pudlák, and Takeuti proved that a collapse of the hierarchy of the theories S^i_2 implies a collapse of PH. Later Buss [16] and Zambella [57] independently strengthened this result by showing that $S_2 = \bigcup_{i=1}^{\infty} S^i_2$ is finitely axiomatizable if and only if PH collapses and this collapse is provable in S_2.

Bounded arithmetic is also closely connected to propositional proof systems. This connection was first developed by Cook [22], who gave a translation of bounded first-order formulas into polynomial-size sequences of propositional formulas. Different and refined translations have later been introduced by Paris and Wilkie [46] as well as by Krajíček and Pudlák [41]. These translations allow the use of logical and in particular model-theoretic machinery to obtain lower bounds to the size of propositional proofs, which constitutes the main objective in propositional proof complexity. In particular, Ajtai [1] successfully used these methods to show super-polynomial lower bounds to the proof size in bounded-depths Frege systems (cf. Theorem 9.4 for the general framework). Together with subsequent improvements [6, 8, 44], this currently forms one of the strongest results about propositional proof systems. Another connection to bounded arithmetic comes from the reflection principles which are arithmetic formulas stating the consistency of propositional proof systems. On the one hand, these formulas are candidates for the separation of arithmetic theories, on the other hand, proving reflection principles in arithmetic theories yields simulations between propositional proof systems. This technique was first used by Krajíček and Pudlák [39] to show the equivalence of extended Frege and substitution Frege systems.

The circle back to computational complexity is completed with the results of Cook and Reckhow [25], who showed that polynomially bounded proof systems exist if and only if NP is closed under complementation. Thus, similarly as the circuit-complexity approach, proving lower bounds to successively stronger systems can be understood as a way to address the P/NP-question by non-uniform methods. In fact, the relationship between proof complexity and computational complexity extends to other complexity classes than NP. Köbler, Messner, and Torán [33] have shown that the problem on the existence of complete sets for promise classes like $\mathsf{NP} \cap \mathsf{coNP}$ or BPP can be reformulated as questions about proof systems.

§2. Overview of the Paper. We start in Sect. 3 by recalling some background information on propositional proof systems and particularly Frege systems and their extensions.

In Sect. 4 we define and investigate natural properties of proof systems which we use throughout this paper. These properties are of logical nature: it should be feasible to carry out basic operations like modus ponens and substitutions in the proof system. Most of these properties have certainly been used before in several contexts. For propositional proof systems, which can be studied both from a proof-complexity as well as from a computational-complexity perspective, we feel that it is important to be precise about the exact conditions that are imposed on proof systems. If complexity theorists state a theorem like

For all propositional proof systems the following holds . . . , then they really mean that this theorem holds for all functions computed by deterministic polynomial-time Turing machines which have as their range the set of tautologies. If, on the other hand, people from proof complexity use this phrase, it is often implicitly understood from the context that the result only holds for some class of meaningful proof systems, operating for example with formulas and enjoying some basic closure properties. Therefore, combining results from both worlds without being conscious about the context in which they are applicable may result in confusion (at least this happened to me once). We therefore try to be rather pedantic in always listing explicitly all assumptions that are made on the proof system.

In Sect. 5 we start to explain the correspondence between bounded arithmetic and propositional proof systems. Section 6 contains a detailed description of the translation of first-order formulas into sequences of propositional formulas as given by Cook [22] and Krajíček and Pudlák [41]. In Sects. 7 and 8 we outline the formalization of syntactic concepts such as propositional formulas and propositional proofs in arithmetic theories.

We then proceed in Sect. 9 with the general correspondence between arithmetic theories and propositional proof systems as defined by Krajíček and Pudlák [41]. In Sect. 10 we explain this correspondence for the theory S^1_2 and the extended Frege system *EF* as well as for extensions of *EF* by additional axioms. Section 11 is again devoted to the general correspondence from [41]. We give a refined analysis of proof systems admitting a corresponding arithmetic theory. We call such proof systems regular and exhibit sufficient conditions for the regularity of propositional proof systems. From this we obtain a purely logical characterization of the degrees of schematic extensions of *EF*.

We conclude in Sects. 12 and 13 with an application to hard tautologies and further observations about the properties from Sect. 4.

§3. Propositional Proof Systems.

Propositional proof systems were defined in a very general way by Cook and Reckhow [25] as polynomial-time functions P which have as their range the set of all tautologies. A string π with $P(\pi) = \varphi$ is called a P-proof of the tautology φ. By $P \vdash_{\leq m} \varphi$ we indicate that there is a P-proof of φ of size $\leq m$. If Φ is a set of propositional formulas we write $P \vdash_* \Phi$ if there is a polynomial p such that $P \vdash_{\leq p(|\varphi|)} \varphi$ for all $\varphi \in \Phi$. If $\Phi = \{\varphi_n \mid n \geq 0\}$ is a sequence of formulas we also write $P \vdash_* \varphi_n$ instead of $P \vdash_* \Phi$.

Proof systems are compared according to their strength by simulations, introduced in [25] and [39]. A proof system S *simulates* a proof system P (denoted by $P \leq S$) if there exists a polynomial p such that for all tautologies φ and P-proofs π of φ there is an S-proof π' of φ with $|\pi'| \leq p(|\pi|)$. If such a proof π' can even be computed from π in polynomial time we say that S *p-simulates* P and denote this by $P \leq_p S$. If the systems P and S mutually (p-)simulate each other, they are called *(p-)equivalent*, denoted by $P \equiv_{(p)} S$. A proof system is *optimal* if it simulates all proof systems.

A prominent example of a class of proof systems is provided by *Frege systems* which are usual textbook proof systems based on axioms and rules. In the context of propositional proof complexity these systems were first studied by Cook and

Reckhow [25], and it was proven there that all Frege systems, i.e., systems using different axiomatizations and rules, are p-equivalent. A different characterization of Frege systems is provided by *Gentzen's sequent calculus* [27], that is historically one of the first and best analysed proof systems. The sequent calculus is widely used, both for propositional and first-order logic, and it is straightforward to verify that Frege systems and the propositional sequent calculus *LK* p-simulate each other [25].

Augmenting Frege systems by the possibility to abbreviate complex formulas by propositional variables, we arrive at the *extended Frege proof system EF*. The extension rule might further reduce the proof size, but it is not known whether *EF* is really stronger than ordinary Frege systems. Both Frege and the extended Frege system are very strong systems for which no non-trivial lower bounds to the proof size are currently known (cf. [11]).

Another way to enhance the power of Frege systems is to allow substitutions not only for axioms but also for all formulas that have been derived in Frege proofs. Augmenting Frege systems by this substitution rule leads to the *substitution Frege system SF*. The extensions *EF* and *SF* were introduced by Cook and Reckhow [25]. While it was already proven there that *EF* is simulated by *SF*, the converse simulation is considerably more involved and was shown independently by Dowd [26] and Krajíček and Pudlák [39].

It is often desirable to further strengthen the proof system *EF* by additional axioms. This can be done by allowing a polynomial-time-computable set Φ as new axioms, i.e., formulas from Φ as well as their substitution instances may be freely used in *EF*-proofs. These schematic extensions of *EF* are denoted by $EF + \Phi$. In this way, we obtain proof systems of arbitrary strength (cf. Proposition 11.7). More detailed information on Frege systems and its extensions can be found in [18] or [34].

§4. Closure Properties of Proof Systems. Although the notion of a propositional proof system was defined by Cook and Reckhow in great generality, propositional proof complexity mostly deals with proof systems satisfying some additional properties. The conditions are usually of logical nature: it should be feasible to carry out basic operations like modus ponens or substitutions in the proof system. These are very natural requirements that are met by most of the studied proof systems. Nevertheless, the general definition of propositional proof systems above permits a great variety of proof systems that violate these conditions.

Definition 4.1. *A proof system P is* closed under modus ponens *if there exists a polynomial p such that for all numbers k and all propositional formulas $\varphi_1, \dots, \varphi_{k+1}$ the following holds. If $P \vdash_{\leq n} \varphi_i$ for $i = 1, \dots, k$ and $P \vdash_{\leq n} \varphi_1 \to \varphi_2 \to \cdots \to \varphi_{k+1}$, then we get $P \vdash_{\leq p(n)} \varphi_{k+1}$.*

This definition not only allows to use modus ponens once with polynomial increase in the proof size, but in fact a polynomial number of times.

If π is a Frege proof of a formula φ, then we can prove substitution instances $\sigma(\varphi)$ of φ by applying the substitution σ to every formula in the proof π. This leads us to the general concept of closure of a proof system under substitutions.

Definition 4.2. *P is* closed under substitutions *if there exists a polynomial q such that $P \vdash_{\leq n} \varphi$ implies $P \vdash_{\leq q(n+|\sigma(\varphi)|)} \sigma(\varphi)$ for all formulas φ and all substitutions σ.*

Modus ponens and substitutions are transformations on proofs which we can also define in a more constructive fashion. As we will need these versions at some places we make the following definition.

Definition 4.3. *A proof system P is* efficiently closed under modus ponens *if there exists a polynomial-time-computable algorithm that takes as input P-proofs $\pi_1, \dots, \pi_k$ of propositional formulas $\varphi_1, \dots, \varphi_k$ together with a P-proof π_{k+1} of $\varphi_1 \to \dots \to \varphi_{k+1}$ and outputs a P-proof of φ_{k+1}.*

Similarly, we say that P is efficiently closed under substitutions *if there exists a polynomial-time procedure that takes as input a P-proof of a formula φ as well as a substitution instance $\sigma(\varphi)$ of φ and computes a P-proof of $\sigma(\varphi)$.*

It also makes sense to consider other properties like closure under conjunctions or disjunctions. A particularly simple property is the following: we say that a proof system *evaluates formulas without variables* if formulas using only constants but no propositional variables have polynomially long proofs. As this is true even for truth-table evaluations, all proof systems simulating the truth-table system evaluate formulas without variables.

We can classify properties of proof systems like those above along the following lines. Some properties are *monotone* in the sense that they are preserved from weaker to stronger systems, i.e., if $P \leq Q$ and P has the property, then also Q satisfies the property. Evaluation of formulas without variables is such a monotone property. Other properties might not be monotone but still *robust* under $\leq$ in the sense that the property is preserved when we switch to a $\leq$-equivalent system. Since we are interested in the degree of a proof system and not in the particular representative of that degree, it is desirable to investigate only robust or even monotone properties. It is straightforward to verify that closure under modus ponens and closure under substitutions are $\leq$-robust properties, whereas the efficient versions of these properties are $\leq_p$-robust.

We remark that Frege systems and their extensions have very good closure properties.

Proposition 4.4. *The Frege system F, the extended Frege system EF, and the substitution Frege system SF are efficiently closed under modus ponens and under substitutions.*

Proof. Modus ponens is available as a rule in F, EF, and SF, hence we have closure under modus ponens. For closure under substitutions let $\varphi_1, \dots, \varphi_k$ be an F-proof of size $\leq m$. If σ is a substitution, then $\sigma(\psi_1), \dots, \sigma(\varphi_k)$ is an F-proof of $\sigma(\varphi_k)$ of size $\leq m|\sigma(\varphi_k)|$. For SF closure under substitutions is immediate, as the substitution rule is available in SF. Finally, for EF this follows from the equivalence $SF \equiv_p EF$. ⊣

The same proposition is also valid for extensions of $EF + \Phi$ by polynomial-time-computable sets of axioms $\Phi \subseteq \mathrm{TAUT}$.

§5. Theories of Bounded Arithmetic. There is a number of different languages for arithmetic theories of which a detailed picture is given in [28]. Here we will only consider the language L introduced by Buss [12], which in addition to the usual ingredients $0, S, +, *, \leq$ contains a number of technical symbols in order to simplify the formalization of syntactic notions with arithmetic formulas.

