Circuit principles and weak pigeonhole variants [1]

Chris Pollett [a,*] Norman Danner [b]

[a] *Department of Computer Science*
San Jose State University
San Jose CA 95192

[b] *Department of Mathematics and Computer Science*
Wesleyan University
Middletown, CT 06549

Abstract

This paper considers the relational versions of the surjective, partial surjective, and multifunction weak pigeonhole principles for PV, Σ^{b}_1, Π^{b}_1, and $B(\Sigma^{\mathsf{b}}_1)$-formulas as well as relativizations of these formulas to higher levels of the bounded arithmetic hierarchy. We show that the partial surjective weak pigeonhole principle for Π^{b}_1 formulas implies that for each k there is a string of length 2^{2n^k} which is hard to block recognize by circuits of size n^k. These principles in turn imply the partial surjective principle for Σ^{b}_1 formulas. We show that the surjective weak pigeonhole principle for $B(\Sigma^{\mathsf{b}}_1)$ formulas in S^1_2 implies our hard-string 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 S^1_2, the multifunction weak pigeonhole principle for such predicates is equivalent to an "iterative" circuit block-recognition principle. A consequence of this is that if S^1_2 proves this principles then RSA is vulnerable to polynomial time attacks.

Key words: Bounded arithmetic, circuit lower bounds, pigeonhole principle, RSA, cryptography

[*] Corresponding author
Email addresses: `pollett@cs.sjsu.edu` (Chris Pollett), `ndanner@wesleyan.edu` (Norman Danner).

[1] An earlier version of this paper has appeared as [25]. We would like to thank the *Theoretical Computer Science* referee for suggesting several corrections and sharpenings of our original results.

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 [1, 2]. 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 polynomial-time (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 by Paris et al. [23] is that $I\Delta_0 + WPHP(\Delta_0)$ proves there are infinitely many primes. The pigeonhole principles in both contexts are intimately related via well known translations of first order bounded arithmetics into sequences of propositional proofs [22, 18].

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 n^2, 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 different formulations 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 every n that is the length of some number there is a string S of length n that is not computed by any circuit with code of length $n-1$. To say that a circuit C computes a string S of length m means that C takes as input a number $i < m$ in binary and outputs the i-th bit of S. Here S^1_2 is a theory which roughly has length induction for NP predicates. It is thus natural to ask whether the other forms of the pigeonhole principle can be connected to circuit principles. Jeřábek's result is for the pigeonhole principle expressed using p-time functions so it is also reasonable 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.

Razborov [26, App. C] has argued 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": for each k there is a set A_k that does not have $O(n^k)$-size circuits (to say that a set A has $O(f(n))$-size circuits means that there are circuits $C_1, C_2, \ldots$ and a constant c such that C_n has size $cf(n)$ and accepts exactly the length n strings in A). It is also still open whether there is a set A defined by a bounded arithmetic formula such that for each $k > 0$ S^1_2 can prove the statement "A does not have $O(n^k)$-size circuits." 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 if we can obtain a smallest string that is not computed by any very small circuit, we can construct a fixed set which does not have $O(n^k)$-size 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 i-th level of polynomial hierarchy. This is because S^3_2 can do the necessary minimization and Paris et al. [23] have shown that S^3_2 proves the weak pigeonhole principle for p-time functions (see Krajíček [17] and Maciel et al. [21] for an exposition and tightenings of the original result). 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. Let $s\,WPHP(\Psi)$, $ps\,WPHP(\Psi)$, and $m\,WPHP(\Psi)$ denote respectively the surjective, partial surjective, and multifunction weak pigeonhole principle for the relations in Ψ. Our first result is that for each k S^1_2 proves $ps\,WPHP(\Pi^{\mathsf{b}}_1)$ implies that for all lengths n there is a string S of length $2n^k$ that is not block-recognized by any circuit with code of length n^k, and that S^1_2 proves that this principle implies $ps\,WPHP(\Sigma^{\mathsf{b}}_1)$. To say that a circuit C M-block-recognizes a string S of length N means that C has $|\lceil N/M \rceil| + M$ input bits and for $b < \lceil N/M \rceil$ and $s < 2^M$, $C(b,s)$ outputs 1 if and only if s is the b-th length-M block of S. We then analyze this proof to give some information about relational versions of the surjective weak pigeonhole principle. In particular, we show that to replace $ps\,WPHP$ with $s\,WPHP$ we must replace Π^{b}_1 with Boolean combinations of Σ^{b}_1 formulas.

