Записки научных
семинаров ПОМИ
Том 304, 2003 г.

Ch. Pollett

$S_{k,\exp}$ DOES NOT PROVE NP =CO−NP UNIFORMLY

Abstract. A notion of a uniform sequent calculus proof is given. It is then shown that a strengthening, $S_{k,\exp}$, of the well-studied bounded arithmetic system S_k of Buss does not prove NP = *co*-NP with a uniform proof. A slightly stronger result that $S_{k,\exp}$ cannot prove $\widehat{\Sigma}^{\mathrm{b}}_{1,k'} = \widehat{\Pi}^{\mathrm{b}}_{1,k'}$ uniformly for $2 \le k' \le k$ is also established. A variation on the technique used is then applied to show that $S_{k,\exp}$ is unable to prove Davis–Putnam–Robinson–Matiyasevich theorem. This result is also without any uniformity conditions. Generalization of both these results to higher levels of the Grzegorczyck Hierarchy are then presented.

1. Introduction

The formalizability of the Davis–Putnam–Robinson–Matiyasevich (DPRM) theorem [12] and the provability of NP = *co*-NP in weak systems of arithmetic are closely connected. Recall the DPRM theorem says that the Σ_1-sets are equivalent to the sets that can be defined by formulas of the form:

$$A = \{x | (\exists \vec{y}) P(x, \vec{y}) = Q(x, \vec{y})\},$$

where P and Q are polynomials with coefficients in $\mathbb{N}$. It is known that the theory $I\Delta_0 + \exp$, which has bounded induction in the language of arithmetic together with an axiom exp for exponentiation, proves the DPRM theorem [5]. By an old folklore result (see Hájek and Pudlák [8]), it is also known if Buss' theories S_k can prove DPRM then $\Sigma^{\mathrm{b}}_{1,k} = \Pi^{\mathrm{b}}_{1,k}$. In particular, when $k = 2$, this implies NP = *co*-NP.

These theories S_k of Buss are interesting theories in which to study the DPRM theorem. All of these theories contain S_1 which is a conservative extension of $I\Delta_0$. They are defined from S_1 by expanding the language to include functions symbols $\#_2, \dots, \#_k$ where the intended meaning of the symbols is $x\#_2 y = 2^{|x||y|}$, and for $k > 2$, $x\#_k y := 2^{|x|\#_{k-1}|y|}$. Given these growth rates are ever increasing but still subexponential, the theories S_k provide a setting in which to study the role of exponentiation in the provability of the DPRM theorem. Moreover, many interesting connections between these theories and complexities classes have been developed (see Krajíček [11], for example) and many complexity arguments have been shown to be formalizable in them [19]. In particular, S_k can prove the consistency of many of the propositional proof systems, such as extended Frege systems, for which lower bounds are unknown. Such systems could potentially be super, and if so, imply NP $=$ *co*-NP. Despite these apparent strengths, in this paper a deductive system $S_{k,\exp}$ that contains S_k is exhibited that cannot prove the DPRM theorem. A notion of uniform sequent calculus proof is also introduced and it is shown that $S_{k,\exp}$ cannot prove NP $=$ *co*-NP with a uniform proof.

The strategy of the proof is as follows: first, the systems $S_{k,\exp}$ in the language with 2^x are defined and developed. These deductive systems have a restricted form of induction inference where all the formulas in the upper and lower sequent must come from the class $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$, the class of bounded formula in our language all of whose quantifier bounding terms do not involve 2^x. These systems are also restricted in that only cuts on $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formulas are allowed in an $S_{k,\exp}$ proof. It is shown by a witnessing argument that the predicates $S_{k,\exp}$ can prove are equivalent to both an exponentially large existential $(\exists x \leq t)$ followed by a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formula (an $E_{1,\exp}(\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp}))$-formula) as well as an exponentially large universal $(\forall x \leq t)$ followed by a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formula (an $U_{1,\exp}(\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp}))$-formula) are precisely the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-predicates. A uniform $S_{k,\exp}$ proof is one which would remain an $S_{k,\exp}$ proof if one replaced everywhere in the proof one of the free variables from the conclusion by a term in the language. If every NP predicate is equivalent to a *co*-NP predicate by such a proof then $S_{k,\exp}$ is said to prove NP$=$*co*-NP uniformly. By a formalized padding argument, it is shown that if $S_{k,\exp}$ could prove NP $=$ *co*-NP by such proofs then, in fact, $S_{k,\exp}$ proves

$$E_{1,\exp}(\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})) = U_{1,\exp}(\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})).$$

This would imply roughly that the polynomial hierarchy (or for $k > 2$, some kind of quasi-polynomial hierarchy) is the same as the class of elementary predicates. However, one can show that these two classes can be diagonalized apart. The DPRM result is proven using a similar argument together with the folklore result mentioned above modified to these theories.

Although the main results of this paper are reasonably strong, the proof techniques that are used are reasonably standard in bounded arithmetic and computational complexity. There are four main aspects of the argument, that we feel are novel and interesting. The first is the theories $S_{k,\exp}$ themselves. They provide nice intermediate theories between the theories S_k which do not have exponentiation and the theory $I\Delta_0 + \exp$. For $k \geq 2$, the theories $S_{k,\exp}$ are also stronger than the theory $S_2(\alpha)$ which Razborov [19] argued can prove Håstad's Switching Lemma [7]. The second aspect of our argument which we feel is novel is the very weak closure properties of the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions that are established and used for our witnessing argument. This technique might be of independent interest and useful in future witnessing arguments. A third aspect of this argument that is interesting is that it illustrates the limitations of padding arguments in weak formal systems. A final aspect of the argument which is interesting is that it generalizes to higher levels of the Grzegorczyk Hierarchy, as is explained next.

The idea of this generalization to higher levels of the Grzegorczyk Hierarchy is that although Gaifman and Dimitracopoulos [5] showed that $I\Delta_0$ in the language with 2^x can prove the DPRM theorem, this is not the complete story. Basically, this result says that this theory can prove exponentially bounded quantifiers in a Σ_1-set can be eliminated in a Diophantine way. By Parikh's theorem, however, this theory cannot prove the existence of any functions of superexponential growth. Thus, one could ask how much induction is needed to show that bounded quantifiers of superexponential size can be eliminated? Obviously, if one has usual induction on bounded formulas in a language with the given superexponential function symbols, then one can probably apply the arguments of Gaifman and Dimitracopoulos. What is shown in this paper is that for finite $m \geq 2$, if symbols for the first $m + 1$ branches of the Ackermann function as well as their inverses are added to the language, then if induction inferences are restricted to only allow formulas not involving the $m+1$st branch symbol and any cut formula in the proof does not involve the $m + 1$st branch symbol, then this system does not suffice to prove DPRM in this language. Further, this system also cannot prove NP = *co*-NP uniformly. Both these arguments use the same ideas as for the $S_{k,\exp}$ case. The unprovability of NP = *co*-NP argument can be pushed slightly further without any work: given their definition, the union of these systems can reason inductively about the primitive recursive functions. One can say this system proves NP = *co*-NP uniformly if any of its fragments can. Then one can show this system cannot prove NP = *co*-NP uniformly. As the techniques used in the arguments of this paper make use of the fact the underlying language only has finitely many symbols, it is unclear how much farther these results can be extended. Still, this line of research seems promising.

This paper is organized as follows: The next section contains the notations and definitions used in this paper. This is followed by a section showing that $\nabla_{1,\exp}$-predicates of $S_{k,\exp}$ are precisely the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-predicates. The NP = *co*-NP result is then presented and the section after that shows the DPRM theorem result. Then the generalizations to higher levels of the Grzegorczyk Hierarchy are given. Finally, a conclusion is presented.

2. Preliminaries

Familiarity with the basic results of bounded arithmetic at the level of say Buss [3] or Krajíček [11] is assumed.

