Circuit Principles and Weak Pigeonhole Variants

Chris Pollett
214 MacQuarrie Hall,
Department of Computer Science
San Jose State University
One Washington Square, San Jose CA 95192.
Email: pollett@cs.sjsu.edu

Norman Danner
Department of Mathematics and Computer Science
Wesleyan University
Middletown, CT 06549-0128
Email: ndanner@wesleyan.edu

Abstract

This paper considers the relational versions of the surjective and multifunction weak pigeonhole principles for PV, Σ^{b}_1 and Θ^{b}_2-formulas. We show that the relational surjective pigeonhole principle for Θ^{b}_2 formulas in S^1_2 implies a circuit block-recognition principle which in turn implies the surjective weak pigeonhole principle for Σ^{b}_1 formulas. We introduce a class of predicates corresponding to poly-log length iterates of polynomial-time computable predicates and show that over R^2_2, the multifunction pigeonhole principle for such predicates is equivalent to an "iterative" circuit block-recognition principle. A consequence of this is that if R^2_3 proves this circuit iteration principle then RSA is vulnerable to quasi-polynomial time attacks.

Id: php.tex,v 1.137 2004/08/30 18:29:27 ndanner Exp

1 Introduction

The weak pigeonhole principle ($WPHP$) states that given a function from a set of size n^2 into a set of size n, there are two elements in the domain that map to the same element in the range. This principle gives one the ability to do a limited amount of counting with regard to the function in question. The weak pigeonhole principle has been used in the context of propositional proof complexity to define sequences of true formulas which do not have short resolution or constant depth Frege proofs [2][3]. It has also been well-studied in the context of first order logic. Here one adds the principle for some class of relations—for instance, the p-time computable relations or the Δ_0 relations—to a weak system of arithmetic and considers what new results are provable in the strengthened system. An early result of this type is that $I\Delta_0 + WPHP(\Delta_0)$ proves there are infinitely many primes [21]. The pigeonhole principles in both contexts are intimately related via well known translations of first order bounded arithmetics into sequences of propositional proofs [20][17].

Besides the traditional injective pigeonhole principle described above, many other flavors have been considered in the literature. These include the surjective pigeonhole principle which says that there is no surjective function from a set of size n onto a set of size $2n$, the bijective pigeonhole principle which combines the injective and surjective principles, and the multifunction pigeonhole principle which is like the injective principle but defined in terms of multifunctions rather than just functions. In weak theories of arithmetic it might not be provable that these pigeonhole notions coincide.

Recently, Jeřábek [12, §3] has shown that the surjective pigeonhole principle for p-time functions is connected with circuit lower bounds. He shows that in bounded arithmetic S^1_2 the surjective weak pigeonhole principle for p-time functions is equivalent to the statement that for each fixed $k > 0$ there is a string of length $2n^k$ which is hard for circuits of size n^k. Here S^1_2 is a theory which roughly has length induction for NP predicates. Given this result it is natural to ask whether the other forms of the pigeonhole principle can be connected to circuit principles. Jeřábek's result was for the pigeonhole principle expressed using p-time functions so it is further natural to try to extend his results to the case where the surjection is expressed as the graph of a function rather than by a function symbol, thereby allowing consideration of functions more complex than p-time.

As Aaronson notes in [1, §4.1], Razborov [24, App. C] argues that Shannon's counting argument cannot obviously be formalized in S^1_2. As a consequence, S^1_2 cannot, at least in a direct way, formalize Kannan's result [14] that there is a set in $\mathsf{NEXP}^{\mathsf{NP}}$ that is not in $\mathsf{P/poly}$. To a large degree, these statements are consequences of Parikh's Theorem which shows that S^1_2 cannot define functions of super-polynomial growth. Nevertheless, it is open whether S^1_2 can prove a "weak Kannan result"—the existence of sets A_k which require circuits of size greater than n^k for each fixed k. It is also still open whether, if for one fixed set A defined by a bounded arithmetic formula, S^1_2 can prove the sequence of statements: "A requires circuits of size greater than n^k", for each fixed $k > 0$. A positive answer to this latter question would imply S^1_2 could prove $\mathsf{P} \neq \mathsf{NP}$, and so, of course, $\mathsf{P} \neq \mathsf{NP}$ would hold in the real world. Jeřábek's result to some extent gives an upper bound on the theory required to prove a weak Kannan result, for once we know a hard string exists, if we can obtain a least such string, we can construct a fixed set which is hard for size-n^k circuits. This kind of argument can be carried out in the theory S^3_2, where S^i_2 is defined roughly as the theory with length induction for the ith level of polynomial hierarchy. This is because S^3_2 can do the necessary minimization and Paris *et al.* [21], as presented in Krajíček [15], have shown that S^3_2 proves the weak pigeonhole principle for p-time functions. It is interesting to ask whether one can make any progress on showing a matching lower bound on the theory required.

The intent of this paper is to show that to some extent all of the questions posed above can be answered. For the remainder of this introduction, $n = |x|$ for some x. Let $sWPHP(\Psi)$ (resp. $mWPHP(\Psi)$) denote the surjective (resp. multifunction) weak pigeonhole principle for the relations in Ψ. We show that over S^1_2, $sWPHP(\Theta^{\mathsf{b}}_2)$ implies there is a string S of length $2n^k$ that is not block-recognized by any circuit (code) of size n^k. That is, there is no such circuit such that for $b < 2n^{k-1}$ and $s < 2^n$, $C(b, s)$ outputs 1 if and only if s is the b-th length-n block of S. Here Θ^{b}_2 (sometimes called $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_1)$ in the literature) is a class of formulas that precisely defines the sets in $P^{\mathsf{NP}}(\log)$, sets computable in polynomial time using at most logarithmically many oracle queries to an NP set. On the other hand, it is also shown that the existence of such a hard string for each k implies $sWPHP(\Sigma^{\mathsf{b}}_1)$. Here the Σ^{b}_1-formulas correspond to the NP-sets. The reason for the slight gap is that specifying the uniqueness of the block that is recognized slightly bumps up the complexity of the pigeonhole principle needed to show the circuit result, but it is not clear how to harness this added complexity in the reverse direction. For this direction, we adapt the proof of Jeřábek's result to amplify a surjection $f : 2^n \to 2^{2n}$ to a surjection from 2^n onto 2^{2n^k} by a circuit that iterates f, but only "remembers" n bits of each computation.

For the multifunction case, let $\mathsf{ITER}(PV, \{||id||^{O(1)}\})$ denote the class of relations which can be computed as poly-log length iterations of a polynomial relation. The precise statement of this requires that when x is in such a set that is defined using a p-time relation, R, the sequence of computation values $R(x, y_1)$, $R(y_1, y_2)$, $\ldots$, $R(y_{t-1}, y_t)$ where t is $O(\log|x|)$, is uniquely defined. Note that just because we can recognize that $R(x, y_1)$ holds in p-time does not imply that there is a p-time function which computes y_1 from x, even if y_1 is polynomially bounded. This iteration principle is similar to one considered in Krajíček [16] in the context of the propositional proof complexity of the surjective pigeonhole principle. $\mathsf{ITER}(PV, \{||id||^{O(1)}\})$ contains PV and like Θ^{b}_2 is contained in the class Σ^{b}_2. We show that over R^2_2, $mWPHP(\mathsf{ITER}(PV, \{||id||^{O(1)}\}))$ is equivalent to the existence of a string $S < 2^{2n^k}$ that is not iteratively block-recognized by any circuit of size n^k. Hence, this principle over R^2_2 also implies $mWPHP(PV)$.