It is natural to ask if one can obtain a result that is closer to involving just p-time functions, as Jeřábek's result does. To this end, we define a class of relations $\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\})$ 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 $R(x, y_1)$, $R(y_1, y_2)$, …, $R(y_{t-1}, y_t)$ 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 by Krajíček in the context of the propositional proof complexity of the surjective pigeonhole principle [15]. $\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\})$ contains PV and is contained in the class Σ^{b}_2. We show that over S^1_2, $m\,WPHP(\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\}))$ 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 S^1_2 also implies $m\,WPHP(PV)$.

Our results 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. Krajíček and Pudlák [19] (see also Thapen [28, 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 $i\,WPHP(PV)$ which allows p-time relations) from a class $\mathcal{C}$ satisfying $\mathsf{P}^{\mathcal{C}} = \mathcal{C}$, then RSA is vulnerable to attacks from $\mathcal{C}$. We apply Krajíček and Pudlák's result to conclude that if S^1_2 proves either of our hard-string principles then RSA is vulnerable to polynomial time attacks. One can somewhat strengthen the theory and still obtain results which we believe are open. For example, if S^2_2 proves our circuit principle, then RSA is vulnerable to attacks computed in the polynomial closure of polynomial local search. These results rely on the fact that $m\,WPHP(PV)$ implies $i\,WPHP(PV)$. It is unknown over S^1_2 whether $s\,WPHP(PV)$ implies $i\,WPHP(PV)$, 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 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 [9], 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 $m\,WPHP(PV)$, and so also $i\,WPHP(PV)$, 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 polynomial local search based attack or showing lower bounds for our iteration principle in S^2_2 might not be much beyond current technology.

As was mentioned earlier, it is known that S^3_2 proves $m\,WPHP(PV)$. One could ask the converse question: if one adds a weak pigeonhole principle to the base theory, how much induction can one prove? Although we do not exactly answer this question, we obtain a related result. We consider the injective pigeonhole principle from $|x|+1$ into $|x|$ which we denote by $i\,WPHP^*$. We view this pigeonhole principle as weaker than the usual one since it applies to lengths. Further, if one has a map from $2x$ into x, one can easily construct a map from $|x|+1$ into $|x|$. We show in this paper that $S^1_2 + i\,WPHP^*(\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_{i+1}))$ is equivalent to S^i_2. From this it can be shown that over S^1_2 if $i\,WPHP^*(\Sigma^{\mathsf{b}}_i)$ is equivalent to $i\,WPHP^*(\Sigma^{\mathsf{b}}_{i+1})$, then the polynomial hierarchy collapses to the $(i+2)$-nd level. In addition to establishing this result, we also extend the hard string principles described earlier up the bounded arithmetic hierarchy.

The format of the rest of this paper is as follows. In the next section we summarize the notations and theories to be discussed in the remainder of the paper. In the third section, we review results concerning the weak pigeonhole principle and prove the relation between $i\,WPHP^*$ and length-induction. In the next two sections we state our circuit principles precisely and prove them equivalent to the surjective and multifunction pigeonhole principles. We conclude with the RSA-related results.