The language L_2 will be used to denote the language with nonlogical symbols: 0, S, $+$, $\cdot$, $\leq$, $\dot{-}$, $\lfloor \frac{1}{2}x \rfloor$, $|x|$, $\mathrm{MSP}(x,i)$, $x\&_2 y$, and $\#$. The symbols 0, $S(x) = x+1$, $+$, $\cdot$, and $\leq$ have the usual meaning. The intended meaning of $x \dot{-} y$ is x minus y if this is greater than zero and zero otherwise, $\lfloor \frac{1}{2}x \rfloor$ is x divided by 2 rounded down, and $|x|$ is $\lceil \log_2(x+1) \rceil$, that is, the length of x in binary notation. $\mathrm{MSP}(x,i)$ stands for 'most significant part' and is intended to mean $\lfloor x/2^i \rfloor$. The function symbol $x\&_2 y$ is intended to return the bit-wise logical AND of x and y. This function will be used to avoid an innermost universal quantifier in the formalization of NP used in this paper. $x\#y$ reads 'x smash y' and is intended to mean $2^{|x||y|}$. The operation $\#$ is also written $\#_2$. In general, $x\#_k y = 2^{|x|\#_{k-1}|y|}$ and the language L_k for $k > 2$ has the symbols of L_{k-1} together with $\#_k$. Finally, $L_{k,\exp}$ is the language L_k together with the nonlogical symbol 2^x intended to represent base 2 exponentiation.

For this paper, it is useful to be able to have a pairing function, as well as to have functions that can project blocks of bits from a number so that a limited amount of sequence coding can be done. These can be defined by L_2-terms as follows: For projection of bits, define the functions $2^{|y|} := 1\#y$, $2^{\min(|y|,x)} := \mathrm{MSP}(1\#y, |y| \dot{-} x)$, $\mathrm{LSP}(x,i) := x \dot{-} \mathrm{MSP}(x,i) \cdot 2^{\min(|x|,i)}$, $\widehat{\beta}_{|t|}(x,w) := \mathrm{MSP}(\mathrm{LSP}(w, Sx|t|), x|t|)$, and $\mathrm{BIT}(i,x) := \widehat{\beta}_1(i,x)$. Here $\widehat{\beta}$ is supposed to project the xth block of $|t|$ bits from w and BIT is supposed to return the ith bit of x. Given these functions to define pairing operations, let $\max(x,y) :=$

$(1 \dot{-} (x \dot{-} y))y + (1 \dot{-} (y \dot{-} x))x$ and set $B = 2^{|\max(x,y)|+1}$. Thus, B will be longer than either x or y. Define an ordered pair as $\langle x, y\rangle := (2^{|\max(x,y)|} + y) \cdot B + (2^{|\max(x,y)|} + x)$. To project out the coordinates from such an ordered pair, use $(w)_1 := \widehat{\beta}_{\lfloor\frac{1}{2}x\rfloor|w|\dot{-}1}(0, \widehat{\beta}_{\lfloor\frac{1}{2}x\rfloor|w|}(0, w))$ and $(w)_2 := \widehat{\beta}_{\lfloor\frac{1}{2}x\rfloor|w|\dot{-}1}(0, \widehat{\beta}_{\lfloor\frac{1}{2}x\rfloor|w|}(1, w))$ which return the left and right coordinates of the pair w. To check if w is a pair the formula

$$ispair(w) := \mathrm{Bit}(w, \lfloor\frac{1}{2}|w|\rfloor \dot{-} 1) = 1 \wedge 2 \cdot |\max((w)_1, (w)_2)| + 2 = |w|$$

is used. The usual properties of this formula as well as the terms listed above are provable in the theories we will consider in this paper [16].

A quantifier of the form $(\forall x \leq t)$ or $(\exists x \leq t)$ where t is a term not containing x is called a *bounded quantifier*. A formula is *bounded* or Δ_0 if all its quantifiers are. A quantifier of the form $(\forall x \leq |t|)$ or of the form $(\exists x \leq |t|)$ is called *sharply bounded* and a formula is *sharply bounded* if all its quantifiers are.

The main results of this paper make use of the following variations on the standard bounded arithmetic hierarchies.

Given languages $L' \subseteq L$ from those mentioned above and a class of L-formulas C, the hierarchy of formulas $E_{i,L'}(C)$ and $U_{i,L'}(C)$ are defined as follows: $E_{1,L'}(C)$ are those formulas of the form $(\exists x \leq t)\phi$ and $U_{1,L'}(C)$ are those formulas of the form $(\forall x \leq t)\phi$ where ϕ is in C and t is an L'-term. $E_{i,L'}(C)$ are those formulas of the form $(\exists x \leq t)\phi$ where $\phi \in U_{i-1,L'}(C)$-formula and t is an L'-term. $U_{i,L'}(C)$ are those formulas of the form $(\forall x \leq t)\phi$ where $\phi \in E_{i-1,L}(C)$ and t is an L'-term. The notation $E_{\infty,L'}(C)$ will be used for $\cup_i(E_{i,L'}(C) \cup U_{i,L'}(C))$. The class of quantifier-free formulas is denoted by *open* (or $open_k$ or $open_{\exp}$ to emphasize the language is L_k or $L_{k,\exp}$).

To indicate that a vector $\vec{b}$ of free variables of a formula ϕ may occur in terms involving any of the symbols of L, sometimes the notation $\phi(\vec{a}; \vec{b})$ will be used. Here the variables $\vec{b}$ may or may not be restricted to L'-terms and $\vec{a}$ are variables which are restricted to L'-terms. For most of this paper L' will be L_k and L will be $L_{k,\exp}$, so the variables $\vec{b}$ in this case would be allowed to occur in terms involving 2^x. If Ψ is a class of L-formulas, then $\Psi[\vec{b}]$ are all those formulas in Ψ of the form $\phi(\vec{a}; \vec{b})$.

The notations E_i and U_i are used when $L' = L$ is understood and C is the class of open formulas, and notations such as $E_{i,k}$, $U_{i,\exp}$, $E_{\infty,k}$, $U_{i,k}[\vec{b}]$ are used for classes such as $E_{i,L_k}(open_k)$, $U_{i,\exp}(open_{\exp})$, $E_{\infty,L_k}(open_k)$, and $U_{i,k}(open_{\exp})[\vec{b}]$. For $i > 0$, a $\widehat{\Sigma}^{\mathrm{b}}_i(C)$-formula (resp. $\widehat{\Pi}^{\mathrm{b}}_i(C)$-formula) is defined to be a $E_{i+1}(C)$-formula (resp. U_{i+1}-formula) whose innermost quantifier is sharply bounded. Again, to emphasize the languages involved notations such as $\widehat{\Sigma}^{\mathrm{b}}_{i,k}$ and $\widehat{\Pi}^{\mathrm{b}}_{i,\mathrm{exp}}$, $\widehat{\Sigma}^{\mathrm{b}}_{1,k}[\vec{b}]$ will be used. Kent and Hodgson [10] (see also Pollett [16])[1] have shown the sets defined by $\widehat{\Sigma}^{\mathrm{b}}_{i,2}$-(resp. $\widehat{\Pi}^{\mathrm{b}}_{i,2}$-)formulas are precisely the Σ^p_i-(resp. Π^p_i-)predicates. Thus, the $\widehat{\Sigma}^{\mathrm{b}}_{1,2}$-formulas correspond to the NP predicates. The proof of this fact was in a language without $x \&_2 y$. Jones and Matiyasevich [9] have shown that any set in NP can be represented as a predicate:

$$\exists y_1 \leq 2^{p(|x|)} \cdots \exists y_n \leq 2^{p(|x|)}[F(x, \vec{y}) = G(x, \vec{y})],$$

where p is a polynomial and F and G are built up from x, $\vec{y}$, using $+$, $\cdot$, and $\&_2$. Since pairing can be defined as an L_k-term and $\&_2$ is one of the operations in L_k, this shows the $E_{i,2}$-predicates are also NP.

Remark 1. The usual formalization of NP in bounded arithmetic theories is in terms of the $\widehat{\Sigma}^{\mathrm{b}}_{1,2}$-predicates. Thus, it is interesting to ask if the theories considered in this paper can prove $\widehat{\Sigma}^{\mathrm{b}}_{1,2} = E_{1,2}$. The author feels that is it is likely one could formalize in $S_{k,\mathrm{exp}}$ that any $\widehat{\Sigma}^{\mathrm{b}}_{1,2}$ machine could be simulated by the appropriate register machine used in Jones and Matiyasevich [9] and then in turn show that such machines can be simulated by $E_{1,2}$-predicates; however, such a proof might be quite long and distract from the main point of this article. Even if this could not be proven in the theories as they will be introduced below, one could always add axioms for each $\widehat{\Sigma}^{\mathrm{b}}_{1,2}$-formula saying it was equivalent to whatever $E_{1,2}$-formula it was equivalent to. These axioms would be $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$-formulas and would not change the proofs of the main results of this paper.