The language L of arithmetic uses the symbols

$$0,\ S,\ +,\ *,\ |.|,\ \lfloor \tfrac{1}{2}.\rfloor,\ \sharp,\ \text{and} \leq\ .$$

0, S, $+$, $*$, $\lfloor \frac{1}{2}.\rfloor$, and $\leq$ are interpreted in the usual way. The intended interpretation of $|x|$ is $\lceil \log_2(x+1)\rceil$, i.e., the number of bits of the binary representation of x, and the smash function $x\sharp y$ is interpreted by $2^{|x|*|y|}$.

Quantifiers of the form $(\forall x \leq t(y)) \dots$ abbreviating $(\forall x)\, x \leq t(y) \to \dots$ and $(\exists x \leq t(y)) \dots$ abbreviating $(\exists x)\, x \leq t(y) \wedge \dots$ with some L-term t not containing the variable x are called *bounded quantifiers.* Because the function symbol $\sharp$ is included in the language, and in the intended interpretation the smash function $\sharp$ has super-polynomial growth rate, that admits exactly polynomial growth in the length of the number, these bounded quantifiers can range over numbers y of length polynomial in the length of x, i.e., over exponentially large sets measured in $|x|$. If the term t is even of the form $t(y) = |s(y)|$ for some term $s(y)$, then the quantifiers are called *sharply bounded.*

Bounded L-formulas are formulas in the language of L containing only bounded quantifiers. As usual, one defines a hierarchy of first-order formulas by counting their quantifier alternations. Doing this for bounded formulas, we count the number of alternations of bounded quantifiers of bounded L-formulas in prenex normal form, but ignoring quantifiers which are sharply bounded. The first level of this hierarchy is formed by L-formulas containing only sharply bounded quantifiers. These formulas are denoted by Σ^b_0. In the following we are particularly interested in Π^b_1- and Σ^b_1-formulas, which are L-formulas in prenex normal form for which only bounded universal and bounded existential quantifiers are allowed, respectively. Using a pairing function, quantifiers of the same type can be combined, and hence a Π^b_1-formula can be assumed to be of the form

$$(\forall y \leq t(x))\, \varphi(x, y)$$

where φ contains only sharply bounded quantifiers. Similarly, Σ^b_1-formulas look like $(\exists y \leq t(x))\, \varphi(x, y)$.

The formula $\varphi(x, y)$ contains only sharply bounded quantifiers which range over sets of numbers of polynomial size measured in the length of x. Furthermore, φ can make use of all number-theoretic functions available in L. As all these functions are easy to compute, $\varphi(x, y)$ can be evaluated in polynomial time for given numbers x and y. Because the existential quantifier $\exists y \leq t(x)$ can be thought of as a suitable polynomial-size witness corresponding to the input x, a Σ^b_1-formula describes an NP-set of natural numbers. But also all NP-sets can be defined by Σ^b_1-formulas, as the next theorem which is a variant of a result of Wrathall [56] (see e.g. [34]) shows.

Theorem 5.1. *Let $\mathbb{N}$ denote the standard model of natural numbers. The subsets of $\mathbb{N}$ definable by Σ^b_1-formulas are exactly the NP-sets. Similarly, the* *subsets of* $\mathbb{N}$ *definable by* Π_1^b*-formulas equal the set of all* coNP*-sets of natural numbers.*

Actually, this correspondence extends to all bounded formulas and sets from the polynomial hierarchy, but we will only need it for Σ_1^b- and Π_1^b-formulas.

Given an L-theory T we say that a formula φ is a Δ_1^b*-formula with respect to* T if there exist a Σ_1^b-formula ψ and a Π_1^b-formula θ such that $T \vdash \varphi \leftrightarrow \psi$ and $T \vdash \varphi \leftrightarrow \theta$.

There is a long history of studying fragments of Peano arithmetic of different strength (see e.g. [28]). The fragment we need here is the theory S_2^1 that comes from a whole collection of weak fragments of PA, that are usually referred to as bounded arithmetic. The theory S_2^1 was introduced by Buss [12] and is axiomatized by a finite set $BASIC$ of axioms describing the interplay of the interpretations of the function symbols S, $+$, $*$, $|.|$, $\lfloor \frac{1}{2} . \rfloor$, $\sharp$, the relation symbol $\leq$ and the constant 0. As usual, a controlled amount of induction is added to these basic axioms. In this case, a version LIND of the induction scheme for the length of numbers is added

$$\varphi(0) \wedge (\forall x)(\varphi(x) \to \varphi(x+1)) \to (\forall x)\varphi(|x|) \ .$$

Instead of this LIND-scheme it is also possible to use the polynomial induction scheme PIND which is defined as

$$\varphi(0) \wedge (\forall x)(\varphi(\lfloor \frac{x}{2} \rfloor) \to \varphi(x)) \to (\forall x)\varphi(x) \ .$$

The theory S_2^1 is then defined as the axiom set $BASIC$ augmented by the induction scheme LIND for all Σ_1^b-formulas. Equivalently, S_2^1 can be characterized as $S_2^1 = BASIC + \Pi_1^b$-LIND as well as by $BASIC + \Sigma_1^b$-PIND and by $BASIC + \Pi_1^b$-PIND. The index 2 in S_2^1 refers to the presence of the function symbol $\sharp$ in the language, which allows a smooth formalization of coding of sequences. This is needed for the formalization of proof systems and polynomial-time computations in S_2^1. The superscript 1 in S_2^1 indicates that LIND for Σ_1^b-formulas is available in the theory. Adding Σ_i^b-LIND to $BASIC$ defines the theories S_2^i.

A central result for the theory S_2^1 is the witnessing theorem of Buss [12]. It describes that the proof-theoretic strength of S_2^1 corresponds to the polynomial-time-computable functions.

Theorem 5.2 (Buss' Witnessing Theorem [12]). *Let* $\varphi(x, y)$ *be a* Σ_1^b*-formula and let* $S_2^1 \vdash (\forall x)(\exists y)\varphi(x, y)$*. Then there exists a polynomial-time-computable function* f *which for every natural number* x *computes a corresponding witness* y*, i.e.,* $\mathbb{N} \models (\forall x)\varphi(x, f(x))$*.*

This theorem probably is the most important instance of a whole collection of witnessing theorems that provide complexity-theoretic characterizations of the provably total functions of various arithmetic theories [15, 19, 43, 45].

§6. A Translation of Arithmetic Formulas into Propositional Formulas. To explain the connection between bounded arithmetic and propositional proof systems we have to translate first-order formulas into propositional formulas. There are essentially two translations from arithmetic formulas into propositional formulas: one was introduced by Paris and Wilkie [46] to transform bounded formulas in the language of $I\Delta_0$ with one extra predicate into propositional logic. The other translation dates back to Cook [22] and was later adapted by Krajíček and Pudlák [41] to translate L-formulas into sequences of quantified propositional formulas.

We will now describe the second translation in detail. But because we do not consider quantified propositional formulas, we will only explain the part of the translation which does not produce bounded quantifiers.

For L-terms t and bounded L-formulas φ we define inductively *bounding polynomials* q_t and q_φ, such that when substituting numbers of length $\leq n$ for the free variables of t or φ, the evaluation of t and φ does not refer to numbers of length $> q_t(n)$ or $> q_\varphi(n)$, respectively. *Bounding polynomials for L-terms* are inductively defined as follows:

1. $q_0(n) = 1$ for all n,
2. $q_x(n) = n$ for a first-order variable x,
3. $q_{S(t)} = q_t + 1$ where t is an L-term,
4. $q_{s+t} = q_s + q_t$ for L-terms s, t,
5. $q_{s\sharp t} = q_s q_t + 1$ for L-terms s, t, and
6. $q_{|t|} = q_{\lfloor \frac{t}{2} \rfloor} = q_t$ for an L-term t.

Using these bounding polynomials for terms we define inductively *bounding polynomials for bounded L-formulas*:

1. $q_{s \leq t} = q_{s=t} = q_s + q_t$ for L-terms s, t,
2. $q_{\neg\varphi} = q_\varphi$ for a bounded L-formula φ,
3. $q_{\varphi\wedge\psi} = q_{\varphi\vee\psi} = q_{\varphi\to\psi} = q_{\varphi\leftrightarrow\psi} = q_\varphi + q_\psi$ for L-formulas φ, ψ, and
4. $q_{(\forall x \leq t)\varphi}(n) = q_{(\exists x \leq t)\varphi}(n) = q_t(n) + q_\varphi(n + q_t(n))$ for a bounded L-formula φ and an L-term t.

Let $\| + \|_m$, $\| * \|_m$, $\| \lfloor \frac{1}{2} \cdot \rfloor \|_m$, $\| \, |\cdot| \, \|_m$ and $\| \sharp \|_m$ be m-tuples of polynomial-size Boolean formulas computing the first m bits of the corresponding functions on inputs of length m.

For each L-term t we now define for $m > q_t(n)$ an m-tuple $\|t\|^n_m$ of propositional formulas. For every free variable x in t we introduce a sequence $p^x_{n-1}, \dots, p^x_0$ of propositional variables which represent the values of the bits of x where p^x_0 takes the value of the least significant bit. By induction on the logical complexity of terms t we define m-tuples of propositional formulas $\|t\|^n_m$ which compute the first m bits of the value of t for inputs of length $\leq n$:

1. $\|0\|^n_m$ is the m-tuple $(\bot, \dots, \bot)$, where $\bot$ stands for a fixed unsatisfiable formula.
2. For a variable x we set $\|x\|^n_m = (\bot, \dots, \bot, p^x_{n-1}, \dots, p^x_0)$ with $m - n$ leading $\bot$.
3. $\|s + t\|^n_m = \| + \|_m (\|s\|^n_m, \|t\|^n_m)$ for L-terms s and t, and
4. analogously for the other L-functions.

An L-formula φ is in *negation implication normal form* (NINF) if φ is in prenex normal form and does not contain the connectives $\to$ or $\leftrightarrow$, and negations occur only directly before atomic formulas. To a formula φ in NINF we assign special propositional variables $\nu^\varphi_0, \nu^\varphi_1, \dots$, called the universal variables of φ, and propositional variables $\varepsilon^\varphi_0, \varepsilon^\varphi_1, \dots$, called the existential variables of φ.

For Σ^b_1- and Π^b_1-formulas φ in NINF we define by induction on the logical complexity of φ propositional translations $\|\varphi\|^n_m$ for $m \geq q_\varphi(n)$. The translation can be extended to Σ^b_1- and Π^b_1-formulas which are not in NINF by transforming these formulas into NINF. The translation is defined as follows.