2 Preliminaries

This paper assumes familiarity with the texts of either Buss [3], Krajíček [17], or Hájek and Pudlák [10]. For completeness, we review the basic notations of bounded arithmetic. The specific bootstrapping we are following is that of Pollett [24], but yields equivalent 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 $2^{|x||y|}$. The original formulations of bounded arithmetic do not usually include $\mathrm{MSP}(x,i)$ and $\dot{-}$, but instead define them with formulas. One slight advantage to our approach is that one can define terms in the language to do a limited amount of sequence coding, which allows us to more directly formulate our principles in the language L_2.

The bounded formulas of L_2 are classified into hierarchies Σ^{b}_i and Π^{b}_i by counting alternations of quantifiers, ignoring sharply-bounded quantifiers, analogous to the hierarchies Σ^0_i and Π^0_i of the arithmetic hierarchy. Here sharply bounded means bounded by a term of the form $|t|$. Formally, a Σ^{b}_0 (Π^{b}_0) formula is one in which all quantifiers are sharply-bounded. The $\Sigma^{\mathsf{b}}_{i+1}$ (Π^{b}_{i+1}) formulas contain the $\Sigma^{\mathsf{b}}_i \cup \Pi^{\mathsf{b}}_i$ formulas and are closed under $\neg A$, $A \supset B$, $B \wedge C$, $B \vee C$, sharply-bounded quantification, and bounded existential (universal) quantification, where A is Π^{b}_{i+1} $(\Sigma^{\mathsf{b}}_{i+1})$ and B and C are $\Sigma^{\mathsf{b}}_{i+1}$ (Π^{b}_{i+1}). The $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_i)$ formulas consist of the closure of the Σ^{b}_i formulas under Boolean connectives and sharply-bounded quantification (Hájek and Pudlák [10, Def. V.4.2]). Buss and Hay [6] show that $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_1)$ corresponds to the complexity class $\mathsf{P}^{\mathsf{NP}}(\log)$. For any class of formulas $\mathcal{C}$, define $B(\mathcal{C})$ to be the closure of $\mathcal{C}$ under Boolean connectives.

The theory *BASIC* is axiomatized by 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* (*length* induction), and -*LLIND* (*length-length* induction) are obtained by taking $A \in \mathcal{C}$ and $t(x)$ to be x, $|x|$, and $\|x\|$, respectively.

The term $\mathrm{Bit}(i, w) := \mathrm{MSP}(w, i) \dot{-} 2 \cdot \lfloor \mathrm{MSP}(w, i)/2 \rfloor$ is the i-th bit of w. The axiom scheme of Comprehension for $A \in \mathcal{C}$ ($\mathcal{C}$-*COMP*) is

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

Sequences can be defined as ordered pairs in which the first component specifies a block size and the second a concatenation of blocks. The predicate $\mathrm{Seq}(s)$ that is true when s is the code of a sequence can be given a Σ^{b}_0-definition. The function $\mathrm{SqBd}(a, b) := 2(2a\#2b)$ is a bound on the value of any sequence of length $|b| + 1$, each of whose components is $< a$, and $\beta(b, w)$ is defined to be the b-th element of the sequence w. $\beta(b, w)$ can be defined as a term in our language, and the basic properties of SqBd and $\beta(b, w)$ can be proved using open length induction. We will sometimes use the notation $(w)_b$ for $\beta(b, w)$. With these terms in hand, we can state the axiom scheme of Replacement for $A \in \mathcal{C}$ ($\mathcal{C}$-*REPL*):

$$\begin{aligned}&\forall x \leq |a|\, \exists y \leq b A(x, y) \supset \\ &\qquad\qquad \exists w \leq \mathrm{SqBd}(b + 1, a) \forall i \leq |a| \left(\beta(i, w) \leq b \wedge A(x, \beta(i, w))\right).\end{aligned}$$

The theories R^i_2, S^i_2 and T^i_2 are obtained from *BASIC* by adding respectively the Σ^{b}_i-*LLIND*, Σ^{b}_i-*LIND*, or Σ^{b}_i-*IND* axiom schema. It is known that $S^{i+1}_2 \supseteq T^i_2 \supseteq S^i_2 \supseteq R^i_2 \supseteq S^{i-1}_2$; R^i_2 (hence S^i_2) proves Σ^{b}_i-*COMP* and that $S^1_2 + \Sigma^{\mathsf{b}}_{i+1}$-*COMP* is equivalent to S^i_2 [4]; S^i_2 proves $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_i)$-*LIND* [4]; and if $R^{i+1}_2 \supseteq T^i_2$ then the polynomial hierarchy collapses [20, 24].