The theories considered in this paper are formulated in the sequent calculus LKB of Buss [3]. $\mathrm{BASIC}_{k,\mathrm{exp}}$ will be used to denote a theory axiomatized by all substitution instances of a finite set of quantifier free axioms for the nonlogical symbols of $L_{k,\mathrm{exp}}$. These axioms are listed in Buss [3] except for MSP, $\dot{-}$, $\&_2$, and 2^x. The axioms for MSP and $\dot{-}$ are listed in Takeuti [21]. The axioms for $x \&_2 y$ are: (1) $x \leq S0, y \leq S0 \to x \&_2 y = x \cdot y$. (2) $a \leq S0, b \leq S0 \to (2c + a) \&_2 (2d + b) = 2(c \&_2 d) + a \&_2 b$. Lastly, the axioms for 2^x are: (1) $2^0 = S0$ and (2) $2^{Sa} = 2^a \cdot SS0$.

[1] Editor's note. Similar results were obtained also by S. Yukna in *Matematicheskaya logika i ee primeneniya*, no. 2:92–107 (1982), no. 3:117–125 (1983), Institut Matematiki i Kibernetiki Akademii Nauk Litovskoĭ SSR, Vil'nyus.

Definition 1. *Let Φ a class of L-formulas not necessarily closed under term substitution. A $\Phi - \overline{\mathrm{IND}}$ inference is an inference*

$$\frac{A(c), \Gamma \to A(Sc), \Delta}{A(0), \Gamma \to A(t), \Delta},$$

where c is an eigenvariable and must not appear in the lower sequent, t is an L-term. A and all the formulas in Γ and Δ are in Φ in both the upper and lower sequent. To emphasize, even $A(t)$ must be in Φ for this to be considered a Φ-$\overline{\mathrm{IND}}$ inference.

Definition 2. *The system $S_{k,\exp}$ is defined as*

$$\mathrm{BASIC}_{k,\exp} + \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp}) - \overline{\mathrm{IND}},$$

where it is further required that all cuts that appear in any $S_{k,\exp}$ derivation must be on $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formulas.

The theory S_k of Buss [3] mentioned in the introduction and abstract uses the language L_k without $x \&_2 y$ and has the axioms of $\mathrm{BASIC}_{k,\exp}$ restricted to this language as well as usual induction for bounded formulas in this language. By cut-elimination for this S_k, one can see S_k is contained in $S_{k,\exp}$. As S_k cannot define 2^x by Parikh's Theorem, this containment is strict.

Remark 2. The main point of the restriction on cut is to prevent derivations of $A(t)$ from $A(a)$ where t involves 2^x. Although $S_{k,\exp}$ can derive statements like $\to (\exists x)x = t$ and $a = t, A(a) \to A(t)$, and from this $(\exists x)x = t, A(a) \to A(t)$; it cannot from this conclude via a cut that $A(a) \to A(t)$. Since $S_{k,\exp}$ proves both S_k and $(\exists y)y = 2^x$, without this restriction on cut $S_{k,\exp}$ would be as strong as $I\Delta_0 + \exp$.

Remark 3. Another important point that will make the system $S_{k,\exp}$ weaker than $I\Delta_0 + \exp$, is that the $L_{k,\exp}$-terms themselves do not define every $\widehat{\Sigma}^{\mathrm{b}}_{1,\exp}$-predicate. Since 2^x is in the language, this could happen if one added symbols for weak variants of the μ-operator to the language or closed the symbols of the language under small amounts of recursion. It is crucial to the argument below that given a $L_{k,\exp}$-term $t(x, w)$, the value of the equation $t(x, w) = 0$ can be computed in polynomial time. This is shown in Lemma 3. The fact that one can do this allows one to diagonalize the classes $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$ and $\widehat{\Sigma}^{\mathrm{b}}_{1,\exp}$ apart.

3. Predicates

For this section let $\Psi := E_{1,\exp}(\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp}))$. The predicate A is said to be $\bigtriangledown_{1,\exp}$ *in a theory* T if $T \vdash A^{\Sigma} \equiv A \equiv A^{\Pi}$ where A^{Σ} is a Ψ-formula and A^{Π} tautologically equivalent to the negation of a Ψ-formula. To indicate which free variables may occur in terms involving 2^x in both A^{Π} and A^{Σ} the notation $\bigtriangledown_{1,\exp}[\vec{b}]$ will be used. The goal of this section is to show the $\bigtriangledown_{1,\exp}[\vec{b}]$-predicates of $S_{k,\exp}$ are precisely $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ and, in general, the $\bigtriangledown_{1,\exp}$-predicates of $S_{k,\exp}$ are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$. The reader who believes these results and who finds witnessing arguments tedious is invited to proceed to the next section.

A $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-*function* is a function whose graph is in $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ and which is bounded by an $L_{k,\exp}$-term in $\vec{b}$ (so the output can be exponentially large as function of $\vec{b}$). Some closure properties of these functions needed for the witnessing argument are now investigated.

Let $(\mu x \leq t)A$ denote the function which returns the least x less than t such that the predicate A holds if it exists and $t+1$ otherwise. Let $\mathrm{cond}(A, x, y)$ denote the function which if A hold returns x and otherwise returns y.

Lemma 1. *Let h be an $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function bounded by an L_k-term, let $s(\vec{b})$ be an $L_{k,\exp}$-term, let f, g be a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions, and let A be a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-predicate. Then the following are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions:* (1) s. (2) $f(\vec{a}, h; \vec{b})$. (3) $\langle f, g\rangle$. (4) $\mu x \leq tA$. (5) $\mathrm{cond}(A, f, g)$.

Proof. Let t_h, t_f, t_g be the $L_{k,\exp}$-terms bounding the output of f, g, and f', respectively. Let A_f, A_g be the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-formulas for their graphs. (1) To $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ define s, consider the equation $s(\vec{b}) = y$ and the the bounding term $t_s = s$. For (2), since t_h is by hypothesis in L_k, $\exists z \leq t_h(A_h(\vec{a}, z; \vec{b}) \wedge A_f(\vec{a}, z; y, \vec{b}))$ will be in $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ and defines the graph of $f(\vec{a}, h; \vec{b}) = y$. (3) To define $y = \langle f, g\rangle$, consider the formula $A_f((y)_1) \wedge A_g((y)_2)$. This formula is equivalent to a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-formula and defines the desired graph. The value y that satisfies this graph can be bounded by $\langle t_f, t_g\rangle$. (4) To define the graph of $\mu x \leq tA$ let $B(v)$ be

$$A(v) \wedge \forall y \leq t(y < v \supset \neg A(v)).$$

The graph of $\mu x \leq tA$ can then be defined as $C(v)$

$$(\exists v \leq t)(B(v) \wedge x = v) \vee (\neg \exists v \leq t)(B(v) \wedge x = t+1).$$

The formula $C(v)$ is easily seen to be equivalent to a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-formula. The term t can be used to bound the graph of the output. (5) The graph of $\mathrm{cond}(A, f, g)$ can be defined as $(A \wedge A_f) \vee (\neg A \wedge A_g)$. This is equivalent to a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-formula and the output of the function can be bounded by $t_f + t_g$ so can be bounded by an $L_{k,\mathrm{exp}}$-term. $\square$

Note that (1) and (2) above allow us to freely substitute L_k-terms into $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions. Nevertheless, it seems hard to directly show the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions are closed under composition. However, Lemma 1 together with the next lemma concerning projecting coordinates from ordered pairs turns out to suffice for the witnessing argument.

Lemma 2. *Let A be a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-predicate, s an $L_{k,\mathrm{exp}}$-term, and let h be a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function. Suppose f, g are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions and so are $(f)_i$, $(g)_i$ for $i = 1, 2$. Then the following are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions:* (1) $(\mathrm{cond}(A, f, g))_i$, $i = 1, 2$. (2) $(\langle f, g\rangle)_i$, $i = 1, 2$. (3) $(f(h))_i$, $i = 1, 2$.

Proof. (1) Using Lemma 1, define $(\mathrm{cond}(A, f, g))_i$ as $\mathrm{cond}(A, (f)_i, (g)_i)$. (2) this follows since f and g are assumed to be $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions. (3) This follows by Lemma 1(2), since by assumption $(f)_i$, $i = 1, 2$ are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions. $\square$

A formula is in $LE_{1,\mathrm{exp}}(\Psi)[\vec{b}]$ if it can be made into a $E_{1,\mathrm{exp}}(\Psi)[\vec{b}]$-formula by padding on dummy quantifiers. A bounding term and witness predicate for $LE_{1,\mathrm{exp}}(\Psi)[\vec{b}]$-formulas is now defined.

