Polynomial Local Search in the Polynomial Hierarchy and Witnessing in Fragments of Bounded Arithmetic

Arnold Beckmann*
Department of Computer Science
Swansea University
Swansea SA2 8PP, UK
a.beckmann@swansea.ac.uk

Samuel R. Buss†
Department of Mathematics
University of California, San Diego
La Jolla, CA 92093-0112, USA
sbuss@math.ucsd.edu

November 5, 2008

Abstract

The complexity class of Π^p_k-polynomial local search (PLS) problems is introduced and is used to give new witnessing theorems for fragments of bounded arithmetic. For $1 \leq i \leq k+1$, the Σ^p_i-definable functions of T^{k+1}_2 are characterized in terms of Π^p_k-PLS problems. These Π^p_k-PLS problems can be defined in a weak base theory such as S^1_2, and proved to be total in T^{k+1}_2. Furthermore, the Π^p_k-PLS definitions can be skolemized with simple polynomial time functions, and the witnessing theorem itself can be formalized, and skolemized, in a weak base theory. We introduce a new $\forall\Sigma^b_1(\alpha)$-principle that is conjectured to separate $T^k_2(\alpha)$ and $T^{k+1}_2(\alpha)$.

1 Introduction

This paper discusses the Σ^b_i-definable functions of the fragments T^{k+1}_2 of bounded arithmetic, for $1 \leq i \leq k+1$, and characterizes these functions in terms of Π^p_k-PLS problems. The Π^p_k-PLS problems are defined in this paper as a kind of polynomial local search, relative to a Π^p_k-definable set of feasible points.

These results complement a number of prior results on the definable functions of bounded arithmetic. For $k \geq 1$, the second author [4] characterized the Σ^b_k-definable functions of S^k_2 as being precisely the functions in

*Supported in part by EPSRC grant EP/D03809X/1.
†Supported in part by NSF grant DMS-0700533.

the class $FP^{\Sigma^p_{k-1}}$. Krajíček [11] proved that the Σ^b_k-definable functions of S^{k-1}_2 are precisely the functions computable by polynomial time algorithms that make $O(\log n)$ witness queries to a Σ^p_{k-1}-oracle. Buss and Krajíček [8] proved that the Σ^b_1-definable functions of T^1_2 are precisely the functions that are (projections of) polynomial local search (PLS) problems. The class PLS was defined by Johnson, Papadimitriou, and Yannakakis [9]. As a number of researchers have noted, this can be generalized to describe the Σ^b_k-definable functions of T^k_2 in terms of the class $\mathrm{PLS}^{\Sigma^p_{k-1}}$, which is defined by replacing the polynomial-time predicates and functions of the class PLS with predicates and functions from $P^{\Sigma^p_{k-1}}$. Since S^{k+1}_2 is $\forall\Sigma^b_{k+1}$-conservative over T^k_2 [5], this also provides a characterization of the Σ^b_k-definable functions of S^{k+1}_2 for $k \geq 1$.

The problem of determining the Σ^b_i-definable functions of T^k_2 (equivalently, of S^{k+1}_2) for $0 < i < k$ has been more difficult, but a couple recent advances have been achieved. Krajíček, Skelley, and Thapen [10] characterized the Σ^b_1-definable functions of T^2_2 in terms of colored PLS problems. They also gave characterizations of the Σ^b_1-definable functions of T^3_2 in terms of a reflection principle, as well as in terms of a kind of recursion called *verifiable recursion*. Skelley and Thapen [20] subsequently gave a characterization of the Σ^b_1-definable functions of T^k_2, for all $k \geq 2$, based on a combinatorial principle for k-turn games. An earlier, more complex, game characterization of the same functions was given by Pudlák [19] using a combinatorial analysis of Herbrand disjunctions.

The present paper gives a characterization of the Σ^b_i-definable functions of T^{k+1}_2 (and hence of its Σ^b_{k+2}-conservative extension S^{k+2}_2) for all $0 < i \leq k+1$, using a relativized notion of polynomial local search problems. Our relativized PLS problems use *polynomial time* computable cost and neighborhood functions; however the set of feasible points can have higher computational complexity. In particular, the class of Π^p_k-PLS problems uses a Π^p_k-predicate to define the set of feasible points. The stopping condition (called the "goal") is defined by a Π^p_{i-1}-predicate. Our first main result states that the (multi)functions which are Σ^b_i-definable in T^{k+1}_2 are precisely the (multi)functions that can be defined as a projection of a Π^p_k-PLS problem with Π^p_{i-1}-goal. (A multifunction is a total relation denoted as a function $y = f(x)$, but allowing a single x to have more than one value for $y = f(x)$.) This is proved by a witnessing lemma, Lemma 5, showing that a T^{k+1}_2-provable sequent of Σ^b_{k+1}-formulas can be witnessed by a Π^p_k-PLS problem. Indeed, S^1_2 can define the Π^p_k-PLS problems, and can prove the that the witnessing property holds. It is important to note

though, that although S^1_2 can define the Π^b_k-PLS problems for all $k \geq 0$, it presumably cannot prove that solutions always exist (otherwise, T^{k+1}_2 would be $\forall\Sigma^b_1$-conservative over S^1_2).

Our second main set of results concern Skolemization. We prove that the Π^p_k-PLS problems used for the witnessing lemma can be defined in the weak base theory S^1_2 in *Skolem form*: this means that the defining properties can be proved in a Skolemized form where the Skolem functions are simple polynomial time computable functions. In addition, Lemma 9 and Theorem 3 give stronger versions of the witnessing properties; namely, the witnessing theorem itself can be proved in Skolemized form.

The paper concludes by using the Skolemized Π^p_k-PLS problems to propose a relativized formula in $\forall\Sigma^b_1(\alpha)$ which is provable in $T^{k+1}_2(\alpha)$ but is conjectured to not be provable in $T^k_2(\alpha)$. Using the Paris-Wilkie translation, this conjecture can be translated into the setting of bounded-depth propositional logic. Namely, we describe sets Ξ_a so that, for $a \in \mathbb{N}$, Ξ_a is a set of sequents of literals. The sets Ξ_a have polynomial size refutations of depth $k-1$ in the Tait-style propositional sequent calculus, but are conjectured to not have quasipolynomial refutations of depth $k - 1\frac{1}{2}$.

The initial work on the results of the present paper was carried out by the first author working in the setting of proof notations to extend the work of [1]. The complete results that are reported below and in [3] were then obtained during a visit to San Diego and in subsequent work. The paper [3] is a companion paper to the present paper, obtaining similar results using proof notations.

2 Π^p_k-polynomial local search problems.

We define a Π^p_k-polynomial local search problem to be a local search problem with a neighborhood function N and a cost function c which are computable in polynomial time, and with a Π^p_k-condition F that defines the intended domain of the search. This is defined formally as follows.

Definition A Π^p_k-PLS problem, with input x, consists of the following:

(1) A polynomial time computable *neighborhood* function $N(x, s)$.

(2) A polynomial time computable, integer valued, *cost* function $c(x, s)$.

(3) A Π^p_k-predicate $F(x, s)$ which defines, for each x, the set $F(x) := \{s : F(x, s)\}$ of *feasible points*. The set of feasible points for an input x must be polynomially bounded, with $F(x, s)$ implying that $|s| \leq d(|x|)$ for some given polynomial d.

(4) A polynomial time *initial point* function $i(x)$ such that $i(x)$ is always a feasible point, i.e., $i(x) \in F(x)$ must hold.

Furthermore, a Π^p_k-PLS problem must satisfy the following four defining conditions (α)-(δ). The first two conditions were already stated above. The third condition, (γ), states that the neighborhood function maps feasible points to feasible points. The fourth condition, (δ), states that the neighborhood function always produces the same point or produces a lower cost point.

(α) $\forall x \forall s(F(x,s) \rightarrow |s| \leq d(|x|))$.

(β) $\forall x(F(x, i(x)))$.

(γ) $\forall x \forall s(F(x,s) \rightarrow F(x, N(x,s)))$.

(δ) $\forall x \forall s(N(x,s) = s \vee c(x, N(x,s)) < c(x,s))$.

The input to the Π^p_k-PLS problem is a value x; a *solution* is a value $s \in F(x)$ such that $N(x,s) = s$.

Let $\mathcal{P}$ be a Π^p_k-PLS problem. Any instance $\mathcal{P}(x)$ must have at least one solution. Indeed, one way to find a solution is start with $s = i(x)$ and iterate the function $s \mapsto N(x,s)$. The conditions (γ) and (δ) ensure that values s remain in $F(x)$ and that the costs $c(s)$ are decreasing. Therefore, a fixed point $s = N(x,s) \in F(x)$ will eventually be reached.

Since F is a Π^p_k-property, the computational complexity of recognizing a valid solution s to $\mathcal{P}(x)$ is, in general, in the class Π^p_k of the polynomial hierarchy. We shall often wish to consider Π^p_k-PLS problems with a lower computational complexity for solutions. For this, we let $0 \leq g \leq k$ and define a *Π^p_k-PLS problem with Π^p_g-goal* to be a Π^p_k-PLS problem with an additional Π^p_g-predicate $G(x,s)$ such that the condition (ϵ) holds:

(ϵ) $\forall x \forall s(G(x,s) \leftrightarrow [F(x,s) \wedge N(x,s) = s])$.

For a Π^p_k-PLS problem with Π^p_g-goal, the property of s being a solution is just the condition $G(x,s)$, and thus is a Π^p_g condition.

In the case of $g = 0$, the class Π^p_g equals P. Hence, the solutions to a Π^p_k-PLS problem with Π^p_0-goal are polynomial time recognizable.

Formalized Π^p_k-PLS problems The definitions of Π^p_k-PLS problems can be readily formalized in a weak fragment of arithmetic. In the present paper, we use S^1_2 as the weak fragment; however, the details of the constructions make it clear that even weaker theories could be used.*

Definition A Π^p_k-PLS problem with Π^p_g-goal is *formalized in* S^1_2 provided

(a) The functions N, i, and c are Σ^b_1-defined by S^1_2.

(b) The predicate F is given by a Π^b_k-formula.

(c) The predicate G is given by a Π^b_g-formula if $g > 0$, or by a Δ^b_1-formula if $g = 0$.

(d) The defining conditions (α)-(ϵ) are provable in S^1_2.

A Π^p_k-PLS problem that is formalized in S^1_2 will sometimes be called a Π^b_k-PLS problem for short (with superscript "b" instead of "p").

Note that S^1_2 can formalize many Π^p_k-PLS problems, but as far as is known, it cannot prove they all have solutions. Instead, we will generally use T^{k+1}_2 to prove the existence of solutions to Π^p_k-PLS problems.

Definition Let a Π^p_k-PLS problem $\mathcal{P}$ be formalized in S^1_2. Then T^{k+1}_2 *proves that* $\mathcal{P}$ *is total* provided that T^{k+1}_2 proves $\forall x \exists s(N(x,s) = s \wedge F(x,s))$.

From (ϵ), it follows that if $\mathcal{P}$ has a Π^p_g-goal, then T^{k+1}_2 *proves* $\mathcal{P}$ *is total* if and only if T^{k+1}_2 proves $\forall x \exists s(G(x,s))$.

Theorem 1 *Let $k \geq 1$ and suppose $\mathcal{P}$ is a Π^p_k-PLS problem which is formalizable in S^1_2. Then T^{k+1}_2 proves that $\mathcal{P}$ is total.*

Proof We argue inside T^{k+1}_2. Suppose x is arbitrary. Since the Σ^b_{k+1}-minimization axioms are consequences of T^{k+1}_2, there is a least value c_0 satisfying

$$\exists s \leq 2^{d(|x|)}(c_0 = c(x,s) \wedge F(x,s)). \tag{1}$$

Choosing any $s_0 \in F(x)$ with $c_0 = c(x,s_0)$, it follows from (δ) that $N(x,s_0) = s_0$, and the theorem is proved. □

*Using S^1_2 (or PV) as a base theory is a good choice in part since it corresponds to the polynomial time complexity of the neighborhood function N, the initial point function i, and the cost function c. However, our constructions also work in weaker settings where N, i and c lie in some lower complexity class such as the log time hierarchy; in this case, we could replace S^1_2 by a base theory that corresponds to a correspondingly simple computational class.

Theorem 2 states that the converse holds as well. Informally, if $y = f(x)$ is a multifunction which is Σ^b_{g+1}-defined by T^{k+1}_2, then there is a Π^b_k-PLS problem with Π^b_g-goal which can be used to define f. For the theorem, let $s \mapsto (s)_0$ denote the projection function so that if s codes a pair $s = \langle a, b\rangle$, then $(s)_0 = a$.

Theorem 2 *Let $k \geq 0$, and $0 \leq g \leq k$. Suppose $A(x, y)$ is a Σ^b_{g+1}-formula and*

$$T^{k+1}_2 \vdash (\forall x)(\exists y)A(x, y).$$

Then there is a Π^b_k-PLS problem $\mathcal{P}$ with Π^b_g-goal G such that S^1_2 proves

$$\forall x \forall s(G(x, s) \rightarrow A(x, (s)_0)).$$

Note the theorem does not imply that S^1_2 can prove that the Π^b_k-PLS problem $\mathcal{P}$ has a solution s for all x. Rather, S^1_2 proves that if there is a solution s satisfying $G(x, s)$, then s provides a witness for the existentially quantified y. Of course, by Theorem 1, T^{k+1}_2 can prove that $\mathcal{P}(x)$ has a solution for all x. But this is, in general, not known for S^1_2.

The case $k = g = 0$ of the theorem is the same as the PLS witnessing theorem for Σ^b_1-defined functions of T^1_2 [8]. We prove Theorem 2 in Section 4. Its proof will be based on a witnessing theorem for sequents of Σ^b_{k+1}-formulas that are provable in T^{k+1}_2.

Strict formulas and bounded arithmetic. A central fact about S^1_2 is that it can Σ^b_1-define exactly the polynomial time functions, and furthermore, S^1_2 can be conservatively extended to a theory $S^1_2(PV)$ that includes all polynomial time functions in its language [4]. We shall work with a fragment of $S^1_2(PV)$, denoted $\hat{S}^1_2$ in the present paper, which is tailored for working with prenex formulas. The theory $\hat{S}^1_2$ was introduced by Pollett [17, 18] and its language, $\hat{L}$, is obtained by extending S^1_2 to include the Σ^b_1-defined function symbols for "most significant part", MSP, and "restricted subtraction", $\dot{-}$. The theory $\hat{S}^1_2$ is strong enough to define versions of the pairing and sequence coding functions that are defined by terms (instead of being just Σ^b_1-defined). One big advantage of working with $\hat{S}^i_2$ and $\hat{T}^i_2$ is that it allows us to assume that free-cut free proofs contain only strict Σ^b_i-formulas (as defined in the next paragraph), and this simplifies the proofs of witnessing theorems by reducing the number of cases to be considered.

A *strict* $\Sigma^b_{=k}$-formula is an $\hat{L}$-formula of the form

$$(\exists x_1 \leq s_1)(\forall x_2 \leq s_2) \cdots (Qx_k \leq s_k)(\overline{Q}y \leq |t|)A(x_1, x_2, \ldots, x_k, y, \vec{a}), \qquad (2)$$

where the quantifiers alternate between existential and universal, and where A is quantifier-free. Strict $\Pi^b_{=k}$-formulas are defined similarly, reversing the roles of universal and existential quantifiers. A *strict* Σ^b_k-formula is defined to be a formula which is strict $\Sigma^b_{=k}$, or is strict $\Sigma^b_{=\ell}$ or strict $\Pi^b_{=\ell}$ for some $\ell < k$, or is quantifier-free.

The theory $\hat{S}^1_2$ is defined to have the length induction (LIND) axioms for strict Σ^b_1 formulas. $\hat{S}^1_2$ is able to prove that any Σ^b_1-formula is equivalent to a strict Σ^b_1-formula, and thus $\hat{S}^1_2$ contains S^1_2. Furthermore, $\hat{S}^1_2$ is conservative over S^1_2.

The theories $\hat{S}^i_2$ and $\hat{T}^i_2$ are defined similarly. Both theories use the same language $\hat{L}$ and the basic (open) axioms as $\hat{S}^1_2$. In addition, $\hat{S}^i_2$ has LIND for strict Σ^b_i-formulas, and $\hat{T}^i_2$ has induction (IND) for the same formulas. The two theories conservatively extend S^i_2 and T^i_2, respectively, and they prove that, for $k \leq i$, any Σ^b_k-formula is equivalent to a strict Σ^b_k-formula. When proving witnessing theorems for T^{k+1}_2, we will be able to assume, via free cut elimination, that every formula in a $\hat{T}^{k+1}_2$-proof is a strict Σ^b_{k+1}-formula.