The last statement can be used to say something either about the likelihood of proving circuit lower bounds in weaker theories or about the security of RSA against various kind of attacks attacks. Krajíček and Pudlák [18] (see also the proof in Thapen [25, Lemma 3.15]) have shown that if there is an algorithm witnessing the injective weak pigeonhole principle for p-time functions (this is contained in $iWPHP(PV)$ which allows p-time relations) from a class $\mathcal{C}$ satisfying $\mathsf{P}^{\mathcal{C}} = \mathcal{C}$, then RSA is vulnerable to attacks from $\mathcal{C}$. Extending R^i_2 by a quasi-polynomial growth rate function, $\#_3$, gives the theories R^i_3. We apply Krajíček and Pudlák's result to conclude that if R^2_3 proves our circuit principle then RSA is vulnerable to quasi-polynomial time attacks. As R^2_3 contains the theories R^2_2 and S^1_2 the same result holds for them if they can prove our circuit principle. One can somewhat strengthen the theory R^2_3 and still obtain results which we believe are open. For example, if R^3_3 proves our circuit principle, then RSA is vulnerable to attacks computed in the polynomial closure of quasi-polynomial local search. All of these result rely on the fact that $mPHP(PV)$ implies $iPHP(PV)$. It is unknown over S^1_2 whether $sPHP(PV)$ implies $iPHP(PV)$, which is why an analogous result does not follow immediately from Jeřábek's result. As far as the authors know, it is open whether RSA is vulnerable to quasi-polynomial local search attacks; the main problem with breaking RSA using such an algorithm would be to find a neighborhood function which could indicate when one was getting closer to the message text. We make the observation here though that Hanika [11] extending work of Ferreira [10] has defined a generalized search class $GLS^\dagger$ which captures the Σ^{b}_1-definable multifunctions of S^3_2. Given that S^3_2 proves $mPHP(PV)$, and so also $iPHP(PV)$, it follows from Krajíček and Pudlák that RSA is vulnerable to attacks from the polynomial closure of $GLS^\dagger$. It also probably follows that there is some generalization of our circuit iteration principle corresponding to these search classes for which S^3_2 can prove lower bounds. Therefore, showing RSA is vulnerable to a quasi-polynomial local search based attack or showing lower bounds for our iteration principle in R^3_3 might not be much beyond current technology.

The format of the rest of this paper is as follows: In the next section the notations and theories to be discussed in the remainder of the paper will be introduced. In the third section, results concerning the weak pigeonhole principle are reviewed. The next two sections prove the results for the surjective and then the multifunction pigeonhole principle. Finally, the last section has the RSA results.

2 Preliminaries

This paper assumes familiarity with Buss [4] or Krajíček [15]. For completeness, the basic notations of bounded arithmetic are quickly reviewed. The specific bootstrapping we are following is from Pollett [23], but yield equivalents theories to the ones in the books just mentioned. The language L_2 contains the non-logical symbols: 0, S, $+$, $\cdot$, $=$, $\leq$, $\dot{-}$, $\lfloor \frac{1}{2}x \rfloor$, $|x|$, $\mathrm{MSP}(x, i)$ and $\#$. The symbols 0, $S(x) = x + 1$, $+$, $\cdot$, and $\leq$ have the usual meaning. The intended meaning of $x \dot{-} y$ is x minus y if this is greater than zero and zero otherwise, $\lfloor \frac{1}{2}x \rfloor$ is x divided by 2 rounded down, and $|x|$ is $\lceil \log_2(x+1) \rceil$, that is, the length of x in binary notation. $\mathrm{MSP}(x, i)$ stands for 'most significant part' and is intended to mean $\lfloor x/2^i \rfloor$. Finally, $x\#y$ reads 'x smash y' and is intended to mean $2^{|x||y|}$.

Natural hierarchies of prenex formulas can be defined in this language by counting alternations of bounded quantifiers. A formula consisting of $i + 1$ alternations of bounded quantifiers, the outermost of the form $\exists x \leq t$ ($\forall x \leq t$, respectively), followed by a matrix of sharply-bounded formulas, is a Σ^{b}_i-formula (Π^{b}_i-formula, respectively). Here sharply bounded means bounded by a term of the form $|t|$. The definition of Σ^{b}_i presented above is sometimes called $strict\Sigma^{\mathsf{b}}_i$ or $\hat{\Sigma}^{\mathsf{b}}_i$ in the literature. For the theories of this paper it is a provably equivalent class to what is usually considered elsewhere such as in Buss [4].

The theory $BASIC$ is axiomatized by all substitution instances of a finite set of quantifier-free axioms for the non-logical symbols of L_2. The theories considered in this paper are obtained from $BASIC$ by adding various forms of the induction scheme

$$A(0) \wedge (\forall x)(A(x) \supset A(Sx)) \supset (\forall x)A(t(x)).$$

$\mathcal{C}$-*IND*, -*LIND*, and -*LLIND* are obtained by taking $A \in \mathcal{C}$ and $t(x)$ to be x, $|x|$, and $||x||$, respectively. We will also have occasion to use the axiom schemes of Comprehension ($\mathcal{C}$-*COMP*):

$$(\exists w \leq 2^{|a|})(\forall i \leq |a|)(A(i,a) \Leftrightarrow Bit(i,w) = 1)$$

and Replacement ($\mathcal{C}$-*REPL*):

$$\forall x \leq |a| \, \exists y \leq bA(x,y) \supset \\ \exists w \leq SqBd(b+1,a) \forall i \leq |a| \big(\\ \beta(i,w) \leq b \wedge A(x, \beta(i,w))\big).$$

The theories R^i_2, S^i_2 and T^i_2 are obtained from $BASIC$ by adding respectively the Σ^{b}_i-*REPL*+Σ^{b}_i-*LLIND*, Σ^{b}_i-*LIND*, or Σ^{b}_i-*IND* axiom schema. The definition of R^i_2 has Σ^{b}_i-*REPL* added because we are working with prenex versions of Σ^{b}_i [23]. It is known that $S^{i+1}_2 \supseteq T^i_2 \supseteq S^i_2 \supseteq R^i_2 \supseteq S^{i-1}_2$ and that R^i_2 proves Σ^{b}_i-*COMP*. It is also known that if $R^{i+1}_2 \supseteq T^i_2$ then the polynomial hierarchy collapses [19][23].

Buss [4, §3] shows that if one adds new function symbols to S^1_2 for each polynomial-time function, together with axioms saying how the functions are recursively defined, one obtains a theory called $S^1_2(PV)$ which is conservative over S^1_2. For convenience, in this paper it will be assumed that these functions symbols are available in the language. We will use the notation FP to denote the defining equational axioms for these function symbols, and PV to denote the relations definable as open formulas involving these functions symbols. Among such functions, we will use the following frequently:

1. $Bit(i,w)$ is the i-th bit of w, $\mathrm{LSP}(w,i)$ is the $|w| - i$ least significant bits of w, and $w[a..b] = \mathrm{LSP}(\mathrm{MSP}(w,a), |w| - b)$ consists of bits a through b inclusive of w.
2. $SqBd(a,b) := 2(2a\#2b)$ is a bound on the value of any sequence of length $|b|+1$, each of whose components is $< a$, and $\beta(b,w)$ is the bth element of the sequence w.
3. $\hat{\beta}(b,n,w) = w[bn..(b+1)n-1]$ is the b-th length n block of bits of w.

Beyond the standard bounded arithmetic formula classes, we next define a class which has appeared in the literature under several different names:

Definition 1 *The class Θ^{b}_2 is the closure of Σ^{b}_1 under Boolean connectives and sharply-bounded quantification.*

Θ^{b}_2 is sometimes called in the literature $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_1)$ or $\Sigma^{\mathsf{b}}_2 \cap \Pi^{\mathsf{b}}_2$. Its sets corresponds to the complexity class $\Theta^p_2 := \mathsf{P}^{\mathsf{NP}}(\log)$[7].

Definition 2 *By $\exists! x \leq tA(x)$ we mean the usual abbreviation*

$$\exists x \leq tA(x) \wedge \forall x, x' \leq t((A(x) \wedge A(x')) \supset x = x').$$

We assume that the reader is familiar with the usual definition of a circuit. The predicate $Circuit(C,n)$ is true if C codes a circuit on n variables and $Output(C,i)$ is the PV-function computing the output of C on input i, where i represents a number in binary (assume some default value if $\forall n \neg Circuit(C,n)$ or $Circuit(C,n)$ but $i \geq 2^n$). These are straightforward to formulate in S^1_2 using the sequence coding available there and have appeared before in the literature [6].