1. $\|s = t\|^n_m = EQ_m\left(\|s\|^n_m, \|t\|^n_m\right)$ with $EQ_m\left(\bar{p}, \bar{q}\right) = \bigwedge_{i=0}^{m-1} p_i \leftrightarrow q_i$
2. $\|s \leq t\|^n_m = LE_m\left(\|s\|^n_m, \|t\|^n_m\right)$
   with $LE_m\left(\bar{p}, \bar{q}\right) = \bigvee_{i=0}^{m-1} \left(\left(\bigwedge_{j=i+1}^{m-1} p_j \leftrightarrow q_j\right) \wedge \neg p_i \wedge q_i\right) \vee EQ_m\left(\bar{p}, \bar{q}\right)$
3. $\|\neg\varphi\|^n_m = \neg\|\varphi\|^n_m$ for atomic formulas φ
4. $\|\varphi \wedge \psi\|^n_m = \|\varphi\|^n_m \wedge \|\psi\|^n_m$
5. $\|\varphi \vee \psi\|^n_m = \|\varphi\|^n_m \vee \|\psi\|^n_m$
6. $\|\left(\forall x \leq t\right)\varphi(x)\|^n_m = \|\neg(x \leq t) \vee \varphi(x)\|^n_m \left(p^x_i/\nu^\varphi_i\right)^{m-1}_{i=0}$,
   where the term t is not of the form $|s|$. The suffix $\left(p^x_i/\nu^\varphi_i\right)^{m-1}_{i=0}$ indicates that the variables $p^x_{m-1}, \dots, p^x_0$ are replaced by the universal variables $\nu^\varphi_{m-1}, \dots, \nu^\varphi_0$. This is necessary for the case that φ contains several universal quantifications over x.
7. $\|\left(\exists x \leq t\right)\varphi(x)\|^n_m = \|(x \leq t) \wedge \varphi(x)\|^n_m \left(p^x_i/\varepsilon^\varphi_i\right)^{m-1}_{i=0}$,
   where the term t is not of the form $|s|$. Again, the substitution $\left(p^x_i/\varepsilon^\varphi_i\right)^{m-1}_{i=0}$ is necessary because the formula that we want to translate might contain more than one existential quantification over x. But as these different existential quantifiers are usually not witnessed by the same element, we need different propositional variables for each quantifier.
8. $\|\left(\forall x \leq |t|\right)\varphi(x)\|^n_m = \bigwedge_{k=0}^{m-1} \|\neg\left(\underline{k} \leq |t|\right) \vee \varphi\left(\underline{k}\right)\|^n_m$, where $\underline{k}$ is some dyadic representation of the natural number k.
9. $\|\left(\exists x \leq |t|\right)\varphi(x)\|^n_m = \bigvee_{k=0}^{m-1} \|\underline{k} \leq |t| \wedge \varphi\left(\underline{k}\right)\|^n_m$

In the following we will omit the explicit reference to the bounding polynomial and write simply $\|\varphi\|^n$ in place of $\|\varphi\|^n_{q(n)}$. Abbreviating further, we will use $\|\varphi(x)\|$ to denote the set $\{\|\varphi(x)\|^n \mid n \geq 0\}$. We will also usually associate first-order formulas $\varphi(\bar{x})$ with free variables with their universally closed counterparts $(\forall\bar{x})\varphi(\bar{x})$. Therefore, the above translation is not only suitable for Π^b_1- but in fact for $\forall\Pi^b_1$-formulas.

The formula $\|\varphi(x)\|^n$ has n propositional variables $p^x_{n-1}, \dots, p^x_0$ corresponding to the bits of x. If $\varphi(x) = (\forall y \leq t)\psi(x, y)$ is a Π^b_1-formula, then additionally the universal variables $\nu^\psi_0, \nu^\psi_1, \dots$ occur in $\|\varphi(x)\|^n$. If $a \in \mathbb{N}$ is a number of length $\leq n$, we denote the bits of a by $\bar{a}$. Substituting $p^x_{n-1}, \dots, p^x_0$ by the constants $\bar{a}$, we arrive at formulas $\|\varphi(x)\|^n(\bar{p}^x/\bar{a})$ with only the universal variables $\nu^\varphi_0, \nu^\varphi_1, \dots$ remaining free. These formulas provide a precise description of the truth value of $\varphi(a)$. We state this in the next theorem which is essentially due to Cook [22]. Its proof is immediate from the construction of the translations $\|.\|$.

Theorem 6.1 (Cook [22]). 1. *For $\varphi \in \Pi^b_1$ or $\varphi \in \Sigma^b_1$ the sequence $\|\varphi\|^n = \|\varphi\|^n_{q(n)}$ consists of propositional formulas which have polynomial size in n. Moreover, the sequence $\|\varphi\|^n$ is polynomial-time constructible, i.e., there* *exists a polynomial-time-computable algorithm that on input 1^n outputs the formula $\|\varphi\|^n$.*

2. *The sequence $\|\varphi\|^n$ propositionally describes the first-order formula φ.*

   More precisely, if $\varphi(x) \in \Pi_1^b$, then for all $a \in \mathbb{N}$ with $|a| \leq n$ the formula $\|\varphi(x)\|^n(\bar{p}^x/\bar{a})$ is a tautology if and only if $\mathbb{N} \models \varphi(a)$. In particular, the formula $\|\varphi(x)\|^n$ is a tautology if and only if $\varphi(a)$ holds for all natural numbers a of length $\leq n$.

   If $\varphi(x) \in \Sigma_1^b$, then for all $a \in \mathbb{N}$ with $|a| \leq n$ the formula $\|\varphi(x)\|^n(\bar{p}^x/\bar{a})$ is satisfiable if and only if $\mathbb{N} \models \varphi(a)$.

§7. Coding Propositional Proofs in Bounded Arithmetic. In order to formalize concepts such as propositional proof systems in L-theories, it is necessary to define polynomial-time computations with L-formulas. As the language L was suitably chosen to include the technical symbols $|.|$, $\lfloor \frac{1}{2}.\rfloor$, and $\sharp$, it is relatively easy to define a pairing function and a coding of finite sets and sequences (cf. [28]). Using this it is possible to code descriptions of Turing machine computations. In particular, using the length induction scheme LIND, the theory S_2^1 can prove the uniqueness of suitably encoded polynomial-time computations, i.e., S_2^1 proves that for all polynomial-time deterministic Turing machines M and all inputs x there exists exactly one computation of $M(x)$. Expressed differently, polynomial-time computations are Δ_1^b-definable in S_2^1 (cf. Chap. V of [28] or Chap. 6 of [34] for the details).

Encoding propositional formulas as numbers in some straightforward way, we can in a theory T speak of propositional formulas, assignments, and proofs. Instead of giving the details of the encoding we will just introduce some notation (similar as in [34]). A more detailed description of these concepts can be found in [17].

First we need to encode propositional formulas as numbers. Let

$$\mathit{Form}$$

be a Σ_0^b-formula such that $\mathbb{N} \models \mathit{Form}(\varphi)$ if and only if φ is the encoding of a propositional formula. Let

$$\mathit{Assign}(\alpha, \varphi)$$

be a Σ_0^b-formula describing that α is the encoding of an assignment of the variables of the propositional formula encoded by φ. Similarly, let the Σ_0^b-formula

$$\mathit{Eval}(\alpha, \varphi, \gamma)$$

describe that γ is an evaluation of the propositional formula φ under the assignment α. By

$$\alpha \models \varphi$$

we denote a first-order description for the fact that α is a satisfying assignment for the formula φ. Using the earlier definitions, $\alpha \models \varphi$ can be expressed as

$$(\exists \gamma) \mathit{Eval}(\alpha, \varphi, \gamma) \wedge \varphi(\gamma) = 1 \ .$$

Since the length of γ can be bounded by a polynomial in the length of φ, this is a Σ_1^b-formula. In the following we will always assume that quantifiers such as $\exists \gamma$

above are implicitly bounded by the quantified formulas. Because the evaluation γ of the formula φ is unique and this uniqueness is provable in S^1_2, i.e.,

$$S^1_2 \vdash \mathit{Eval}(\alpha, \varphi, \gamma_1) \wedge \mathit{Eval}(\alpha, \varphi, \gamma_2) \to \gamma_1 = \gamma_2 \ ,$$

it follows that

$$(\forall \gamma)\mathit{Eval}(\alpha, \varphi, \gamma) \to \varphi(\gamma) = 1$$

is a Π^b_1-definition of $\alpha \models \varphi$ which is in S^1_2 provably equivalent to the above Σ^b_1-definition, hence $\alpha \models \varphi$ is Δ^b_1 with respect to S^1_2 (cf. [34] Sect. 9.3 for the details).

Now we are ready to formalize tautologies. For this let $\mathit{Taut}(\varphi)$ be an L-formula asserting that all assignments satisfy the formula φ, i.e.,

$$(\forall \alpha)\mathit{Assign}(\alpha, \varphi) \to \alpha \models \varphi \ .$$

Because $\alpha \models \varphi$ has a Π^b_1-definition and *Assign* is a Σ^b_0-formula, this definition of *Taut* is a Π^b_1-formula.

Finally, we need to code propositional proofs. For a propositional proof system P let

$$\mathit{Prf}_P(\pi, \varphi)$$

be an L-formula describing that π is the encoding of a correct P-proof of the propositional formula encoded by φ. Because P is a polynomial-time-computable function, Prf_P is definable by a Σ^b_1-formula. But like all polynomial-time-computable functions the predicate Prf_P also has a Π^b_1-definition. Moreover, these definitions can be chosen in such a way that the theory S^1_2 proves their equivalence, hence Prf_P is Δ^b_1-definable with respect to S^1_2.

§8. Consistency Statements. The consistency of a proof system is described by the *consistency statement* of a proof system

$$\mathit{Con}(P) = (\forall \pi)\neg \mathit{Prf}_P(\pi, \bot) \ .$$

A somewhat stronger formulation of consistency is given by the *reflection principle* of a propositional proof system P which is defined by the L-formula

$$\mathit{RFN}(P) = (\forall \pi)(\forall \varphi)\mathit{Prf}_P(\pi, \varphi) \to \mathit{Taut}(\varphi) \ .$$

From the remarks in the previous section it follows that $\mathit{Con}(P)$ and $\mathit{RFN}(P)$ are $\forall\Pi^b_1$-formulas.

These two consistency notions are compared by the following well-known observation, contained e.g. in [34]:

Proposition 8.1. *Let P be a proof system that is closed under substitutions and modus ponens and evaluates formulas without variables. Assume further that these properties are provable in S^1_2. Then $S^1_2 \vdash \mathit{RFN}(P) \leftrightarrow \mathit{Con}(P)$.*

Proof. Assume $S^1_2 \vdash \mathit{RFN}(P)$. In particular, this means that

$$S^1_2 \vdash (\forall \pi)\mathit{Prf}_P(\pi, \bot) \to \mathit{Taut}(\bot) \ .$$

Because $\mathit{Taut}(\bot)$ is false in S^1_2, this implies $S^1_2 \vdash (\forall \pi)\neg \mathit{Prf}_P(\pi, \bot)$, which means $S^1_2 \vdash \mathit{Con}(P)$.

For the opposite implication assume $S^1_2 \not\vdash RFN(P)$. Hence there exists a model M of S^1_2 and a propositional formula $\varphi(\bar{p})$ such that

$$M \models (\exists\pi)Prf_P(\pi, \varphi(\bar{p})) \wedge \neg\, Taut(\varphi(\bar{p})) \ .$$

By the definition of *Taut* this means that there exists an assignment α such that

$$M \models (\exists\pi)Prf_P(\pi, \varphi(\bar{p})) \wedge \alpha \not\models \varphi(\bar{p}) \ .$$

Let α map the variables $\bar{p}$ of $\varphi(\bar{p})$ to the tuple $\bar{a}$. Hence $\varphi(\bar{a})$ is a false formula without variables. By assumption S^1_2 proves that $\neg\varphi(\bar{a})$ is provable in P. Because P is provably closed under substitutions, we get

$$M \models (\exists\pi)Prf_P(\pi, \varphi(\bar{a})) \wedge (\exists\pi')Prf_P(\pi', \neg\varphi(\bar{a})) \ .$$

By closure of P under modus ponens in S^1_2 we obtain $M \models (\exists\pi)Prf_P(\pi, \bot)$. Hence $Con(P)$ fails in M, and because $M \models S^1_2$, the theory S^1_2 does not prove the consistency principle of P. $\dashv$

Very often we will consider propositional descriptions of the reflection principle. These can be simply obtained by translating $RFN(P)$ to a sequence of propositional formulas using the translation $\|.\|$.

DEFINITION 8.2. *A propositional proof system P has the* reflection property *if* $P \vdash_* \|RFN(P)\|^n$.