- If $A(\vec{a}; \vec{b}) \in \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ then $t_A = 0$ and $\mathrm{WIT}_A(\vec{a}; w, \vec{b}) := A(\vec{a}; \vec{b}) \wedge w = 0$.
- If $A(\vec{a}; \vec{b}) \in \Psi[\vec{b}]$ is of form $\exists x \leq tB$ then $t_A := t$ and $\mathrm{WIT}_A(\vec{a}; w, \vec{b}) := w \leq t \wedge B$.
- If $A(\vec{a}) \in E_{1,\mathrm{exp}}(\Psi)[\vec{b}]$ is of the form $(\exists x_1 \leq t_1)(\exists x_2 \leq t_2)B$, then $t_A := \langle t_1, t_2\rangle$ and

$$\mathrm{WIT}_A(\vec{a}; w, \vec{b}) := \mathit{ispair}(w) \wedge (w)_1 \leq t_1 \wedge (w)_2 \leq t_2 \wedge$$

$$B(\vec{a}; (w)_1, (w)_2, \vec{b}).$$

For a cedent $\Gamma = \{A_1, \ldots, A_N\}$ of $LE_{1,\mathrm{exp}}(\Psi)[\vec{b}]$-formulas, let $\wedge\Gamma$ denote their conjunction and $\vee\,\Gamma$ their disjunction. Let Δ be another such cedent. Following Pollett [17] one can define $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ witness predicates

$\mathrm{WIT}_{\wedge\Gamma}(\vec{a};w,\vec{b})$, $\mathrm{WIT}_{\vee\Delta}(\vec{a};w,\vec{b})$ for such cedents and also terms t_Γ and t_Δ such that $S_{k,\exp} \vdash \Gamma \to \Delta$ iff

$$S_{k,\exp} \vdash (\exists w \le t_\Gamma)\mathrm{WIT}_{\wedge\Gamma}(\vec{a};w,\vec{b}) \to (\exists w \le t_\Delta)\mathrm{WIT}_{\vee\Delta}(\vec{a};w,\vec{b}).$$

The predicate $\mathrm{WIT}_{\vee\Delta}(\vec{a};w,\vec{b})$ in Pollett [17] has the property that if Δ is empty then it is $\neg(0=0)$. If Δ consists of just one formula A then $\mathrm{WIT}_{\vee\Delta}(\vec{a};w,\vec{b}) = \mathrm{WIT}_A(\vec{a};w,\vec{b})$ and if $\Delta = A, \Delta'$ then

$$\mathrm{WIT}_{\vee\Delta}(\vec{a};w,\vec{b}) = \mathrm{WIT}_A(\vec{a};(w)_1,\vec{b}) \vee \mathrm{WIT}_{\vee\Delta'}(\vec{a};(w)_2,\vec{b}).$$

The predicate $\mathrm{WIT}_{\wedge\Gamma}(\vec{a};w,\vec{b})$ is defined similarly using conjunction rather than disjunction. In the case where Γ is empty, it is defined as $0=0$.

Theorem 1. *Suppose*

$$S_{k,\exp} \vdash \Gamma \to \Delta$$

where Γ, Δ are cedents of $LE_{1,\exp}(\Psi)[\vec{b}]$-formulas. Then there is a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function f such that

$$\mathbb{N} \models \mathrm{WIT}_{\wedge\Gamma}(\vec{a};w,\vec{b}) \supset \mathrm{WIT}_{\vee\Delta}(\vec{a};f(\vec{a};w,\vec{b}),\vec{b}).$$

Further, the projections of f needed to witness the individual existentials of the formulas (the projections of f) in Δ are also $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions.

Proof. This is proved by induction on the number of inferences in a $S_{k,\exp}$-proof of $\Gamma \to \Delta$. By the restriction on cut, it can be assumed that all the sequents in the proof are in $LE_{1,\exp}(\Psi)[\vec{b}]$. By Buss [3], it can also be assumed that the proof is in free variable normal form and restricted by parameters. So the elimination inference for a free variable in the proof is not a cut, ($\forall \le$:left), or ($\exists \le$:right) inference and also the term substituted into the conclusion of an induction inference only involves parameter variables of the endsequent. In the base case, the proof consists of a sequent $A \to A$, where A is an atomic formula, an equality axiom, or an $\mathrm{BASIC}_{k,\exp}$ axiom. In each of these cases the witness predicate for each of the formulas in the sequent is of the form $A \wedge w = 0$. So a witness for Δ can be constructed as pairings of the 0-function which is a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function. It is easy to verify that the projections of such pairings of the 0-functions are also $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions. The weak inferences, structural inferences, and cut can be handled in essentially the same way as in the S^i_2 case of the witnessing argument in Buss [3]. The $(cut - rule)$

will be shown below to illustrate why the *cond* function was needed. The remaining cases are the bounded quantifier rules and induction. The ($\exists \leq$:left) and ($\exists \leq$:right) – the ($\forall \leq$:left) and ($\forall \leq$:right) are similar, but simpler – and, the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-$\overline{\mathrm{IND}}$ case are shown.

(Cut rule case) Suppose the inference is:

$$\frac{\Gamma \rightarrow A, \Delta \qquad A, \Gamma \rightarrow \Delta}{\Gamma \rightarrow \Delta}.$$

The induction hypothesis gives $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions g and h whose relevant projections are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions such that

$$\mathbb{N} \models \mathrm{WIT}_{\wedge\Gamma}(\vec{a}; w, \vec{b}) \rightarrow \mathrm{WIT}_{A\vee\Delta}(\vec{a}; g(\vec{a}; w, \vec{b}), \vec{b})$$

$$\mathbb{N} \models \mathrm{WIT}_{A\wedge\Gamma}(\vec{a}; w, \vec{b}) \rightarrow \mathrm{WIT}_{\vee\Delta}(\vec{a}; h(\vec{a}; w, \vec{b}), \vec{b}).$$

As was mentioned above since the proof is in free variable normal form no free variable will be eliminated by this cut. Define the function k as

$$k(\vec{a}; v, w, \vec{b}) := \mathrm{cond}(A, v, w)$$

Notice k is a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function because A must be a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ formula by the restriction on cuts that every $S_{k,\exp}$ derivation must satisfy. Define the function f to be

$$f(\vec{a}; w, \vec{b}) := k(\vec{a}; (g(\vec{a}; w, \vec{b}))_1, h(\vec{a}; \langle 0, w\rangle, \vec{b}), \vec{b}).$$

By Lemmas 1 and 2, and by the assumption that the projections of g and h are defined, f is in $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ and so are its projections. The function f also satisfies:

$$\mathbb{N} \models \mathrm{WIT}_{\wedge\Gamma}(\vec{a}; w, \vec{b}) \rightarrow \mathrm{WIT}_{\vee\Delta}(\vec{a}; f(\vec{a}; w, \vec{b}), \vec{b}).$$

($\exists \leq$:left case) Suppose the inference is:

$$\frac{c \leq t, A(c), \Gamma \rightarrow \Delta}{(\exists x \leq t)A(x), \Gamma \rightarrow \Delta}.$$

The induction hypothesis gives a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function g whose relevant projections are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions such that

$$\mathbb{N} \models \mathrm{WIT}_{c\leq t\wedge A\wedge\Gamma}(\vec{a}; w, c, \vec{b}) \supset \mathrm{WIT}_{\vee\Delta}(\vec{a}; g(\vec{a}; w, c, \vec{b}), c, \vec{b}).$$

There are three subcases to consider. In each case, we need to determine a value for c and then run g on that value.