3 Pigeonhole principles

In this paper, the following variants of the weak pigeonhole principle will be considered:

$iPHP(R)^m_n(\vec{z})$**:**

$$n < m \wedge \forall x < \exists! y < nR(x,y,\vec{z}) \supset \\ \exists x_1, x_2 < m \exists y < n \big(\\ x_1 \neq x_2 \wedge R(x_1,y,\vec{z}) \wedge R(x_2,y,\vec{z})\big)$$

$mPHP(R)^m_n(\vec{z})$**:**

$$n < m \wedge \forall x < m \exists y < nR(x,y,\vec{z}) \supset \\ \exists x_1, x_2 < m \exists y < n \big(\\ x_1 \neq x_2 \wedge R(x_1,y,\vec{z}) \wedge R(x_2,y,\vec{z})\big)$$

$sPHP(R)^m_n(\vec{z})$**:**

$$n < m \wedge \forall x < n \exists! y < mR(x,y,\vec{z}) \supset \\ \exists y < m \forall x < n \neg R(x,y,\vec{z})$$

where R is some predicate. These are respectively the injective, surjective, and multifunction variants of the principle. For any of these variants, the notation $PHP(R)^m_n$ will be used when there are no parameter variables or when the parameter variables are clear. The notation $PHP(\mathcal{C})^m_n$ will be used for the class of formulas $PHP(R)^m_n$ where $R \in \mathcal{C}$. The notation $WPHP(\mathcal{C})$ will be used for $PHP(\mathcal{C})^{n^2}_n$. When we refer to the scheme $v\,WPHP(R)^m_n$ for $v = s$, i, or m, we mean all instances of the corresponding sentence in which terms are substituted for m and n. When $\mathcal{C} = FP$, we understand the parameter list to have length 0, R to be a function symbol $f \in FP$, and $R(x,y)$ to be $f(x) = y$. We now make a few observations about the relations between the various principles.

Proposition 1 *$BASIC$ proves the following equivalences:*

(a) $mPHP(R)^m_n$ is equivalent to $mPHP'(R)^m_n$ where $mPHP'(R)^m_n$ is

$$\forall x_1, x_2 < m \forall y < n(R(x_1,y) \wedge R(x_2,y) \supset x_1 = x_2) \supset \\ \neg(n < m) \vee \exists x < m \forall y < n \neg R(x,y)\,.$$

(b) $sPHP(R)^m_n$ is equivalent to

$$\forall y < n \exists x < mR(x,y) \supset mPHP'(R)^m_n.$$

(c) $iPHP(R)^m_n$ is equivalent to

$$\forall x < m \forall y_1 < n \forall y_2 < n \big(\\ R(x,y_1) \wedge R(x,y_2) \supset y_1 = y_2\big) \supset mPHP(R)^m_n.$$

Proof. This argument will also hold if we had written parameter variables. The statement (a) follows because $mPHP'(R)^m_n$ is just the contrapositive of $mPHP(R)^m_n$. (b) follows because the condition $\forall y < n \exists x < m R(x,y)$ says R is a total multifunction from $y < n$ to $x < m$ and the premise of $mPHP'(R)^m_n$ guarantees this multifunction is a function. Finally, (c) follows since the condition

$$\forall x < m \forall y_1 < n \forall y_2 < n (R(x,y_1) \wedge R(x,y_2) \supset y_1 = y_2)$$

says R is a partial function from $x < m$ to $y < n$ and the premise of $mPHP(R)^m_n$ guarantees this function is total. □

Corollary 2 *BASIC proves $mPHP(R)^m_n$ implies both $sPHP(R)^m_n$ and $iPHP(R)^m_n$.*

Proposition 3 *For each pigeonhole variant $v = m, s, i$, the theories $T^1_2(R)$ and $S^2_2(R)$ prove that $vPHP^{n^2}_n(R) \supset vPHP^{2n}_n(R)$,*

Proof. (Sketch) The $T^1_2(R)$ results follows from the $S^2_2(R)$ results since the formulas in question are boolean combinations of Σ^{b}_2-formulas and $S^2_2(R)$ is conservative over $T^1_2(R)$ for such formulas. The basic idea of the proof for $S^2_2(R)$ is to show $\neg vPHP^{2n}_n(R) \supset \neg vPHP^{n^2}_n(R)$. To do this in each case one iterates $|n|$ times the $2n$ into n function or multifunction (or n onto $2n$ function) violating $vPHP^{n^2}_n(R)$. □

It is unknown whether $mPHP(\Sigma^{\mathsf{b}}_1)^m_n$ is equivalent to $vPHP(\Sigma^{\mathsf{b}}_1)^m_n$ over S^1_2 for $v = s$ or i. Paris, Wilkie, Woods [21] showed that $S_2 \vdash mWPHP(\Delta_0)$, where Δ_0 is the class of bounded formulas. Krajíček [15] has sharpened this to:

Lemma 4 *$T^2_2(R) \vdash mWPHP(R)$. Hence, $T^2_2 \vdash mWPHP(PV)$ and in particular $T^2_2 \vdash sWPHP(PV)$.*

4 Surjective pigeonhole principle and block-recognition

Jeřábek shows in [12] that over S^1_2, the surjective weak pigeonhole principle is equivalent to the claim that there is a string hard string of length $2n^k$ for circuits of size n^k. The following can be shown to be equivalent to Jeřábek's result; the main difference is the notation, which here corresponds to the notation we will use for our later results:

Theorem 5 *([12, Lemma 3.2, Proposition 3.5]) Let $n = |z|$. Over S^1_2, the scheme $sWPHP(FP)^{n^2}_n$ is equivalent to the scheme*

$$\exists S < 2^{2n^k} \forall C < 2^{n^k} \exists i < 2n^k \big(\mathit{Circuit}(C, |2n^k|) \supset \mathit{Output}(C,i) \neq \mathit{Bit}(i,S) \big)$$

for $k = 0, 1, \ldots$.

We begin by giving modified versions of Jeřábek's results for relational versions of the surjective weak pigeonhole principle. To simplify the notation a bit in this section, we often abuse notation and write $C(i)$ to denote $\mathit{Output}(C,i)$.

Definition 3 *Let C be a circuit on $|\lceil m/n \rceil| + n$ input variables. We say that C n-block-recognizes $S < 2^m$ for all $i < \lceil m/n \rceil$ and $s < 2^n$, $C(i,s)$ is true iff $s = \hat{\beta}(i,n,S)$.*

The predicate $\mathit{Fits}(C,S,m,n)$ says that $C(\cdot,\cdot)$ has the right shape for n-block-recognizing $S < 2^m$:

$$\mathit{Circuit}(C, |\lceil m/n \rceil| + n) \wedge S < 2^m.$$

Let $\mathit{BlockRec}(C,S,m,n)$ be the formula that says C n-block-recognizes $S < 2^m$:

$$\mathit{Fits}(C,S,m,n) \wedge \\ \forall i < \lceil m/n \rceil \, \exists! s < 2^n \big(C(i,s) \wedge C(i, \hat{\beta}(i,n,S)) \big).$$

Proposition 6 *Let $n = |z|$. For each $k > 0$, $S^1_2 + sWPHP(\Theta^{\mathsf{b}}_2)$ proves $\exists S < 2^{2n^k} \forall C < 2^{n^k} \neg \mathit{BlockRec}(C,S,2n^k,n)$.*

Proof. Reason in S^1_2. Existence of S in

$$\forall C < 2^{n^k} \exists! S < 2^{2n^k} \Big[\\ \big[\mathit{Fits}(C,S,2n^k,n) \wedge \\ \forall i < 2n^{k-1} (\exists! s < 2^n C(i,s) \wedge C(i,\hat{\beta}(i,n,S)))\big] \vee \\ \big[(\neg \mathit{Fits}(C,S,2n^k,n) \vee \\ \exists i < 2n^{k-1} \neg \exists! s < 2^n C(i,s)\big) \wedge S = 0\big]\Big]$$

is provable using PV-$REPL$ as follows: Fix $C < 2^{n^k}$. If C is not of the correct shape, then S will be 0 and the result holds. So assume $\mathit{Circuit}(C, |2n^{k-1}| + n)$ and $\forall i < 2n^{k-1} \exists! s < 2^n C(i,s)$. Using Σ^{b}_1-$COMP$, one can show one can define any block of S bit-by-bit. Then using Σ^{b}_1-$REPL$ one can define all the blocks in a single string. Uniqueness of S follows by proving length induction first on the bits in two strings in a block and then by length induction on the blocks. Since the predicate in brackets is Θ^{b}_2, one can apply $sWPHP(\Theta^{\mathsf{b}}_2)$ to conclude that there is some $S < 2^{2n^k}$ such that for all $C < 2^{n^k}$ the predicate in brackets fails. Then in particular, the first disjunct must fail, which completes the proof of the Theorem. □

As a corollary to the proof, we have the following result:

Proposition 7 *Let $n = |z|$. For each $k > 0$, $S^1_2 + sWPHP(FP)$ proves $\exists S < 2^{2n^k} \forall C < 2^{n^k} \neg \mathit{BlockRec}(C,S,2n^k,|n|)$.*

Proof. The same argument applies, but now we note that the condition on C is PV because the quantifiers in the uniqueness criterion are sharply bounded, so $sWPHP(PV)$ applies. But then this condition defines a PV-function, so only the functional version of $sWPHP$ is needed. □

Theorem 8 *Let T be the theory obtained from S^1_2 by adding the axiom*

$$\exists S < 2^{2n^k} \forall C < 2^{n^k} \neg \mathit{BlockRec}(C,S,2n^k,n)$$

for each $n = |z|$ and $k > 0$. Then T proves $sWPHP(\Sigma^{\mathsf{b}}_1)$.