Sequence coding. It is well-known that S^1_2 can define Gödel sequence coding functions based on efficient representations of sequences. If $w \geq 0$ codes a sequence, we write $(w)_i$ for the i-th entry in w, starting with $i = 0$. That is, $w = \langle (w)_0, (w)_1, \ldots, (w)_n \rangle$, where the length of w, denoted $\mathit{Len}(w)$, is equal to $n+1$. The binary function $*$ is used to concatenate two sequences. We often use the letter $\mathfrak{a}$ or $\mathfrak{b}$ to denote a tuple, or sequence, of values. For $i \geq 0$, we write $\mathfrak{a}_i$ for the i-th element of the tuple $\mathfrak{a}$. The notation $\langle \mathfrak{a} \rangle$ indicates the Gödel number of the sequence, namely the code $\langle \mathfrak{a}_0, \mathfrak{a}_1, \ldots, \mathfrak{a}_n \rangle$ where $\mathfrak{a}$ has $n+1$ elements. Thus, $\langle \mathfrak{a} \rangle * \langle \mathfrak{b} \rangle$ indicates the Gödel number of the sequence containing the elements of $\mathfrak{a}$ followed by the elements of $\mathfrak{b}$; this can also be more succinctly denoted as $\langle \mathfrak{a}, \mathfrak{b} \rangle$.

As already mentioned, in the strict variants of bounded arithmetic with the MSP and $\dot{-}$ functions, it is often possible to use terms to denote the needed sequence coding functions including the binary functions $(w)_i$ and $v * w$. For this, it is enough if the sequence has fixed length entries, possibly padded with leading zeros [17, 18].

It is important that the Gödel numbers for sequences are efficient encodings. In particular, in our constructions, the feasible solutions s for Π^b_k-PLS problems will be sequences. To make sequence coding simpler and efficient, the feasible solutions for any fixed Π^b_k-PLS problem $\mathcal{P}$ will usually be required to be sequences of a fixed length. In addition, the entries will be polynomially bounded by the input x to $\mathcal{P}$. This will ensure that it

is possible to pick a polynomial d so that condition (α) is satisfied, provably in S^1_2.

Skolemized PLS problems. We now discuss the formalization of Π^p_k-PLS problems that use Skolemized versions of the principles (α)-(ϵ). Since the proof of Theorem 2 does not use Skolemized PLS problems, Skolemized PLS problems will not be considered again until Section 5. Thus, the reader may wish to skip the rest of the present section on first reading.

Suppose that $\hat{S}^1_2$ proves a strict formula

$$(\forall \vec{a})(\exists x_1 \leq s_1)(\forall x_2 \leq s_2) \cdots (Qx_\ell \leq s_\ell) A(x_1, x_2, \ldots, x_\ell, \vec{a}), \tag{3}$$

where A is a quantifier-free formula and where, w.l.o.g., the terms s_j do not contain any of the variables x_i. In some cases, $\hat{S}^1_2$ can prove a Skolemized form of (3); namely, there may be $\hat{L}$-terms $t_i(\vec{a}, x_2, x_4, \ldots, x_{i-1})$ for i odd, such that $\hat{S}^1_2$ proves

$$\begin{aligned}(\forall \vec{a})(\forall x_2 \leq s_2)(\forall x_4 \leq s_4) \cdots (\forall x_{\ell-1} \leq s_{\ell-1}) \qquad (4)\\ [t_1(\vec{a}) < s_1 \wedge t_3(\vec{a}, x_2) < s_3 \wedge \cdots \wedge t_\ell(\vec{a}, x_2, x_4, \ldots, x_{\ell-1}) < s_\ell \wedge \\ A(t_1(\vec{a}), x_2, t_3(\vec{a}, x_2), x_4, t_5(\vec{a}, x_2, x_4), \ldots, t_\ell(\vec{a}, x_2, x_4, \ldots, x_{\ell-1}), \vec{a})],\end{aligned}$$

where here the notation assumes ℓ is odd so that Qx_ℓ is an existential quantifier. (For ℓ even, the definition is modified in the obvious fashion, namely with the same definition, but letting $A(\cdots)$ incorporate the last universal quantifier.) Note that the Skolemized formula (4) logically implies (3). The converse is, of course, not always true. However, we prove later that, in many situations, $\hat{S}^1_2$ can prove Skolemized versions of the conditions (α)-(ϵ) that define a Π^b_k-PLS problem.

When Skolemizing a Π^b_k-PLS problem, we will always be in the situation that the functions N and c are defined by $\hat{L}$-terms, and that the predicates $F(x, s)$ and $G(x, s)$ are strict Π^b_k- and strict Π^b_g-formulas, respectively. To Skolemize the formulas (α), (β) and (ϵ), we first put them in prenex form. There is a unique natural way to put (α) and (β) in prenex form, namely, pulling out the quantifiers in F one at a time. The equation (ϵ) needs to be rewritten before it can be Skolemized, since the $\leftrightarrow$ connective is neither monotone nor antimonotone in its arguments. Thus, (ϵ) must be replaced by the two formulas

(ϵ') $\forall x \forall s(G(x, s) \rightarrow [F(x, s) \wedge N(x, s) = s])$.

(ϵ'') $\forall x \forall s([F(x, s) \wedge N(x, s) = s] \rightarrow G(x, s))$.

The formulas (γ), (ϵ') and (ϵ'') are universal closures of boolean combinations of strict Σ^b_k- and Π^b_k-formulas. These must be converted to prenex form before they can be Skolemized. The prenex form of (γ) is chosen conservatively, as follows. The *level* of an bounded quantifier $(\exists x{\leq}t)$, respectively $(\forall x{\leq}t)$, is defined to equal i if the quantifier is the outermost connective of a strict $\Sigma^b_{=i}$-subformula, respectively of a strict $\Pi^b_{=i}$-formula. A bounded quantifier is called *essentially existential* if when prenex operations are applied, the quantifier becomes (or, remains) existential. Otherwise, the quantifier is *essentially universal*. Boolean combinations of strict formulas are converted to prenex form by using prenex operations to move quantifiers one at a time to the front of the formula, bringing quantifiers to the front in order of ϵ-level (highest level first, of course), and for quantifiers at a given level i, bringing out the essentially universal quantifiers before the essentially existential ones.

The prenex forms of (ϵ') and (ϵ'') are chosen a bit differently. For these, we match up quantifiers level-by-level, starting with the outer quantifiers. Specifically, let a Π^b_k-PLS problem with Π^b_g goal be given. A quantifier at level i in G is defined to have *ϵ-level* equal to $i+k-g$. This means the outermost quantifier in G has ϵ-level k. For a quantifier in F, the ϵ-level is just equal to its level. Then, (ϵ') and (ϵ'') are converted to prenex form by bringing out quantifiers in order of ϵ-level, essentially universal ones before essentially existential ones.

As an example of prenexification, suppose $F(x,s)$ is the formula

$$(\forall y_1{\leq}t_1)(\exists y_2{\leq}t_2)(\forall y_3{\leq}t_3)\cdots F_0(\vec{y},x,s).$$

Then the prenexification of (γ) is

$$\begin{aligned}(\forall y'_1{\leq}t'_1)(\exists y_1{\leq}t_1)(\forall y_2{\leq}t_2)(\exists y'_2{\leq}t'_2)(\forall y'_3{\leq}t'_3)(\exists y_3{\leq}t_3)\cdots \\ \cdots(F_0(\vec{y},x,s)\rightarrow F_0(\vec{y}',N(x,s),s)),\end{aligned} \tag{5}$$

where the terms t'_i are the same as t_i but with variables y_j replaced by variables y'_j.[†]

Section 5 will discuss how to Skolemize Π^b_k-PLS problems in more detail.

Definition A Π^p_k-PLS problem with Π^p_g-goal is *formalized in Skolem form in* $\hat{S}^1_2$ provided

(a) The functions N, i, and c are all defined by $\hat{L}$-terms,

[†]One could also assume, without loss of generality, that the terms t_i do not involve the variables y_j. In that case, t_i and t'_i are the same term.

(b) The predicates F and G are given by strict Π^b_k- and strict Π^b_g-formulas, respectively,

(c) Skolemized versions of the defining conditions (α)-(δ), (ϵ'), and (ϵ'') are provable in $\hat{S}^1_2$, where the Skolem functions are given by $\hat{L}$-terms.

The earlier theorem applies also to Π^b_k-PLS problems formalized in Skolem form:

Theorem 3 *Let $k \geq 0$, and $0 \leq g \leq k$. Suppose A is a Σ^b_{g+1}-formula and*

$$T^{k+1}_2 \vdash (\forall x)(\exists y)A(x, y).$$

Then there is a Π^b_k-PLS problem $\mathcal{P}$ with Π^b_g-goal G which is formalized in Skolem form in $\hat{S}^1_2$, such that $\hat{S}^1_2$ proves that

$$\forall x \forall s(G(x, s) \rightarrow A(x, (s)_0)). \tag{6}$$

Furthermore, there is a Skolemization of (6)*, with $\hat{L}$-terms as Skolem functions, which is provable in $\hat{S}^1_2$.*

Theorem 3 will be proved in Section 5.

3 Constructions of Π^b_k-PLS problems

As preparation for the proofs of Theorems 2 and 3, this section introduces several constructions for composing Π^b_k-PLS problems, and defines Π^b_k-PLS problems for deciding Π^p_k-properties.

We adopt the following conventions for feasible elements $s \in F(x)$. The purpose of these conventions is to aid the modular design of Π^b_k-PLS problems, especially of Π^b_k-PLS problems that define functions or multifunctions. When designing a Π^b_k-PLS problem $\mathcal{P}$, we shall ensure that any $s \in F(x)$ codes a sequence of length exactly ℓ for some fixed ℓ that depends on $\mathcal{P}$. Furthermore, s will have length > 2 and be equal to $\langle x, y, \ldots \rangle$, where x is the input value. Then we always have $(s)_0 = x$ by convention, so that s specifies explicitly the input x. This allows us to simplify the notations for the neighborhood and cost functions by defining $N(s) = N((s)_0, s)$ and $c(s) = c((s)_0, s)$. Furthermore, if s is a solution to $\mathcal{P}$, so that $N(s) = s$ and $s \in F(x)$, then the value $y = (s)_1$ will be the *output* of $\mathcal{P}(x)$.

This last convention allows us to regard $\mathcal{P}$ as a multifunction $x \mapsto y$. In general, $\mathcal{P}$ defines only a multifunction rather than a function, since there may be multiple solutions to $\mathcal{P}(x)$ and hence multiple possible values $y = (s)_1$ for solutions s. We write $y = \mathcal{P}(x)$ to denote that y is one of the possible output values for $\mathcal{P}(x)$; in other words,

$$y = \mathcal{P}(x) \;\Leftrightarrow\; \exists s(F(x,s) \wedge N(s) = s \wedge y = (s)_1).$$

Since condition (α) implies that the set $F(x)$ of feasible points is polynomially bounded, and since the cost function c is polynomial time computable, we can assume w.l.o.g. that every Π^p_k-PLS problem has associated polynomial bounds $\mathrm{maxc}(x)$ and $\mathrm{maxout}(x)$ such that $c(x,s) < \mathrm{maxc}(x)$ and such that the output value y satisfies $y < \mathrm{maxout}(x)$. Both $\mathrm{maxc}(x)$ and $\mathrm{maxout}(x)$ can be taken to be strictly increasing functions; in fact they can be taken to be of the form $2^{p(|x|)}$ for some polynomial p with non-negative integer coefficients. Indeed, w.l.o.g., $\mathrm{maxout}(x) = 2^{d(|x|)} \geq x$.

Polynomial time functions as Π^b_k-PLS problems. Let $y = f(x)$ be a polynomial time function. For $k \geq 0$, f can be coded by a Π^b_k-PLS problem as follows. The initial function is defined as $i(x) = \langle x, f(x)\rangle$. $F(x,s)$ is defined to hold iff $s = \langle x, f(x)\rangle$. The neighborhood functions is simply $N(x,s) = s$, and the cost function is $c(x,s) = 0$. It is easy to check that this defines a Π^b_k-PLS problem such that the unique output possible for $\mathcal{P}(x)$ is the value $y = f(x)$.

Combining Π^b_k-PLS problems. The *composition* of two PLS problems, $\mathcal{P} = \mathcal{P}_2 \circ \mathcal{P}_1$, is defined so that $y = \mathcal{P}(x)$ iff there is a y_1 so that $y_1 = \mathcal{P}_1(x)$ and $y = \mathcal{P}_2(y_1)$. The *pairing* of two PLS problems $\mathcal{P} = \langle \mathcal{P}_1, \mathcal{P}_2\rangle$ is defined by requiring that $y = \mathcal{P}(x)$ holds iff $y = \langle y_1, y_2\rangle$ for some $y_1 = \mathcal{P}_1(x)$ and some $y_2 = \mathcal{P}_2(x)$.

Composition and pairing, and other similar constructions, can be unified into a single construction we call *fg*-combination. Let f and g be polynomial time functions. The *fg-combination* of $\mathcal{P}_1$ and $\mathcal{P}_2$ is defined by

$$\mathcal{P}(x) = f(\langle \mathcal{P}_1(x), \mathcal{P}_2(g(x, \mathcal{P}_1(x)))\rangle),$$

where the two occurrences of $\mathcal{P}_1(x)$ must denote the same value. Namely, $\mathcal{P}$ is the multifunction defined so that $y = \mathcal{P}(x)$ holds iff there is some $u = \mathcal{P}_1(x)$ and some $v = \mathcal{P}_2(g(x,u))$ such that $y = f(\langle u, v\rangle)$. By choosing f and g appropriately, it is easy to use *fg*-combination to define the

composition and the pairing of $\mathcal{P}_1$ and $\mathcal{P}_2$. As another simple example of the power of *fg*-combination, recall that the *Cond* function is defined by

$$Cond(x, y, z) \;=\; (1 \dot{-} x) \cdot y + (1 \dot{-} (1 \dot{-} x)) \cdot z,$$

so that $Cond(x, y, z)$ equals y if $x = 0$ and equals z otherwise. Then, $\mathcal{P}(x) = Cond(\mathcal{P}_1(x), \mathcal{P}_2(x), \mathcal{P}_3(x))$ can be defined by using pairing to define $\mathcal{Q} = \langle P_2, P_3 \rangle$, and then setting $\mathcal{P} = f(\langle P_1, Q \rangle)$, where f is the polynomial time function $f(u) = Cond((u)_0, ((u)_1)_0, ((u)_1)_1)$. The latter step is a use of *fg*-combination with $g(x, y) = x$.

Suppose $\mathcal{P}_1$ and $\mathcal{P}_2$ are Π^p_k-PLS problems. Their *fg*-combination is formally defined as a Π^p_k-PLS problem as follows. For $\ell = 1, 2$, let $\mathcal{P}_\ell$ be defined in terms of i_ℓ, N_ℓ, c_ℓ, d_ℓ, and F_ℓ. We define the feasible set $F(x)$ for the *fg*-combination $\mathcal{P}$ of $\mathcal{P}_1$ and $\mathcal{P}_2$ so that

$$\begin{aligned}
\langle x, 0, 0, \mathfrak{a}, \vec{0} \rangle \in F(x) \quad &\Leftrightarrow \quad F_1(x, \langle \mathfrak{a} \rangle) \\
\langle x, 0, 1, \mathfrak{a}, \mathfrak{b} \rangle \in F(x) \quad &\Leftrightarrow \quad F_1(x, \langle \mathfrak{a} \rangle) \wedge N_1(\langle \mathfrak{a} \rangle) = \langle \mathfrak{a} \rangle \wedge \mathfrak{b}_0 = g(x, \mathfrak{a}_1) \wedge F_2(\langle \mathfrak{b} \rangle) \\
\langle x, y, 2, \mathfrak{a}, \mathfrak{b} \rangle \in F(x) \quad &\Leftrightarrow \quad F_1(x, \langle \mathfrak{a} \rangle) \wedge N_1(\langle \mathfrak{a} \rangle) = \langle \mathfrak{a} \rangle \wedge \mathfrak{b}_0 = g(x, \mathfrak{a}_1) \wedge F_2(\langle \mathfrak{b} \rangle) \\
&\qquad \wedge N_2(\langle \mathfrak{b} \rangle) = \langle \mathfrak{b} \rangle \wedge y = f(\langle \mathfrak{a}_1, \mathfrak{b}_1 \rangle).
\end{aligned}$$

and so that $s \notin F(x)$ for all other s. The intuitive meaning of the above definition of $F(x)$ is that a feasible point $s = \langle x, y, z, \mathfrak{a}, \mathfrak{b} \rangle$ either has (a) $z = 0$ and $\mathfrak{a}$ is a feasible point for $\mathcal{P}_1(x)$, or (b) $z = 1$ and $\mathfrak{b}$ is a feasible point for $\mathcal{P}_2(g(x, \mathcal{P}_1(x)))$, or (c) $z = 2$ and y is the output value. In the first case, (a), $\mathfrak{b} = \vec{0}$ is used as padding so that all feasible points are sequences of the same length.