In the first case, $(\exists x \leq t)A(x) \in E_{1,\exp}(\Psi)[\vec{b}]$. If w witnesses $(\exists x \leq t)A(x) \wedge \Gamma$ then $((w)_1)_1$ is a value for c such that $A(c)$ holds and $((w)_1)_2$ is a witness for $A(c)$. So let $f(\vec{a}; w, \vec{b}) := g(\vec{a}; \langle\langle 0, ((w)_1)_2, (w)_2\rangle\rangle, ((w)_1)_1, \vec{b})$. Then

$$\mathbb{N} \models \mathrm{WIT}_{(\exists x \leq t)A \wedge \Gamma}(\vec{a}; w, \vec{b}) \supset \mathrm{WIT}_{\vee\Delta}(\vec{a}; f(\vec{a}; w, \vec{b}), \vec{b}).$$

In the second case, $(\exists x \leq t)A(x) \in \Psi[\vec{b}]$. If w witnesses $(\exists x \leq t)A(x) \wedge \Gamma$, then $(w)_1$ is a value for b such that $A(b)$ holds. Let

$$f(\vec{a}; w, \vec{b}) := g(\vec{a}; \langle\langle 0, 0, (w)_2\rangle\rangle, (w)_1, \vec{b}).$$

Then

$$\mathbb{N} \models \mathrm{WIT}_{(\exists x \leq t)A \wedge \Gamma}(\vec{a}; w, \vec{b}) \supset \mathrm{WIT}_{\vee\Delta}(\vec{a}; f(\vec{a}; w, \vec{b}), \vec{b}).$$

The last subcase is when $(\exists x \leq t)A(x) \in \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$. In this case c might not occur in a term involving 2^x so it might be better to write g as $g(\vec{a}, c; w, \vec{b})$. Whether it does or not, one can define f as above except rather than use $((w)_1)_1$ or $(w)_1$ for c use the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function $\mu x \leq t \neg A(x)$.

For each of the cases by Lemma 1, the resulting function will be in $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$ and its projection can be defined using that g's projections are defined and Lemma 2.

($\exists \leq$:right case) Suppose the inference is:

$$\frac{\Gamma \to A(t), \Delta}{t \leq s, \Gamma \to (\exists x \leq s)A(x), \Delta}.$$

The induction hypothesis gives a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function g whose relevant projections are $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions such that

$$\mathbb{N} \models \mathrm{WIT}_{\wedge\Gamma}(\vec{a}; w, \vec{b}) \supset \mathrm{WIT}_{A(t)\vee\Delta}(\vec{a}; g(\vec{a}; w, \vec{b}), \vec{b}).$$

The definition of WIT implies

$$\mathbb{N} \models \mathrm{WIT}_{t \leq s \wedge \Gamma}(\vec{a}; w, \vec{b}) \supset t \leq s \wedge \mathrm{WIT}_{\wedge\Gamma}(\vec{a}; (w)_2, \vec{b}).$$

So if $A \in \Psi[\vec{b}]$ define $f := \langle\langle t, (g(\vec{a}; (w)_2, \vec{b}))_1\rangle, (g(\vec{a}; (w)_2, \vec{b}))_2\rangle$. If $A \in \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ and s involves 2^x define $f := \langle t, (g(\vec{a}; (w)_2, \vec{b}))_2\rangle$. For all other

A define $f := g(\vec{a};(w)_2,\vec{b})$. These functions are all $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$ and given Lemma 1 and the assumption that the relevant projections of g are in $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$, so are the projections of f. Finally, note in each case

$$\mathbb{N} \models \mathrm{WIT}_{t\leq s\wedge\Gamma}(\vec{a};w,\vec{b}) \supset \mathrm{WIT}_{(\exists x\leq s)A(x)\vee\Delta}(\vec{a};f(\vec{a};w,\vec{b}),\vec{b}).$$

($\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-$\overline{\mathrm{IND}}$ case) Suppose the inference is:

$$\frac{A(c),\Gamma \to A(Sc),\Delta}{A(0),\Gamma \to A(t),\Delta}.$$

Since both the upper and lower sequents involve only $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-formulas, a witness function for both the upper and lower sequents just needs to map the number of formulas in the antecedent pairings of 0 into the number of formulas in the succedent pairings of 0. This could be done by an L_2-term and so is a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function. Similarly, the appropriate projections of these pairings can be done by L_2-terms and so will be $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function. Since the proof is restricted by parameters, there is no need have to worry that t introduces any new parameter variables that are not in the endsequent.

This completes the cases that remained to be shown and the proof. □

Corollary 1. (1) *The $\nabla_{1,\exp}[\vec{b}]$-predicates of $S_{k,\exp}$ are precisely the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-predicates.* (2) *The $\nabla_{1,\exp}$-predicates of $S_{k,\exp}$ are precisely the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-predicates.*

Proof. (2) follows easily from (1) so only (1) is shown. (1) By definition any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-predicate will be $\nabla_{1,\exp}[\vec{b}]$ in $S_{k,\exp}$. On the other hand, if $S_{k,\exp}$ proves A is $\nabla_{1,\exp}[\vec{b}]$, let A_Σ and $\neg A_\Pi$ be tautologicaly equivalent to $\Psi[\vec{b}]$-formulas and suppose $A_\Pi \Leftrightarrow A \Leftrightarrow A_\Sigma$ is provable in $S_{k,\exp}$. Consider

$$B(\vec{a},y;\vec{b}) := (\neg A_\Pi(\vec{a};\vec{b}) \wedge y=0) \vee (A_\Sigma(\vec{a};\vec{b}) \wedge y=1).$$

Certainly, $S_{k,\exp}$ proves $(\exists y \leq 1)B(\vec{a},y;\vec{b})$ and proves B is equivalent to some $\Psi[\vec{b}]$-formula. So by Theorem 1, there is a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-function f such that

$$\mathbb{N} \models \mathrm{WIT}_B(\vec{a};f(\vec{a};\vec{b}),\vec{b}).$$

Further, by Theorem 1, the projections of f needed to witness the individual existentials of B are also $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-functions. So A, which is equivalent to $(f)_1 = 1$, will be a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}[\vec{b}]$-predicate. □

4. Main result

Lemma 3. *Let $t(x, w)$ be an $L_{k,\exp}$-term. Then the predicate $t(x, w) = 0$ is computable in polynomial time.*

Proof. The idea is to take the input numbers and encode them as polynomial lengthed sequences of a kind that it is easy to check if such a sequence represents zero or not. Then for each operation in the term, rather than calculating its value directly, instead an output sequence (again of polynomial length) is calculated in terms of its input sequences. To be precise, a *stack code* is defined inductively as: (1) the sequence $\langle \ulcorner 0 \urcorner \rangle$ (intended to mean 0) is a stack code, (2) if v and w are stack codes then so are the sequences: $\langle \ulcorner + \urcorner, v, w\rangle$ (intended to mean $v + w$), $\langle \ulcorner - \urcorner, v\rangle$ (intended to mean $-v$), and $\langle \ulcorner 2^x \urcorner, v\rangle$ (intended to mean 2^v), (3) nothing else is a stack code. Here $\ulcorner\urcorner$ is used to represent a fixed Gödel code for the given symbol. The intended value of a stack code is the number one gets by evaluating the sequence according to the intended meaning of the symbols. Note there might be many stack codes for a number. For instance, $2^{2^0} + (-2^0)$ and 2^0 rewritten as stack code sequences would both represent 1. Given a number x one can find a polynomial length stack code for it in polynomial time by determining the "on" bits in its binary representation and coming up with stack codes for each of them. For each symbol in $L_{k,\exp}$, one can verify that if one has stack codes for each of its inputs then a stack code for its output can be constructed in polynomial time in the size of its inputs (see Pollett [18] for more details on this). Moreover, one can do the initial encoding and the computation of each operation in the term so that after applying each operation the stack code has the form (rewritten in a simplified infix notation)

$$(2^{\bar{x}_0} \pm 2^{\bar{x}_1} \cdots \pm 2^{\bar{x}_n}),$$

where $\bar{x}_i$ are stack codes for numbers $x_0 > x_1 > \cdots > x_n$. Given that stack codes can be computed in this normal form, after computing the stack code for $t(x, w)$ one can check whether it is the same sequence as $\langle \ulcorner 0 \urcorner \rangle$ to compute whether $t(x, w) = 0$. $\square$