Buss [3, §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 let PV denote the set of equations over terms in the expanded language. We will without further comment identify $\Sigma^{\mathsf{b}}_0(PV)$ with PV. Among such functions, we will use the following "bit-manipulation" functions frequently:

(1) $\mathrm{LSP}(w, i) = x - 2^i\, \mathrm{MSP}(x, i)$ is the i least significant bits of w;

(2) $w[a..b] = \mathrm{LSP}(\mathrm{MSP}(w, a), b)$ consists of bits a through b inclusive of w;
(3) $\hat{\beta}(b, n, w) = w[bn..(b+1)n - 1]$ is the b-th length n block of bits of w.
(4) $vw = 2^{|w|}v + w$ is the concatenation of the bits of v and w.

We now summarize the notations and types of formulas that will occur frequently in this paper.

Definition 1 *$n \in Log$ abbreviates $\exists z(n = |z|)$. "Log-bounded" quantifiers are defined as expected; e.g., $\forall n \in Log \cdots$ abbreviates $\forall n(n \in Log \supset \cdots)$.*

Definition 2

(1) By $\exists^{\leq 1} x \leq tA(x)$ we mean the formula

$$\forall x \leq t \forall x' \leq t\Big((A(x) \wedge A(x')) \supset x = x'\Big).$$

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

$$\exists x \leq tA(x) \wedge \exists^{\leq 1} x \leq tA(x).$$

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 $i \geq 2^n$). By abuse of notation, we will frequently write $C(i)$ for the predicate $Output(C, i) = 1$. These are straightforward to formulate in S^1_2 using the sequence coding available there and have appeared before in the literature, such as in Buss [5].

3 Pigeonhole principles

We begin by defining variants of the pigeonhole principle with the domain and range parametrized:

$iPHP(R)^m_n$:

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

$mPHP(R)^m_n$:

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

$sPHP(R)^m_n$:

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

$psPHP(R)^m_n$:

$$\forall \vec{z}\Big[n < m \wedge \forall x < n \exists^{\leq 1} y < m R(x, y, \vec{z}) \supset \exists y < m \forall x < n \neg R(x, y, \vec{z})\Big]$$

where R is some predicate. The first three principles are frequently referred to as the functional, basic, and onto (or dual) principles, respectively. However, we will refer to these principles as the injective, multifunction, surjective, and partial surjective principles, as we feel that these names more directly convey the intended meanings. The scheme $psPHP$ is essentially Thapen's alternative definition of the surjective principle that states that there is no surjection from a subset of n onto m [28, Definition 3.1(4)] and is equivalent to $mPHP$ as we will note below. When we wish to refer to one of the schemes without concern for which one, we shall refer to $vPHP(R)^m_n$, where $v = i$, m, s, or ps. For a set of predicates $\mathcal{C}$ the notation $vPHP(\mathcal{C})^m_n$ will be used for the class of formulas $vPHP(R)^m_n$ where $R \in \mathcal{C}$. The notation $vWPHP(R)$ will be used for $\forall n\, vPHP(R)^{n^2}_n$ and similarly for $vWPHP(\mathcal{C})$. When $\mathcal{C} = FP$, we understand the relations to range over the equations $f(x, \vec{z}) = y$ for f is a PV-function symbol. In this situation, the injective and multifunction principles are equivalent, as are the surjective and partial surjective principles. We now make a few observations about the relations between the various principles.