Proof. It suffices to argue in S^1_2 that if there is a Σ^{b}_1-relation $R(x,y)$ that is the graph of a surjection f from 2^n onto 2^{2n} (where without loss of generality $n = |z|$ for some z), then $\forall S < 2^{2n^k} \exists C < 2^{n^k} \mathit{BlockRec}(C,S,2n^k,n)$. Note that assuming the existence of such a relation R does *not* allow us to assume there is a function symbol in PV for f, since we do not have that S^1_2 *proves* that R is the graph of a function. Say that R has the form $\exists z < 2^{n^\ell} R_0(x,y,z)$ where R_0 is PV and let C_0 be a circuit on variables $x_0,\ldots,x_{n-1},y_0,\ldots,y_{2n-1},z_0,\ldots,z_q$ ($q = n^\ell$) that outputs 1 exactly when $R_0(x,y,z)$ holds (here, $Bit(i,x) = x_i$, etc.). Furthermore, let ℓ' be such that C_0 has size $O(n^{\ell'})$. We will use C_0 to construct circuits $G_i(u,x,y,w)$ where $u < 2^i$, $x,y < 2^n$, and w is a sequence of length i, each of whose elements has size bounded by $2n+q$. G_i is intended to represent a surjection from 2^n onto $2^{2^i n}$ by repeatedly applying f to x and taking the left- or right-half of the result according to the bits of u. Our final circuit C will be obtained by fixing i and "hard-coding" w. Specifically, the predicate computed by G_i is defined as follows:

$$\begin{aligned} G_0(u,x,y,w) :=&(u=0) \wedge (x=y)\\ G_{i+1}(u,x,y,w) :=& u < 2^{i+1} \wedge\\ & G_i(\mathrm{DMSB}(u),\\ &\quad \mathrm{cond}(\mathrm{MSB}(u), w[n..2n-1],\\ &\qquad w[0..n-1]),\\ &\quad y, \mathrm{MSP}(w, 2n+q)) \wedge\\ & C_0(x, w[0..2n-1],\\ &\quad w[2n..2n+q-1]). \end{aligned}$$

where $\mathrm{DMSB}(u) = \mathrm{LSP}(a, |a|-1)$ is a with its most significant bit deleted, $\mathrm{MSB}(a) = \mathrm{MSP}(a,1)$ is the most significant bit of a, and $\mathrm{cond}(b,c,d)$ is either c or d as per whether $b=0$ or $b=1$. Formally, we are defining a function $\bar{G}(i)$, where $\bar{G}(i)$ is the code of the circuit computing the predicate G_i; $\bar{G}(i+1)$ is defined recursively from the code returned by $\bar{G}(i)$. Thus, when we write $G_i(u,x,y,w)$, we really mean $Output(\bar{G}(i),u,x,y,w)$. Following Jeřábek, if $r = ||z||$ for some z and $i < r$, then $G_i(u,x,y,z)$ is Σ^{b}_1-definable and we can prove

1. For any $S < 2^{2^r n}$,

   $$\begin{gathered}\exists e < SqBd(n, 2^{2^{r-i}}) \exists w < SqBd(i(2n{+}q), 2^{2^{r-i}})\\ \forall u < 2^i \forall v < 2^{r-i} G_i(u, (e)_v, \hat{\beta}(2^i v{+}u, n, S), (w)_v),\end{gathered}$$

   (Σ^{b}_1-*LIND* on $i \le r$). Since $r = ||z||$ and $i \le r$, this predicate is in fact Σ^{b}_1. In particular, taking $i = r$ we have that

   $$\exists e < 2^n \exists w < 2^{r(2n+q)} \forall u < 2^r G_r(u, e, \hat{\beta}(u,n,S), w).$$

2. $$\begin{gathered}\forall i \forall u < 2^{i+1} \forall e < 2^n \forall y,y' < 2^{2n} \forall w,w' < 2^{i(2n+q)} [\\ (G_i(u,e,y,w) \wedge G_i(u,e,y',w')) \supset y = y'].\end{gathered}$$

   (Π^{b}_1-*LIND* on i).

3. The size of G_i is $O(in^{\ell'+1})$.

Now suppose that $S < 2^{2n^k}$ and let $r = \left|2n^{k-1}\right| = (k-1)|n| + 1$, so that $2^r n = 2n^k$. Then as we just showed, there are (provably in S^1_2) e and w such that $G_r(\cdot, e, \cdot, w)$ n-block-recognizes S. Let $C_r(i,s) = G_r(i,e,s,w)$. The size of C_r is $\le c((k-1)|n| + 1)n^{\ell'+1} \le c'kn^{\ell'+2}$ for some c and c'. Furthermore, any circuit of size m can be given a code of size $\le 2m(|m|+1)$. Thus, if we take k large enough so that $n^k \ge 2c'kn^{\ell'+2}(\left|c'kn^{\ell'+2}\right|+1)$, then $C_{(k-1)|n|+1} < 2^{n^k}$ is a circuit that n-block recognizes S. Since S was chosen arbitrarily, this completes the proof. □

Combining Propositions 6 and 7 with Theorem 8, we have the following inclusions of theories, where the block-recognition axiom schemes range over all $n = |z|$ and $k > 0$:

$$\begin{gathered} S^1_2 + s\,WPHP(\Theta^{\mathsf{b}}_2) \supseteq\\ S^1_2 + \exists S < 2^{2n^k} \forall C < 2^{n^k} \neg BlockRec(C, S, 2n^k, n) \supseteq\\ S^1_2 + s\,WPHP(\Sigma^{\mathsf{b}}_1) \supseteq S^1_2 + s\,WPHP(PV) \supseteq\\ S^1_2 + s\,WPHP(FP) \supseteq\\ S^1_2 + \exists S < 2^{2n^k} \forall C < 2^{n^k} \neg BlockRec(C, S, 2n^k, |n|). \end{gathered}$$

We have the following curious corollary:

Corollary 9

$$\begin{gathered} S^1_2 + \forall S < 2^{2n^k} \exists C < 2^{n^k} BlockRec(C, S, 2n^k, |n|) \supseteq\\ S^1_2 + \forall S < 2^{2n^k} \exists C < 2^{n^k} BlockRec(C, S, 2n^k, n). \end{gathered}$$

where the block-recognition axiom schemes range over all $n = |z|$ and $k > 0$.

Note that the "obvious" approach to proving this fails. One can construct an n-block-recognizer by combining many copies of an $|n|$-block-recognizer, each of size n^k; however, the resulting circuit (code) will possibly have size greater than n^k.

Referring back to Theorem 5, we see what appears to be a peculiarity. Jeřábek shows that $s\,WPHP(FP)^{n^2}_n$ is equivalent to the claim that for each k there is $S < 2^{2n^k}$ that is not computed by any circuit (code) of size n^k. This in turn seems to be equivalent to the claim that for each k there is S of size $< 2n^k$ that is not 1-block recognized by any circuit (code) of size n^k, since 1-block-recognition is obviously equivalent to computability. However, this is not quite the case. Suppose that $S < 2^{2n^k}$ is computed by $C < 2^{n^k}$. A circuit $C'(i,b)$ that 1-block-recognizes S may be built from $C(i)$ by comparing b to $C(i)$ and outputting 0 or 1 as appropriate. However, if C' is obtained from C by adding c new gates to do this calculation, then $|C'| \ge |C| + c\,|n|$, and so it may not be the case that $|C'| < n^k$. Thus being computable by a circuit (code) of size $< n^k$ does not imply being 1-block-recognizable by a circuit (code) of size $< n^k$.