At some places we need the more efficient version of this definition that short P-proofs of $\|RFN(P)\|^n$ are constructible.

DEFINITION 8.3. *We say that a propositional proof system P has the* strong reflection property *if there exists a polynomial-time algorithm that on input* 1^n *outputs a P-proof of* $\|RFN(P)\|^n$.

There is a subtle problem with Definitions 8.2 and 8.3 that is somewhat hidden in the definitions. Namely, the formula Prf_P describes the computation of some Turing machine computing the function P. However, the provability of the formulas $\|RFN(P)\|^n$ with polynomial-size P-proofs might depend on the actual choice of the Turing machine computing P. We will illustrate this by an example which unfortunately has to be postponed until Sect. 13 (Proposition 13.2). Nevertheless, this observation tells us that we should understand the meaning of Definition 8.2 in the following, more precise way: a propositional proof system P has the reflection property if there exists a deterministic polynomial-time Turing machine M computing the function P such that for a suitable Δ^b_1-formalization Prf_P of the computation of M with respect to S^1_2 we have

$$P \vdash_* \|(\forall\pi)(\forall\varphi)Prf_P(\pi, \varphi) \rightarrow Taut(\varphi)\|^n \ .$$

The same applies to Definition 8.3. For this definition of reflection we can show the robustness of the reflection principle under p-simulations, namely:

PROPOSITION 8.4. *Let P and Q be p-equivalent proof systems. Then P has (strong) reflection if and only if Q has (strong) reflection.*

PROOF. Let g compute a p-simulation of P by Q, and assume that Q has strong reflection. We want to show strong reflection for P. Consider the following polynomial-time Turing machine that computes the proof system P: at input

π, we first evaluate $g(\pi)$ and then $Q(g(\pi))$. Clearly, $Q \circ g$ computes P. We claim that Q proves the reflection principle $\mathit{RFN}(P)$ with respect to the Turing machine $Q \circ g$. This follows, as Q proves $\mathrm{rng}(Q) \subseteq \mathrm{TAUT}$, and therefore also $\mathrm{rng}(Q \circ g) \subseteq \mathrm{TAUT}$ by simply proving $\mathit{RFN}(Q)$ and ignoring the computation of g. Because $P \equiv_p Q$, we also get polynomial-size P-proofs of $\mathit{RFN}(P)$ with respect to $Q \circ g$.

The proof for reflection instead of strong reflection proceeds analogously. ⊣

§9. The Correspondence Between Arithmetic Theories and Propositional Proof Systems. Now we have made sufficient preparations to treat the main topic of this paper, namely the correspondence between arithmetic theories and propositional proof systems. Rather than concentrating on specific theories and corresponding proof systems, we will pursue a general axiomatic approach. Krajíček and Pudlák introduced in [41] a general correspondence between L-theories T and propositional proof systems P. Pairs (T, P) from this correspondence possess in particular the following two properties:

1. For all Π_1^b-formulas $\varphi(x)$ with $T \vdash (\forall x)\varphi(x)$ we have $P \vdash_* \|\varphi(x)\|^n$.
2. T proves the correctness of P, i.e., $T \vdash \mathit{RFN}(P)$. Furthermore P is the strongest proof system for which T proves the correctness, i.e., $T \vdash \mathit{RFN}(Q)$ for a proof system Q implies $Q \leq P$.

Actually, [41] contains a stronger formulation, namely properties 1 and 2 are required to be provable in S_2^1. These properties then take the following form:

3. For all Π_1^b-formulas $\varphi(x)$ with $T \vdash (\forall x)\varphi(x)$ the theory S_2^1 proves the formula $(\forall n)(\exists \pi_n)\mathit{Prf}_P(\pi_n, \|\varphi(x)\|^{|n|})$.
4. T proves the correctness of P, i.e., $T \vdash \mathit{RFN}(P)$.

From Buss' witnessing theorem for S_2^1 (Theorem 5.2) it follows that a proof π_n of $\|\varphi(x)\|^{|n|}$ can be computed in polynomial time from the number n. Therefore condition 3 implies condition 1.

It is then even possible to derive the second part of property 2 as a consequence of 3 and 4 (cf. [47]), i.e., if T and P fulfill the conditions 3 and 4, then every proof system Q with $T \vdash \mathit{RFN}(Q)$ is p-simulated by P, and this p-simulation is provable in S_2^1. In contrast, we only stated the weak simulation $Q \leq P$ in condition 2.

For many applications conditions 1 and 2 are sufficient. Therefore we make the following definition:

Definition 9.1. *A propositional proof system P is called* regular *if there exists an L-theory T such that properties 1 and 2 are fulfilled for (T, P).*

Occasionally, a strengthened version of regularity is needed, but still weaker than properties 3 and 4.

Definition 9.2. *We call a propositional proof system P* strongly regular *if there exists an L-theory T such that the following two properties are fulfilled for (T, P).*

5. *Let $\varphi(x)$ be a Π_1^b-formula such that $T \vdash (\forall x)\varphi(x)$. Then there exists a polynomial-time-computable function which on input 1^n outputs a P-proof of $\|\varphi(x)\|^n$.*

6. *$T \vdash RFN(P)$, and if $T \vdash RFN(Q)$ for some proof system Q, then $Q \leq_p P$.*

In comparison to regularity conditions 1 and 2 we gave these axioms a constructive formulation: in condition 5 P-proofs are polynomial-time constructible and in 6 we have p-simulations instead of $\leq$. Clearly, conditions 3 and 4 imply the strong regularity conditions 5 and 6 which in turn imply the regularity conditions 1 and 2. In Sect. 11 we will discuss sufficient conditions for the regularity and strong regularity of propositional proof systems.

If T is an L-theory such that there exists a regular proof system P satisfying conditions 1 and 2, then P is unique up to $\leq$-equivalence by property 2. Conversely, if P is a proof system for which there exists an L-theory T satisfying conditions 3 and 4, then the $\forall\Pi_1^b$-consequences of T are determined by P. This is the content of the next theorem which is essentially contained in [41].

THEOREM 9.3. 1. *Let T be an L-theory and P_1, P_2 be proof systems such that both (T, P_1) and (T, P_2) satisfy conditions 1 and 2. Then $P_1 \equiv P_2$.*

2. *Let $T \supseteq S_2^1$ be an L-theory and P a proof system such that conditions 3 and 4 are satisfied for (T, P). Then the theories T and $S_2^1 + RFN(P)$ have the same set of $\forall\Pi_1^b$-consequences.*

PROOF. Part 1 follows immediately from condition 2 for (T, P_1) and (T, P_2).

For part 2 let T be an extension of S_2^1 and let P be a proof system such that conditions 3 and 4 hold. As $S_2^1 \subseteq T$ and $T \vdash RFN(P)$, all $\forall\Pi_1^b$-consequences of $S_2^1 + RFN(P)$ are also provable in T.

For the other inclusion let $\varphi(x)$ be a Π_1^b-formula such that $T \vdash (\forall x)\varphi(x)$. By condition 3 this implies $S_2^1 \vdash (\forall n)(\exists \pi_n) Prf_P(\pi_n, \|\varphi(x)\|^{|n|})$. Using the reflection principle of P we infer $S_2^1 + RFN(P) \vdash (\forall n)\, Taut(\|\varphi(x)\|^{|n|})$. By induction on the logical complexity of φ we can show

$$S_2^1 \vdash (\forall n)\, Taut(\|\varphi(x)\|^{|n|}) \to (\forall x)(|x| \leq |n| \to \varphi(x))$$

and hence we obtain $S_2^1 + RFN(P) \vdash (\forall x)\varphi(x)$. ⊣

Before we continue the investigation of regular systems we will give an informal discussion on the properties of the correspondence between arithmetic theories and propositional proof systems.

Let us start with the second axiom, as this is easier and shorter to explain. Part 2 of the correspondence expresses that from the knowledge of the theory T the proof system P is an optimal proof system. This can be used to show simulations between proof systems. Namely, to show $Q \leq P$ for a regular proof system P it suffices to prove $RFN(Q)$ in the theory T associated with P. In this way it was shown for example that the substitution Frege system SF is simulated by the extended Frege system EF [26, 39]. For this it is enough to verify that $S_2^1 \vdash RFN(SF)$ which is considerably simpler than to give a direct propositional simulation [39].

Part 1 of the correspondence is called the *simulation of T by P*. Its main application is the uniform construction of P-proofs. We will explain this in some more detail. If some $\forall\Pi_1^b$-formula φ is T-provable, then, as $\mathbb{N}$ is a model of T, we have in particular $\mathbb{N} \models \varphi$. Hence by Theorem 6.1 the sequence $\|\varphi\|^n$ contains only tautologies. Moreover, by part 1 of the correspondence the tautologies of this sequence have polynomial-size P-proofs. Usually these P-proofs are also constructible in polynomial time as follows. The T-proof of φ is given in some first-order sequent calculus suitable for the language L. The first-order sequent calculus proof of φ is then translated to a sequence of propositional proofs in some propositional sequent calculus, which is a propositional counterpart of the first-order calculus. The translation proceeds by replacing each application of a first-order rule by an application of the corresponding propositional rule. As the first-order rules are often more flexible than their propositional versions, it is necessary to fill in the gaps between the steps. If carefully done, this results in a sequence of propositional proofs of polynomial size in the respective propositional calculus, which then has to be transformed into a sequence of P-proofs. We will sketch this procedure for the correspondence of S^1_2 and EF in Sect. 10.

If one replaces condition 1 by the stronger condition 3, then P-proofs for the sequence $\|\varphi\|^n$ are always constructible in polynomial time. This follows from condition 3 because Buss' witnessing theorem applied to

$$S^1_2 \vdash (\forall n)(\exists \pi_n) \mathit{Prf}_P(\pi_n, \|\varphi(x)\|^{|n|})$$

yields a polynomial-time-computable function f that on input n produces the P-proof π_n.

As it usually is easier to show the validity of a first-order principle in some theory than to explicitely construct sequences of propositional proofs, the correspondence provides an elegant method to construct short propositional proofs. Therefore theories of bounded arithmetic and propositional proof systems are often seen in analogy to the correspondence of Turing machines to Boolean circuits as the uniform and respective non-uniform realization of the same concept.

Additionally, the correspondence also allows to show lower bounds to the length of propositional proofs. This requires some model-theoretic machinery which we will describe next. Let M be a non-standard model of $\mathit{Th}(\mathbb{N})$ and let $n \in M$ be a non-standard element. Then we define the cut M_n in the model M as

$$M_n = \{b \in M \mid |b| \leq n^k \text{ for some } k \in \mathbb{N}\} \enspace .$$

The next theorem offers a model-theoretic way to show lower bounds to the length of propositional proofs.