The initial point function for $\mathcal{P}$ is defined by $i(x) = \langle x, 0, 0 \rangle * i_1(x) * \langle \vec{0} \rangle$. The neighborhood function $N(s)$ is defined in terms $N_1(s)$ and $N_2(s)$ so as to satisfy:

$$\begin{aligned}
N(\langle x, 0, 0, \mathfrak{a}, \vec{0} \rangle) \;&=\; \begin{cases} \langle x, 0, 0 \rangle * N_1(\langle \mathfrak{a} \rangle) * \langle \vec{0} \rangle & \text{if } N_1(\langle \mathfrak{a} \rangle) \neq \langle \mathfrak{a} \rangle \\ \langle x, 0, 1, \mathfrak{a} \rangle * i_2(g(x, \mathfrak{a}_1)) & \text{if } N_1(\langle \mathfrak{a} \rangle) = \langle \mathfrak{a} \rangle \end{cases} \\
N(\langle x, 0, 1, \mathfrak{a}, \mathfrak{b} \rangle) \;&=\; \begin{cases} \langle x, 0, 1, \mathfrak{a} \rangle * N_2(\langle \mathfrak{b} \rangle) & \text{if } N_2(\langle \mathfrak{b} \rangle) \neq \langle \mathfrak{b} \rangle \\ \langle x, f(\mathfrak{a}_1, \mathfrak{b}_1), 2, \mathfrak{a}, \mathfrak{b} \rangle & \text{if } N_2(\langle \mathfrak{b} \rangle) = \langle \mathfrak{b} \rangle \end{cases} \\
N(\langle x, y, 2, \mathfrak{a}, \mathfrak{b} \rangle) \;&=\; \langle x, y, 2, \mathfrak{a}, \mathfrak{b} \rangle.
\end{aligned}$$

Let $\overline{g}(a, b)$ be an $\hat{L}$-term so that $\overline{g}$ dominates g in that sense that $\overline{g}(a, b) \geq g(a', b')$ whenever $a \geq a'$ and $b \geq b'$. The cost function for $\mathcal{P}$ is defined so

that

$$\begin{aligned} c(\langle x,0,0,\mathfrak{a},\vec{0}\rangle) &= 1+\mathrm{maxc}_2(\overline{g}(x,\mathrm{maxout}_1(x)))+c_1(\langle\mathfrak{a}\rangle) \\ c(\langle x,0,1,\mathfrak{a},\mathfrak{b}\rangle) &= 1+c_2(\langle\mathfrak{b}\rangle) \\ c(\langle x,y,2,\mathfrak{a},\mathfrak{b}\rangle) &= 0. \end{aligned}$$

It is straightforward to check that $\mathcal{P}$ is indeed a Π^p_k-PLS problem with conditions (α)-(ϵ) all satisfied. Furthermore, the entire construction can be formalized in S^1_2. That is to say, if $\mathcal{P}_1$ and $\mathcal{P}_2$ are formalized Π^b_k-PLS problems, then so is $\mathcal{P}$.

Pseudo-iteration of Π^b_k-PLS problems. The proofs of Theorems 2 and 3 will be based on witnessing lemmas, and the crucial step for the proofs of the witnessing lemmas uses iteration of Π^b_k-PLS functions to handle the case of an induction inference. Given a Π^b_k-PLS problem $\mathcal{P}_1$, it is entirely straightforward to define a Π^b_k-PLS problem $\mathcal{P}$ that computes a function defined from $\mathcal{P}_1$ by limited iteration *on notation*. This, however, is not sufficient for our purposes; instead we must define iterations of exponential length.

The problem with defining iterations of exponential length is that feasible points in $F(x)$ are polynomially bounded, so no $s \in F(x)$ can encode the entire computation of all the steps of the iteration. Indeed, there is seemingly no way to define the "true" iteration of $\mathcal{P}_1$. Instead, we use a side property H, that is preserved by iteration of $\mathcal{P}_1$ to indirectly describe the result of an exponentially long iteration. We call this a "pseudo-iteration"' since the output values may not be obtainable by a true iteration, but rather only need to satisfy the property H. In general, for a Π^p_k-PLS problem, the side property H will be in Π^p_k.

Let $\mathcal{P}_1$ be a Π^p_k-PLS problem, $H(i,x,z)$ be a Π^p_k-predicate, and p_H a polynomial. Further suppose that

(ι_0) For all x, $H(0,x,x)$ holds.

(ι_1) For all i, x, y, if $H(i,x,y)$ holds, then $|y| \le p_H(|x|+|i|)$.

(ι_2) For all i, x, y, if $z = \mathcal{P}_1(y)$ and $H(i,x,y)$, then $H(i+1,x,z)$.

We wish to define a Π^p_k-PLS problem $\mathcal{P}$ so that when $y = \mathcal{P}(\langle m,x\rangle)$, then $H(m,x,y)$ holds. This problem $\mathcal{P}$ will be denoted by $\mathit{PsIter}[\mathcal{P}_1,H]$. The intent is that x is the input value on which $\mathcal{P}_1$ will be iterated, and m is

the number of iterations. The intuition is that we wish to compute values $y_0, y_1, \ldots, y_m$ such that $y_0 = x$, and $y_{i+1} = \mathcal{P}_1(y_i)$ for all $i \geq 0$; at the end, y_m is a desired value $y_m = \mathcal{P}(\langle m, x\rangle)$. However, (ι_0)-(ι_2) allow more generality, namely any y satisfying $H(m, x, y)$ is a valid output value for the multifunction $\mathcal{P}(\langle m, x\rangle)$. The condition (ι_0) allows the iteration to start with value x. The condition (ι_1) imposes a polynomial bound on the values obtained by iteration. The condition (ι_2) ensures that all iteration values satisfy H.

$\mathcal{P} = \mathit{PsIter}[\mathcal{P}_1, H]$ is formally defined as follows. Let $\mathcal{P}_1$ be defined using F_1, i_1, N_1, and c_1. The feasible states s for $\mathcal{P}$ will have the format $s = \langle\langle m, x\rangle, \mathfrak{a}_0, i, \mathfrak{a}\rangle$ to indicate that $\langle\mathfrak{a}\rangle$ codes a state for the computation of the i-th iteration of $\mathcal{P}_1$ on input x. A state $s = \langle\langle m, x\rangle, y, m, \mathfrak{a}\rangle$ will be used for the final state, where y is the output value. The set F of feasible points for $\mathcal{P}$ is defined so that $s \in F(\langle m, x\rangle)$ iff $s = \langle\langle m, x\rangle, y, i, \mathfrak{a}\rangle$ and

$$i \leq m \wedge H(i, x, y) \wedge [i < m \rightarrow F_1(\langle\mathfrak{a}\rangle) \wedge y = \mathfrak{a}_0]. \tag{7}$$

The initial function is defined by $i(\langle m, x\rangle) = \langle\langle m, x\rangle, x, 0\rangle * i_1(x)$. The neighborhood function N is defined so that, for $s = \langle\langle m, x\rangle, y, i, \mathfrak{a}\rangle$,

$$N(s) \;=\; \begin{cases} \langle\langle m, x\rangle, y, i\rangle * N_1(\langle\mathfrak{a}\rangle) & \text{if } i < m \text{ and } N_1(\langle\mathfrak{a}\rangle) \neq \langle\mathfrak{a}\rangle \\ \langle\langle m, x\rangle, \mathfrak{a}_1, i+1\rangle * i_1(\mathfrak{a}_1) & \text{if } i < m \text{ and } N_1(\langle\mathfrak{a}\rangle) = \langle\mathfrak{a}\rangle \end{cases}$$

and $N(s) = s$ in all other cases. Finally, the cost function is defined by defining $c(\langle\langle m, x\rangle, y, i, \mathfrak{a}\rangle)$ to equal $c_1(\langle\mathfrak{a}\rangle) + (m - i) * \mathrm{maxc}_1(2^{p_H(|m|+|x|)})$ when $i < m$, and letting $c(s) = 0$ in all other cases.

It is straightforward to check that the above definition of $\mathcal{P}$ correctly defines the pseudo-iteration of $\mathcal{P}_1$. Furthermore, if P_1 is a Π^b_k-PLS problem formalized in S^1_2 and if (ι_0)-(ι_2) are provable in S^1_2, then $\mathcal{P}$ is a Π^b_k-PLS problem formalizable in S^1_2.

Deciding Π^p_k- and Σ^p_k-properties. We next describe how a Π^b_k-PLS problem can decide the validity of a Σ^b_k-formula and, when valid, provide a witness value. Let $A(\vec{x})$ be a strict Σ^b_k-formula $A(\vec{x}) = (\exists y{\leq}t(\vec{x}))B(y, \vec{x})$. We shall define a Π^b_k-PLS problem $\mathcal{P}_A$ such that $\mathcal{P}_A(\langle\vec{x}\rangle)$ equals $\langle 0, t(\vec{x})+1\rangle$ if A is false, and equals $\langle 1, i\rangle$ if $A(\vec{x})$ is true and i is the least value such that $B(i, \vec{x})$ holds.

Similarly, for a strict Π^b_k-formula $A'(\vec{x}) = (\forall y{\leq}t'(\vec{x}))B'(y, \vec{x})$, the Π^b_k-PLS problem $\mathcal{P}_{A'}$ will be defined so that $\mathcal{P}_{A'}(\langle\vec{x}\rangle)$ equals $\langle 1, t'(\vec{x}) + 1\rangle$ if $A'(\vec{x})$ is true, and equals $\langle 0, i\rangle$ if $A'(\vec{x})$ is false with i the least value such that $B'(i, \vec{x})$ is false.

The definitions proceed by induction on $k \geq 0$. For the base case, $k = 0$, the formula A is sharply bounded, and $\mathcal{P}_A$ can be evaluated in polynomial time.

For $k \geq 1$, let $A(\vec{x})$ be the Σ^b_k-formula above. The induction hypothesis is that we have already defined $\mathcal{P}_B$, a Π^b_{k-1}-PLS problem such that $\mathcal{P}_B(\langle y, \vec{x}\rangle)$ equals $\langle i, j\rangle$ with i equal to 1 or 0 depending on whether $B(y, \vec{x})$ is true or false, respectively. We define a Π^b_k-PLS problem $\mathcal{Q}$ so that

$$\mathcal{Q}(\langle\langle\vec{x}\rangle, \langle 0, i\rangle\rangle) = \begin{cases} \langle\langle\vec{x}\rangle, \langle 0, i+1\rangle\rangle & \text{if } (\mathcal{P}_B(\langle i, \vec{x}\rangle))_0 = 0 \\ \langle\langle\vec{x}\rangle, \langle 1, i\rangle\rangle & \text{otherwise} \end{cases} \tag{8}$$

$$\mathcal{Q}(\langle\langle\vec{x}\rangle, \langle 1, i\rangle\rangle) = \langle\langle\vec{x}\rangle, \langle 1, i\rangle\rangle. \tag{9}$$

The intuition is that, by (pseudo)iterating $\mathcal{Q}$ for $(t+1)$ times, we obtain the value $\langle\langle\vec{x}\rangle, \langle 1, y\rangle\rangle$ where y is the least value $\leq t(x)$ such that $B(y, \vec{x})$ holds, or if no such y exists, we obtain $\langle\langle\vec{x}\rangle, \langle 0, t(\vec{x})+1\rangle\rangle$. The initial value for the pseudo-iteration of $\mathcal{Q}$ is $\langle\langle\vec{x}\rangle, \langle 0, 0\rangle\rangle$. Accordingly, we define $\mathcal{R}(\langle\vec{x}\rangle)$ using the $(t+1)$-fold pseudo-iteration of $\mathcal{Q}$ and composition, as

$$\mathcal{R}(\langle\vec{x}\rangle) = PsIter[\mathcal{Q}, H](\langle t(\vec{x})+1, \langle\langle\vec{x}\rangle, \langle 0, 0\rangle\rangle\rangle).$$

Then $\mathcal{P}_A$ is defined using composition by setting $\mathcal{P}_A(\langle\vec{x}\rangle) = (\mathcal{R}(\langle\vec{x}\rangle))_1$.

The side condition H for the pseudo-iteration of $\mathcal{Q}$ is defined so that

$$H(j, \langle\langle\vec{x}\rangle, \langle 0, 0\rangle\rangle, \langle\langle\vec{x}\rangle, \langle 1, i\rangle\rangle) \Leftrightarrow i < j \wedge B(i, \vec{x}) \wedge (\forall i' < i)(\neg B(i', \vec{x}))$$

$$H(i, \langle\langle\vec{x}\rangle, \langle 0, 0\rangle\rangle, \langle\langle\vec{x}\rangle, \langle 0, i\rangle\rangle) \Leftrightarrow (\forall i' < i)(\neg B(i', \vec{x})).$$

And, $H(i, u, v)$ is false for any other inputs. Note that $H \in \Pi^b_k$.

It is easy to check that this definition of $\mathcal{P}_A$ correctly decides the truth of $A(\vec{x})$ and correctly finds the minimal witness when $A(\vec{x})$ is true. It is also easy to verify that (ι_0)-(ι_2) are provable in S^1_2. Thus $\mathcal{P}_A$ is formalizable in S^1_2.

The definition of $\mathcal{P}_{A'}$ for a strict Π^b_k-formula A' is dual.

4 The Witnessing Proof

This section is devoted to the proof of Theorem 2. By Parikh's theorem [15], the value of y in the statement of Theorem 2 can be bounded by a term $t(x)$. In addition, by the equivalence of T^{k+1}_2 and $\hat{T}^{k+1}_2$, it will suffice to prove the theorem for $\hat{T}^{k+1}_2$. Thus, it will suffice to prove the following theorem:

Theorem 4 *Let $k \geq 0$, and $0 \leq g \leq k$. Suppose $A(x,y)$ is a strict Π^b_g-formula and*

$$\hat{T}^{k+1}_2 \vdash (\forall x)(\exists y \leq t) A(x,y).$$

Then there is a Π^b_k-PLS problem $\mathcal{P}$ with Π^b_g-goal G that is formalized in S^1_2, such that S^1_2 proves

$$\forall \vec{x} \forall s (G(x,s) \rightarrow A(x,(s)_0)).$$

Remark: Since the formula A is now assumed to be in Π^b_g, instead of only in Σ^b_{g+1}, Theorem 4 also holds if we replace the conclusion with $\forall \vec{x} \forall s (G(x,s) \rightarrow A(x,s))$, namely with $(s)_0$ replaced by s.

The rest of the section gives the proof of Theorem 4 and thereby of Theorem 2. Fix $k \geq 0$. The proof will be based on a witnessing lemma for sequents $\Gamma \longrightarrow \Delta$ of strict Σ^b_{k+1}-formulas.

Suppose C is a strict Σ^b_{k+1}-formula which is not in $\Pi^b_k \cup \Sigma^b_k$, so that $C(\vec{c}) = (\exists z \leq r(\vec{c})) D(z, \vec{c})$ where $D \in \Pi^b_{=k}$. Then we define $Wit_C(u, \vec{c})$ to be the Π^b_k-formula

$$u \leq r(\vec{c}) \wedge D(u, \vec{c}).$$

On the other hand, if $C \in \Pi^b_k \cup \Sigma^b_k$, we define $Wit_C(u, \vec{c})$ to be just the formula C. In this case, C is said to be *self-witnessing*.