We conclude this section with a variant of $s\,WPHP$. Let $s\,WPHP(D,R)^m_n$ be the following principle:

$$\begin{gathered}(n < m \wedge \forall x < n(D(x) \supset \exists! y < mR(x,y))) \supset\\ \exists y < m \forall x < n(D(x) \supset \neg R(x,y)).\end{gathered}$$

In other words, R cannot be the graph of a surjective function from D (a subset of $\{0,\ldots,n\}$) onto $\{0,\ldots,m\}$. $s\,WPHP(\mathcal{D},\mathcal{C})$ is the set of principles $s\,WPHP(D,R)^{n^2}_n$ for $D \in \mathcal{D}$ and $R \in \mathcal{C}$. This bears some similarity to Thapen's alternative version of a multifunction pigeonhole principle which states that a function cannot be a surjection from a subset of n onto m [25, Definition 3.1(4)]. There, however, the complexity of the domain is left unspecified, and it is not certain what the exact relationship between the two principles is. The proof of the following is similar to that of Proposition 6:

Proposition 10 *Let $n = |z|$. For each $k > 0$, $S^1_2 + sWPHP(\Theta^b_2, PV)$ proves*

$$\exists S < 2^{2n^k} \forall C < 2^{n^k} \neg BlockRec(C, S, 2n^k, n).$$

5 *mWPHP* and Iteration

The next definition will be used to state circuit principles connected with *mWPHP*.

Definition 4 *Given a class $\mathcal{C}$ of formulas and a set τ of terms, $\mathsf{ITER}(\mathcal{C}, \tau)$ consists of formulas of the form*

$$Iter(R, B, E, z_1, \ldots, z_n, s, t) :=$$
$$\exists w \leq SqBd(s, 2^{\min(t+1,|r|)}) Comp(R, B, E, w, \vec{z}, s, t)$$

where $R(i, u, v, \vec{z}) \in \mathcal{C}$, r, $B(\vec{z})$ and $E(\vec{z})$ are terms, and $Comp(R, B, E, w, \vec{z}, s, t)$ is

$$Seq(w) \wedge Len(w) = t + 2 \wedge$$
$$\forall i \leq t\Big(\beta(i, w) \leq s \wedge R(i, \beta(i, w), \beta(i + 1, w), \vec{z}) \wedge$$
$$\forall v \leq s(R(i, \beta(i, w), v, \vec{z}) \supset v = \beta(i + 1, w))\Big) \wedge$$
$$\beta(0, w) = B(\vec{z}) \wedge \beta(t + 1, w) = E(\vec{z}).$$

It is permissible that R not depend on all of the variables $\vec{z}$; when this is a case for a specific R (such as *Out*, in Definition 5), we will omit mention of the unused variables. Formally we should declare the parameters upon which R depends and rewrite *Comp* to list only those parameters, but we will instead informally refer to R "depending" on z_i or not (and similarly for B and E).

The predicate *Iter* is related to a predicate in Krajíček [16] which was studied in the context of propositional proof complexity. Where it is clear that a suitable r can be found so that $t + 1 < |r|$ then, we will sometimes just write 2^{t+1} for $2^{\min(t+1,|r|)}$. The latter form is introduced only because the exponential function is not necessarily total in bounded arithmetic theories. The intuition behind $Iter(R, B, E, \vec{z}, s, t)$ is that it verifies that there is a $(t + 1)$-stepped computation from initial value $B(\vec{z})$ to final value $E(\vec{z})$ each step of which follows uniquely from the previous according to R. The values at each step are bounded by s. It should be observed that if s is of polynomial length then the ability to verify in p-time that a string for the $(i+1)$-st step follows from a string for i-th step does not entail that there is a p-time function computing the $(i + 1)$-st step from the i-th step.

Write $\{||id||^{O(1)}\}$ for the set of terms of the form $||t||^m$ for some term t and some fixed number m in the language. The following lemmas establish the basic properties of $\mathsf{ITER}(\mathcal{C}, \tau)$.

Lemma 11

1. *The theory S^1_2 proves that $\mathsf{ITER}(PV, \{||id||^{O(1)}\})$ contains the PV predicates.*
2. *For $R(i, u, v, j, \vec{z}) \in PV$, any terms $B(j, \vec{z})$ and $E(j, \vec{z})$, and any term $h(\vec{z})$, there is $R^*(i, u, v, \vec{z}) \in PV$ and terms $B^*(\vec{z})$ and $E^*(\vec{z})$ such that R^2_2 proves*

$$\forall j \leq |h(\vec{z})|\, Iter(R, B, E, j, \vec{z}, s, ||t||^m) \Leftrightarrow$$
$$Iter(R^*, B^*, E^*, \vec{z}, s(|h| + 1), ||t||^m).$$

In other words, $\mathsf{ITER}(PV, \{||id||^{O(1)}\})$ is closed under sharply bounded universal quantification.

Proof. (1) Suppose $R(\vec{z})$ is a PV predicate. Consider the predicate $R^*(i, a, b, \vec{z})$ defined as

$$(i = i \wedge a = 0 \wedge b = 0 \wedge R(\vec{z})).$$

Then $Iter(R^*, 0, 0, \vec{z}, 1, ||t||^m)$ will compute the same predicate as $R(\vec{z})$ (regardless of t).

(2) The left-hand-side says that for each $j \leq |h|$, R "maps" $B(j, \vec{z})$ to $E(j, \vec{z})$ in $||t||^m$ steps. R^* will map the *sequence* $\langle B(0, \vec{z}), \ldots, B(|h|, \vec{z})\rangle$ to the sequence $\langle E(0, \vec{z}), \ldots, E(|h|, \vec{z})\rangle$ in the same number of steps. By Σ^b_2-*REPL* the left-hand-side is equivalent to

$$\exists w \leq SqBd(SqBd(s, 2^{||t||^m+1}), 2^{|h|+1}) \forall j \leq |h|$$
$$Comp(R, B, E, \beta(j, w), j, \vec{z}, s, ||t||^m).$$

Let $R^*(i, u, v, \vec{z})$ be the predicate

$$u \leq SqBd(s, 2^{|h|}) \wedge v \leq SqBd(s, 2^{|h|}) \wedge$$
$$Seq(u) \wedge Seq(v) \wedge$$
$$\forall j \leq |h|\, (R(i, \beta(j, u), \beta(u, v), j, \vec{z}).$$

Let $B^* = \langle B(0, \vec{z}), \ldots, B(|h|, \vec{z})\rangle$ and $E^* = \langle E(0, \vec{z}), \ldots, E(|h|, \vec{z})\rangle$. Then the sequence W defined by $\beta(i, W) = \langle \beta(0, \beta(i, w)), \ldots, \beta(|h|, \beta(i, w))\rangle$ is a witness to the right-hand-side; since W is computable in polynomial-time from w, it is definable in $R^2_2 \supseteq S^1_2$. □

Lemma 12 *R^2_2 proves $Uniq(||t||^m)$ for fixed m where $Uniq(a)$ is the formula*

$$Comp(R, B, E_1, w_1, \vec{z}, s, a) \wedge$$
$$Comp(R, B, E_2, w_2, \vec{z'}, s, a) \supset w_1 = w_2 \wedge E_1 = E_2$$

where $z'_i = z_i$ if R or B depends on z_i.