Lemma 4. (1) *For any $E_{1,\exp}$-formula $A(a)$, there is a $E_{1,2}$-formula (notice the 2) $U_A(a, z)$ such that there is a $L_{k,\exp}$-term t_A for which*

$$\mathrm{BASIC}_{k,\exp} \vdash U_A(a, t_A(a)) \equiv A(a).$$

(2) *There is a $\widehat{\Sigma}^{\mathrm{b}}_{1,\exp}$-formula $U_\infty(e, a)$ such that for any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-*

formula $A(a)$ *there is a number* e_A *for which*

$$\mathbb{N} \models U_\infty(e_A, a) \equiv A(a).$$

Proof. (1) For this proof, the separation of variables by a semi-colon into those that do not occur in terms involving 2^x and those that do will not be indicated. Using the work in Pollett [16] one can show that $\mathrm{BASIC}_{k,\exp}$ can prove enough properties of the pairing functions and block coding to carry out the argument that is now presented. Using $\mathrm{K}_\neg(x) := 1 \dot{-} x$, $\mathrm{K}_\vee(x,y) := x + y$, and $\mathrm{K}_\leq(x,y) := \mathrm{K}_\neg(y \dot{-} x)$, one can write any open formula $A(x,\vec{y})$ as an equation $f(x,\vec{y}) = 0$ where $f \in L_k$. So any $E_{1,\exp}$-formula $\phi(x)$ is provably equivalent in $\mathrm{BASIC}_{k,\exp}$ to one of the form

$$(\exists y \leq t_1)(t_2(x,y) = 0)$$

where the t_i's are in $L_{k,\exp}$. Fix a coding scheme for the 11 non-$\#_k$ symbols of $L_{k,\exp}$ as well as for the 2 variables x, y. The symbols $\#_k$, $k > 2$ will be broken down into suboperations $\#^i_k$, $1 \leq i \leq k-1$, described below, each of which is defined in terms of the other symbols in the language. Creating codes for these suboperations will give a total of $k' := \frac{k(k-1)}{2} + 13$ symbols to code. Use $\ulcorner\urcorner$ to denote the code for some symbol, i.e., $\ulcorner = \urcorner$ is the code for $=$. Choose a coding so that all codes require less than $|k'|$ bits and we use 0 as $\ulcorner NOP \urcorner$ meaning no operation. Thus, if one tries to project out operations beyond the end of the code of the term one naturally just projects outs $\ulcorner NOP \urcorner$'s. The code for a term t is a sequence of blocks of length $|k'|$ that write out t in postfix order. So $x + y_1$ would be coded as the three blocks $\ulcorner x \urcorner \ulcorner y_1 \urcorner \ulcorner + \urcorner$. The code for a $E_{1,\exp}$-formula will be $\langle\langle \ulcorner t_1 \urcorner, \ulcorner t_2 \urcorner \rangle\rangle$. $U_A(x,z)$ is obtained from the formula

$$(\exists w \leq z)(\exists y_1 \leq z)(\forall j \leq |e|)\phi(e,j,x,\vec{y})$$

after pairing is applied and converting the $(\forall j \leq |e|)$ into the finite appropriate finite conjunction. The conjunction will be finite since for any fixed A, its code will be a finite number. Here ϕ consists of a statement saying w is a tuple of the form $\langle\langle w_1, w_2 \rangle\rangle$ together with statements saying each w_m codes a postfix computation of t_m in $e = \langle\langle \ulcorner t_1 \urcorner, \ulcorner t_2 \urcorner \rangle\rangle$. If $z' := MSP(z, \lfloor \frac{1}{2} x \rfloor |z|)$ (roughly, the square root of z) is used as the block size, this amounts to checking conditions for each m

$$[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner x \urcorner \supset \widehat{\beta}_{|z'|}(j, w_m) = x] \wedge$$
$$[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner + \urcorner \supset$$

$$
\begin{aligned}
&\widehat{\beta}_{|z'|}(j, w_m) = \widehat{\beta}_{|z'|}(j \dot{-} 2, w_m) + \widehat{\beta}_{|z'|}(j \dot{-} 1, w_m)] \wedge \cdots \\
&\qquad\qquad\qquad\qquad\cdots \\
&[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner \# \urcorner \supset \\
&|\widehat{\beta}_{|z'|}(j, w_m)| = S(|\widehat{\beta}_{|z'|}(j \dot{-} 2, w_m)||\widehat{\beta}_{|z'|}(j \dot{-} 1, w_m)|) \\
&\wedge LSP(\widehat{\beta}_{|z'|}(j, w_m), |\widehat{\beta}_{|z'|}(j, w_m)| \dot{-} 1) = 0] \wedge \\
&[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner \#_3^2 \urcorner \supset \\
&[\widehat{\beta}_{|k'|}(j \dot{-} 1, \ulcorner t_m \urcorner) = \ulcorner \#_3^1 \urcorner \wedge \\
&|\widehat{\beta}_{|z'|}(j, w_m)| = S(|\widehat{\beta}_{|z'|}(j \dot{-} 1, w_m)|) \\
&\wedge LSP(\widehat{\beta}_{|z'|}(j, w_m), |\widehat{\beta}_{|z'|}(j, w_m)| \dot{-} 1) = 0] \wedge \\
&[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner \#_3^2 \urcorner \supset \\
&|\widehat{\beta}_{|z'|}(j, w_m)| = S(||\widehat{\beta}_{|z'|}(j, w_m)||||\widehat{\beta}_{|z'|}(j, w_m)||) \\
&\wedge \mathrm{LSP}(\widehat{\beta}_{|z'|}(j, w_m), |\widehat{\beta}_{|z'|}(j, w_m)| \dot{-} 1) = 0] \cdots \\
&[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner 2^x \urcorner \supset \\
&|\widehat{\beta}_{|z'|}(j, w_m)| = S(\widehat{\beta}_{|z'|}(j \dot{-} 1, w_m)) \\
&\wedge LSP(\widehat{\beta}_{|z'|}(j, w_m), |\widehat{\beta}_{|z'|}(j, w_m)| \dot{-} 1) = 0] \\
&\qquad\qquad\qquad\qquad\cdots \\
&[\widehat{\beta}_{|k'|}(j, \ulcorner t_m \urcorner) = \ulcorner NOP \urcorner \supset \widehat{\beta}_{|z'|}(j, w_m) = \widehat{\beta}_{|z'|}(j \dot{-} 1, w_m)].
\end{aligned}
$$

Notice how a valid code of a term involving $\#_3$ has this operation coded as two operations $\ulcorner \#_3^1 \urcorner$ followed by $\ulcorner \#_3^2 \urcorner$. For $\#_k$ uses codes $k-1$ sub-operations $\#_k^i$ in a similar manner. The formula ϕ also has a condition $y_1 \leq \widehat{\beta}_{|z'|}(|e|, w_m) \wedge$ to bound the existential quantifier to the value of t_1. It should be observed that none of the conditions mentioned make use of the $\#_3, \ldots, \#_k$, or 2^x functions. Finally, ϕ has a condition saying $\widehat{\beta}_{|z'|}(|e|, w_2) = 0$. Since $\mathrm{BASIC}_{k,\mathrm{exp}}$ can prove simple facts about projections from pairs, it can prove by induction on the complexity of the terms in any $\widehat{\Sigma}^{\mathrm{b}}_{1,\mathrm{exp}}$-formula A that $U(e_\phi, x, t_A(x)) \equiv A(x)$ provided $t_A(x)$ is large enough to bound the codes of the computations of t_1 and t_2. $t_A(x)$ can be chosen to be an $L_{k,\mathrm{exp}}$-term because the sequences involved in the above are of finite length and at each step in the computation of a term the size of a given intermediate value can grow by at most a function of 2^x times the size of the previous values.

(2) First, in view of Lemma 3, given a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formula A, the $open_{\exp}$ matrix of A can be replaced by a $E_{1,2}$-formula. Thus, A can be converted into a $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$-formula. So from now on it is assumed that A is from $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$. Consider the following valid kind of quantifier replacement

$$(\forall x \leq s)(\exists y \leq t(x,a))A(x,y,a) \Leftrightarrow$$
$$(\exists w \leq 2 \cdot (t^*(s,a)\#(2^s)))(\forall x \leq s)A(x, \dot{\beta}(x, |t^*(s,a)|, t, w)),$$

where t and s are in L_k, t^* is an inductively defined nondecreasing term bounding t and $\dot{\beta}(x,|t|,s,w) := \min(\widehat{\beta}(x,|t|,w),s)$ where $\min(x,y) := (1 \dot{-} (x \dot{-} y))x + 1 \dot{-} (y \dot{-} x))y$. Using this kind of replacement and the fact that $|2^s| - 1 = s$, any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$-formula can be shown equivalent to a $\widehat{\Sigma}^{\mathrm{b}}_{1,\exp}$-formula. Moreover, the term bounding the outermost existential can be bounded by applying $\#$ to a finite number of terms of the form 2^s where s is in L_k. Thus, any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$-formula A can be coded by the triple, coding in the same fashion as in (1), the three terms in the $\widehat{\Sigma}^{\mathrm{b}}_{1,\exp}$-formula obtained by doing the above kind of quantifier exchanges to A. Thus, A would be equivalent to the formula

$$U'(e,x,z) := (\exists w \leq z)(\exists y_1 \leq z)(\forall j \leq |e|)(\forall y_2 \leq |z|)\phi(e,j,x,\vec{y})$$

after an appropriate term was substituted for z. Here ϕ is an appropriately modified version of the ϕ from (1). Given that the largest of the three terms is the outermost existential and as just mentioned it can be be bounded by applying $\#$ to a finite number of terms of the form 2^s where s is in L_k, one can find a fixed term t of growth rate 2^{2^s} for some L_k-term s which can bound the size of the computations needed to calculate the value of any three such terms. Using this t and applying pairing to U' to make it a $\widehat{\Sigma}^{\mathrm{b}}_{1,\exp}$-formula we get any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}$-formula (and, hence, also as remarked at the beginning of this proof, any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formula) A is equivalent to $U_\infty(e_A,x) := U'(e_A,x,t)$ for an appropriately chosen code e_A. □

Lemma 5. $E_{1,\exp} \neq \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$.