Proposition 3 *The following inclusions and equivalence of theories hold over BASIC for any class of formulas $\mathcal{C}$:*

(1) $sPHP(B(\mathcal{C}))^m_n \supseteq psPHP(\mathcal{C})^m_n \supseteq sPHP(\mathcal{C})^m_n$.
(2) $iPHP(B(\mathcal{C}))^m_n \supseteq mPHP(\mathcal{C})^m_n \supseteq iPHP(\mathcal{C})^m_n$.
(3) $psPHP(\mathcal{C})^m_n \equiv mPHP(\mathcal{C})^m_n$.

In particular, if $\mathcal{C}$ is closed under Boolean connectives then the schemes $vPHP(\mathcal{C})^m_n$ are all equivalent for $v = i$, m, s, or ps.

PROOF. Most of the inclusions are immediate; for example, if $R(x, y, \vec{b})$ is a graph of a surjection from n onto $m > n$, then it is the graph of a partial surjection, and if $R(x, y, \vec{b})$ is the graph of an injective multifunction $x \mapsto y$

from m into $n < m$, then $R(y, x, \vec{b})$ is the graph of a partial surjective function $y \mapsto x$ from n onto m. For the first inclusion of (1), suppose that $R(x, y, \vec{b}) \in \mathcal{C}$ is a graph of a partial surjection from n onto $m > n$. Define $R^*(x, y, \vec{z})$ to hold if $R(x, y, \vec{z}) \vee (\neg R(x, y, \vec{z}) \wedge y = 0)$. Then R^* is a Boolean combination of $\mathcal{C}$-predicates and $R^*(x, y, \vec{b})$ is the graph of a surjection from n onto m. For the first inclusion of (2), if $R(x, y, \vec{z}) \in \mathcal{C}$ is a graph of an injective multifunction, then $R(x, y, \vec{b}) \wedge \forall y' < y \neg R(x, y', \vec{b})$ is a graph of an injective function that is in $B(\mathcal{C})$.

Proposition 4 $\mathit{BASIC} + s\mathit{WPHP}(\Pi^{\mathsf{b}}_i) \supseteq \mathit{BASIC} + s\mathit{WPHP}(\Sigma^{\mathsf{b}}_i)$.

PROOF. Suppose $R(x, y)$ is a Σ^{b}_i graph of a surjection f from 2^n onto 2^{2n} for some $n \in \mathit{Log}$ with $R_0 \in PV$. Let $R'(x, y)$ be the Π^{b}_i predicate $\forall y' < 2^{2n}(R(x, y') \supset y' = y)$. Suppose $x < 2^n$, $y < 2^{2n}$, and $R(x, y)$; we will show $R'(x, y)$ as well. Take any $y' < 2^{2n}$ such that $R(x, y')$. Then since R is the graph of a function when restricted to domain 2^n and range 2^{2n}, it must be that $y' = y$. Now suppose that in addition $R'(x, y_1)$ for some $y_1 < 2^{2n}$. Since $y < 2^{2n}$ and $R(x, y)$, we have that $y = y_1$. In other words, if $x < 2^n$ and $y < 2^{2n}$, then $R(x, y)$ holds iff $R'(x, y)$ does. So R' is a Π^{b}_i graph of a surjection from 2^n onto 2^{2n}.

Proposition 5 *S^1_2 proves*

$$\forall n \forall m \in \mathit{Log}\Big(\quad v\mathit{PHP}(\forall i < m R(\hat{\beta}(i, n, x), \hat{\beta}(i, 2n, y), \vec{z}))^{(mn)^2}_{mn} \supset v\mathit{PHP}(R)^{n^2}_n\Big).$$