Proof. First, note that *Comp* is equivalent to a Π^b_1 formula so $Uniq(|x|)$ will be equivalent to a Σ^b_1 formula. Also, given the definition of *Comp* uniqueness of w in $Comp(R, B, E, w, \vec{z}, s, a)$ guarantees uniqueness of E. Let $Comp'(R, B, w, \vec{z}, s, a)$ be the same predicate as *Comp* except where the last conjunct checking that the final value of the sequence is E has been discarded. Let $Uniq'(a)$ be

$$\forall w_1, w_2 \leq SqBd(s, 2^a)\big[$$
$$Comp'(R, B, w_1, \vec{z}, s, a) \wedge Comp'(R, B, w_2, \vec{z}, s, a) \supset$$
$$w_1 = w_2\big].$$

Given our discussion this will be a Π^b_2-formula and $Uniq'(a) \supset Uniq(a)$, so it suffices to prove $Uniq'(|x|)$ to complete the proof. The theory S^1_2 proves $Uniq'(0)$ since any sequence satisfying the $Comp'$ expression will in this case consist of only two elements, the first elements must be x and the third conjunct in the definition of $Comp'$ forces the uniqueness of the second block. This third conjunct in the definition of $Comp'$ can also be used to show $Uniq'(a) \supset Uniq'(Sa)$; the relevant fact is that

$$Comp'(R, B, w, \vec{z}, s, Sa) \supset$$
$$Comp'(R, B, FRONT(w), \vec{z}, s, a).$$

Here $FRONT(w)$ is the p-time function that returns all but the last element of w. The point is $Uniq'(a)$ guarantees the uniqueness of $FRONT(w)$ since it has size less than $SqBd(s, 2^a)$, and the third conjunct will guarantee the uniqueness of the element that is added to $FRONT(w)$ to obtain w. Hence, w will also be unique. Thus using Π_2^{b}-$LLIND$ and standard speed-up of induction techniques [23], the theory R_2^2 proves $Uniq'(||x||^m)$ and hence $Uniq(||x||^m)$. □

Definition 5

1. *Let $Out(i, u, v, b, C)$ be the predicate that is true when C is a circuit on $|i|+|u|+|v|+|b|$ variables and $C(i, u, v, b)$ is true.*
2. *Let $IterBlockRec(C, S, c, x, t)$ be*

$$\forall b < n^{k-1}\Big(\mathit{Iter}(Out, c, \hat\beta(b, 2\,|x|\,, S), b, C, c, S, 2^{|c|}, t)\Big).$$

By Lemma 11, this is an iteration predicate. Note that Out depends only on the parameters b and C.

3. *Let $CompOutput(w, C, S, c, b, x, t)$ be*

$$Comp(Out, c, \hat\beta(b, 2\,|x|\,, S), w, b, C, c, S, 2^{|c|}, t)$$

so that $IterBlockRec(C, S, c, x, t)$ is

$$\forall b < n^{k-1}\exists w \le SqBd(2^{|c|}, 2^{t+1})\Big(CompOutput(w, C, S, c, b, x, t)\Big).$$

Theorem 13 *Let $n = |x|$. For $k > 1$, $||t||^j$ in $\{||id||^{O(1)}\}$, the theory R_2^2+$mWPHP(\mathsf{ITER}(PV, \{||id||^{O(1)}\}))$ proves the principle*

$$\exists S < 2^{2n^k}\forall C < 2^{n^k-2n}\forall c < 2^{2n} \neg \mathit{IterBlockRec}(C, S, c, x, ||t||^j).$$

The use of two separate variables C and c is a notational convenience: we could replace them by a single variable C' of size 2^{n^k} and use MSP *and* LSP *to obtain values for these two variables.*

Proof. Reason in R_2^2, and suppose that

$$\forall S < 2^{2n^k}\exists C < 2^{n^k-2n}\exists c < 2^{2n}\Big[\forall b < n^{k-1}\exists w \le SqBd(2^{2n}, 2^{||t||^j+1}) CompOutput(w, C, S, c, b, x, ||t||^j)\Big].$$

Using Lemma 11, the expression in square brackets is equivalent in R_2^2 to an $\mathsf{ITER}(PV, \{||id||^{O(1)}\})$ predicate. So by $mWPHP(\mathsf{ITER}(PV, \{||id||^{O(1)}\}))$ there are $S_1 \ne S_2 < 2^{2n^k}$, $C < 2^{n^k-2n}$, $c < 2^{2n}$ such that

$$\forall b < n^{k-1}\exists w \le SqBd(2^{2n}, 2^{||t||^j+1})\Big(CompOutput(w, S, C, c, b, x, ||t||^j)\Big)$$

for $i = 1, 2$. Fix any $b < n^{k-1}$. By Lemma 12, there is a unique pair (w, v) such that $Comp(Out, c, v, w, b, C, c, S_i, 2^{|c|}, ||t||^j)$ for $i = 1, 2$ (note that Out does not depend on S), and so we conclude that for each $b < n^{k-1}$ we have $\hat\beta(b, 2n, S_1) = \hat\beta(b, 2n, S_2)$. In other words, the b-th blocks of S_1 and S_2 are equal. Since b was chosen arbitrarily, all blocks of S_1 and S_2 are the same. By induction on the number of blocks, one shows that this implies that $S_1 = S_2$, a contradiction. □

Theorem 14 *Let $n = |x|$. Let T be the theory R_2^2 extended by the axioms*

$$\exists S < 2^{2n^k}\forall C < 2^{n^k-2n}\forall c < 2^{2n} \neg \mathit{IterBlockRec}(S, C, c, x, ||t||^j)$$

for each $k > 1$, $||t||^j$ in $\{||id||^{O(1)}\}$. Then (a) T proves $mWPHP(PV)$ and (b) T proves $mWPHP(\mathsf{ITER}(PV, \{||id||^{O(1)}\}))$.

Proof. (a) Assume that $R(x, y)$ is a PV-formula that is the graph of an injective multifunction from 2^{2n} into 2^n. Define $AMP'(S, c, j, x, w)$ to be the conjunction of the following statements:

1. $S < 2^{2^{j+1}n}$;
2. w is a sequence of length $j + 1$;
3. For $0 \le i \le j$, $\beta(i, w)$ is a sequence of length 2^i;
4. For $0 \le i \le j$ and $0 \le \ell < 2^i$, $|\beta(\ell, \beta(i, w))| \le 2n$;
5. For $0 \le i < j$ and $0 \le \ell < 2^i$,
$$R(\beta(2\ell, \beta(i + 1, w)), \mathrm{MSP}(\beta(\ell, \beta(i, w)), n));$$
6. For $0 \le i < j$ and $0 \le \ell < 2^i$,
$$R(\beta(2\ell + 1, \beta(i + 1, w)), \mathrm{LSP}(\beta(\ell, \beta(i, w)), n));$$
7. $\beta(0, \beta(0, w)) = c$;
8. For $0 \le \ell < 2^j$, $\beta(\ell, \beta(j, w)) = \hat\beta(\ell, 2n, S)$.

In other words, w is a "triangle" that consists of $j + 1$ rows, where the i-th row has 2^i blocks, and each block is of length at most $2n$; the 0-th row is c and the $(j + 1)$-st row is S. The i-th row is formed by using R to "compress" the blocks of the $(i + 1)$-st row. Let $AMP(S, c, j, x)$ be the predicate $\exists w \le SqBd(SqBd(2^{2n}, 2^{2^j-1}), 2^j)AMP'(S, c, j, x, w)$. As usual exponentials are 'cut-off', in this case by a term of the form $||r||$ for some r, so AMP is (equivalent to) a Σ_1^{b} formula over $BASIC$. By Π_2^{b}-$LLIND$ on j, one can show that $\forall S < 2^{2^{j+1}n}\exists c < 2^{2n}AMP(S, c, j, x)$ and hence conclude $\forall S < 2^{2^{k|n|+1}n}\exists c < 2^{2n}AMP(S, c, k\,|n|\,, x)$. For the induction step, given $S < 2^{2^{j+2}n}$, use R to compress adjacent length-n blocks in pairs to get $S' < 2^{2^{j+1}n}$ and then apply the induction hypothesis to get c such that $AMP(S', c, j, x)$. To show $AMP(S, c, j, x)$, take the sequence (triangle) w' given by $AMP(S', c, j, x)$ and add a new row consisting of the length-$2n$ blocks of S.

Now fix $S < 2^{2n^k}$ and take c such that $AMP(S, c, k|n|, x)$. Let $C(i, u, v, b)$ be the circuit that computes the predicate

$$R\Big(v, \mathrm{cond}\big(Bit((k-1)|n| - i, b), \mathrm{MSP}(u, n), \mathrm{LSP}(u, n)\big)\Big) \wedge (i = 0 \supset u = c).$$

Take any $b < n^{k-1}$ (the number of length-$2n$ blocks in S). Let w be the sequence (triangle) given by $AMP(S, c, k|n|, x)$ and define a new sequence v by $\beta(i, v)) = \beta(\mathrm{MSP}(b, i), \beta(i, w))$. In other words, v consists of the blocks in w starting at c and traversing the triangle to end at the b-th block of S in the last row. Then v is a sequence of $k|n|$ starting at c, ending at $\hat{\beta}(b, 2n, S)$ and for which $C(i, \beta(i, v), \beta(i+1, v))$ for each i; this follows from $AMP(S, c, k|n|, x)$. Uniqueness of each step follows from the fact that R is injective. As in the proof of Theorem 8, take k large enough so that we can assume $C < 2^{n^k - 2n}$; then by chasing definitions, we see that we have proved

$$\forall S < 2^{2n^k} \exists C < 2^{n^k - 2n} \exists c < 2^{2n} \; IterBlockRec(C, S, c, x, k\,||x||),$$

completing the proof of (a).