If Γ is the antecedent $A_0, \ldots, A_{m-1}$, then $Wit_\Gamma(u, \vec{c})$ is defined to be the Π^b_k-formula which asserts that u is the code of a sequence of length m such that, for $0 \leq i < m$, $(u)_i$ witnesses A_i. Suppose the succedent Δ is $B_0, \ldots, B_{p-1}$. The witnessing predicate $Wit_\Delta(u, \vec{c})$ will be defined to state that u provides a value for i and a witness for the formula $B_i(\vec{c})$; namely, that u is a sequence of length 2, $u = \langle i, v \rangle$, and that $Wit_{B_i}(v, \vec{c})$ holds. More specifically, Wit_Δ is the Π^b_k-formula

$$\bigvee_{i=0}^{p-1} ((u)_0 = i \wedge Wit_{B_i}((u)_1, \vec{c})).$$

Lemma 5 *Let $k \geq 0$. Suppose $\hat{T}^{k+1}_2$ proves a sequent $\Gamma \longrightarrow \Delta$ containing only strict Σ^b_{k+1}-formulas, with $\vec{c}$ as free variables. Then there is a Π^b_k-PLS problem $\mathcal{P}$ which is formalized in S^1_2 such that S^1_2 proves*

$$Wit_\Gamma(u, \vec{c}) \wedge v = \mathcal{P}(\langle u, \vec{c} \rangle) \rightarrow Wit_\Delta(v, \vec{c}).$$

Proof Lemma 5 is proved by induction on the number of lines in a free-cut free sequent calculus $\hat{T}^{k+1}_2$-proof P of $\Gamma \longrightarrow \Delta$. We take the Gentzen sequent

calculus to be formalized as in [7]. Note that every formula appearing in P is a strict Σ^b_{k+1}-formula. The base case is the case where P consists of a single initial inference, which must either be a BASIC axiom, an equality axiom, or a logical initial sequent $A \to A$ with A atomic. Any of these initial sequents contains only atomic formulas, for which witnesses are trivial. In addition, any initial sequent for $\hat{T}^{k+1}_2$ is also an initial sequent for $\hat{S}^1_2$. Thus, the lemma is easily seen to hold for any initial sequent.

The induction step of the proof of Lemma 5 splits into cases based on the last inference of P. To consider a simple case first, suppose that the final inference of P is an $\vee$*:right* inference:

$$\frac{\Gamma \longrightarrow \Delta, B, C}{\Gamma \longrightarrow \Delta, B \vee C}$$

By the induction hypothesis there is a Π^b_k-PLS problem $\mathcal{Q}$ which witnesses the upper sequent, so that S^1_2 proves

$$Wit_\Gamma(u, \vec{c}) \wedge v = \mathcal{Q}(\langle u, \vec{c}\rangle) \rightarrow Wit_{\Delta,B,C}(v, \vec{c}).$$

By the free-cut free property, the formula $B \vee C$ is quantifier-free (and hence polynomial time). A witnessing function for the lower sequent can be informally defined as follows: the function is computed by first checking whether $B \vee C$ holds, and then if not, invoking $\mathcal{Q}$ to find a witness for a formula in Δ. More formally, a Π^b_k-PLS problem $\mathcal{P}$ witnessing the lower sequent can be defined in terms of $\mathcal{Q}$ by

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; \begin{cases} \langle p, 0\rangle & \text{if } B(\vec{c}) \vee C(\vec{c}) \\ \mathcal{Q}(\langle u, \vec{c}\rangle) & \text{otherwise} \end{cases}$$

where p is the number of formulas in Δ and thus $\mathcal{P}(\langle u, \vec{c}\rangle) = \langle p, 0\rangle$ serves to witness the formula $B \vee C$ when it is true.

For another example of a propositional inference, suppose the final inference of P is a $\neg$*:left* inference:

$$\frac{\Gamma \longrightarrow \Delta, A}{\neg A, \Gamma \longrightarrow \Delta}$$

Note A must be quantifier-free and thus self-witnessing. Let $\mathcal{Q}$ be the Π^b_k-PLS problem given by the induction hypothesis which witnesses the upper sequent. The Π^b_k-PLS problem $\mathcal{P}$ can be defined as

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; Q(\langle cdr(u), \vec{c}\rangle),$$

where $cdr(u) = \langle u_1, \ldots u_{m-1} \rangle$ if $u = \langle u_0, u_1, \ldots, u_{m-1} \rangle$, i.e., it equals the rest of u after the first entry. It is easy to check that S^1_2 proves that if u witnesses the antecedent $\neg A, \Gamma$ and if $v = \mathcal{P}(\langle u, \vec{c} \rangle)$, then v witnesses the succedent Δ.

The other cases where the last inference of P is a propositional inference are similar and we omit them here. Likewise, the weak structural inferences (exchange and contraction) are also quite easy; we do only the case of *Contraction:right*. In this case, the final inference of P is

$$\frac{\Gamma \longrightarrow \Delta_1, A, A, \Delta_2}{\Gamma \longrightarrow \Delta_1, A, \Delta_2}$$

Let p_1 be the number of formulas in Δ_1, and let $\mathcal{Q}$ be the Π^b_k-PLS problem for the upper sequent given by the induction hypothesis. Define $\mathcal{P}$ to witness the lower sequent by letting f be the function

$$f(\langle i, v \rangle) \;=\; \begin{cases} \langle i, v \rangle & \text{if } i \leq p_1 \\ \langle i - 1, v \rangle & \text{otherwise} \end{cases}$$

and defining $\mathcal{P}$ by composition as $\mathcal{P}(\langle u, \vec{c} \rangle) = (f \circ Q)(\langle u, \vec{c} \rangle)$.

Next we consider the quantifier inferences. Suppose the final inference of P is an $\exists \leq$*:right* inference

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

Let $\mathcal{Q}$ be the Π^b_k-PLS problem for the upper sequent as given by the induction hypothesis; we need to define $\mathcal{P}$ for the lower sequent. If $A(x)$ is in Π^b_{k-1}, then witnesses for $(\exists x \leq t) A(x)$ are trivial, and we can use composition to define $\mathcal{P}$ by $\mathcal{P}(\langle u, \vec{c} \rangle) = \mathcal{Q}(\langle cdr(u), \vec{c} \rangle)$. Here the function cdr is used to remove the unneeded witness for $s \leq t$. For the case where A is not in Π^b_{k-1}, the formula $(\exists x \leq t) A(x)$, if it needs to be witnessed, should be witnessed by the value of s. Without loss of generality, s involves only the free variables $\vec{c}$. Define

$$f(\vec{c}, \langle i, v \rangle) \;=\; \begin{cases} \langle i, v \rangle & \text{if } i < p \\ \langle p, s(\vec{c}) \rangle & \text{otherwise} \end{cases}$$

where p is the number of formulas in Δ. Then set $\mathcal{P}(\langle u, \vec{c} \rangle) = f(\vec{c}, \mathcal{Q}(\langle cdr(u), \vec{c} \rangle))$.

Next, suppose the last inference of P is a $\forall \leq$*:left* inference

$$\frac{A(s), \Gamma \longrightarrow \Delta}{s \leq t, (\forall x{\leq}t)A(x), \Gamma \longrightarrow \Delta}$$

Since the proof is free-cut free, the principal formula $(\forall x{\leq}t)A(x)$ must be in Π^b_k and thus is self-witnessing. Let $\mathcal{Q}$ be given by the induction hypothesis as the Π^b_k-PLS problem that witnesses the upper sequent. Then define $\mathcal{P}(\langle u, \vec{c}\rangle) = \mathcal{Q}(\langle \mathit{cdr}(u), \vec{c}\rangle)$. It is easy to see that $\mathcal{P}$ satisfies the desired properties.

Now suppose the final inference of P is a $\forall{\leq}$*:right* inference

$$\frac{b \leq t, \Gamma \longrightarrow \Delta, A(b)}{\Gamma \longrightarrow \Delta, (\forall x{\leq}t)A(x)}$$

where b is an eigenvariable and appears only as indicated. The induction hypothesis gives a Π^b_k-PLS problem $\mathcal{Q}(u, b, \vec{c})$ witnessing the upper sequent. We need to define $\mathcal{P}(u, \vec{c})$ witnessing the lower sequent. Of course, $\mathcal{P}$ will invoke $\mathcal{Q}$, but for this it needs a value for $b \leq t$ that makes $A(b)$ false, if any such b exists. Let $(\forall x{\leq}t)A$ be in $\Pi^b_{=\ell}$ for some $\ell \leq k$. By the construction in Section 3, there is a Π^b_ℓ-PLS problem $\mathcal{P}_{\forall A}(\langle \vec{c}\rangle)$ which either outputs $\langle 0, b\rangle$ for the least value $b \leq t$ such that $\neg A(b)$ or, if there is no such b, outputs the value $\langle 1, t+1\rangle$. The Π^b_k-PLS problem $\mathcal{P}$ that witnesses the lower sequent of the $\forall{\leq}$:right inference can now be defined by

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; \begin{cases} \langle p, 0\rangle & \text{if } \mathcal{P}_{\forall A}(\langle c\rangle) = \langle 1, t+1\rangle \\ \mathcal{Q}(\langle \langle 0\rangle * u, (\mathcal{P}_{\forall A}(\langle \vec{c}\rangle))_1, \vec{c}\rangle) & \text{otherwise} \end{cases}$$

where Δ contains p formulas.

Next consider the case where P ends with an $\exists{\leq}$*:left* inference

$$\frac{b \leq t, A(b), \Gamma \longrightarrow \Delta}{(\exists x{\leq}t)A(x), \Gamma \longrightarrow \Delta}$$

with b the eigenvariable. If $A \in \Pi^b_{k-1}$ then this case is handled very similarly to the case of a $\forall{\leq}$:right inference, and we omit the argument. So, suppose $A \in \Pi^b_k \setminus \Pi^b_{k-1}$. A witness v for the formula $(\exists x{\leq}t)A(x)$ is thus a value for x which is $\leq t$ and which makes $A(x)$ true. Let $\mathcal{Q}$ be the Π^b_k-PLS problem that witnesses the upper sequent. Then a Π^b_k-PLS problem $\mathcal{P}$ for the lower sequent can be defined by

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; \mathcal{Q}(\langle \langle 0, 0\rangle * \mathit{cdr}(u), (u)_0, \vec{c}\rangle).$$

Here the value $(u)_0$ extracts the witness for the principal formula $(\exists x{\leq}t)A$ from u, and the values "$0, 0$" give the trivial witnesses for the first two formulas of the antecedent in the upper sequent.

Now suppose the final inference of P is a *cut*:

$$\frac{\Gamma \longrightarrow \Delta, A \qquad A, \Gamma \longrightarrow \Delta}{\Gamma \longrightarrow \Delta}$$

Let $\mathcal{Q}_1$ and $\mathcal{Q}_2$ be the two Π^b_k-PLS problems given by the induction hypothesis for the upper left and upper right sequents, respectively. The intuitive idea behind defining $\mathcal{P}$ is that it first invokes $\mathcal{Q}_1$; that produces either a witness for a formula in Δ or a witness for A. In the latter case, the witness for A is used to invoke $\mathcal{Q}_2$ and this then produces a witness for Δ. More formally, let $\mathcal{Q}'_2$ be defined by

$$\mathcal{Q}'_2(\langle u, \vec{c}, v\rangle) \;=\; \begin{cases} v & \text{if } (v)_0 < p \\ \mathcal{Q}_2(\langle\langle (v)_1\rangle * u, \vec{c}\rangle) & \text{otherwise} \end{cases}$$

where p is the number of formulas in Δ. Then define $\mathcal{P}$ as

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; \mathcal{Q}'_2(\langle u, \vec{c}, \mathcal{Q}_1(\langle u, \vec{c}\rangle)\rangle).$$

To understand the above definitions, note that in the definition of $\mathcal{Q}'_2$, the value v is intended to equal the value output by $\mathcal{Q}_1$, and thus v will be a witness for the antecedent Δ, A. The property $(v)_0 < p$ means that v witnesses the truth of one of the formulas in Δ, and $(v)_0 = p$ implies that v witnesses the truth of A. In the latter case, $\langle (v)_1\rangle * u$ then witnesses the antecedent A, Γ.

Finally consider the case where the final inference of P is an induction inference

$$\frac{A(b), \Gamma \longrightarrow \Delta, A(b+1)}{A(0), \Gamma \longrightarrow \Delta, A(t)}$$

The induction hypothesis gives a Π^b_k-PLS problem $\mathcal{Q}(\langle u, b, \vec{c}\rangle)$ which witnesses the upper sequent. We will define $\mathcal{P}$ to witness the lower sequent by using pseudo-iteration on a variant $\mathcal{P}_1$ of $\mathcal{Q}$. For an input value $\langle u, \vec{c}\rangle$ to $\mathcal{P}$ where $Wit_{A(0),\Gamma}(u, \vec{c})$, the pseudo-iteration will produce intermediate values $\langle v, i, \vec{c}, w\rangle$ which satisfy the property H defined as follows, where the intent is that $v = cdr(u)$ and $z = \langle p, (u)_0\rangle$:

$H(j, \langle v, 0, \vec{c}, z\rangle, \langle v, i, \vec{c}, w\rangle)$

$\Leftrightarrow \quad (\, Wit_\Gamma(v, \vec{c}) \wedge Wit_{\Delta, A(0)}(z, \vec{c}) \rightarrow Wit_\Gamma(v, \vec{c}) \wedge Wit_{\Delta, A(b)}(w, i, \vec{c})) \wedge i = j.$

The condition $Wit_{\Delta,A(b)}(w,i,\vec{c})$ means that either $(w)_0 < p \wedge Wit_{\Delta}(w,\vec{c})$ or $(w)_0 = p \wedge Wit_{A(b)}((w)_1,i,\vec{c})$, where p is the number of formulas in Δ and where i gives the value for the free variable b. The fact that $Wit_{\Gamma}(v,\vec{c})$ appears also on the righthand side of the implication H is unimportant for now, but will be needed in Section 5.3 when we prove Lemma 9, the Skolemized version of Lemma 5.

To initialize the pseudo-iteration, define $f(\langle u,\vec{c}\rangle) = \langle cdr(u),0,\vec{c},\langle p,(u)_0\rangle\rangle$. Note that $Wit_{A(b),\Gamma}(u,0,\vec{c})$ implies that $(u)_0$ witnesses $A(0)$, and hence that $Wit_{\Delta,A(b)}(\langle p,(u)_0\rangle,0,\vec{c}\rangle)$ holds and further that $H(0,f(\langle u,\vec{c}\rangle),f(\langle u,\vec{c}\rangle))$ is true. The function $\mathcal{P}_1$ to be (pseudo)iterated is defined so that

$$\mathcal{P}_1(\langle v,i,\vec{c},w\rangle) \;=\; \begin{cases} \langle v,i+1,\vec{c},w\rangle & \text{if } (w)_0 < p \\ \langle v,i+1,\vec{c},\mathcal{Q}(\langle\langle(w)_1\rangle * v,i,\vec{c}\rangle)\rangle & \text{otherwise.} \end{cases}$$

Finally, we define $\mathcal{P}$ by

$$\mathcal{P}(\langle u,\vec{c}\rangle) \;=\; (PsIter[\mathcal{P}_1,H](\langle t(\vec{c}),f(\langle u,\vec{c}\rangle)\rangle))_{\ell+2},$$

where ℓ is the number of variables in $\vec{c}$. Note $PsIter[\mathcal{P}_1,H](\langle t(\vec{c}),f(\langle u,\vec{c}\rangle)\rangle)$ outputs a tuple $\langle v,t,\vec{c},w\rangle$, and that the function $(\cdots)_{\ell+2}$ extracts the value w, which witnesses the antecedent $\Delta,A(t)$. It is straightforward to check that S^1_2 proves the requisite conditions (ι_0)-(ι_2) and proves that $\mathcal{P}$ serves as a witness function for the lower sequent of the induction inference.

That completes the proof of Lemma 5. □

We can now finish the proof of Theorem 4, and thus Theorem 2. As first step, convert the formula $A(x,y)$ into an equivalent (strict) formula $A^*(x,y)$ so that $A^*(x,y)$ is in $\Pi^b_{=k}$; to do this, simply add vacuous quantifiers at the end of the bounded quantifiers of A. If the hypotheses of Theorem 4 hold, then $\hat{T}^{k+1}_2$ proves the sequent

$$\longrightarrow (\exists y{\leq}t(x))A^*(x,y).$$

The antecedent of this sequent is empty and this is trivially witnessed by the empty sequence $\langle\rangle$. Thus, by Lemma 5, there is a Π^b_k-PLS problem $\mathcal{Q}$ such that S^1_2 proves

$$v = \mathcal{Q}(\langle\langle\rangle,x\rangle) \rightarrow Wit_{(\exists y\leq t)A^*}(v,x). \tag{10}$$

Here the condition $Wit_{(\exists y\leq t)}A^*(v,x)$ means that $v = \langle 0,v_1\rangle$ for a value $v_1 \leq t$ such that $A(x,v_1)$ holds.