Proof. Suppose the class of $E_{1,\exp}$ predicates and $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$ predicates were the same. Then as $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$ predicates are close under complement $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k} = E_{1,\exp} = U_{1,\exp} = \widehat{\Pi}^{\mathrm{b}}_{1,\exp} = \widehat{\Sigma}^{\mathrm{b}}_{\infty,\exp}$ and so $\neg U_\infty(a,a)$ would be in $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$. But this formula is easily seen not to be equivalent to any $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-formula. □

Definition 3. *A derivation system T proves a sequent $\Gamma(a) \rightarrow \Delta(a)$ uniformly by proof P if one can substitute any term t in the language for the variable a everywhere in P and still obtain a valid derivation in T. T proves classes Ψ and Φ equivalent uniformly if for every formula $\phi \in \Phi$ there is some formula $\psi \in \Psi$ such that $\phi \Leftrightarrow \psi$ has a uniform proof in T.*

By examining the rules of inference allowed in an $S_{k,\exp}$ proof, one can check that the only kinds of inferences that could cause a proof to fail to be uniform are cut-inferences and $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-$\overline{\mathrm{IND}}$ inferences. As a free-cut free $\mathrm{BASIC}_{k,\exp}$ derivation uses no induction and has cut only on open formulas, any $\mathrm{BASIC}_{k,\exp}$ derivation is uniform. Thus, Lemma 4(1) has a uniform $S_{k,\exp}$-proofs.

Theorem 2. *$S_{k,\exp}$ cannot prove $E_{1,k'} = U_{1,k'}$ uniformly where $2 \leq k' \leq k$. Since, as argued in the preliminaries, the $E_{1,2}$-sets are precisely the predicates in* NP*, this means $S_{k,\exp}$ cannot prove* NP $=$ *co*-NP *uniformly.*

Proof. Suppose $S_{k,\exp}$ proves $E_{1,k'} = U_{1,k'}$ uniformly. This means that for each $E_{1,k'}$-formula C there is some $U_{1,k'}$-formula D such that $S_{k,\exp} \vdash C \equiv D$ uniformly. Let $A(x) := \exists y \leq t(x) D(x,y)$ be an arbitrary $E_{1,\exp}$-formula in one variable. Let $U_A(x,z)$ be the formula from Lemma 4. So U_A is in $E_{1,2} \subseteq E_{i,k'}$, and, thus, by assumption, provably equivalent to some $U_{1,k'}$-formula $U'_A(x,z)$ in $S_{k,\exp}$ by a uniform proof. Since these proofs were all uniform, $S_{k,\exp}$ proves

$$A \equiv U_A(x, t_A(x)) \equiv U'_A(x, t_A(x)),$$

where t_A is the bounding term on U_A in Lemma 4. The last formula is a $U_{1,\exp}$-formula. Hence, it follows that $S_{k,\exp}$ proves

$$E_{1,\exp} = U_{1,\exp} = \widehat{\Sigma}^{\mathrm{b}}_{\infty,\exp} \supseteq \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp}).$$

Further, by the first equality above every $E_{1,\exp}$ set would be $\bigtriangledown_{1,\exp}$ in $S_{k,\exp}$. As the $\bigtriangledown_{1,\exp}$-formulas of $S_{k,\exp}$ are precisely $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$ by Corollary 1, one also gets that $E_{1,\exp} = \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$. But this contradicts Lemma 5. $\square$

5. DPRM Lower Bound

In this section, it is shown that $S_{k,\exp}$ cannot prove the DPRM theorem. To show this, we begin with a proof of the folklore observation mentioned in the introduction:

Lemma 6. *If $S_{k,\exp}$ proves the DPRM theorem then $S_{k,\exp}$ proves $E_{1,\exp} = U_{1,\exp}$.*

Proof. To see this, suppose $S_{k,\exp}$ proves the DPRM theorem. Then for every $U_{1,\exp}$-formula $A(\vec{x})$ there is a formula $F(\vec{x}) := (\exists\vec{y})P(\vec{x},\vec{y}) = Q(\vec{x},\vec{y})$ where P, Q are polynomials such that $S_{k,\exp} \vdash A \equiv F$. In particular, $S_{k,\exp}$ proves $A \rightarrow (\exists\vec{y})P(\vec{x},\vec{y}) = Q(\vec{x},\vec{y})$. By Parikh's theorem [15], since $S_{k,\exp}$ is a bounded theory one can bound the $\vec{y}$'s by an $L_{k,\exp}$-term t giving an $E_{1,\exp}$-formula F_2. Note $F_2 \supset F \supset A$ so $A \Leftrightarrow F_2$ completing the proof. □

Theorem 3. *$S_{k,\exp}$ does not prove the DPRM theorem.*

Proof. By the previous lemma, if $S_{k,\exp}$ proves the DPRM theorem then

$$E_{1,\exp} = U_{1,\exp} = \widehat{\Sigma}^{\mathrm{b}}_{\infty,\exp}.$$

On the other hand, the things that $S_{k,\exp}$ proves are both $E_{1,\exp}$ and $U_{1,\exp}$ are contained in its $\bigtriangledown_{1,\exp}$-predicates. Hence,

$$E_{1,\exp} = \widehat{\Sigma}^{\mathrm{b}}_{\infty,\exp} \subseteq \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})).$$

As the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,\exp}$-predicates contain the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$-predicates, it follows that $E_{1,\exp} = \widehat{\Sigma}^{\mathrm{b}}_{\infty,k}(open_{\exp})$. But this contradicts Lemma 5. □

6. Generalizations

In this section, it will be argued that the results of this paper can be generalized to finite levels of the Grzegorczyk Hierarchy.

Consider the following variation of the branches of the Ackermann function defined for $n \geq 2$: (1) $h_2(x) = 2^x$, (2) $h_{n+1}(0) = h_n(0)$, and (3) $h_{n+1}(Sx) = h_n(h_{n+1}(x))$. As an example of these growth rate, it is not hard to verify that $h_3(x)$ will be an x-high stack of 2's with a 0 in as the final exponent. One can define a rounded down inverse function, $h_n^{-1}(x)$, for each of the h_n functions, as the unique number satifying $h_n(h_n^{-1}(x)) \leq x < h_n(h_n^{-1}(x+1))$. It is not hard to show that these growth rates are elementary equivalent with the usual Ackermann branches $h'_0(x) = x+1$ and $h'_{n+1} = h'^{(x)}_n(x)$, so will not effect the definition of the levels of the Grzegorczyk Hierarchy. These levels are defined for $n > 2$ as the classes, $\mathcal{E}_n$, which are the closure under composition of the functions h_{n-1}, $x - y$, x^y, and the operation of bounded μ-recursion. The union of these finite levels gives precisely the primitive recursive functions. See Odifreddi [14] for more information about this hierarchy.

Let $L_{\mathcal{E}_2} := L_{2,\exp}$, and for $n > 2$ define $L_{\mathcal{E}_{n+1}} := L_{\mathcal{E}_n} \cup \{h_n, h_n^{-1}\}$. Next denote by $L^-_{\mathcal{E}_n}$, the language $L_{\mathcal{E}_n}$ less the symbols for 2^x and h_m for $3 \leq m \leq n$. Observe that the functions h_n^{-1} are each p-time computable, so the sets given by $E_{1,L^-_{\mathcal{E}_n}}$-formulas will still be NP-sets. Also, observe a μ operation bounded by a $L_{\mathcal{E}_n}$-term can be defined using an $L_{\mathcal{E}_n}$ bounded existential followed by a $L_{\mathcal{E}_n}$ bounded universal and with a $L_{\mathcal{E}_n}$ bounded μ-operator you can simulate as a $0-1$-function the value of a $L_{\mathcal{E}_n}$ bounded quantifier. Thus, it is straightforward to show that the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}$-sets are precisely the sets in $\mathcal{E}_n$.