We now describe how to modify the proof of (a) to obtain a proof of (b). Let $Q := Iter(R, B, E, x, y, \vec{z}, s, ||t||^m)$ be a predicate such that $\neg m\,WPHP(Q)$. We are assuming that the injection from 2^{2n} to 2^n is on the variables x and y which are among the parameter variables of R, B, and E. We use the R in this Q to create a modified version of AMP, essentially where we have inserted between each step in the old AMP the iterations need to compute Q . Let $clen := ||t||^m + 3$. The new version of AMP' asserts:

1. $S < 2^{2^{j+1}n}$;
2. w is a sequence of length $j \cdot clen + 1$;
3. For $0 \le i \le j$, let $i' := i \cdot clen$; then $\beta(i', w)$ is a sequence of length 2^i and for $0 \le \ell < 2^i$, $|\beta(\ell, \beta(i', w))| \le 2n$.
4. For $0 \le i \le j$, and $i \cdot clen < i' < (i+1) \cdot clen$, $\beta(i', w)$ is a sequence of length 2^{i+1}, and for $0 \le \ell < 2^{i+1}$, $|\beta(\ell, \beta(i', w))| \le s$, and $R(i', \beta(\ell, \beta(i'+1, w)), \beta(\ell, \beta(i', w)))$;
5. For $0 < i < j$ and $0 \le \ell < 2^i$, let $x_{i,2\ell} := \beta(2\ell, \beta((i+1) \cdot clen, w))$, $Ly_{i,\ell} := \mathrm{MSP}(\beta(\ell, \beta(i \cdot clen, w)), n)$, $b_{i,2\ell} := \beta(2\ell, \beta((i+1) \cdot clen - 1, w))$, and $e_{i,2\ell} := \beta(2\ell, \beta(i \cdot clen + 1, w))$. Then $b_{i,2\ell} = Ly_{i,\ell} * B(x_{i,2\ell}, Ly_{i,\ell}, \vec{z})$ and $e_{i,2\ell} = Ly_{i,\ell} * E(x_{i,2\ell}, Ly_{i,\ell}, \vec{z})$. Here, $*$ denotes concatenation; we need this extra data when we construct the circuit that iteratively block-recognizes S.
6. For $0 \le i < j$ and $0 \le \ell < 2^i$, let $x_{i,2\ell+1} := \beta(2\ell+1, \beta((i+1) \cdot clen, w))$, $Ry_{i,\ell} := \mathrm{LSP}(\beta(\ell, \beta(i \cdot clen, w)), n)$, $b_{i,2\ell+1} := \beta(2\ell+1, \beta((i+1) \cdot clen - 1, w))$, and $e_{i,2\ell+1} := \beta(2\ell+1, \beta(i \cdot clen + 1, w))$. Then $b_{i,2\ell+1} = Ry_{i,\ell} * B(x_{i,2\ell+1}, Ry_{i,\ell}, \vec{z})$ and $e_{i,2\ell+1} = Ry_{i,\ell} * E(x_{i,2\ell+1}, Ry_{i,\ell}, \vec{z})$;
7. $\beta(0, \beta(0, w)) = c$;
8. For $0 \le \ell < 2^j$, $\beta(\ell, \beta(j \cdot clen, w)) = \hat{\beta}(\ell, 2n, S)$.

So this formula asserts that w is a "triangle of grids" that consists of $j+1$ grids. The i-th grid has 2^i columns and $||t||^m + 3$ rows. The last row of each grid corresponds to to a row of the triangle from the PV case. The immediately prior row consists of blocks of the form $B(x, y)$, where $x < 2^{2n}$ is the value in same column and next row and $y < 2^n$ is the value x is mapped to by Q. Then within a column, one traverses row-by-row by applying R. The new formula AMP is defined from this AMP' as before with a larger (but still polynomial bound) for w. Given that the universals above will be sharply bounded in R^2_2, this AMP is still equivalent to a Σ^{b}_1-formula. So one can prove $\forall S < 2^{2^{j+1}n} \exists c < 2^n AMP(S, c, j, \cdot ||t||^m, x)$ by induction on j in R^2_2 and hence conclude $\forall S < 2^{n^k} \exists c < 2^n AMP(S, c, k\,||x|| \cdot ||t||^m, x)$. The induction step is handled by using the fact that since $\neg m\,WPHP(Q)$, there is some unique sequence that makes Q an an injective map from 2^{2n} into 2^n. So given $S < 2^{2^{j+2}n}$, apply Q to the length-$2n$ blocks of S to obtain length-n blocks, and concatenate these to get $S' < 2^{2^{j+1}n}$. Apply the induction hypothesis to find c such that $AMP(S', c, j\,||t||^m, x)$. Let w' be the sequence such that $AMP'(S', c, j\,||t||^m, x, w')$ and now append the $clen$-row by 2^{j+1}-column "grid" that has the length-$2n$ blocks of S as the last row, and the the computation sequence of R in each column.

Now given $S < 2^{2n^k}$ we need a circuit $C(i, u, v, b)$ that recognizes a path through this "triangle of grids" that starts at c and ends at $\hat{\beta}(b, 2n, S)$. When $i = i' \cdot clen$, we are transitioning from the last row of a grid (corresponding to the rows of the triangle from the PV case); the circuit verifies that $\mathrm{LSP}(v, n) = B(u, \mathrm{MSP}(v, |v| - n))$. This is why we need to keep extra copies of the Ly's and Ry's in all the cells of the grid; without them, we could not perform this verification "locally." When $i = i' \cdot clen + 1$, we are transitioning from one grid to the next. The circuit just verifies that left- or right-half of v is $\mathrm{MSP}(u, n)$ according to the $((k-1)|n| - i')$-th bit of b. Finally, if $i = i' \cdot clen + i''$ with $i'' > 1$, we are transitioning according to R, so the circuit verifies that $R(i'', \mathrm{LSP}(v, n), \mathrm{LSP}(u, n))$. □

It would be interesting to know if $m\,WPHP(PV)$ implies $m\,WPHP(\mathsf{ITER}(PV, \{||id||^{O(1)}\}))$ over some non-trivial theory. To show this would seem to involve showing that from an iterated relation PV defining a injective multifunction from n^2 to n, one could somehow do away with the iteration and find a PV relation defining a injective multifunction from n^2 to n relation. It is not clear how this could be done.

6 Iteration and RSA

In this section, the provability of

$$\exists S < 2^{2n^k} \forall C < 2^{n^k - 2n} \forall c < 2^{2n} \; \neg IterBlockRec(C, S, c, x, k)$$

in R^2_3 and R^3_3 is connected to the security of RSA. To state our results, we define the class $qPLS$ and recall the definition of RSA.

The class PLS for polynomial search was defined by Johnson, Papadimitriou, and Yannakakis [13] and was shown to contain several interesting optimization problems. Buss and Krajíček [8] showed that the Σ^{b}_1 provably total multifunctions of T^1_2 can be characterized as the composition of a projection function with a PLS multifunction. By a quasi-polynomial, we mean a function of the form $2^{(\log n)^k}$ for some k. A natural generalization of PLS to quasi-polynomial time can be defined as follows:

Definition 6 *A* qPLS *problem consists of a quasi-polynomial time cost function c, a quasi-polynomial time neighborhood function N, and a quasi-polynomially bounded set of quasi-polynomial time solutions, defined by a predicate F. For an input x, the set $\{s : F(x,s)\}$ is the set of feasible solutions, the mapping $s \mapsto c(x,s)$ assigns a cost to each solution, and the mapping $s \mapsto N(x,s)$ maps solutions to solutions. The multifunction f defined by the* qPLS *problem is given by the relation $f(x) = y$ iff $F(x,y)$ and $c(x, N(x,y)) < c(x,y)$.*

Define $x\#_3 y$ as $2^{|x|\#|y|}$. Let R^i_3, S^i_3, and T^i_3 be the theories obtained from R^i_2,S^i_2, and T^i_2 by adding this symbol and its defining axiom. A straightforward generalization of Buss and Krajíček [8] shows that the Σ^{b}_1 provably total multifunctions of T^1_3 can be characterized as the composition of a projection function with a qPLS multifunction (see Pollett [22] for results of this type). A straightforward generalization of Buss [4] shows that the Σ^{b}_1-definable functions of S^1_3 are the quasi-polynomial time functions.