Let F, N, c, i be the components of the problem $\mathcal{Q}$. By our conventions, the feasible points in $F(x)$ are all Gödel numbers of sequences of length at least three. We define a Π^b_k-PLS problem $\mathcal{Q}'$ which works by modifying the results of $\mathcal{Q}$ slightly. Namely, define the set of feasible points $F'(\langle\langle\rangle, x\rangle)$ for $\mathcal{Q}'(\langle\langle\rangle, x\rangle)$ by

$$F'(\langle\langle\rangle, x\rangle, s) \;\Leftrightarrow\; F(\langle\langle\rangle, x\rangle, s) \vee (\mathit{Len}(s) = 1 \wedge (s)_0 \leq t(x) \wedge A^*(x, (s)_0)).$$

The neighborhood function N' for $\mathcal{Q}'$ is defined so that, for any $s = \langle z, y, \mathfrak{a}\rangle$, $N'(s)$ is defined as

$$N'(\langle z, y, \mathfrak{a}\rangle) \;=\; \begin{cases} N(\langle z, y, \mathfrak{a}\rangle) & \text{if } N(\langle z, y, \mathfrak{a}\rangle) \neq \langle z, y, \mathfrak{a}\rangle \\ \langle (y)_1 \rangle & \text{otherwise.} \end{cases}$$

and setting $N'(s) = s$ for any other s, including any s encoding a sequence of length one. That is to say, N' acts like N, except that it maps any solution of $\mathcal{Q}$ to a sequence of length one containing the witness for A produced by the output of $\mathcal{Q}$. Similarly, the cost function for $\mathcal{Q}'$ is defined by letting $c'(s) = 0$ for any s coding a sequence of length one, and letting $c'(s) = c(s) + 1$ for all other s. The initial function i' for $\mathcal{Q}'$ is defined to equal the initial function of $\mathcal{Q}$, $i'(z) = i(z)$.

Finally, to complete the proof of Theorem 4, the Π^b_k-PLS problem $\mathcal{P}$ is defined by letting $\mathcal{P}(x) = \mathcal{Q}'(\langle\langle\rangle, x\rangle)$ using essentially the constructions for composing PLS problems described in Section 3. The Π^b_g-goal for $\mathcal{P}$ is defined to be

$$G(x, s) \;\Leftrightarrow\; \mathit{Len}(s) = 1 \wedge (s)_0 \leq t(x) \wedge A(x, (s)_0). \tag{11}$$

It is easy to verify that $\mathcal{P}$ satisfies the desired properties for Theorem 4, including that S^1_2 can prove properties (α)-(ϵ).
Q.E.D. Theorems 4 and 2.

5 The Skolemized Witnessing Theorem

This section sketches the proof Theorem 3. The proof is similar in spirit to the proof of Theorem 2; however, Lemma 5 must be modified to state that its conclusion is *Skolemizable* in S^1_2 rather than just *provable* in S^1_2. The proof of Theorem 3 has three parts. First, more care must be taken with the definitions of the Π^b_k-PLS problems so that the functions i, N, and c are given by $\hat{L}$-terms and that the conditions (α)-(γ) can be Skolemized with $\hat{L}$-terms. Second, we must establish that the Π^b_k-PLS problems $\mathcal{P}_A$ that

decide the validity of A can be used in a way that allows, in effect, resources to be "doubled". What this means is that formulas such as $A \to A \wedge A$ can be Skolemized with $\hat{L}$-terms — in the presence of the Π^b_k-PLS problem $\mathcal{P}^*_A$. Third, Lemma 9 is proved by induction on the number of lines in a free-cut free proof.

For the rest of the paper, when we say a formula A is "Skolemized" or "Skolemizable", we mean there is a Skolemization A_{SK} of the prenexification of A, with $\hat{L}$-terms as Skolem functions, so that $\hat{S}^1_2$ proves A_{SK}.

5.1 Skolemizing constructions of PLS problems.

This section proves that the constructions of Π^b_k-PLS problems in Section 3 preserve the property that the conditions (α), (β) and (γ) can be Skolemized.[‡] As a first step, we observe that it is essentially trivial to Skolemize the condition (α). Namely, suppose that $\mathcal{P}$ is a Π^b_k-PLS problem with components F, d, N, i, c, G, and then define F', as a replacement for F, by

$$F'(x,s) \;\Leftrightarrow |s| \leq d(|x|) \wedge F(x,s).$$

By the provability of (α), S^1_2 proves that $F'(x,s)$ is equivalent $F(x,s)$. Replacing $F(x,s)$ with $F'(x,s)$ leaves $\mathcal{P}$ unchanged (provably in S^1_2), and the condition (α) becomes

$$\forall x \forall s (F'(x,s) \to |s| \leq d(|x|)).$$

Since the definition of F' includes the condition $|s| \leq d(|x|)$ explicitly, this formula can be Skolemized by simply replacing all the universal quantified variables in F' with the constant 0.

We consider the constructions in Section 3 one at a time. First, consider the encoding of a polynomial time function f as a PLS problem. Under the further assumption that f is expressed by a $\hat{L}$-term, it is clear that the functions i, N, c are all expressible by $\hat{L}$-terms. The feasible set is definable by a term, and the conditions (β) and (γ) contain no quantifiers to Skolemize.

Second, consider the fg-combination where now the functions f and g are both required to be defined by $\hat{L}$ terms. The functions i, N, and c for the fg-combination $\mathcal{P}$ of $\mathcal{P}_1$ and $\mathcal{P}_2$ are easily expressed as $\hat{L}$-terms using the $\hat{L}$-terms for functions operating on sequence coding functions and for

[‡]We do not need to worry about Skolemizing the conditions (ϵ') and (ϵ'') since none of the constructions in Section 3 have goal predicates. Skolemization of these two conditions will be handled as a special case when we complete the proof of Theorems 3 and 8.

the *Cond* function along with the $\hat{L}$-terms for the functions i_ℓ, N_ℓ, and c_ℓ ($\ell = 1, 2$). The Skolem functions for condition (β) for $\mathcal{P}_1$ can also serve as the Skolem functions for (β) for $\mathcal{P}$. Furthermore, it is straightforward to check that terms for the Skolem functions for the condition (γ) for $\mathcal{P}$ can readily be defined from the Skolem functions for the conditions (β) and (γ) for $\mathcal{P}_1$ and $\mathcal{P}_2$ using $\hat{L}$-terms for sequence coding and definitions by cases.

Third, suppose $\mathcal{P} = \mathit{PsIter}[\mathcal{P}_1, H]$ and that $\mathcal{P}_1$ is formalized in Skolem form. In order to prove $\mathcal{P}$ can be formalized in Skolem form, we must make the extra assumption that (ι_0)-(ι_2) can be Skolemized. It can be assumed without loss of generality that (ι_1) can be Skolemized, since otherwise we can replace H with H' defined by

$$H'(i, x, y) \;\Leftrightarrow\; H(i, x, y) \wedge |y| \leq p_H(|x| + |i|).$$

However, we must explicitly assume that

(ι_0) $H(0, x, x)$, and

(ι_2) $F_1(\langle \mathfrak{a} \rangle) \wedge N_1(\langle \mathfrak{a} \rangle) = \langle \mathfrak{a} \rangle \wedge H(i, x, \mathfrak{a}_0) \rightarrow H(i + 1, x, \mathfrak{a}_1)$.

can be Skolemized.

Recall that the set of feasible points F for $\mathcal{P}$ is defined by (7). The condition (β) can be Skolemized using the $\hat{L}$-terms that Skolemize condition (β) for F_1, and the $\hat{L}$-terms that Skolemize condition (ι_0). We still need to show that (γ) can be Skolemized with $\hat{L}$-terms for this definition of F. Recall the two cases for the definition of the neighborhood function for $\mathcal{P}$ in Section 3. In the first case, $i < m$ and $N_1(\langle \mathfrak{a} \rangle) \neq \langle \mathfrak{a} \rangle$. In this case, the formula (γ) becomes equivalent to

$$F_1(\langle \mathfrak{a} \rangle) \wedge H(i, x, \mathfrak{a}_0) \rightarrow F_1(N_1(\langle \mathfrak{a} \rangle)) \wedge H(i, x, \mathfrak{a}_0),$$

since $(N_1(\langle \mathfrak{a} \rangle))_0 = \mathfrak{a}_0$. The Skolemization of this formula is easy from the fact that, since $\mathcal{P}_1$ is assumed to be formalized in Skolem form, the formula $F_1(\langle \mathfrak{a} \rangle) \rightarrow F_1(N_1(\langle \mathfrak{a} \rangle))$ is Skolemized. In the second case, $i < m$ and $N_1(\langle \mathfrak{a} \rangle) = \langle \mathfrak{a} \rangle$. Then ($\gamma$) becomes equivalent to

$$F_1(\langle \mathfrak{a} \rangle) \wedge H(i, x, \mathfrak{a}_0) \rightarrow F_1(i_1(\mathfrak{a}_1)) \wedge H(i + 1, x, \mathfrak{a}_1), \tag{12}$$

where we have used the fact that $(i_1(\mathfrak{a}_1))_0 = \mathfrak{a}_1$. The formula (12) is Skolemizable, since both equation (ι_2) and the condition (β) for F_1 are Skolemizable.

Fourth, consider the case where $\mathcal{P}_A$ is chosen to decide the truth of a Σ^b_k- or Π^b_k-formula A. Since we allow only $\hat{L}$-terms to serve as Skolem functions,

it is necessary to slightly modify the construction in Section 3 by having the inductive definition of the $\mathcal{P}_A$ problems start with A quantifier-free (instead of starting with A sharply bounded). This modification allows the Π^b_0-PLS problem $\mathcal{P}_A(\vec{x})$ that equals $\langle 1,0\rangle$ or $\langle 0,0\rangle$ depending on whether $A(\vec{x})$ is true or false to be defined by an $\hat{L}$-term.

With this modification, the rest of the construction in Section 3 goes through without any changes. There is one extra level of (pseudo)iteration but no increase in the complexity of the definitions of the feasible sets.

To prove that the problems $\mathcal{P}_A$ can be Skolemized, we argue by induction on k. In the induction step, where A is the strict Σ^b_k-formula $(\exists y{\leq}t(\vec{x}))B(y,\vec{x})$, $\mathcal{P}_A$ is defined in Section 3 from $\mathcal{P}_B$ using pseudo-iteration. The induction hypothesis is that $\mathcal{P}_B$ is defined in Skolem form. Let $\mathcal{Q}$ and H be as defined at the end of Section 3. The formula (ι_0) for $PsIter[\mathcal{Q},H]$ is trivially Skolemizable. Thus, it will suffice to show that the formula (ι_2) for $PsIter[\mathcal{Q},H]$ can be Skolemized. In view of the three cases in the definition of $\mathcal{Q}$ in equations (8) and (9), this means we must show that the following three formulas are Skolemizable:

$$\begin{aligned}\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 1, t(\vec{x})+1\rangle \wedge H(i,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i\rangle\rangle) \\ \rightarrow H(i+1,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 1,i\rangle\rangle)\end{aligned} \tag{13}$$

and

$$\begin{aligned}\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 0, j\rangle \wedge H(i,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i\rangle\rangle) \\ \rightarrow H(i+1,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i+1\rangle\rangle)\end{aligned} \tag{14}$$

and

$$H(j,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 1,i\rangle\rangle) \rightarrow H(j+1,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 1,i\rangle\rangle).$$

It is clear from the definition of H that the third formula is trivially Skolemizable; so we need to show (13) and (14) are Skolemizable. Here the formula $\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 1, t(\vec{x})+1\rangle$ represents the condition that, for some $\mathfrak{a}$,

$$F_B(\langle\mathfrak{a}\rangle) \wedge N_B(\langle\mathfrak{a}\rangle) = \langle\mathfrak{a}\rangle \wedge \mathfrak{a}_0 = \langle i,\vec{x}\rangle \wedge \mathfrak{a}_1 = \langle 1, t(\vec{x})+1\rangle \tag{15}$$

where F_B and N_B are the feasible set and the neighborhood function for $\mathcal{P}_B$. The formula $\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 0, j\rangle$ represents a similar formula.

Suppose B is atomic. In this case, unwinding the definitions of F_B and N_B in (15) gives that (15) is equivalent to $B(i,\vec{x})$. Similarly, $\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 0, j\rangle$ is equivalent to $\neg B(i,\vec{x})$. Equations (13) and (14) become

$$B(i,\vec{x}) \wedge H(i,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i\rangle\rangle) \rightarrow H(i+1,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 1,i\rangle\rangle)$$

and

$$\neg B(i,\vec{x}) \wedge H(i,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i\rangle\rangle) \\ \rightarrow H(i+1,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i+1\rangle\rangle).$$

Referring back to the definition of H at the end of Section 3, both of these are easily Skolemizable.

Now, suppose $B(y,\vec{x})$ is non-atomic, and is thus of the form $(\forall z{\leq}t_2(y,\vec{x}))C(z,y,\vec{x})$. The condition $\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 1, t_2(i,\vec{x})+1\rangle$ is again equivalent to $B(i,\vec{x})$, so (13) is again Skolemizable. However, $\mathcal{P}_B(\langle i,\vec{x}\rangle) = \langle 0,j\rangle$ is now equivalent to

$$j \leq t_2(i,\vec{x}) \wedge \neg C(j,i,\vec{x}) \wedge (\forall z{<}j)C(z,i,\vec{x}).$$

Equation (14) becomes

$$j \leq t_2(i,\vec{x}) \wedge \neg C(j,i,\vec{x}) \wedge (\forall z{<}j)C(z,i,\vec{x}) \wedge H(i,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i\rangle\rangle) \\ \rightarrow H(i+1,\langle\langle\vec{x}\rangle,\langle 0,0\rangle\rangle,\langle\langle\vec{x}\rangle,\langle 0,i+1\rangle\rangle).$$

From the definition of H, this is Skolemizable iff the following implication is:

$$j \leq t_2(i,\vec{x}) \wedge \neg C(j,i,\vec{x}) \wedge (\forall z{<}j)C(z,i,\vec{x}) \\ \wedge(\forall y{<}i)(\exists z{\leq}t_2(y,\vec{x}))\neg C(z,y,\vec{x}) \\ \rightarrow (\forall y{\leq}i)(\exists z{\leq}t_2(y,\vec{x}))\neg C(z,y,\vec{x}).$$

And, it is straightforward to see that this is Skolemizable.

A dual argument shows that $\mathcal{P}_{A'}$ can be Skolemized when A' is of the form $(\forall y{\leq}t')B'(y,\vec{x})$.

5.2 Witness doubling

Section 5.3 will prove that the conclusion of Lemma 5 can be strengthened to conclude that

$$Wit_\Gamma(u,\vec{c}) \wedge v = \mathcal{P}(\langle u,\vec{c}\rangle) \rightarrow Wit_\Delta(v,\vec{c})$$

can be Skolemized. More precisely, this means that $\hat{S}^1_2$ can prove the Skolemization of

$$Wit_\Gamma(u,\vec{c}) \wedge F(\langle u,\vec{c}\rangle,s) \wedge N(s) = s \rightarrow Wit_\Delta((s)_1,\vec{c}). \tag{16}$$

for some set of $\hat{L}$-terms as Skolem functions.

As a special case of this, consider the tautology $A \to (A \wedge A)$. This is not, in general, Skolemizable, unless $P = NP$. However, by the Skolemizability of (16), if $A(\vec{c})$ is a strict Π^b_k-formula, and taking Γ to be A and Δ to consist of a single formula equivalent to $A \wedge A$, it should be possible to find an PLS problem P so that $v = \mathcal{P}(\vec{c}) \wedge A \to A \wedge A$ is Skolemizable. In fact, as the next theorem states, $\mathcal{P} = \mathcal{P}_A$ suffices.