Let $\mathrm{BASIC}_{\mathcal{E}_n}$ be $\mathrm{BASIC}_{2,\exp}$ extended by additional open axioms for the symbols h_m and h_m^{-1} for $3 \leq m \leq n$ and define $I\mathcal{E}_{n,n+1}$ as $\mathrm{BASIC}_{\mathcal{E}_{n+1}}$ together with the inference $\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}(open_{L_{\mathcal{E}_{n+1}}})$-$\overline{\mathrm{IND}}$ and the restriction on cuts to be only on $\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}(open_{L_{\mathcal{E}_{n+1}}})$-formulas. A predicate that is provably equivalent to both a $\widehat{\Sigma}^{\mathrm{b}}_{1,L_{\mathcal{E}_{n+1}}}(\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}(open_{L_{\mathcal{E}_{n+1}}}))$-formula and the negation of such a formula is called a $\bigtriangledown_{1,L_{\mathcal{E}_{n+1}}}$-predicate. By the same kind of argument as in the $S_{k,\exp}$ case one case show the $\bigtriangledown_{1,L_{\mathcal{E}_{n+1}}}$-predicates of $I\mathcal{E}_{n,n+1}$ are precisely the $\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}(open_{L_{\mathcal{E}_{n+1}}})$-sets. As in the $S_{k,\exp}$ case, one can also show that for any $E_{1,L_{\mathcal{E}_{n+1}}}$-formula $A(x)$ there is an $E_{1,L^-_{\mathcal{E}_{n+1}}}$-formula $U_A(x,z)$ and a $L_{\mathcal{E}_{n+1}}$-term t_A such that $I\mathcal{E}_{n,n+1}$ proves $A(x) \Leftrightarrow U_A(x,t_A)$. this is because one can verify in a computation that $y = h_n(x)$ by the equation $h_n^{-1}(y) = x$. Also, one can generalize the stack code idea to show that equations $t(x,w) = 0$ in the language of $L_{\mathcal{E}_{n+1}}$ can be evaluated in polynomial time in the inputs. Thus,

$$\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}(open_{L_{\mathcal{E}_{n+1}}}) = \widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}.$$

Recall it was just argued that the $E_{1,L^-_{\mathcal{E}_{n+1}}}$-sets are just the sets in NP. So one can argue in the same fashion as in the $S_{k,\exp}$ case that if $I\mathcal{E}_{n,n+1}$ proves either NP $=$ *co*-NP uniformly or the $DPRM$ theorem then

$$\widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_n}}(open_{L_{\mathcal{E}_{n+1}}}) = \widehat{\Sigma}^{\mathrm{b}}_{\infty,L_{\mathcal{E}_{n+1}}}.$$

This would imply, however, that $\mathcal{E}_n = \mathcal{E}_{n+1}$ which is well known to be false [14]. Thus, it can be concluded that:

Theorem 4. *For $n > 2$, $I\mathcal{E}_{n,n+1}$ cannot prove* NP $=$ *co*-NP *uniformly and also cannot prove the* DPRM *theorem.*

7. Conclusion

In this section, further avenues of research are suggested.

One obvious first avenue would be to find out how much the requirement of uniform proof of NP=*co*-NP can be weakened. It is reasonable to wonder as well how far the results of the last section can be continued higher up into the extended Grzegorczyk Hierarchy. The arguments of this paper are reasonably insensitive to expansions of the underlying language by function symbols of subexponential growth. This gives reasonably strong evidence that $I\Delta_0 + \exp$ might be the weakest theory able to prove the DPRM theorem in a language with exponentiation. Nevertheless, it might be interesting to consider stronger theories than $S_{k,\exp}$ that are weaker than $I\Delta_0 + \exp$ by adding other axiom schemas such as restricted forms of comprehension or replacement axioms. This paper also leaves open whether $S_{k,\exp}$ can prove $\Sigma^{\mathrm{p}}_i = \Pi^{\mathrm{p}}_i$ uniformly for any $i > 1$.

As a final comment on lines of further research, it would be interesting to get an unconditional result concerning the provability of NP $\neq$ *co*-NP in bounded arithmetic. Razborov [20] has shown that assuming the existence of pseudorandom number generators secure against attacks by quasi-polynomial sized circuit families that $S^2_2(\alpha)$ cannot prove superpolynomial lower bounds on circuit size for NP-predicates. Here α is a second order predicate symbol with a polynomial bounded domain. In view of the results paper, it seems likely to the author there are models of $S^2_2(\alpha)$ in which NP $\neq$ *co*-NP, so there might be hope of constructing models in which pseudorandom number generators of the appropriate strength exist.

8. Acknowledgements

The author would like to thank Arnold Beckmann and Jan Johannsen for e-mail conversations related to an earlier version of this paper.

References

1. L. M. Adleman and K. Manders, *The computational complexity of decision procedures for polynomials.* in: *Proceedings of the Sixteenth Annual Symposium on the Foundations of Computer Science* (1975), pp. 169–177.
2. L. M. Adleman and K. Manders, *Diophantine Complexity.* in: *Proceedings of the Seventeenth Annual Symposium on the Foundations of Computer Science* (1976), pp. 81–88..
3. S. R. Buss, *Bounded Arithmetic.* Bibliopolis, Napoli (1986).

4. P. Clote and G. Takeuti, *First order bounded arithmetic and small boolean circuit complexity classes.* in: *Feasible Mathematics* II, P. Clote and J. Remmel (eds.), Birkhäuser, Boston (1995), pp. 154–218.
5. H. Gaifman and C. Dimitracopoulos, *Fragments of Peano's arithmetic and the MRDP theorem.* Monographie 30 de L'Enseignement Mathématique (1982), pp. 187–206.
6. A. Grzegorczyck, *Some classes of recursive functions.* — Rozpr. Mat. **4**, 1–45 (1953).
7. J. Håstad, *Almost optimal lower bounds for small depth circuits.* in: *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing* 1987, pp. 6–20.
8. P. Hájek and P. Pudlák, *Metamathematics of First-Order Arithmetics.* Springer-Verlag (1993).
9. J. P. Jones and Yu. Matiyasevich, *Register machine proof of the theorem on exponential diophantine representation.* — J. Symbolic Logic, **49**, 818–829 (1984).
10. C. F. Kent and B.R. Hodgson, *An arithmetical characterization of NP.* — Theor. Computer Sci. **21**, 255–267 (1982).
11. J. Krajíček, *Bounded Arithmetic, Propositional Logic and Complexity Theory.* Cambridge University Press (1995).
12. Ю. В. Матиясевич, *Диофантовость перечислимых множеств.* — Доклады АН СССР **191**, 279–282 (1970).
13. Ю. В. Матиясевич, *Десятая проблема Гильберта.* Физматлит, Москва (1993).
14. P. G. Odifreddi, *Classical recursion Theory.* Vol. II Elsevier (1999).
15. R. Parikh, *Existence and feasibility in arithmetic.* — J. Symbolic Logic **36**, 494–508 (1971).
16. C. Pollett, *Structure and definability in general bounded arithmetic theories.* — Annals Pure Appl. Logic **100**, 189–245 (1999).
17. C. Pollett, *Multifunction algebras and the provability of* PH $\downarrow$. — Annals Pure Appl. Logic **104**, 279–303 (2000).
18. C. Pollett, *On the Bounded Version of Hilbert's Tenth Problem.* — Arch. Math. Logic **42**, No. 3, 469–488 (2003).
19. A. A. Razborov, *Bounded arithmetic and lower bounds in Boolean complexity.* in: *Feasible Mathematics.* II, P. Clote and J. Remmel (eds.), Birkhauser (1995), pp. 344–386..
20. A. A. Razborov, *Lower bounds for propositional proofs and independence results in bounded arithmetic.* in: *Proceedings of* 20*th International Symposium on the Mathematical Foundations of Computer Science*, Springer-Verlag (1995), p. 105.
21. G. Takeuti, *RSUV isomorphisms.* in: *Arithmetic, Proof Theory, and Computational Complexity*, P. Clote and J. Krajíček (eds.), *Oxford Logic Guides* **23**, Clarendon Press, Oxford (1993), p. 364–386.

214 MacQuarrie Hall
Department of Computer Science
San Jose State University
1 Washington Square,
San Jose CA 95192
pollett@cs.sjsu.edu

Поступило 3 мая 2003 г.