Recall what an instance of RSA is:

Definition 7 *An instance of RSA consists of a modulus $n = pq$ for two large primes p and q, exponents e and d which are mutual inverse modulo $(p-1)(q-1)$, a message $m < n$, and a ciphertext $c < n$ such that $c \equiv m^e \bmod n$ and $m \equiv c^d \bmod n$. The RSA instance is solved (hence, vulnerable) if given n, e, and c, one can compute m.*

We are now ready to present the main result of this section.

Theorem 15 *Let $n = |x|$. (a) If for any k, R^2_3 proves $\exists S < 2^{2n^k} \forall C < 2^{n^k - 2n} \forall c < 2^{2n} \neg IterBlockRec(S, C, c, x, k)$ then RSA is vulnerable to quasi-polynomial time based attacks. (b) If for any k, R^3_3 proves $\exists S < 2^{2n^k} \forall C < 2^{n^k-2n} \forall c < 2^{2n} \neg IterBlockRec(S, C, c, x, k)$ then RSA is vulnerable to polynomial time in* qPLS *based attacks.*

Proof. Both (a) and (b) are proved essentially the same way. By Buss, Krajíček, and Takeuti [9] it is known that R^3_3 is Σ^{b}_3 conservative over S^2_2, and by Buss [5], S^2_3 is Σ^{b}_2-conservative over T^1_3. Similarly, by Buss, Krajíček, and Takeuti [9], R^2_3 is Σ^{b}_2-conservative over S^1_3. Let T be either R^2_3 or R^3_3. Then if T proves $\exists S < 2^{2n^k} \forall C < 2^{n^k-2n} \forall c < 2^{2n} \neg IterBlockRec(S, C, c, x, k)$, then by Theorem 14, T proves $m\,WPHP(PV)$ so by Proposition 1, it also proves $i\,WPHP(PV)$ and thus $i\,WPHP(FP)$. The latter consists of formulas of the form:

$$\exists x < n^2 f(x, c) \geq n \vee \\ \exists x_1, x_2 < n^2 (x_1 \neq x_2 \wedge f(x_1, c) = f(x_2, c))$$

which are Σ^{b}_1-formulas. Hence, by the previously mentioned conservation results, one has in the case of R^2_3 that S^1_3 proves $i\,WPHP(FP)$ and in the case of R^3_3 that T^1_3 proves $i\,WPHP(FP)$. Using the witnessing arguments used to show the characterizations of Σ^{b}_1-definability in these latter theories one can say the following: (a) for R^2_3, there is a quasi-polynomial time function g which when given inputs c, a such that $\forall x < a^2 f(x,c) < a$ outputs $x_1 < x_2 < a^2$ such that $f(x_1, c) = f(x_2, c)$. (b) for R^3_3, g can be computed as a a projection of a qPLS problem. By Krajíček and Pudlák [18] there is polynomial time algorithm using g as an oracle which solves RSA. □

References

[1] S. Aaronson. Is P versus NP formally independent? *Bulletin of the European Association for Theoretical Computer Science*, pages 109–136, October 2003.

[2] M. Ajtai. The complexity of the pigeonhole principle. *Combinatorica*, 14(4):417–433, 1994.

[3] P. Beame, R. Impagliazzo, J. Krajíček, T. Pitassi, P. Pudlák, and A. Woods. Exponential lower bounds for the pigeonhole principle. In *Proceedings of the 24th Annual ACM Syposium on the Theory of Computing (Victoria, 1992)*, pages 200–221, New York, 1992. ACM Press.

[4] S. R. Buss. *Bounded Arithmetic.* Bibliopolis, Naples, 1986.

[5] S. R. Buss. Axiomatizations and conservation results for fragments of bounded arithmetic. In *Logic and Computation (Pittsburgh, PA, 1987)*, volume 106 of *Contemp. Math.*, pages 57–84. Amer. Math. Soc., Providence, RI, 1990.

[6] S. R. Buss. Bounded arithmetic, complexity, and cryptography. *Theoria*, 63:147–167, 1997.

[7] S. R. Buss and L. Hay. On truth-table reducibility to SAT. *Inform. and Comput.*, 91(1):86–102, 1991.

[8] S. R. Buss and J. Krajíček. An application of Boolean complexity to separation problems in bounded arithmetic. *Proc. London Math. Soc. (3)*, 69(1):1–21, 1994.

[9] S. R. Buss, J. Krajíček, and G. Takeuti. Provably total functions in bounded arithmetic theories R^i_3, U^i_2 and V^i_2. In *Arithmetic, Proof Theory, and Computational Complexity (Prague, 1991)*, volume 23 of *Oxford Logic Guides*, pages 116–161. Oxford Univ. Press, New York, 1993.

[10] F. Ferreira. What are the $\forall\Sigma^{\mathsf{b}}_1$-consequences of T^1_2 and T^2_2? *Ann. Pure Appl. Logic*, 75(1-2):79–88, 1995. Proof theory, provability logic, and computation (Berne, 1994).

[11] J. Hanika. *Search Problems and Bounded Arithmetic.* PhD thesis, Charles University, 2004.

[12] E. Jeřábek. Dual weak pigeonhole principle, Boolean complexity, and derandomization. *Annals of Pure and Applied Logic*, 129(1–3):1–37, 2003.

[13] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *J. Comput. System Sci.*, 37(1):79–100, 1988. 26th IEEE Conference on Foundations of Computer Science (Portland, OR, 1985).

[14] R. Kannan. Circuit-size lower bounds and nonreducibility to sparse sets. *Inform. and Control*, 55(1-3):40–56, 1982.

[15] J. Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*, volume 60 of *Encyclopedia of Mathematics and its Applications.* Cambridge University Press, Cambridge, 1995.

[16] J. Krajíček. Dual weak pigeonhole principle, pseudo-surjective functions, and provability of curcuit lower bounds. *J. Symbolic Logic*, 69(1):265–286, 2004.

[17] J. Krajíček and P. Pudlák. Quantified propositional calculi and fragments of bounded arithmetic. *Z. Math. Logik Grundlag. Math.*, 36(1):29–46, 1990.

[18] J. Krajíček and P. Pudlák. Some consequences of cryptographical conjectures for S^1_2 and EF. *Inform. and Comput.*, 140(1):82–94, 1998.

[19] J. Krajíček, P. Pudlák, and G. Takeuti. Bounded arithmetic and the polynomial hierarchy. *Ann. Pure Appl. Logic*, 52(1-2):143–153, 1991. International Symposium on Mathematical Logic and its Applications (Nagoya, 1988).

[20] J. Paris and A. Wilkie. Counting problems in bounded arithmetic. In *Methods in Mathematical Logic (Caracas, 1983)*, volume 1130 of *Lecture Notes in Math.*, pages 317–340. Springer-Verlag, Berlin, 1985.

[21] J. B. Paris, A. J. Wilkie, and A. R. Woods. Provability of the pigeonhole principle and the existence of infinitely many primes. *J. Symbolic Logic*, 53(4):1235–1244, 1988.

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

[23] C. Pollett. Structure and definability in general bounded arithmetic theories. *Ann. Pure Appl. Logic*, 100(1-3):189–245, 1999.

[24] A. A. Razborov. Bounded arithmetic and lower bounds in Boolean complexity. In *Feasible Mathematics II (Ithaca, NY, 1992)*, volume 13 of *Progr. Comput. Sci. Appl. Logic*, pages 344–386. Birkhäuser Boston, Boston, MA, 1995.

[25] N. Thapen. *The Weak Pigeonhole Principle in Models of Bounded Arithmetic*. PhD thesis, Oxford University, 2002.