Theorem 6 *Let $A(\vec{x})$ be a strict Π^b_k- or Σ^b_k-formula. Then $\hat{S}^1_2$ can prove Skolemized versions of*

$$v = \mathcal{P}_A(\langle \vec{x} \rangle) \wedge A \to A \wedge A$$

and

$$v = \mathcal{P}_A(\langle \vec{x} \rangle) \wedge (A \vee A) \to A$$

with $\hat{L}$-terms as Skolem functions.

Proof The theorem is trivial if A is quantifier-free, since there are no quantifiers to be Skolemized. For quantified formulas, the proof is by induction on $k \geq 0$, where the case $k = 0$ corresponds to A having a single, sharply bounded quantifier. The base case is where A is quantifier-free, and it is convenient to view this as the $k = -1$ case.

The formula $A \to (A \wedge A)$ is equivalent to $(\neg A \vee \neg A) \to \neg A$, and the former can be Skolemized if and only if the latter can. This duality means that it will suffice to prove the induction step under the assumption that the outermost quantifier of A is existential. Thus, we henceforth assume that $A(\vec{x})$ is equal to $(\exists y{\leq}t(\vec{x}))B(y, \vec{x})$.

Referring back to the definition of $\mathcal{P}_A$ in terms of H, $\mathcal{Q}$, and $\mathcal{R}$, we need to prove that

$$H(t(\vec{x}) + 1, \langle \langle \vec{x} \rangle, \langle 0, 0 \rangle, \langle \langle \vec{x} \rangle, v \rangle) \wedge A \to A \wedge A \tag{17}$$

and

$$H(t(\vec{x}) + 1, \langle \langle \vec{x} \rangle, \langle 0, 0 \rangle, \langle \langle \vec{x} \rangle, v \rangle) \wedge (A \vee A) \to A \tag{18}$$

are Skolemizable. The condition $H(t(\vec{x})+1, \langle \langle \vec{x} \rangle, \langle 0, 0 \rangle, \langle \langle \vec{x} \rangle, v \rangle)$ holds if and only if $v = \langle j, i \rangle$ for some j, i, with $j \in \{0, 1\}$ and

$$\begin{aligned}[j = 0 \wedge i = t(\vec{x}) + 1 \wedge (\forall y{\leq}t(\vec{x}))(\neg B(y, \vec{x}))] \vee \\ [j = 1 \wedge i \leq t(\vec{x}) \wedge B(i, \vec{x}) \wedge (\forall i'{<}i)(\neg B(i', \vec{x}))].\end{aligned}$$

The definitions of the Skolem functions for (17) and (18) split into two cases depending on the value of j. For $j = 0$, we need to show that the formulas

$$(\forall y{\leq}t)(\neg B(y, \vec{x})) \wedge A \to A \wedge A$$

and

$$(\forall y{\leq}t)(\neg B(y,\vec{x})) \wedge (A \vee A) \rightarrow A$$

are Skolemizable. These are readily Skolemizable with identity functions by noting that $(\forall y{\leq}t)(\neg B(y,\vec{x})) \wedge A \rightarrow \bot$ is Skolemizable with identity functions, since A is $(\exists y{\leq}t)B(y,\vec{x})$. For $j = 1$, it suffices to show that

$$i \leq t(\vec{x}) \wedge B(i,\vec{x}) \wedge A \rightarrow A \wedge A$$

and

$$i \leq t(\vec{x}) \wedge B(i,\vec{x}) \wedge (A \vee A) \rightarrow A$$

are both Skolemizable. Both are readily seen to be Skolemizable.
Q.E.D. Theorem 6 □

5.3 Skolemized witnessing of free-cut free proofs

The proof of Theorem 3 is based on Theorem 8 and Lemma 9 below. The latter strengthens Lemma 5 by showing that the conclusion can be Skolemized in $\hat{S}^1_2$ with $\hat{L}$-terms. First, we state a well-known lemma which states that cut inferences preserve Skolemizability.

Lemma 7 *Let $k \geq 0$. Suppose that the formulas*

$$A \rightarrow B \vee C \quad \textit{and} \quad C \wedge D \rightarrow E$$

are provable in $\hat{S}^1_2$ in Skolemized form with $\hat{L}$-terms as Skolem functions. Then

$$A \wedge D \rightarrow B \vee E$$

is also provable in $\hat{S}^1_2$ in Skolemized form with $\hat{L}$-terms as Skolem functions.

Proof Without loss of generality, the formulas $A, \ldots, E$ are prenex formulas, and no variable is quantified twice in the formulas $A, \ldots, E$. Let $A_0, \ldots, E_0$ be the maximal quantifier-free subformulas of $A, \ldots, E$. The Skolemization hypothesis implies that there are substitutions σ_1 and σ_2 such that (i) the domain of σ_1, respectively σ_2, is the set of essentially existentially quantified variables in $A \rightarrow B \vee C$, respectively $C \wedge D \rightarrow E$; (ii) for each essentially existential quantified variable x in the formula $A \rightarrow B \vee C$ (resp., $C \wedge D \rightarrow E$), the term $x\sigma_1$ (resp., $x\sigma_2$) involves only universally quantified variables from the formula at the same or higher level; and (iii) the formulas

$$(A_0 \rightarrow B_0 \vee C_0)\sigma_1 \quad \text{and} \quad (C_0 \wedge D_0 \rightarrow E_0)\sigma_2$$

are theorems of $\hat{S}^1_2$. Since no variable is quantified twice, σ_1 and σ_2 have disjoint domains (by the usual convention, a substitution acts as the identity function on objects outside its domain). Let C be $\forall x_1 \exists x_2 \forall x_3 \cdots Q x_\ell C_0$ where the notation is suppressing the bounds on the quantifiers. Define the substitutions π_i so that π_i has domain $\{x_i\}$ with $\pi_i(x_i) = x_i\sigma_1$ for even values of i, and $\pi_i(x_i) = x_i\sigma_2$ for odd values of i. Then set

$$\rho = (\sigma_1 \cup \sigma_2)\pi_\ell \pi_{\ell-1} \cdots \pi_3 \pi_2 \pi_1.$$

The substitution ρ is an instance of σ_1 and σ_2 so

$$(A_0 \to B_0 \vee C_0)\rho \quad \text{and} \quad (C_0 \wedge D_0 \to E_0)\rho$$

and thus $(A_0 \wedge D_0 \to B_0 \vee E_0)\rho$ are all theorems of $\hat{S}^1_2$. Furthermore, it is clear that ρ respects the levels of variables in that if x is a essentially existential variable at level i, then $\rho(x)$ is an $\hat{L}$-term involving only essentially universal variables at levels $\geq i$. Therefore, ρ provides the desired Skolemization of $A \wedge D \to B \vee E$. □

Note that the proof of Lemma 7 shows how to define the Skolem functions for $A \wedge D \to B \vee E$ explicitly from the Skolem functions for $A \to B \vee C$ and $C \wedge D \to E$.

Theorem 8 *Let $k \geq 0$, and $0 \leq g \leq k$. Suppose $A(x, y)$ is a strict Π^b_g-formula and*

$$\hat{T}^{k+1}_2 \vdash (\forall x)(\exists y \leq t) A(x, y).$$

Then there is a Π^b_k-PLS problem $\mathcal{P}$ with Π^b_g-goal G that is formalized in Skolem form in $\hat{S}^1_2$, such that $\hat{S}^1_2$ proves

$$\forall \vec{x} \forall s (G(x, s) \to A(x, (s)_0)).$$

Furthermore, $\hat{S}^1_2$ proves a Skolemized form of this formula with $\hat{L}$-terms as Skolem functions.

The proof of Theorem 8 will be based on the next lemma.

Lemma 9 *Let $k \geq 0$. Suppose $\hat{T}^{k+1}_2$ proves a sequent $\Gamma \longrightarrow \Delta$ containing only strict Σ^b_{k+1}-formulas, with $\vec{c}$ as free variables. Then there is a Π^b_k-PLS problem $\mathcal{P}$ which is formalized in $\hat{S}^1_2$ in Skolem form such that S^1_2 proves*

$$Wit_\Gamma(u, \vec{c}) \wedge F(\langle u, \vec{c} \rangle, s) \wedge N(s) = s \to Wit_\Delta((s)_1, \vec{c}).$$

where F and N define the feasible points and the neighborhood function for $\mathcal{P}$. Furthermore, $\hat{S}^1_2$ can prove a Skolemized version of this formula, with $\hat{L}$-terms as Skolem functions.

Proof (of Lemma 9.) The proof of Lemma 9 proceeds by induction on the number of steps in a free-cut free $\hat{T}_2^{k+1}$-proof P. The proof splits into cases based on the final inference in the proof. Generally, the arguments are similar to those in the proof of Lemma 5, but now care must be taken to show the Skolemization properties hold. We discuss only the harder cases, and leave the easier cases for the reader.

The cases where the $\hat{T}_2^{k+1}$-proof is either a single initial sequent, or ends with a propositional rule, are very simple with arguments similar to those in Lemma 5. The first non-trivial case is when the final inference in the proof P is a *Contraction:left* inference:

$$\frac{\Gamma_1, A, A, \Gamma_2 \longrightarrow \Delta}{\Gamma_1, A, \Gamma_2 \longrightarrow \Delta}$$

Let Γ' be the upper antecedent Γ_1, A, A, Γ_2. The induction hypothesis is that there is a Π_k^b-PLS problem $\mathcal{Q}$, formalized in Skolem form, so that $\hat{S}_2^1$ proves a Skolemized version of

$$F_{\mathcal{Q}}(\langle u, \vec{c}\rangle, s) \wedge N_{\mathcal{Q}}(s) = s \wedge Wit_{\Gamma'}(u, \vec{c}) \rightarrow Wit_{\Delta}((s)_1, \vec{c}). \tag{19}$$

Let Γ be the lower antecedent Γ_1, A, Γ_2. Suppose u witnesses Γ. Then $u = \langle u_0, \ldots, u_{p_1}, \ldots, u_{p_1+p_2}\rangle$, where p_1 and p_2 are the number of formulas in Γ_1 and Γ_2. So u_{p_1} witnesses A. Define $u' = \langle u_0, \ldots, u_{p_1}, u_{p_1}, \ldots, u_{p_1+p_2}\rangle$. Since we are using sequences with fixed length entries, the mapping $u \mapsto u'$ is definable with an $\hat{L}$-term, and $\hat{S}_2^1$ proves $Wit_{\Gamma}(u, \vec{c}) \rightarrow Wit_{\Gamma'}(u', \vec{c})$. Unfortunately, $\hat{S}_2^1$ may not prove this in Skolemized form, since it may not be able to prove

$$Wit_A(u_{p_1}, \vec{c}) \rightarrow Wit_A(u_{p_1}, \vec{c}) \wedge Wit_A(u_{p_1}, \vec{c})$$

in Skolemized form (this is an open problem, in fact). To circumvent this, we invoke the Π_k^b-PLS problem $\mathcal{P}_{Wit_A}$ so as to use witness doubling property of Theorem 6. Accordingly, we define

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; (\langle \mathcal{Q}(\langle u, \vec{c}\rangle), \mathcal{P}_{Wit_A}(\langle (u)_{p_1}, \vec{c}\rangle)\rangle)_0.$$

In effect, $\mathcal{P}$ calculates $\mathcal{P}_{Wit_A}$ merely in order to discard the value. More precisely, the output value of $\mathcal{P}_{Wit_A}$ is discarded, but the final feasible point in its computation is still available to aid the Skolemization. Let F and N define the feasible points and the neighborhood function for $\mathcal{P}$. By the conventions for definition by *fg*-composition, the condition $F(\langle u, \vec{c}\rangle, s) \wedge$

$N(s) = s$ means that $s = \langle\langle u, \vec{c}\rangle, v, 2, \mathfrak{a}, \mathfrak{b}\rangle$, where $\mathfrak{a}$ and $\mathfrak{b}$ are intended to code final feasible points for $\mathcal{P}_{Wit_A}$ and $\mathcal{Q}$, and thus satisfy

$$F_{P_{Wit_A}}(\langle(u)_{p1}, \vec{c}\rangle, \langle\mathfrak{a}\rangle) \wedge N_{P_{Wit_A}}(\langle\mathfrak{a}\rangle) = \langle\mathfrak{a}\rangle$$

and

$$F_{\mathcal{Q}}(\langle u, \vec{c}\rangle, \langle\mathfrak{b}\rangle) \wedge N_{\mathcal{Q}}(\langle\mathfrak{b}\rangle) = \langle\mathfrak{b}\rangle.$$

By Theorem 6 and the Skolemizability of (19), and using the construction of the proof of Lemma 7, it follows that

$$F(\langle u, \vec{c}\rangle, s) \wedge N(s) = s \wedge Wit_\Gamma(u, \vec{c}) \rightarrow Wit_\Delta(v, \vec{c}).$$

is Skolemizable. This completes the argument for the case of an Contraction:left inference.

The case of *Contraction:right* is as simple as in the proof of Lemma 5 and in fact does not even use Theorem 6. We omit this case here.

Now suppose the final inference is a $\forall\leq$*:right* inference

$$\frac{b \leq t, \Gamma \longrightarrow \Delta, A(b)}{\Gamma \longrightarrow \Delta, (\forall x{\leq}t)A(x)}$$

We use the same construction for this case as in the proof of Lemma 5. Let $\mathcal{Q}(u, b, \vec{c})$ be the Π^b_k-PLS function for the upper sequent given by the induction hypothesis; then the Π^b_k-PLS problem $\mathcal{P}$ for the lower sequent is defined by *fg*-combination as

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; \begin{cases} \langle p, 0\rangle & \text{if } \mathcal{P}_{\forall A}(\langle c\rangle) = \langle 1, t+1\rangle \\ \mathcal{Q}(\langle\langle 0\rangle * u, (\mathcal{P}_{\forall A}(\langle\vec{c}\rangle))_1, \vec{c}\rangle) & \text{otherwise} \end{cases}$$

Let F and N define the feasible points and the neighborhood function for $\mathcal{P}$. Unwinding the definition of $\mathcal{P}$, the condition

$$F(\langle u, \vec{c}\rangle, s) \wedge N(s) = s \tag{20}$$

states that s is of the form $\langle\langle u, \vec{c}\rangle, v, 2, \mathfrak{a}, \mathfrak{b}\rangle$, where $\mathfrak{a}$ codes a final feasible point of $\mathcal{P}_{\forall A}$, and if $(\mathfrak{a}_1)_1 = 1$ then $\mathfrak{b}$ codes a final feasible point for $\mathcal{Q}$. That is to say, (20) states that

$$\begin{aligned} &[(v)_0 = p \wedge (\forall x{\leq}t(\vec{c}))A(x) \wedge v = \langle p, 0\rangle] \\ &\quad \vee [(v)_0 < p \wedge (\mathfrak{a}_1)_1 \leq t(\vec{c}) \wedge \neg A((\mathfrak{a}_1)_1) \wedge (\forall x{<}(\mathfrak{a}_1)_1)A(x) \wedge \\ &\qquad F_{\mathcal{Q}}(\langle\langle 0\rangle * u, (\mathfrak{a}_1)_1, \vec{c}\rangle, \langle\mathfrak{b}\rangle) \wedge N_{\mathcal{Q}}(\langle\mathfrak{b}\rangle) = \langle\mathfrak{b}\rangle \wedge v = \mathfrak{b}_1]. \end{aligned} \tag{21}$$

Here $F_{\mathcal{Q}}$ and $N_{\mathcal{Q}}$ define feasible points and the neighborhood function for $\mathcal{Q}$. By the induction hypothesis,

$$b \leq t(\vec{c}) \wedge \mathit{Wit}_{\Gamma}(u, \vec{c}) \wedge F_{\mathcal{Q}}(\langle\langle 0\rangle * u, b, \vec{c}\rangle, \langle\mathfrak{b}\rangle) \wedge N_{\mathcal{Q}}(\langle\mathfrak{b}\rangle) = \langle\mathfrak{b}\rangle \rightarrow \mathit{Wit}_{\Delta}(\mathfrak{b}_1, \vec{c}) \vee A(b) \tag{22}$$

is Skolemizable. To finish the $\forall\leq$:right case, we must show that

$$(21) \wedge \mathit{Wit}_{\Gamma}(u, \vec{c}) \rightarrow \mathit{Wit}_{\Delta,(\forall\leq t)A}(v, \vec{c}) \tag{23}$$

is Skolemizable. The $\hat{L}$-terms for the Skolem functions are defined by cases. If $(v)_0 = p$, the Skolem functions are just the identity functions that suffice for the implication $(\forall x{\leq}t)A \rightarrow (\forall x{\leq}t)A$. If $(v)_0 < p$, the Skolem functions are defined as for the Skolemization of equation (22) using $b = (\mathfrak{a}_1)_1$ and noting that the formula $A(b)$ at the end of (22) is replaced by the formula $\neg A((\mathfrak{a}_1)_1)$ in the hypothesis (21) of (23).