Theorem 9.4 (Krajíček, Pudlák [40]). *Let P be a regular proof system and let T be the theory corresponding to P. Assume further that P is closed under modus ponens and substitutions by constants, and let $\varphi(x)$ be a Π^b_1-formula. Then the following two conditions are equivalent:*

1. *For every model $M \models \mathit{Th}(\mathbb{N})$ and every non-standard element $a \in M \setminus \mathbb{N}$, $|a| = n$, there exists a model $N \supseteq M_n$ such that $N \models T \cup \{\neg\varphi(a)\}$ and N preserves P-proofs, i.e., if $M_n \models \mathit{Prf}_P(\pi, \psi)$ for some π, ψ, then also $N \models \mathit{Prf}_P(\pi, \psi)$.*
2. *There does not exist a sequence of pairwise distinct natural numbers a_i, $i \in \mathbb{N}$, of length $n_i = |a_i|$ such that $P \vdash_* \|\varphi(x)\|^{n_i}(\bar{p}^x/\bar{a}_i)$.*

Proof. For the forward implication let a_i, $i \in \mathbb{N}$, be pairwise distinct natural numbers and let $n_i = |a_i|$. Assume that $\|\varphi(x)\|^{n_i}(\bar{p}^x/\bar{a}_i)$ have P-proofs of length

$\leq n_i^k$ for some $k \in \mathbb{N}$, i.e.,

$$\mathbb{N} \models (\exists \pi)|\pi| \leq n_i^k \wedge \mathit{Prf}_P(\pi, \|\varphi(x)\|^{n_i}(\bar{p}^x/\bar{a}_i)) \ .$$

By compactness there exist a model $M \models \mathit{Th}(\mathbb{N})$ and a non-standard element $a \in M \setminus \mathbb{N}$, $|a| = n$, such that

$$M \models (\exists \pi)|\pi| \leq n^k \wedge \mathit{Prf}_P(\pi, \|\varphi(x)\|^{n}(\bar{p}^x/\bar{a})) \ .$$

Let now N be a model satisfying the conditions from 1. Because a and π are elements from M_n, and N preserves P-proofs, we obtain from the validity of $\mathit{Prf}_P(\pi, \|\varphi(x)\|^{n}(\bar{p}^x/\bar{a}))$ in M_n also

$$N \models \mathit{Prf}_P(\pi, \|\varphi(x)\|^{n}(\bar{p}^x/\bar{a})) \ .$$

N is a model of T and hence $N \models \mathit{RFN}(P)$, which together with the previous line gives $N \models \mathit{Taut}(\|\varphi(x)\|^{n}(\bar{p}^x/\bar{a}))$. On the other hand, $N \models \neg\varphi(a)$ yields an assignment α such that $N \models (\alpha \models \neg\|\varphi(x)\|^{n}(\bar{p}^x/\bar{a}))$, which gives a contradiction.

For the reverse implication let $M \models \mathit{Th}(\mathbb{N})$ and $a \in M \setminus \mathbb{N}$ with $|a| = n$. Assume that for all $N \supseteq M_n$, $N \models T$ we have $N \models \varphi(a)$. Then we infer

$$\mathit{Diag}(M_n) \cup T \vdash \varphi(a) \ ,$$

where the diagram $\mathit{Diag}(M_n)$ contains all closed L-formulas using constants from M_n, that are valid in M_n. By compactness there exist a tuple $\bar{b} \in M_n$ and a formula $\psi(a, \bar{b}) \in \mathit{Diag}(M_n)$ such that $T \vdash \psi(a, \bar{b}) \to \varphi(a)$. Hence $T \vdash (\forall x, \bar{y})\, \psi(a, \bar{y}) \to \varphi(a)$. As this is a $\forall\Pi_1^b$-formula, there exist polynomial-size P-proofs of the formulas

$$(*) \qquad \|\psi(x, \bar{y}) \to \varphi(x)\|^{n,\bar{m}} = \|\psi(x, \bar{y})\|^{n,\bar{m}} \to \|\varphi(\bar{p}^x)\|^{n} \ .$$

Because $\bar{b} \in M_n$, we have in particular $|\bar{b}| \leq |a|^k$ for some $k \in \mathbb{N}$. Therefore the P-proofs of the formulas $(*)$ have proofs of size polynomial in n.

Because $M \models \mathit{Th}(\mathbb{N})$ and for the non-standard elements a and $\bar{b}$ we have $M \models \psi(a, \bar{b})$, there exists by compactness an infinite sequence of standard elements $\mathbb{N} \models \psi(a_i, \bar{b}_i)$. As the formulas $\psi(a_i, \bar{b}_i)$ are contained in $\mathit{Diag}(M_a)$, their $\|.\|$-translations have polynomial-size P-proofs. Because P is closed under modus ponens and substitutions by constants, we get polynomial-size P-proofs of the formulas $\|\varphi(x)\|^{|a_i|}(\bar{p}^x/\bar{a}_i)$ by substituting $a_i, \bar{b}_i$ into the P-proofs of the formulas $(*)$. ⊣

Intuitively, the above theorem states, that proving a super-polynomial lower bound to the proof size of regular proof systems is equivalent to some model-theoretic task. Namely, a sequence of tautologies arising from an arithmetic formula φ is hard for a regular proof system (cf. Sect. 12), if and only if for every non-standard element a in some model $M \models \mathbb{N}$, we can construct a model of T that extends $M_{|a|}$ and falsifies $\varphi(a)$. In particular, this opens the way to employ model-theoretic techniques such as forcing to obtain lower bounds to the proof size [3, 30, 34, 54]. This is the general set-up of Ajtai's famous result on bounded-depth Frege systems [1], as well as of recent approaches to obtain lower bounds for strong systems [35, 36, 37].

§10. The Correspondence Between S^1_2 and EF. Probably the most important instance of the general correspondence from the last section is the correspondence between S^1_2 and EF, of which this section offers a brief description. We start with property 1 of the correspondence, which states the simulation of S^1_2 by EF. We will only sketch the proof as a complete presentation is very tedious. The theorem is essentially contained in [22], but for the theory PV instead of S^1_2. A complete proof is contained in [34].

Theorem 10.1 (Cook [22], Buss [12]). *Let φ be a Π^b_1-formula. Then $S^1_2 \vdash (\forall\bar{x})\varphi(\bar{x})$ implies $EF \vdash_* \|\varphi(\bar{x})\|^n$. In fact, the EF-proofs of $\|\varphi(\bar{x})\|^n$ can be constructed in polynomial time.*

Proof. The proof proceeds in three main steps. In the first step we fix a first-order sequent calculus *LKB*, which extends the usual propositional sequent calculus *LK* by rules for the introduction of quantifiers, both bounded and unbounded. Additionally, for all axioms A from *BASIC*, sequents $\longrightarrow A$ are introduced, and the polynomial induction scheme PIND is formalized by some suitable inference rule. The sequent calculus *LKB* is defined in such a way that for any formula B we have $S^1_2 \vdash B$ if and only if the sequent $\longrightarrow B$ has an $LKB + \Sigma^b_1$-PIND-proof from the initial sequents corresponding to *BASIC.*

For the second step assume now that, as in the hypothesis of this theorem, $\varphi(\bar{x})$ is a Π^b_1-formula such that $S^1_2 \vdash (\forall\bar{x})\varphi(\bar{x})$. By the first step above this means that there exists an $LKB + \Sigma^b_1$-PIND-proof π of $\longrightarrow (\forall x)\varphi(x)$ from the sequents for *BASIC.* By Gentzen's cut-elimination theorem [27], adapted to the *LKB*-calculus [12], it follows that the proof π can be chosen in such a way that all formulas occurring in π are Σ^b_1 or Π^b_1.

In the third step we transform the *LKB*-proof π from the second step to a sequence of propositional EF-proofs. The idea of this simulation of S^1_2 by EF is to choose a bounding polynomial q that bounds all formulas in π and then translate every formula B occurring in π to $\|B\|^m_{q(m)}$. This is possible as all formulas B in π are Σ^b_1- or Π^b_1-formulas. This itself might not produce valid EF-proofs, but filling the gaps by polynomial-size EF-derivations results in the desired EF-proofs of $\|\varphi\|^m_{q(m)}$. Particularly this third step presents some nontrivial technical details, which, however, we will omit altogether. ⊣

Examining the proof of this theorem it turns out that the theorem is still valid if both the theory S^1_2 and the proof system EF are enhanced by further axioms. In particular, to add the reflection principle of a propositional proof system will be of central interest for the following section. We formulate this version of Theorem 10.1 in the following corollary.

Corollary 10.2. *Let Φ be a polynomial-time-decidable set of true Π^b_1-formulas, i.e., $\mathbb{N} \models \varphi$ for all $\varphi \in \Phi$. Then the proof system $EF + \|\Phi\|$ simulates the theory $S^1_2 + \Phi$, i.e., for all Π^b_1-formulas ψ, provability of $(\forall\bar{x})\psi(\bar{x})$ in S^1_2 implies that $EF + \|\Phi\|$-proofs for $\|\psi(\bar{x})\|^n$ can be constructed in polynomial time.*

Proof. Adding the formulas Φ as axioms to the theory S^1_2 corresponds to enhancing the first-order sequent calculus *LKB* from the first step of the previous proof by the initial sequents $\longrightarrow \varphi$ for all formulas $\varphi \in \Phi$. The transformation of these sequents into $EF + \|\Phi\|$-proofs in the third step of the last proof does not present any problem, as the $\|.\|$-translations of all formulas from Φ are available in the proof system. $\dashv$

Before we can come to part 6 of the correspondence between S^1_2 and EF, we need a technical lemma which describes that EF can evaluate the $\|.\|$-translations of the first-order formula *Taut*. The proof proceeds by induction on the logical complexity of formulas.

Lemma 10.3 (Krajíček, Pudlák [41]). *For all propositional formulas φ we have $EF \vdash_* \| Taut(\varphi)\|^{|\varphi|} \to \varphi$. Moreover, EF-proofs of these formulas are constructible in polynomial time.*

We continue with property 6 of the correspondence.

Theorem 10.4 (Krajíček, Pudlák [41]). *$S^1_2 \vdash RFN(EF)$.*

Proof. We have to show $S^1_2 \vdash (\forall\pi)(\forall\varphi) Prf_{EF}(\pi,\varphi) \to Taut(\varphi)$. Assume that $\pi = (\varphi_1, \varphi_2, \dots, \varphi_n = \varphi)$ is an EF-proof of φ and $S^1_2 \vdash Prf_{EF}(\pi,\varphi)$. We have to show $S^1_2 \vdash Taut(\varphi)$, which is by definition

$$S^1_2 \vdash (\forall\alpha) Assign(\alpha,\varphi) \to \alpha \models \varphi \ .$$

Assume that in the proof π the propositional variables $\bar{p}$ occur together with the extension variables $\bar{q}$. Consider the formula

$$\theta(\alpha, i) = (\exists\beta) Assign(\beta, \bar{q}) \wedge \alpha \cup \beta \models \bigwedge_{j=1}^{i} \varphi_j$$

expressing that the assignment α can be extended to an assignment to the extension variables $\bar{q}$ that satisfies the first i formulas from the proof π.

Formulas and proofs are coded by numbers, using a pairing function, which at least doubles the numbers in each application. Therefore, the PIND-induction scheme, available in S^1_2, enables us to use induction on the numbers coding the proof steps φ_i, i.e., we can argue by induction on the number of steps. Hence by verifying the correctness of the EF-axioms and rules in S^1_2 we can prove the formula $\theta(\alpha, n)$ by induction on i in $\theta(\alpha, i)$. Because the extension variables do not occur in $\varphi_n = \varphi$ we have shown $\alpha \models \varphi$. As this was shown for all assignments α, we obtain $Taut(\varphi)$. $\dashv$

In order to generalize this theorem to schematic extensions of EF we need the following lemma:

Lemma 10.5. *$S^1_2 \vdash (\forall x)\varphi(x) \to (\forall y)\, Taut(\|\varphi(x)\|^{|y|})$ for all Π^b_1-formulas $\varphi(x)$.*

Proof. The lemma could be proved by induction on the logical complexity of φ. However, we can also derive it from the results proved so far. Namely, let $\varphi(x)$ be a Π^b_1-formula such that $S^1_2 \vdash (\forall x)\varphi(x)$. As the proof of Theorem 10.1 formalizes in the theory S^1_2, we get

$$S^1_2 \vdash (\forall y)(\exists\pi) Prf_{EF}(\pi, \|\varphi(x)\|^{|y|}) \ .$$

Using Theorem 10.4 we obtain $S^1_2 \vdash (\forall x)\varphi(x) \to (\forall y)\, Taut(\|\varphi(x)\|^{|y|})$, as claimed. $\dashv$

Examining the proof of Theorem 10.4 again for extensions $EF + \|\Phi\|$ we get:

Corollary 10.6. *Let Φ be a polynomial-time-decidable set of true Π^b_1-formulas. Then $S^1_2 + \Phi \vdash RFN(EF + \|\Phi\|)$.*

Proof. The proof proceeds again by induction on i in the formula $\theta(\alpha, i)$ defined in the proof of Theorem 10.4. The only difference is that in the induction step for the case that φ_i is a formula of the form $\|\psi\|^n$ with $\psi \in \Phi$, we use the formula ψ, which is available as an axiom in $S^1_2 + \Phi$, to derive $Taut(\|\psi\|^n)$ by Lemma 10.3. This suffices to prove $\theta(\alpha, i)$. ⊣

To check property 6 for S^1_2 and *EF*, it remains to show that S^1_2 cannot prove the consistency of any proof system stronger than *EF*. This is stated in the next theorem.