PROOF. We'll prove the proposition just for the case $v = ps$. Let $n, m \in \mathit{Log}$ and suppose $R(x, y, \vec{b})$ is the graph of a partial surjection f from 2^n onto 2^{2n}, where $\vec{b}$ is a list of fixed parameters. Let $R'(x, y, \vec{b})$ be the predicate $\forall i < m\, R(\hat{\beta}(i, n, x), \hat{\beta}(i, 2n, y), \vec{b})$. We want to show that $R'(x, y, \vec{b})$ is the graph of a partial surjection from 2^{mn} onto 2^{2mn}. If $R'(x, y_1, \vec{b})$ and $R'(x, y_2, \vec{b})$, then by induction on $i < m$ show that $\hat{\beta}(i, 2n, y_1) = \hat{\beta}(i, 2n, y_2)$ using the fact that $R(x, y, \vec{b})$ is the graph of a partial function; conclude $y_1 = y_2$. To show surjectivity, given y we use PV-REPL and surjectivity of f to obtain a sequence w of length-n strings such that for all $i < m$ $R(\beta(i, w), \hat{\beta}(i, 2n, y))$; we then define x to be the concatenation of the strings in w by PV-COMP.

Proposition 6 *For each pigeonhole variant $v = m, s, i$, the theory $S^1_2(R)$ proves that $v\mathit{PHP}(\Sigma^{\mathsf{b}}_1(R))^{n^2}_n \supset v\mathit{PHP}(R)^{2n}_n$,*

PROOF. (Sketch) The basic idea of the proof for $S^1_2(R)$ is to show $\neg vPHP(R)^{2n}_n \supset \neg vPHP(\Sigma^{\mathsf{b}}_1(R))^{n^2}_n$. To do this in each case one iterates $|n|$ times the $2n$ into n function or multifunction (or n onto $2n$ function) violating $vPHP(R)^{n^2}_n$.

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 et al. [23] showed that $S_2 \vdash iWPHP(\Delta_0)$, where Δ_0 is the class of bounded formulas, and a variation on that proof shows that $T^{i+2}_2 \vdash iWPHP(\Sigma^{\mathsf{b}}_i)$ for $i \geq 1$. Maciel et al. [21] have sharpened this to show that $T^2_2(R) \vdash mWPHP(R)$ and hence $T^2_2 \vdash mWPHP(PV)$ and in particular $T^2_2 \vdash sWPHP(PV)$.

Turning to the relation between bounded arithmetic and the polynomial hierarchy, Krajíček et al. [20] have shown if $S^{i+1}_2 = S^{i+2}_2$, then the polynomial hierarchy collapses to the $(i+3)$-rd level. We next show that a similar result can be had for theories based on weak pigeonhole principles. For theories such as S^i_2 that are based on length-induction, it is reasonable to consider pigeonhole principles where the numbers involved are lengths. Let $iWPHP^*(R)$ be $\forall n\, iPHP(R)^{|n|+1}_{|n|}$, and define $mWPHP^*(R)$ similarly; let $iWPHP^*(\mathcal{C})$ and $mWPHP^*(\mathcal{C})$ be the obvious extensions to classes of formulas.

Proposition 7 *For $i \geq 1$, $S^1_2 + mWPHP^*(\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_i)) \subseteq S^i_2$.*

PROOF. Suppose A is $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_i)$ and $\neg mWPHP(A)^{|n|+1}_{|n|}$. By $\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_i)$-$COMP$ define $w < 2^{(|n|+1)|n|}$ so that $\forall x < |n|+1 \forall y < |n|\, (A(x,y) \Leftrightarrow \mathrm{Bit}(x\,|n|+y, w) = 1)$. Let $R(x,y,z,m)$ be the predicate that is true when $|z| = (m+1)m$ and $\mathrm{Bit}(xm+y, z) = 1$. Then $R(x,y,w,|n|)$ is a PV-relation with parameters that is the graph of an injective multifunction from $|n|+1$ into $|n|$. But $S^1_2 \vdash mWPHP(PV)^{|n|+1}_{|n|}$ (for example, Cook and Reckhow's proof [8] is easily formalized), so this is a contradiction.