That completes the case of a $\forall\leq$:right inference. The other quantifier inferences are similar, so we omit them.

Suppose the final inference of P is a *cut* inference

$$\frac{\Gamma \longrightarrow \Delta, A \qquad A, \Gamma \longrightarrow \Delta}{\Gamma \longrightarrow \Delta}$$

The construction begins the same way as for the *cut* inference case of the proof of Lemma 5. Let $\mathcal{Q}_1$ and $\mathcal{Q}_2$ be given by the induction hypothesis and define $\mathcal{P}$ from these by *fg*-combination exactly as before by

$$\mathcal{Q}_2'(\langle u, \vec{c}, v\rangle) \;=\; \begin{cases} v & \text{if } (v)_0 < p \\ \mathcal{Q}_2(\langle\langle (v)_1\rangle * u, \vec{c}\rangle) & \text{otherwise} \end{cases}$$

and

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; \mathcal{Q}_2'(\langle u, \vec{c}, \mathcal{Q}_1(\langle u, \vec{c}\rangle)\rangle).$$

Unwinding the definitions, the final feasible point property for $\mathcal{P}$, $F(\langle u, \vec{c}\rangle, s) \wedge N(s) = s$, states that s is of the form $\langle\langle u, \vec{c}\rangle, w, 2, \mathfrak{a}, \mathfrak{b}\rangle$ where $\mathfrak{a}$ and $\mathfrak{b}$ are intended to code the final states of computations for $\mathcal{Q}_1$ and $\mathcal{Q}_2$ and that the following condition holds:

$$\begin{aligned} F_1(\langle u, \vec{c}\rangle, \langle\mathfrak{a}\rangle) \wedge N_1(\langle\mathfrak{a}\rangle) = \langle\mathfrak{a}\rangle \wedge ([(\mathfrak{a}_1)_0 < p \wedge w = \mathfrak{a}_1] \\ \vee [(\mathfrak{a}_1)_0 = p \wedge F_2(\langle\langle(\mathfrak{a}_1)_1\rangle * u, \vec{c}\rangle, \langle\mathfrak{b}\rangle) \wedge N_2(\langle\mathfrak{b}\rangle) = \langle\mathfrak{b}\rangle \wedge w = \mathfrak{b}_1]). \end{aligned} \tag{24}$$

By the induction hypothesis for $\mathcal{Q}_1$, the formula

$$[\mathit{Wit}_\Gamma(u,\vec{c}) \wedge F_1(\langle u,\vec{c}\rangle,\langle\mathfrak{a}\rangle) \wedge N_1(\langle\mathfrak{a}\rangle) = \langle\mathfrak{a}\rangle] \rightarrow \\ ([(\mathfrak{a}_1)_0 = p \wedge \mathit{Wit}_A((\mathfrak{a}_1)_1,\vec{c})] \vee [(\mathfrak{a}_1)_0 < p \wedge \mathit{Wit}_\Delta(\mathfrak{a}_1,\vec{c})]). \tag{25}$$

is Skolemizable. Similarly, the induction hypothesis for $\mathcal{Q}_2$ implies the same holds for

$$[\mathit{Wit}_A((\mathfrak{a}_1)_1,\vec{c}) \wedge \mathit{Wit}_\Gamma(u,\vec{c}) \wedge F_2(\langle\langle(\mathfrak{a}_1)_1\rangle * u,\vec{c}\rangle,\langle\mathfrak{b}\rangle) \wedge N_2(\langle\mathfrak{b}\rangle) = \langle\mathfrak{b}\rangle] \\ \rightarrow \mathit{Wit}_\Delta(\mathfrak{b}_1,\vec{c}). \tag{26}$$

Combining (24) and the induction hypothesis for $\mathcal{Q}_1$ yields that

$$\mathit{Wit}_\Gamma(u,\vec{c}) \wedge F(\langle u,\vec{c}\rangle,s) \wedge N(s) = s \wedge (\mathfrak{a}_1)_0 < p \rightarrow \mathit{Wit}_\Delta((s)_1,\vec{c})$$

is Skolemizable, where $\mathfrak{a}_1 = (s)_4$ of course. Combining (24) and the two induction hypotheses yields that

$$\mathit{Wit}_\Gamma(u,\vec{c}) \wedge \mathit{Wit}_\Gamma(u,\vec{c}) \wedge F(\langle u,\vec{c}\rangle,s) \wedge N(s) = s \wedge (\mathfrak{a}_1)_0 = p \rightarrow \mathit{Wit}_\Delta((s)_1,\vec{c})$$

is likewise Skolemizable. Putting the last two equations together shows that

$$\mathit{Wit}_\Gamma(u,\vec{c}) \wedge \mathit{Wit}_\Gamma(u,\vec{c}) \wedge F(\langle u,\vec{c}\rangle,s) \wedge N(s) = s \rightarrow \mathit{Wit}_\Delta((s)_1,\vec{c})$$

is Skolemizable. We are almost done, except that there are two occurrences of $\mathit{Wit}_\Gamma(u,\vec{c})$ in the last formula, instead of only one. To fix this, we use the same technique as in the case of a *Contraction:left* inference. Define $\mathcal{P}_2$ as

$$\mathcal{P}_2(\langle u,\vec{c}\rangle) \;=\; (\langle\mathcal{P}(\langle u,\vec{c}\rangle), P_{\mathit{Wit}_\Gamma}(\langle u,\vec{c}\rangle)\rangle)_0.$$

Letting F_2 and N_2 define the feasible points and the neighborhood function for $\mathcal{P}_2$, and arguing as in the *Contraction:left* case, we obtain

$$\mathit{Wit}_\Gamma(u,\vec{c}) \wedge F_2(\langle u,\vec{c}\rangle,s) \wedge N_2(s) = s \rightarrow \mathit{Wit}_\Delta((s)_1).$$

Thus Lemma 9 holds for $\Gamma \longrightarrow \Delta$ using the Π^b_k-PLS problem $\mathcal{P}_2$.

Finally, suppose the final inference of P is an induction inference:

$$\frac{A(b),\Gamma \longrightarrow \Delta, A(b+1)}{A(0),\Gamma \longrightarrow \Delta, A(t)}$$

Let $\mathcal{Q}$ be given by the induction hypothesis. Let $X(v, \vec{c}, z)$ be the formula $Wit_\Gamma(v, \vec{c}) \wedge Wit_{\Delta, A(0)}(z, \vec{c})$, and give the side condition H the same definition as used for this case in Lemma 5:

$$\begin{aligned} & H(j, \langle v, 0, \vec{c}, z\rangle, \langle v, i, \vec{c}, w\rangle) \\ & \quad \Leftrightarrow \quad (X(v, \vec{c}, z) \rightarrow Wit_\Gamma(v, \vec{c}) \wedge Wit_{\Delta, A(b)}(w, i, \vec{c})) \wedge i = j. \end{aligned}$$

Likewise, define the initialization function $f(\langle u, \vec{c}\rangle) = \langle cdr(u), 0, \vec{c}, \langle p, (u)_0\rangle\rangle$ exactly as before. Recall that $\mathcal{P}_1$ was defined as

$$\mathcal{P}_1(\langle v, i, \vec{c}, w\rangle) \;=\; \begin{cases} \langle v, i+1, \vec{c}, w\rangle & \text{if } (w)_0 < p \\ \langle v, i+1, \vec{c}, \mathcal{Q}(\langle\langle (w)_1\rangle * v, i, \vec{c}\rangle)\rangle & \text{otherwise.} \end{cases}$$

For technical reasons that will be clear in a moment, we cannot define $\mathcal{P}$ by pseudo-iteration on $\mathcal{P}_1$; instead, similar to the construction used for the Contraction:left and Cut inferences, we define

$$\mathcal{P}_2(\langle v, i, \vec{c}, w\rangle) \;=\; (\langle \mathcal{P}_1(v, i, \vec{c}, w), \mathcal{P}_{Wit_\Gamma}(\langle v, \vec{c}\rangle)\rangle)_0.$$

Note that the $\mathcal{P}_2$ is defined so as to compute, but then discard, the value $\mathcal{P}_{Wit_\Gamma}$. As before, the point of this is to include extra conditions in the formula F defining the feasible points of $\mathcal{P}$ that will allow Skolemization of an "extra" occurrence of Wit_Γ. The Π^b_k-PLS problem $\mathcal{P}$ is now defined using pseudo-iteration of $\mathcal{P}_2$ instead of $\mathcal{P}_1$: Let $\mathcal{P}_3$ be defined by

$$\mathcal{P}_3(\langle u, \vec{c}\rangle) \;=\; PsIter[\mathcal{P}_2, H](\langle t(\vec{c}), f(\langle u, \vec{c}\rangle)\rangle) \tag{27}$$

and set

$$\mathcal{P}(\langle u, \vec{c}\rangle) \;=\; (\mathcal{P}_3(\langle u, \vec{c}\rangle))_{\ell+2}.$$

In order to show $\mathcal{P}$ is defined in Skolem form, we must show that, for the definition of the pseudo-iteration of $\mathcal{P}_2$, the conditions (ι_0) and (ι_2) can be Skolemized. This is trivial for (ι_0). Let F_j and N_j define the feasible points and the neighborhood function for $\mathcal{P}_j$, where $j = 1, 2$. For (ι_2), it is required that

$$\begin{aligned} F_2(\langle v, i, \vec{c}, w\rangle, s) \wedge N_2(s) = s \wedge H(i, \langle v, 0, \vec{c}, z\rangle, \langle v, i, \vec{c}, w\rangle) \\ \rightarrow H(i, \langle v, 0, \vec{c}, z\rangle, (s)_1) \end{aligned} \tag{28}$$

is Skolemizable. The conventions for encoding feasible points for *fg*-combinations mean that the hypothesis $F_2(\langle v, i, \vec{c}, w\rangle, s) \wedge N_2(s) = s$ implies that

$$s \;=\; \langle\langle v, i, \vec{c}, w\rangle, \langle v, i+1, \vec{c}, w'\rangle, 2, \mathfrak{a}, \mathfrak{b}\rangle,$$

where $\mathfrak{a}$ encodes the final feasible point of a computation of $\mathcal{P}_{Wit_\Gamma}(\langle v, \vec{c}\rangle)$ and $\mathfrak{b}$ encodes the final feasible point of the computation of $\mathcal{P}_1(\langle v, i, \vec{c}, w\rangle)$. Of course, the definition of $\mathcal{P}_1$ means that if $(w)_0 = p$ then $\mathfrak{b}$ further includes a subsequence $\mathfrak{c}$ which encodes the final feasible point of a computation of $\mathcal{Q}$.

The Skolem functions for (28) can be defined by two cases. The first case is $(w)_0 < p$. In this case, from the definition of $\mathcal{P}$, we have $w' = w$, so the Skolem functions for the quantifiers in H are just identity functions. The second case is $(w)_0 = p$. We have, from the presence of $\mathfrak{a}$ in s, that

$$F_{Wit_\Gamma}(\langle v, \vec{c}\rangle, \langle \mathfrak{a}\rangle) \wedge N_{Wit_\Gamma}(\langle \mathfrak{a}\rangle) = \langle \mathfrak{a}\rangle \wedge Wit_\Gamma(\langle v, \vec{c}\rangle) \rightarrow Wit_\Gamma(\langle v, \vec{c}\rangle) \wedge Wit_\Gamma(\langle v, \vec{c}\rangle)$$

is Skolemizable. Also, from the presence of $\mathfrak{c}$ and the induction hypothesis for $\mathcal{Q}$, we have that

$$F_1(\langle v, i, \vec{c}, w\rangle, \langle \mathfrak{b}\rangle) \wedge N_1(\langle \mathfrak{b}\rangle) = \langle \mathfrak{b}\rangle \wedge Wit_{A(b),\Gamma}(\langle\langle (w)_1\rangle * v, i, \vec{c}\rangle) \rightarrow Wit_{\Delta,A(b)}((\mathfrak{b}_1)_{\ell+2}, i+1, \vec{c})$$

is similarly Skolemizable. Note that $\mathfrak{b}_1 = (s)_1 = \langle v, i+1, \vec{c}, w'\rangle$; also, recall that ℓ is the number of variables in $\vec{c}$, so $((s)_1)_{\ell+2} = w'$. In addition, it is easy to prove, and Skolemize using $\hat{L}$-terms, the property that

$$Wit_{A(b)}((w)_1, i, \vec{c}) \wedge Wit_\Gamma(v, \vec{c}) \rightarrow Wit_{A(b),\Gamma}(\langle\langle (w)_1\rangle * v, i, \vec{c}\rangle).$$

Continuing to use the condition $(w)_0 = p$, the formula (28) can be expanded in more detail as being equivalent to

$$\begin{aligned}[F_2(\langle v, i, \vec{c}, w\rangle, s) \wedge N_2(s) = s \wedge \\ (X(v, \vec{c}, z) \rightarrow Wit_\Gamma(v, \vec{c}) \wedge Wit_{A(b)}((w)_1, i, \vec{c}))] \\ \rightarrow (X(v, \vec{c}, z) \rightarrow Wit_\Gamma(v, \vec{c}) \wedge Wit_{\Delta,A(b)}(((s)_1)_{\ell+2}, i+1, \vec{c})).\end{aligned} \tag{29}$$

Skolem functions for (29) can readily be defined using $\hat{L}$-terms and the Skolem functions for the previous three formulas and Lemma 7.

The above showed that $\mathcal{P}$ is definable by $\hat{S}^1_2$ in Skolem form. We now need to establish that Lemma 9 holds for the sequent $A(0), \Gamma \longrightarrow \Delta, A(t)$. From (27) and the definitions of f and H, we have that the condition $F_3(\langle u, \vec{c}\rangle, s) \wedge N_3(s) = s$ is equivalent to

$$Wit_\Gamma(cdr(u), \vec{c}) \wedge Wit_{A(0)}(((u)_0, \vec{c})) \rightarrow Wit_\Gamma(cdr(u), \vec{c}) \wedge Wit_{\Delta,A(t)}(((s)_1)_{\ell+2}, \vec{c}).$$

From this,

$$F_3(\langle u, \vec{c}\rangle, s) \wedge N_3(s) = s \wedge \mathit{Wit}_{A(0),\Gamma}(u, \vec{c}) \rightarrow \mathit{Wit}_{\Delta,A(t)}(((s)_1)_{\ell+2}, \vec{c})$$

is immediately seen to be Skolemizable. This suffices to prove the lemma for the case of an induction inference, and thereby completes the proof of Lemma 9. □

The proof of Theorem 8 from Lemma 9 uses the same construction as the proof of Theorem 4 from Lemma 5. Suppose the hypothesis of Theorem 8 holds and let A^* be as in the proof of Theorem 4. By Lemma 9, there is a Π^b_k-PLS problem $\mathcal{Q}$, formalized in Skolem form in $\hat{S}^1_2$ such that the formula

$$F(\langle\langle\rangle, x\rangle, s) \wedge N(s) = s \rightarrow \mathit{Wit}_{(\exists y \leq t)A*}((s)_1, x) \tag{30}$$

is Skolemizable, where F and N define the feasible points and the neighborhood function of $\mathcal{Q}$. (Note this is just a restatement of equation (10) with $v = (s)_1$.) Construct the Π^b_k-PLS problem $\mathcal{Q}'$ from $\mathcal{Q}$ exactly the same as in the proof of Theorem 4. We need to check that equation (γ) for $\mathcal{Q}'$ can be Skolemized. We can assume w.l.o.g. that $F(s) \rightarrow \mathit{Len}(s) > 1$ is Skolemizable. Indeed, for ℓ the sequence length of feasible points, the atomic formula expressing $\mathit{Len}(s) = \ell$ can be included as a conjunct of $F(s)$. The Skolemization of (γ) $F'(s) \rightarrow F'(N(s))$ splits into three cases, namely, (1) $N(s) = s$, (2) $N(s) \neq s \wedge \mathit{Len}(s) = \mathit{Len}(N(s)) > 1$, or (3) $\mathit{Len}(s) > 1 \wedge \mathit{Len}(N(s)) = 1$. The first case is trivial. The second case uses the same Skolem functions as are used in the Skolemization of (γ) for $\mathcal{Q}$, namely the Skolemization of $F(s) \rightarrow F(N(s))$. The third case uses the Skolem functions used in the Skolemization of (30).