Theorem 10.7 (Krajíček, Pudlák [41]). *Let P be a propositional proof system such that $S^1_2 \vdash RFN(P)$. Then EF p-simulates P.*

As before we state the general result for extensions of *EF*. We postpone the proof to the next section.

Theorem 10.8. *Let Φ be a polynomial-time-decidable set of true Π^b_1-formulas, and let P be a propositional proof system such that $S^1_2 + \Phi \vdash RFN(P)$. Then $EF + \|\Phi\|$ p-simulates P.*

Combining Corollaries 10.2 and 10.6 and Theorem 10.8 we obtain

Theorem 10.9. *Let Φ be a polynomial-time-decidable set of true Π^b_1-formulas. Then the proof system $EF + \|\Phi\|$ is strongly regular and corresponds to the theory $S^1_2 + \Phi$. In particular, the system $EF + \|\Phi\|$ has the strong reflection property.*

§11. Regular Proof Systems. Using the results from Buss [12] and Krajíček and Pudlák [41] which we explained in the previous section, we will now exhibit sufficient conditions for the regularity of a propositional proof system. From the definition of a regular system, as given in Sect. 9, it is clear that regular proof systems have the reflection property. Furthermore, a combination of the properties of proof systems introduced in Sect. 4 guarantees the regularity of the system, namely:

Theorem 11.1.

1. *Let P be a proof system such that $EF \leq P$ and P has the reflection property and is closed under substitutions and modus ponens. Then P is regular and corresponds to the theory $S^1_2 + RFN(P)$. In particular, we have $EF + \|RFN(P)\| \equiv P$.*
2. *If P is a proof system such that $EF \leq_p P$ and P has the strong reflection property and is efficiently closed under substitutions and modus ponens, then P is strongly regular and corresponds to the theory $S^1_2 + RFN(P)$. In particular, we have $EF + \|RFN(P)\| \equiv_p P$.*

Before we prove this theorem we apply it to obtain a characterization of extensions of *EF* in terms of an easy combination of closure properties.

Corollary 11.2. *For all proof systems $P \geq_p EF$ the following conditions are equivalent:*

1. *P is p-equivalent to a proof system of the form $EF + \|\varphi\|$ with a true Π^b_1-formula φ.*

2. *P is p-equivalent to a proof system of the form $EF + \|\Phi\|$ with some polynomial-time-decidable set of true Π^b_1-formulas Φ.*
3. *P has the strong reflection property and is efficiently closed under modus ponens and substitutions.*

Proof. Item 1 trivially implies item 2. For the implication 2 ⇒ 3 let $P \equiv_p EF + \|\Phi\|$. Then the closure properties of $EF + \|\Phi\|$ are transferred to P. Similarly, strong reflection for $EF + \|\Phi\|$ (Theorem 10.9) is transferred to P by Proposition 8.4.

The implication 3 ⇒ 1 follows by the second part of Theorem 11.1 above. ⊣

The equivalence of items 1 and 2 in the above corollary expresses some kind of compactness for extensions of EF: systems of the form $EF + \|\Phi\|$ are always equivalent to a system $EF + \|\varphi\|$ with a single arithmetic formula φ. The equivalence to item 3 shows that these systems have a robust logical definition, independent of the particular axiomatization chosen for EF.

The proof of Theorem 11.1 requires a series of lemmas which are also of independent interest. The first lemma is an efficient version of the deduction theorem for EF.

Lemma 11.3 (Deduction theorem for EF). *There exists a polynomial-time procedure that takes as input an EF-proof of a formula ψ from a finite set of tautologies Φ as extra assumptions, and produces an EF-proof of the implication $(\bigwedge_{\varphi\in\Phi} \varphi) \to \psi$.*

Proof. For every Frege rule

$$R_i = \frac{\psi_1 \quad \dots \quad \psi_r}{\psi}$$

in EF we fix a Frege proof π_i of the tautology

$$((q \to \psi_1) \wedge \cdots \wedge (q \to \psi_r)) \to (q \to \psi) \ .$$

For $r = 0$ this also includes the case that R_i is an axiom scheme.

Let Φ be a finite set of tautologies, and let $(\theta_1, \dots, \theta_k)$ be an EF-proof of ψ of size $\leq m$ that uses the formulas Φ as extra axioms. Let $m' = \sum_{\varphi\in\Phi} |\varphi|$. By induction on j we construct proofs of the implications

$$(\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_j \ .$$

We distinguish three cases on how the formula θ_j was derived.

If θ_j was inferred from $\theta_{j_1}, \dots, \theta_{j_r}$ by the rule R_i, then we can get from π_i a Frege proof of size $O(m' + |\theta_j| + \sum_{l=1}^r |\theta_{j_l}|)$ of the tautology

$$(((\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_{j_1}) \wedge \cdots \wedge ((\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_{j_r})) \to ((\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_j) \ .$$

Combining all the earlier proved implications $(\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_{j_l}$, $l = 1, \dots, r$, by conjunctions and using modus ponens, we get the desired implication $(\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_j$ in a proof of size $O(m + m')$.

If θ_j is one of the formulas from Φ, then we get $(\bigwedge_{\varphi\in\Phi} \varphi) \to \theta_j$ in a proof of size $O(m')$.

Let now θ_j be derived by the extension rule, i.e., $\theta_j = (q \leftrightarrow \theta)$ with a new variable q. In this case we also use the extension rule to get $q \leftrightarrow \theta$ and then derive $(\bigwedge_{\varphi\in\Phi} \varphi) \to (q \leftrightarrow \theta)$ in a proof of size $O(m' + |\theta|)$.

It easily checked that all transformations can be executed efficiently, hence the *EF*-proof of $(\bigwedge_{\varphi\in\Phi} \varphi) \to \psi$ can be constructed in polynomial time. ⊣

We just remark that there are different ways to formalize deduction for strong proof systems, and these deduction properties seem to be quite powerful, as they allow the characterization of the existence of optimal and even polynomially bounded proof systems [10].

The next lemma starts the comparison between schematic extensions of *EF* and strong proof systems with sufficient closure properties. The full comparison will, in fact, require a series of lemmas (up to Lemma 11.9).

LEMMA 11.4. *Let P be a proof system such that $EF \leq P$ and P is closed under substitutions and modus ponens. Let Φ be some polynomial-time set of tautologies such that $P \vdash_* \Phi$. Then $EF + \Phi \leq P$.*

PROOF. Let $EF + \Phi \vdash_{\leq m} \varphi$. This means that there are substitution instances $\psi_1, \dots, \psi_k$ of formulas from Φ such that we have an *EF*-proof of φ from $\psi_1, \dots, \psi_k$. Using the deduction theorem for *EF* (Lemma 11.3) we get polynomial-size *EF*-proofs of $(\bigwedge_{i=1}^k \psi_i) \to \varphi$. By induction on k we can show that these *EF*-proofs can be transformed into polynomial-size *EF*-proofs of $(\psi_1 \to (\psi_2 \to \dots (\psi_{k-1} \to (\psi_k \to \varphi))\dots))$. The hypothesis $P \geq EF$ gives us also polynomial-size P-proofs of these formulas. Since $P \vdash_* \Phi$ and P is closed under substitutions, we get polynomial-size P-proofs of ψ_i for $i = 1, \dots, k$. Finally, using the closure of P under modus ponens we obtain a polynomial-size P-proof of φ. ⊣

Making stronger assumptions we can improve the simulation of $EF + \Phi$ by P from the last lemma to a p-simulation, namely:

LEMMA 11.5. *Let P be a proof system such that $EF \leq_p P$ and P is efficiently closed under substitutions and modus ponens. Let Φ be some polynomial-time set of tautologies such that P-proofs of all formulas from Φ can be constructed in polynomial time. Then $EF + \Phi \leq_p P$.*

PROOF. As also the deduction property for *EF* holds in an efficient version (Lemma 11.3), the assumptions guarantee that all steps in the proof of Lemma 11.4 can be efficiently executed. ⊣

Lemmas 11.4 and 11.5 are mostly used in the following form:

COROLLARY 11.6. 1. *Let P be a proof system with the reflection property such that $EF \leq P$ and P is closed under substitutions and modus ponens. Then $EF + \|RFN(P)\| \leq P$.*

2. *If the proof system $P \geq_p EF$ has the strong reflection property and P is efficiently closed under under substitutions and modus ponens, then we get the p-simulation $EF + \|RFN(P)\| \leq_p P$.*

Further comparing the proof systems $EF + \|RFN(P)\|$ and P, we now come to the converse simulation, established in [39]. This reduction is even a p-simulation, and no assumptions on P are necessary.

PROPOSITION 11.7 (Krajíček, Pudlák [39]). *For every proof system P we have $P \leq_p EF + \|RFN(P)\|$.*

PROOF. Let π be a P-proof of φ. Because $RFN(P)$ is available as an axiom we get by substitution a polynomial-size $EF + \|RFN(P)\|$-proof of

$$\|Prf_P(x,y)\|(\bar{p}^x/\bar{\pi}, \bar{p}^y/\bar{\varphi}) \to \|Taut(y)\|(\bar{p}^y/\bar{\varphi}) \ ,$$

where the suffix $(\bar{p}^x/\bar{\pi})$ indicates, that the propositional variables for x are substituted by the bits of π, and similarly for $(\bar{p}^y/\bar{\varphi})$. $\|Prf_P(x,y)\|(\bar{p}^x/\bar{\pi}, \bar{p}^y/\bar{\varphi})$ can be evaluated in EF to $\top$, giving a polynomial-size proof of $\|Taut(y)\|(\bar{p}^y/\bar{\varphi})$ in the proof system $EF + \|RFN(P)\|$. From this we get by Lemma 10.3 a polynomial-size EF-proof of the tautology φ. As these proofs can be constructed in polynomial time, we get the $\leq_p$-reduction. ⊣

The previous proposition can be seen as a propositional version of property 2 of the correspondence to arithmetic theories and documents the importance of the proof systems $EF + \|RFN(P)\|$.

For later use we now prove a lemma which is very similar to Proposition 11.7.