Proposition 8 *For $i \geq 1$, $S^i_2 \subseteq S^1_2 + iWPHP^*(B(\Sigma^{\mathsf{b}}_i))$.*

PROOF. Suppose that A is Σ^{b}_i, $A(0)$ and $\forall x(A(x) \supset A(x+1))$ hold, but $\neg A(|b|)$ for some b. Define $R(x,y)$ to hold if

$$(\forall x' \leq |b|\, (x' \leq x \supset A(x')) \wedge y = x) \vee \\ (\exists x' \leq |b|\, (x' \leq x \wedge \neg A(x')) \wedge y = x - 1).$$

Then $R(x,y)$ is the graph of a function from $|b|+1$ into $|b|$. By $iWPHP^*(B(\Sigma^{\mathsf{b}}_i))$ there must be $x_1 < x_2$ and y such that $R(x_1,y)$ and $R(x_2,y)$. Chasing the definition of R, it must be that for all $x' \leq x_1$ $A(x')$ holds and there is $x'' \leq x_2$ such that $A(x'')$ fails. From the former we have that $y = x_1$ and from the latter $y = x_2 - 1$, so $x_2 = x_1 + 1$. Since $A(x')$ holds for all $x' \leq x_1$ and there is $x'' \leq x_2 = x_1 + 1$ such that $A(x'')$ fails, it must be that $A(x_2)$ fails. But then $A(x_1)$ holds but $A(x_1 + 1)$ does not, contradicting our assumption.

Combining Propositions 7 and 8 with Proposition 3 yields the following result:

Theorem 9 *For $i \geq 1$, $S^1_2 + v\,WPHP^*(\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_i)) \equiv S^i_2$ for $v = s$, ps, i, or m. In particular, if $S^i_2 \vdash v\,WPHP^*(\Sigma^{\mathsf{b}}_0(\Sigma^{\mathsf{b}}_{i+1}))$ for some v, then the polynomial hierarchy collapses to the $(i+2)$-nd level.*

4 The partial surjective pigeonhole principle and block-recognition

Jeřábek [12] shows that over S^1_2, the surjective weak pigeonhole principle is equivalent to the claim that there is a string of length n that is hard for circuits with codes of length $n - 1$. 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 10 *(Jeřábek [12, Lemma 3.2, Proposition 3.5]) Over S^1_2, the scheme $s\,WPHP(FP)$ is equivalent to*

$$\forall n \in Log \exists S < 2^n \forall C < 2^{n-1} \exists i < n \Big(\\ Circuit(C, |n|) \supset Output(C, i) \neq \mathrm{Bit}(i, S) \Big)$$

We begin by giving modified versions of Jeřábek's results for relational versions of the partial surjective weak pigeonhole principle. Before getting to the precise formulation of the result, let us consider what kind of circuit principle we should expect. In one direction, the weak pigeonhole principle fails, and we wish to take advantage of having the graph of a partial surjection from 2^n onto 2^{2n} in hand. Jeřábek has a function f represented by some circuit, which he iterates to amplify into a surjection from 2^n onto $2^{2^r n}$ for appropriate r which he then uses to show that every large string can be computed by some small circuit. In the relational case, since we have the graph of f, we are given x and y as input and can recognize when $f(x) = y$, but not necessarily compute f itself. Thus instead of expecting to compute the bits of a very large string S, we expect to be able to recognize length-n blocks of S. This will be our circuit principle: one that formulates that the circuit recognizes each length-n block of S (as opposed to each bit, which would essentially be computing S). For the other direction (that we can prove our circuit principle from a weak pigeonhole principle), we want to apply the pigeonhole principle to a statement that associates to every circuit C the unique string S that C

block-recognizes. Unfortunately, there is not necessarily such a unique string. But what is the case is that any circuit block-recognizes *at most* one string, and hence our "no partial surjection" formulation of the pigeonhole principle will be adequate.