Define the Π^b_k-PLS problem $\mathcal{P}$ by $\mathcal{P}(x) = \mathcal{Q}'(\langle\langle\rangle, x\rangle)$ as before. The Π^b_g-goal $G(x, s)$ is again defined by (11). We still need to show that (ϵ) and (ϵ') for $\mathcal{P}$ are Skolemizable. By the fact that

$$F'(x, s) \wedge N(s) = s \;\leftrightarrow\; \mathit{Len}(s) = 1 \wedge (s)_0 < t(x) \wedge A^*(x, (s)_0)$$

is provable in Skolem form (more precisely, each direction of the implication is provable in Skolem form), it suffices to show that both $A(x, y) \rightarrow A^*(x, y)$ and $A^*(x, y) \rightarrow A(x, y)$ are Skolemizable. The Skolem forms of these two formulas are picked using the ϵ-level of the quantifiers; that is to say, each quantifier in A is matched with the corresponding quantifier in A^*, and the vacuous quantifiers of A^* are brought out last. Thus, both of these formulas are trivially Skolemized with identity functions.

This completes the proof of Theorems 8 and 3. □

6 Towards relativized $\forall\Sigma^b_1$-separations

One of the central open problems for bounded arithmetic is whether the hierarchy of bounded arithmetic theories S^i_2 and T^i_2 is proper. Some conditional results are known; specifically, it is known [14, 6, 21] that if T^i_2 equals S^i_2, then the polynomial hierarchy collapses, provably in S^i_2. This result relativizes, yielding that $S^i_2(\alpha)$ is distinct from $T^i_2(\alpha)$.

However, no similar conditional or relativized results are known for the $\forall\Sigma^b_1$-consequences of S^i_2 or of T^i_2. These theories, however, have been characterized in terms of propositional proof complexity. In fact there are two such characterizations. One, by Krajíček and Pudlák [13], relates the $\forall\Sigma^b_1$-consequences of S^i_2 or T^i_2 to uniform provability in (tree-like or dag-like, respectively) proofs in quantified propositional logic, where quantifier alternation is restricted to i levels. The other characterization applies to relativized theories of bounded arithmetic, and uses a construction that goes back to Paris and Wilkie [16]; it relates the $\forall\Sigma^b_1(\alpha)$-consequences of $T^i_2(\alpha)$ to provability in a bounded depth proof system.

The Skolemized PLS problems, as described in the previous section, can give new $\forall\Sigma^b_1(\alpha)$-principles that are candidates for separating $S^{k+1}_2(\alpha)$ and $T^{k+1}_2(\alpha)$. For the rest of this section, we fix a value $k \geq 0$ and work with Π^b_k-PLS problems that have a Π^b_0-goal G.

Consider the prenexification of (γ) as given by equation (5). Let F be a strict $\Pi^b_{=k}$-formula of the form

$$(\forall y_1 \leq t_1)(\exists y_2 \leq t_2)\cdots(Qy_k \leq t_k)F_0(\vec{y}, x, s),$$

where F_0 is a new predicate symbol adjoined to the language, and where there is no last sharply bounded quantifier present. A Skolemization of (γ) uses functional substitutions for the existentially quantified variables, for instance, $y_1 \mapsto g_1(y'_1)$, $y'_2 \mapsto g_2(y'_1, y_2)$, $y_3 \mapsto g_3(y'_1, y_2, y'_3)$, etc. Thus, the Skolemization of (γ) has the following form.

$$\begin{aligned}(\forall y'_1 \leq t'_1)(\forall y_2 \leq t_2)(\forall y'_3 \leq t'_3)\cdots \\ (g_1(y'_1) \leq t_1 \wedge g_2(y'_1, y_2) \leq t'_2 \wedge g_3(y'_1, y_2, y'_3) \leq t_3 \wedge \cdots \wedge \\ (F_0(g_1(y'_1), y_2, g_3(y'_1, y_2, y'_3), \ldots, x, s) \rightarrow \\ F_0(y'_1, g_2(y'_1, y_2), y'_3, \ldots, x, N(x, s))))).\end{aligned} \tag{31}$$

Let $\gamma_{SK}(x, s)$ denote the formula (31). The formula $\forall x \forall s \gamma_{SK}(x, s)$ is a $\forall\Pi^b_1(\vec{g})$-formula. Clearly, $\forall x \forall s \gamma_{SK}(x, s) \vDash (\gamma)$.

Similar constructions Skolemizing (α), (β), (ϵ), and (ϵ'), give formulas α_{SK}, β_{SK}, ϵ'_{SK}, and ϵ''_{SK}. For the relativization of (α), we

set $d(n) = n$ without loss of generality. Let $\Psi(\vec{g}, i, N, c, F_0, G)$ be the $\forall\Pi^b_1(\vec{g}, i, N, c, F_0, G)$-formula

$$(\forall x)(\forall s)\alpha_{SK} \wedge (\forall x)\beta_{SK} \wedge (\forall x)(\forall s)\gamma_{SK} \wedge (\delta) \wedge (\forall x)(\forall s)\epsilon'_{SK} \wedge (\forall x)(\forall s)\epsilon''_{SK}.$$

In the definition of Ψ, the symbols for the functions $\vec{g}$, i, N, and c and the predicates F_0 and G are understood to be new symbols added to the language of bounded arithmetic. The functions $\vec{g}$ are the functions used for Skolemizing (α)-(ϵ'') (of course, with different g's for each Skolemized formula.) Then, by the relativized version of Theorem 1, $T^{k+1}_2(\vec{g}, i, N, c, F_0, G)$ proves

$$\Psi \;\rightarrow\; (\forall x)(\exists y{\leq}s)G(x, y), \tag{32}$$

for s the term $2^{d(|x|)} = 2^{|x|}$. Note that the symbols $\vec{g}, i, N, c, F_0, G$ are adjoined to the language of bounded arithmetic and are allowed to be used freely in induction formulas. This is entirely reasonable, since Theorem 3 shows that Π^b_k-PLS problems can be, without loss of generality, formalized in Skolem form: the functions used in the Skolem form are all given by $\hat{L}$-terms and thus may be used freely in induction formulas.

Since Ψ is a $\forall\Pi^b_1$-formula, (32) is equivalent to a $\forall\exists\Sigma^b_1$-sentence of the form $(\forall x)(\exists x')\Psi_M$ with Ψ_M bounded. The unbounded existential quantifier $\exists x'$ comes from the outer universal quantifiers of Ψ. This quantifier can be bounded by a term involving only x: this can be done on general principles by Parikh's theorem or, in the present case, an a priori bound can be obtained by the fact that the bound $d(|x|) = |x|$ is used. Thus x' can be bounded linearly in terms of x, i.e., $|x'| \leq c \cdot |x|$ for some constant c. Thus, formula (32) may be replaced by a $\forall\Sigma^b_1$-formula of the form $(\forall x)(\exists y{\leq}r(x))\Phi(x, y)$ for some term $r(x)$ and some Δ^b_0-formula Φ.

Conjecture 10 *The $\forall\Sigma^b_1$-formula (32) is not provable in $T^k_2(\vec{g}, i, N, c, F_0, G)$ or in $S^{k+1}_2(\vec{g}, i, N, c, F_0, G)$.*

The conjecture can be sharpened slightly by using a single unary predicate α to encode simultaneously all of $\vec{g}, i, N, c, F_0, G$. This is done in the usual way, letting α encode the predicates F_0 and G directly, and encode the functions $\vec{g}, i, N, c$ via their bit graphs. This results in a formula $\Psi^*(\alpha)$ that expresses the same conditions as $\Psi(\vec{g}, i, N, c, F_0, G)$, and a formula $G^*(\alpha, x, y)$ that expresses the same condition as $G(x, y)$. By Theorem 1, we again have that $T^{k+1}_2(\alpha)$ proves the $\forall\Sigma^b_1$-formula

$$\Psi^*(\alpha) \rightarrow \forall x\exists y G^*(\alpha, x, y). \tag{33}$$

Conjecture 10 can be equivalently expressed as stating that $T_2^k(\alpha)$ and $S_2^{k+1}(\alpha)$ do not prove (33). Like (32), the formula (33) can be replaced by a $\forall\Sigma_1^b$-formula of the form $(\forall x)(\exists y{\leq}r(x))\Phi^*(x,y,\alpha)$.

Conjecture 10 can be extended to a conjecture about bounded depth Frege proofs. Our bounded depth Frege proof system is formulated as a propositional Tait-style sequent calculus using connectives $\neg$, $\wedge$, and $\vee$. W.l.o.g., the negation signs are applied only to variables, so that a propositional formula consists of $\wedge$ and $\vee$ connectives applied to literals (a literal is a variable or a negated variable). The depth of a formula is defined to be the number of alternations of $\wedge$'s and $\vee$'s. Thus a literal is depth zero, and a disjunction or a conjunction of literals is a formula of depth one, etc. The depth of a sequent is defined to equal the maximum depth of the formulas in the sequent.

Letting $a \in \mathbb{N}$, define the formulas Ω_a to be the Paris-Wilkie propositional formulas $[\![(\exists y{\leq}r(a))\Phi^*(a,y,\alpha)]\!]$ which give propositional formulas that express the condition that (33) is true for $x = a$. The Paris-Wilkie translation is defined in the usual way: bounded quantifiers become conjunctions or disjunctions, atomic subformulas of the form $\alpha(s)$ are replaced by propositional variables p_i where i is the numeric value of s, and other atomic formulas are replaced by just *True* or *False*. Krajíček [12, §9.1] describes the Paris-Wilkie translation in more detail: the end result is that Σ_ℓ^b and Π_ℓ^b-formulas become quasipolynomial size, depth $\ell + \frac{1}{2}$ formulas (also called Σ-depth ℓ formulas). These formulas have depth $\ell + 1$, but the lowest level of boolean gates have polylogarithmic fanin (and hence only count as depth $\frac{1}{2}$).

The formulas Ω_a are quasipolynomial-size disjunctions of small (polylogarithmic-size) conjunctions; they thus have depth $1\frac{1}{2}$.

As is well-known, the Paris-Wilkie translation also applies to the T_2^{k+1}-proof of (33). This yields that there are tree-like propositional refutations of the formulas $\neg\Omega_a$ in which all formulas are quasipolynomial size and depth $k + 1\frac{1}{2}$, and in which each sequent has only a constant number of formulas, and which have height polylogarithmic in a. Using Lemmas 5 and 6 of Beckmann-Buss [2] (which are based on constructions of Krajíček and Razborov), this implies that there are quasipolynomial size, depth $k-\frac{1}{2}$, dag-like sequent calculus refutations of Ω_a.

In fact, we can do a little bit better than this: there are polynomial size, depth $k-1$, dag-like sequent calculus refutations of $\neg\Omega_a$. To prove this, consider the $T_2^{k+1}(\vec{g}, i, N, c, F)$-proof of (32). Referring back to the proof of Theorem 1, the main step in the proof is a use of minimization on the

formula (1). This minimization principle is proved by using induction with respect to the variable c on the formula

$$\neg(\exists c_0 \leq c)(\exists s \leq 2^{d(|x|)})(c_0 = c(x,s) \wedge F(x,s)).$$

Since $F(x,s)$ was chosen to have k bounded quantifiers, but no sharply bounded quantifier, the Paris-Wilkie translation transforms this to a depth $k+1$ formula. Likewise, the Skolemizations of the formulas (α)-(ϵ'') are all bounded formulas with $k+1$ blocks of bounded quantifiers, and no sharply bounded quantifiers. Thus these formulas, along with the rest of the T_2^{k+1}-proof, are transformed to depth $k+1$ propositional formulas. Finally, since $d(|x|) = |x|$, the proof obtained by the Paris-Wilkie translation is only polynomial size, instead of quasi-polynomial size. Since the proof is also tree-like, has height polylogarithmic in a, and contains only a constant number of formulas in each sequent, it follows from Theorem 10 of [2] that $\neg\Omega_a$ has polynomial size, depth $k-1$, dag-like sequent calculus refutations.

Conjecture 11 *The formulas $\neg\Omega_a$ do not have quasipolynomial size, depth $k - 1\frac{1}{2}$ dag-like sequent calculus refutations.*

Note that, by the correspondence given by the Paris-Wilkie translation, Conjecture 11 is a restatement of Conjecture 10 in a non-uniform setting.

The outermost connective of $\neg\Omega_a$ is a conjunction. By putting each conjunct in a separate sequent, and then replacing $\vee$'s with commas, the formula $\neg\Omega_a$ can be replaced by an equivalent set Ξ_a of sequents. Note that each sequent in Ξ_a contains only literals, so Ξ_a is a set of sequents of depth 0 and each sequent contains only polylogarithmically many literals. Conjecture 11 is then equivalent to stating that the sets Ξ_a do not have quasipolynomial size, depth $k - 1\frac{1}{2}$ dag-like sequent calculus refutations.

If Conjecture 11 could be established for $k > 1$, these sequents would be the first example of sequents of depth $< k$ that have quasipolynomial size constant depth refutations, but do not have quasipolynomial size depth k refutations.

References

[1] K. Aelig and A. Beckmann, *On the computational complexity of cut reduction*, in Proc. 23rd Annual IEEE Symp. on Logic in Computer Science (LICS'08), 2008, pp. 284–293.

[2] A. Beckmann and S. R. Buss, *Separation results for the size of constant-depth propositional proofs*, Annals of Pure and Applied Logic, 136 (2005), pp. 30–55.

[3] ——, *Characterization of definable search problems in bounded arithmetic via proof notations*. In preparation, 2008.

[4] S. R. Buss, *Bounded Arithmetic*, Bibliopolis, 1986. Revision of 1985 Princeton University Ph.D. thesis.

[5] ——, *Axiomatizations and conservation results for fragments of bounded arithmetic*, in Logic and Computation, proceedings of a Workshop held Carnegie-Mellon University, 1987, vol. 106 of Contemporary Mathematics, American Mathematical Society, 1990, pp. 57–84.

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

[7] ——, *An introduction to proof theory*, in Handbook of Proof Theory, S. R. Buss, ed., North-Holland, 1998, pp. 1–78.

[8] S. R. Buss and J. Krajíček, *An application of Boolean complexity to separation problems in bounded arithmetic*, Proc. London Math. Society, 69 (1994), pp. 1–21.

[9] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis, *How easy is local search?*, J. Comput. System Sci., 37 (1988), pp. 79–100.

[10] J. Krajíček, A. Skelley, and N. Thapen, *Np search problems in low fragments of bounded arithmetic*, Journal of Symbolic Logic, 72 (2007), pp. 649–672.

[11] J. Krajíček, *Fragments of bounded arithmetic and bounded query classes*, Transactions of the A.M.S., 338 (1993), pp. 587–598.

[12] ——, *Bounded Arithmetic, Propositional Calculus and Complexity Theory*, Cambridge University Press, Heidelberg, 1995.

[13] J. Krajíček and P. Pudlák, *Quantified propositional calculi and fragments of bounded arithmetic*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik, 36 (1990), pp. 29–46.

[14] J. Krajíček, P. Pudlák, and G. Takeuti, *Bounded arithmetic and the polynomial hierarchy*, Annals of Pure and Applied Logic, 52 (1991), pp. 143–153.

[15] R. J. Parikh, *Existence and feasibility in arithmetic*, Journal of Symbolic Logic, 36 (1971), pp. 494–508.

[16] J. B. Paris and A. J. Wilkie, *Counting problems in bounded arithmetic*, in Methods in Mathematical Logic, Lecture Notes in Mathematics #1130, Springer-Verlag, 1985, pp. 317–340.

[17] C. Pollett, *Arithmetic Theories with Prenex Normal Form Induction*, PhD thesis, University of California, San Diego, 1997.

[18] ——, *Structure and definability in general bounded arithmetic theories*, Annals of Pure and Applied Logic, 100 (1999), pp. 189–245.

[19] P. Pudlák, *Consistence and games — in search of combinatorial principles*, in Logic Colloquium '03, Association for Symbolic Logic, AK Peters, 2006, pp. 244–281.

[20] A. Skelley and N. Thapen, *The provable total search problems of bounded arithmetic*. Typeset manuscript, 2007.

[21] D. Zambella, *Notes on polynomially bounded arithmetic*, Journal of Symbolic Logic, 61 (1996), pp. 942–966.