LEMMA 11.8. *Let P be a proof system and let Φ be some polynomial-time set of tautologies. Then $EF + \Phi \vdash_* \|RFN(P)\|^n$ implies $P \leq EF + \Phi$.*

PROOF. Let π be a P-proof of φ. Because $EF + \Phi \vdash_* \|RFN(P)\|^n$ and $EF + \Phi$ is closed under substitutions, we get a polynomial-size $EF + \Phi$-proof of

$$\|Prf_P(x,y)\|(\bar{p}^x/\bar{\pi}, \bar{p}^y/\bar{\varphi}) \to \|Taut(y)\|(\bar{p}^y/\bar{\varphi}) \ .$$

$\|Prf_P(x,y)\|(\bar{p}^x/\bar{\pi}, \bar{p}^y/\bar{\varphi})$ can be evaluated in EF to $\top$, giving a polynomial-size $EF + \Phi$-proof of $\|Taut(y)\|(\bar{p}^y/\bar{\varphi})$. From this we get again by Lemma 10.3 a polynomial-size EF-proof of the tautology φ. Combining these proofs by modus ponens we get an $EF + \Phi$-proof of φ. ⊣

Note that the reduction in the last lemma is only $\leq$, as the $EF + \Phi$-proofs of $\|RFN(P)\|^n$ are not assumed to be constructible in polynomial time. However, if we make this assumption we can draw the stronger conclusion $P \leq_p EF + \Phi$:

LEMMA 11.9. *Let P be a proof system and Φ be some polynomial-time set of tautologies. If $EF + \Phi$-proofs of $\|RFN(P)\|^n$ can be generated in polynomial time, then $P \leq_p EF + \Phi$.*

PROOF. Given a P-proof π of a formula φ we start by generating the $EF + \Phi$-proof of $\|RFN(P)\|^{|\pi|,|\varphi|}$. Careful analysis of the proof of Lemma 11.8 then shows that all transformations can be efficiently performed. Therefore we get the p-simulation. ⊣

Lemma 11.9 enables us to give an easy proof of Theorem 10.8 from Sect. 10.

PROOF OF THEOREM 10.8. Let P be a proof system such that $S^1_2 + \Phi \vdash RFN(P)$. As $RFN(P)$ is a $\forall\Pi^b_1$-formula we conclude with Corollary 10.2 $EF + \Phi \vdash_* \|RFN(P)\|$. As these proofs can be constructed in polynomial time we infer with Lemma 11.9 the simulation $P \leq_p EF + \|\Phi\|$. ⊣

Finally, we can now give the proof of the main theorem of this section.

Proof of Theorem 11.1. To prove part 1 of the theorem let P be a proof system such that $EF \leq P$ and P has reflection and is closed under substitutions and modus ponens. By Corollary 11.6 we have $EF + \|RFN(P)\| \leq P$, and Proposition 11.7 gives $P \leq_p EF + \|RFN(P)\|$. Hence $EF + \|RFN(P)\|$ and P are $\leq$-equivalent.

Next we have to check the axioms of the correspondence for $S^1_2 + RFN(P)$ and P. Suppose φ is a $\forall\Pi^b_1$-formula such that $S^1_2 + RFN(P) \vdash \varphi$. By Corollary 10.2 we get $EF + \|RFN(P)\| \vdash_* \|\varphi\|^n$. As we already know that $EF + \|RFN(P)\|$ is simulated by P, we obtain $P \vdash_* \|\varphi\|^n$. This proves part 1 of the correspondence.

It remains to verify the second part. Clearly $S^1_2 + RFN(P) \vdash RFN(P)$. Assume now $S^1_2 + RFN(P) \vdash RFN(Q)$ for some proof system Q. By Corollary 10.2 this implies $EF + \|RFN(P)\| \vdash_* \|RFN(Q)\|$. Now we can apply Lemma 11.8 and Corollary 11.6 to conclude $Q \leq EF + \|RFN(P)\| \leq P$.

Careful analysis of the proof of the first part reveals, that all steps can actually be performed efficiently under the stronger assumptions of part 2 of the theorem. This yields a proof of the second part of the theorem. ⊣

§12. Hard Tautologies.

In [34] a sequence of tautologies φ_n is called *hard for a proof system P*, if φ_n is constructible in polynomial time, i.e., there exists a polynomial-time-computable function that produces φ_n on input 1^n, and $P \not\vdash_* \varphi_n$. The search for hard sequences for a given proof system constitutes the main objective in propositional proof complexity, and all current lower bounds to the proof length have been obtained by establishing the hardness of some particular sequence of tautologies for the respective proof system. In particular, for resolution Haken [29] showed the hardness of the sequence expressing the pigeon-hole principle, and this was subsequently extended to the Nullstellensatz system [5], polynomial calculus [20, 50], and further systems (cf. [7, 47, 55] for surveys of this development).

While such sequences corresponding to combinatorial principles have proved to be hard for weak systems, they admit polynomial-size proofs in strong systems like Frege systems and their extensions [13]. In fact, it seems difficult to even come up with suggestions for viable candidates for hard sequences for Frege systems [11]. Lemma 11.8 above indicates that the reflection principles $RFN(P)$ for presumably strong systems P are good choices for hard tautologies for EF and its extensions, and in fact for all proof systems. Namely, to obtain a hard sequence for some proof system P, it suffices to choose a proof system Q such that $Q \not\leq EF + RFN(P)$. Then $RFN(Q)$ is hard for P, as otherwise $P \vdash_* RFN(Q)$ and hence by Proposition 11.7 also $EF + RFN(P) \vdash_* RFN(Q)$, which implies $Q \leq EF + RFN(P)$ by Lemma 11.8. Unfortunately, however, the reflection principles do not appear to be easily susceptible to a combinatorial analysis, and thus it seems difficult to prove lower bounds for them.

A novel approach that constructs hard tautologies from pseudo-random generators was independently suggested by Krajíček [35] and Alekhnovich, Ben-Sasson, Razborov, and Wigderson [2]. This approach was shown to be effective for weak systems [2, 37, 51] and connects to cryptographic primitives for strong systems (cf. [36, 51]) for an overview).

In view of Proposition 11.7 it is probably not surprising, that the search for hard tautologies connects to the existence of optimal proof systems, namely the question, whether there exists a strongest proof system that simulates all propositional proof systems. This question was posed by Krajíček and Pudlák [39], and both necessary and sufficient conditions for the existence of optimal proof systems point towards the difficulty of the problem [33, 39, 53]. The next theorem from [34] collects some of the most important information on optimal proof systems.

Theorem 12.1 (Krajíček [34]). *For all proof systems $P \geq EF$ which are closed under substitutions and modus ponens the following conditions are equivalent:*

1. *There exists a sequence of tautologies hard for P.*
2. *The proof system P is not optimal.*
3. *There is a proof system Q such that $P \not\vdash_* \|RFN(Q)\|^n$.*

Proof. To prove the implication $1 \Rightarrow 2$, let φ_n be a sequence of hard tautologies for P. Consider the proof system $Q = EF + \{\varphi_n \mid n \geq 0\}$. Because $P \not\vdash_* \varphi_n$ and $Q \vdash_* \varphi_n$, we have $P \not\geq Q$. Therefore the system P is not optimal.

For the implication $2 \Rightarrow 3$ let P be a non-optimal proof system. Hence there exists a proof system Q such that $Q \not\leq P$. Then $\|RFN(Q)\|^n$ is a sequence of hard tautologies for P. Assume on the contrary $P \vdash_* \|RFN(Q)\|^n$. Since $P \geq EF$ is closed under substitutions and modus ponens we get by Lemma 11.4 and Proposition 11.7 the simulations $P \geq EF + \|\mathrm{RFN(Q)}\| \geq Q$, contradicting $Q \not\leq P$.

The implication $3 \Rightarrow 1$ is trivial, and hence the proof is complete. ⊣

§13. Properties of Proof Systems Revisited. The results from the previous section allow us to compare the properties of propositional proof systems that we introduced in Sect. 4. In particular, we want to know whether these properties are independent from each other. With regard to closure under substitutions and closure under modus ponens we observe the following.

Proposition 13.1. *Assume that the extended Frege proof system is not optimal. Then there exist proof systems which are closed under substitutions but not under modus ponens.*

Proof. We use the assumption of the non-optimality of EF to get a polynomial-time-constructible sequence of tautologies ψ_n with $EF \not\vdash_* \psi_n$ by Theorem 12.1. We may assume that the formulas ψ_n do not contain implications.

Let φ_n be an arbitrary polynomial-time-constructible sequence of tautologies with polynomially long EF-proofs. We define the system Q as

$$Q(\pi) = \begin{cases} \varphi & \text{if } \pi = 0\pi' \text{ and } \pi' \text{ is an } EF\text{-proof of } \varphi \\ \sigma(\varphi_n \to \psi_n) & \text{if } \pi = 10^n 1\sigma \text{ for some substitution } \sigma \\ \top & \text{otherwise.} \end{cases}$$

Because EF is closed under substitutions, this is also true for Q according to the second line of its definition. From $EF \vdash_* \varphi_n$ and $EF \leq_p Q$ we get $Q \vdash_* \varphi_n$. We also have $Q \vdash_* \varphi_n \to \psi_n$ according to the definition of Q. By hypothesis we have $EF \not\vdash_* \psi_n$. Substitution instances of $\varphi_n \to \psi_n$ are different from the formulas

ψ_n, because the former are implications, whereas the latter do not contain the connective $\to$. Therefore also $Q \nvdash_* \psi_n$ and hence Q is not closed under modus ponens. ⊣

Candidates for proof systems that are closed under modus ponens but not under substitutions come from extensions of Frege systems by polynomial-time-computable sets $\Phi \subseteq \mathrm{TAUT}$ as new axioms. Clearly these systems are closed under modus ponens. In [10], however, we exhibit a suitable hypothesis, involving disjoint NP-pairs, which guarantees that these proof systems are not even closed under substitutions by constants for suitable choices of Φ.

Finally, we describe the promised example that the reflection property of a proof system P is sensitive to the choice of the Turing machines which are used to evaluate the P-proofs (cf. the remark at the end of Sect. 8).

Proposition 13.2. *Assume that the extended Frege proof system is not p-optimal. Then there exists a proof system $Q \equiv_p EF$ such that S^1_2 does not prove the reflection principle of Q, i.e., $S^1_2 \nvdash (\forall \pi)(\forall \varphi) \mathit{Prf}_Q(\pi, \varphi) \to \mathit{Taut}(\varphi)$ for some suitable choice of the Turing machine that computes Q and is used for the formula Prf_Q.*

Proof. If EF is not p-optimal, then there exists a proof system R such that $R \not\leq_p EF$. We define the system P as $EF + \|RFN(R)\|$. By Proposition 11.7 we have $R \leq_p P$ and therefore also $P \not\leq_p EF$. We now define the system Q as

$$Q(\pi) = \begin{cases} \varphi & \text{if } \pi = 0\pi' \text{ and } \pi' \text{ is an } EF\text{-proof of } \varphi \\ P(\pi') & \text{if } \pi = 1\pi' \text{ and } P(\pi') \in \{\top, \bot\} \\ \top & \text{otherwise.} \end{cases}$$

Then EF and Q are $\leq_p$-equivalent because EF $\leq_p$-reduces to Q via $\pi \mapsto 0\pi$ and the opposite reduction $Q \leq_p EF$ is given by

$$\pi \mapsto \begin{cases} \pi' & \text{if } \pi = 0\pi' \\ \pi_0 & \text{if } \pi = 1\pi' \end{cases}$$

where π_0 is a fixed EF-proof of $\top$. We have to show that S^1_2 does not prove the formula $RFN(Q)$ where for the predicate Prf_Q we use the canonical Turing machine M according to the above definition of Q, i.e., on input $0\pi'$ the machine M checks whether π' is a correct EF-proof and on input $1\pi'$ the machine M evaluates $P(\pi')$. Assume on the contrary that $S^1_2 \vdash_* RFN(Q)$. Because of line 2 of the definition of Q this means that S^1_2 can prove that there is no P-proof of $\bot$, i.e., S^1_2 proves the consistency statement of P. The system P is closed under substitutions by constants and modus ponens. Therefore $\mathit{Con}(P)$ and $RFN(P)$ are equivalent in S^1_2 by Proposition 8.1. Together with $S^1_2 \vdash \mathit{Con}(P)$ this yields $S^1_2 \vdash RFN(P)$, and hence by Theorem 10.7 we obtain $P \leq_p EF$, contradicting the choice of P. Thus S^1_2 proves $RFN(EF)$, but not $RFN(Q)$. ⊣

Acknowledgements. I am indebted to Jan Krajíček for many helpful and stimulating discussions on the topic of this paper. I also thank Sebastian Müller for detailed suggestions on how to improve the presentation of the paper.

REFERENCES

[1] Miklós Ajtai, *The complexity of the pigeonhole-principle*, ***Combinatorica***, vol. 14 (1994), no. 4, pp. 417–433.

[2] Michael Alekhnovich, Eli Ben-Sasson, Alexander A. Razborov, and Avi Wigderson, *Pseudorandom generators in propositional proof complexity*, ***SIAM Journal on Computing***, vol. 34 (2004), no. 1, pp. 67–88.

[3] Jeremy Avigad, *Forcing in proof theory*, ***The Bulletin of Symbolic Logic***, vol. 10 (2004), no. 3, pp. 305–333.

[4] Theodore Baker, John Gill, and Robert Solovay, *Relativizations of the P=?NP question*, ***SIAM Journal on Computing***, vol. 4 (1975), pp. 431–442.

[5] Paul W. Beame, Russel Impagliazzo, Jan Krajíček, Toniann Pitassi, and Pavel Pudlák, *Lower bounds on Hilbert's Nullstellensatz and propositional proofs*, ***Proc. London Mathematical Society***, vol. 73 (1996), no. 3, pp. 1–26.

[6] Paul W. Beame, Russel Impagliazzo, Jan Krajíček, Toniann Pitassi, Pavel Pudlák, and Alan Woods, *Exponential lower bounds for the pigeonhole principle*, ***Proc. 24th ACM symposium on theory of computing***, 1992, pp. 200–220.

[7] Paul W. Beame and Toniann Pitassi, *Propositional proof complexity: Past, present, and future*, ***Current trends in theoretical computer science: Entering the 21st century*** (G. Paun, G. Rozenberg, and A. Salomaa, editors), World Scientific Publishing, 2001, pp. 42–70.

[8] Paul W. Beame, Toniann Pitassi, and Russel Impagliazzo, *Exponential lower bounds for the pigeonhole principle*, ***Computational Complexity***, vol. 3 (1993), no. 2, pp. 97–140.

[9] Olaf Beyersdorff, *Classes of representable disjoint NP-pairs*, ***Theoretical Computer Science***, vol. 377 (2007), no. 1–3, pp. 93–109.

[10] ———, *The deduction theorem for strong propositional proof systems*, ***Proc. 27th conference on foundations of software technology and theoretical computer science***, Lecture Notes in Computer Science, vol. 4855, Springer-Verlag, Berlin Heidelberg, 2007, pp. 241–252.

[11] Maria Luisa Bonet, Samuel R. Buss, and Toniann Pitassi, *Are there hard examples for Frege systems?*, ***Feasible mathematics II*** (P. Clote and J. Remmel, editors), Birkhäuser, 1995, pp. 30–56.

[12] Samuel R. Buss, ***Bounded arithmetic***, Bibliopolis, Napoli, 1986.

[13] ———, *Polynomial size proofs of the propositional pigeonhole principle*, ***The Journal of Symbolic Logic***, vol. 52 (1987), pp. 916–927.

[14] ———, *Axiomatizations and conservation results for fragments of bounded arithmetic*, ***Logic and Computation, Contemporary Mathematics***, vol. 106 (1990), pp. 57–84.

[15] ———, *The witness function method and provably recursive functions of Peano arithmetic*, ***Proc. 9th international congress on logic, methodology and philosophy of science***, 1994, pp. 29–68.

[16] ———, *Relating the bounded arithmetic and polynomial-time hierarchies*, ***Annals of Pure and Applied Logic***, vol. 75 (1995), pp. 67–77.

[17] ———, *First order proof theory of arithmetic*, ***Handbook of proof theory*** (Samuel R. Buss, editor), Elsevier, Amsterdam, 1998, pp. 79–147.

[18] ———, *An introduction to proof theory*, ***Handbook of proof theory*** (Samuel R. Buss, editor), Elsevier, Amsterdam, 1998, pp. 1–78.

[19] Mario Chiari and Jan Krajíček, *Witnessing functions in bounded arithmetic and search problems*, ***The Journal of Symbolic Logic***, vol. 63 (1998), no. 3, pp. 1095–1115.

[20] M. Clegg, J. Edmonds, and R. Impagliazzo, *Using the Groebner basis algorithm to find proofs of unsatisfiability*, ***Proc. 28th ACM symposium on theory of computing***, 1996, pp. 174–183.

[21] Stephen A. Cook, *The complexity of theorem proving procedures*, ***Proc. 3rd annual ACM symposium on theory of computing***, 1971, pp. 151–158.

[22] ———, *Feasibly constructive proofs and the propositional calculus*, ***Proc. 7th annual ACM symposium on theory of computing***, 1975, pp. 83–97.

[23] ———, *Theories for complexity classes and their propositional translations*, ***Complexity of computations and proofs*** (Jan Krajíček, editor), Quaderni di Matematica, 2005, pp. 175–227.

[24] Stephen A. Cook and Phuong Nguyen, ***Logical foundations of proof complexity***, Cambridge University Press, 2009, To appear, Preprint available from http://www.cs.toronto.edu/˜sacook.

[25] Stephen A. Cook and Robert A. Reckhow, *The relative efficiency of propositional proof systems*, ***The Journal of Symbolic Logic***, vol. 44 (1979), no. 1, pp. 36–50.

[26] Martin Dowd, *Model-theoretic aspects of* $P{\neq}NP$, Unpublished manuscript, 1985.

[27] Gerhard Gentzen, *Untersuchungen über das logische Schließen*, ***Mathematische Zeitschrift***, vol. 39 (1935), pp. 68–131.

[28] Petr Hájek and Pavel Pudlák, ***Metamathematics of first-order arithmetic***, Perspectives in Mathematical Logic, Springer-Verlag, Berlin Heidelberg, 1993.

[29] Amin Haken, *The intractability of resolution*, ***Theoretical Computer Science***, vol. 39 (1985), pp. 297–308.

[30] Wilfrid Hodges, ***Building models by games***, Dover Publications, 2006.

[31] Emil Jeřábek, *Dual weak pigeonhole principle, Boolean complexity, and derandomization*, ***Annals of Pure and Applied Logic***, vol. 129 (2004), pp. 1–37.

[32] Richard M. Karp, *Reducibility among combinatorial problems*, ***Complexity of computer computations*** (R. E. Miller and J. W. Thatcher, editors), Plenum Press, 1972, pp. 85–103.

[33] Johannes Köbler, Jochen Messner, and Jacobo Torán, *Optimal proof systems imply complete sets for promise classes*, ***Information and Computation***, vol. 184 (2003), no. 1, pp. 71–92.

[34] Jan Krajíček, ***Bounded arithmetic, propositional logic, and complexity theory***, Encyclopedia of Mathematics and Its Applications, vol. 60, Cambridge University Press, Cambridge, 1995.

[35] ———, *On the weak pigeonhole principle*, ***Fundamenta Mathematicae***, vol. 170 (2001), pp. 123–140.

[36] ———, *Tautologies from pseudo-random generators*, ***The Bulletin of Symbolic Logic***, vol. 7 (2001), no. 2, pp. 197–212.

[37] ———, *Dual weak pigeonhole principle, pseudo-surjective functions, and provability of circuit lower bounds*, ***The Journal of Symbolic Logic***, vol. 69 (2004), no. 1, pp. 265–286.

[38] ———, *Implicit proofs*, ***The Journal of Symbolic Logic***, vol. 69 (2004), no. 2, pp. 387–397.

[39] Jan Krajíček and Pavel Pudlák, *Propositional proof systems, the consistency of first order theories and the complexity of computations*, ***The Journal of Symbolic Logic***, vol. 54 (1989), no. 3, pp. 1063–1079.

[40] ———, *Propositional provability and models of weak arithmetic*, ***Proc. 3rd workshop on computer science logic***, Lecture Notes in Computer Science, vol. 440, Springer-Verlag, Berlin Heidelberg, 1990, pp. 193–210.

[41] ———, *Quantified propositional calculi and fragments of bounded arithmetic*, ***Zeitschrift für mathematische Logik und Grundlagen der Mathematik***, vol. 36 (1990), pp. 29–46.

[42] ———, *Some consequences of cryptographical conjectures for* S^1_2 *and* EF, ***Information and Computation***, vol. 140 (1998), no. 1, pp. 82–94.

[43] Jan Krajíček, Pavel Pudlák, and Gaisi Takeuti, *Bounded arithmetic and the polynomial hierarchy*, ***Annals of Pure and Applied Logic***, vol. 52 (1991), pp. 143–153.

[44] Jan Krajíček, Pavel Pudlák, and Alan Woods, *Exponential lower bounds to the size of bounded depth Frege proofs of the pigeonhole principle*, ***Random Structures and Algorithms***, vol. 7 (1995), no. 1, pp. 15–39.

[45] Jan Krajíček, Alan Skelley, and Neil Thapen, *NP search problems in low fragments of bounded arithmetic*, ***The Journal of Symbolic Logic***, vol. 72 (2007), no. 2, pp. 649–672.

[46] Jeff Paris and Alec J. Wilkie, *Counting problems in bounded arithmetic*, ***Methods in mathematical logic, proc. 6th latin american symposium***, 1985, pp. 317–340.

[47] Pavel Pudlák, *The lengths of proofs*, ***Handbook of proof theory*** (Samuel R. Buss, editor), Elsevier, Amsterdam, 1998, pp. 547–637.

[48] Alexander A. Razborov, *Unprovability of lower bounds on the circuit size in certain fragments of bounded arithmetic*, ***Izv. Ross. Akad. Nauk Ser. Mat.***, vol. 59 (1995), no. 1, pp. 201–224.

[49] ———, *Lower bounds for propositional proofs and independence results in bounded arithmetic*, ***Automata, languages, and programming: 23rd international colloquium, icalp '96***, Lecture Notes in Computer Science, vol. 1099, Springer, 1996, pp. 48–62.

[50] ———, *Lower bounds for the polynomial calculus*, ***Computational Complexity***, vol. 7 (1998), no. 4, pp. 291–324.

[51] ———, *Pseudorandom generators hard for k-DNF resolution and polynomial calculus resolution*, Preprint, 2003.

[52] Alexander A. Razborov and Steven Rudich, *Natural proofs*, ***Proc. 26th ACM symposium on theory of computing***, 1994, pp. 204–213.

[53] Zenon Sadowski, *On an optimal propositional proof system and the structure of easy subsets of TAUT*, ***Theoretical Computer Science***, vol. 288 (2002), no. 1, pp. 181–193.

[54] Gaisi Takeuti and Masahiro Yasumoto, *Forcing on bounded arithmetic II*, ***The Journal of Symbolic Logic***, vol. 63 (1998), no. 3, pp. 860–868.

[55] Alasdair Urquhart, *The complexity of propositional proofs*, ***The Bulletin of Symbolic Logic***, vol. 1 (1995), pp. 425–467.

[56] Celia Wrathall, *Rudimentary predicates and relative computation*, ***SIAM Journal on Computing***, vol. 7 (1978), no. 2, pp. 149–209.

[57] Domenico Zambella, *Notes on polynomially bounded arithmetic*, ***The Journal of Symbolic Logic***, vol. 61 (1996), no. 3, pp. 942–966.

INSTITUT FÜR INFORMATIK
HUMBOLDT-UNIVERSITÄT ZU BERLIN
UNTER DEN LINDEN 6, 10099 BERLIN, GERMANY
E-mail: beyersdo@informatik.hu-berlin.de