Definition 11 *Let C be a circuit on $|\lceil m/n\rceil|+n$ input variables. We say that C* n-block-recognizes *$S < 2^m$ if 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 $S < 2^m$:

$$\mathit{Fits}(C, S, m, n) \wedge \forall i < \lceil m/n\rceil \left(\exists^{\leq 1} s < 2^n C(i, s) \wedge C(i, \hat{\beta}(i, n, S))\right).$$

Note that $\mathit{BlockRec}(C, S, m, n)$ is Π^{b}_1.

Proposition 12 *$S^1_2 + ps\,WPHP(\Pi^{\mathsf{b}}_1)$ proves the following principle for $k = 0, 1, \ldots$:*

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

PROOF. $\forall C < 2^{n^k} \exists^{\leq 1} S < 2^{2n^k} \mathit{BlockRec}(C, S, 2n^k, n)$ is provable in S^1_2 by length induction on the bits in each block of the string, then on the blocks. The proposition now follows from $ps\,WPHP(\Pi^{\mathsf{b}}_1)$.

The use of k here is not a triviality just because n^k is a length when n is. Specifically, one might be tempted to restate the result for only $k = 1$, in which case one obtains

$$S^1_2 + ps\,WPHP(\Pi^{\mathsf{b}}_1) \vdash \forall n \in \mathit{Log} \exists S < 2^{2n} \forall C < 2^n \neg \mathit{BlockRec}(C, S, 2n, n). \quad (*)$$

However, because n is used to specify the sizes of the blocks, this statement does not imply the one in Proposition 12. Consider trying to prove it implies the statement in the proposition. Assume that for some $n \in \mathit{Log}$ every $S < 2^{2n^k}$ is n-block-recognized by some circuit (code) $< 2^{n^k}$. Since n is a lengths, so is n^k. However, to conclude that $(*)$ fails for $n^k \in \mathit{Log}$ we would need a circuit code $C' < 2^{n^k}$ that n^k-block recognizes S; it is not obvious how to construct such a circuit from the one that we are given that n-block recognizes S.

As a corollary to the proof of Proposition 12, we have the following result:

Proposition 13 *$S^1_2 + psWPHP(FP)$ proves the following principle for $k = 0, 1, \ldots$:*

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

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 $psWPHP(PV)$ applies. But then this condition defines a PV-function, so only the functional version of $psWPHP$ is needed.

Lemma 14 *([17, Lemma 9.2.2]) Let $R(x_0, \ldots, x_{k-1})$ be a PV-relation. Then there is a polynomial p such that*

$$S^1_2 \vdash \forall \vec{m} \in Log \exists C < 2^{p(\vec{m})}[Circuit(C, m_0 + \cdots + m_{k-1}) \wedge \\ \forall x_0 < 2^{m_0} \ldots x_{k-1} < 2^{m_{k-1}}(C(\vec{x}) \Leftrightarrow R(\vec{x}))].$$

Theorem 15 *Let T be the theory obtained from S^1_2 by adding the axioms*

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

for $k = 0, 1, \ldots$. Then T proves $psWPHP(\Sigma^{\mathsf{b}}_1)$.

PROOF. It suffices to argue in S^1_2 that if there is a Σ^{b}_1-relation $R(x, y, \vec{x'})$ and parameters $\vec{b}$ such that $R(x, y, \vec{b})$ is the graph of a partial surjection $f(x)$ from 2^n onto 2^{2n} for some $n \in Log$, then $\forall S < 2^{2n^k} \exists C < 2^{n^k} BlockRec(C, S, 2n^k, n)$. By taking $m = \max\{1, \lceil \max_i\{|b_i|\} \rceil / n\}$ in Proposition 5 we can assume $|b_i| \leq n$ for each i. Note that even if we were to assume that f were total, we would not be able to assume there is a function symbol for f, since we do not have that S^1_2 *proves* that $R(x, y, \vec{b})$ is the graph of a function.

Say that R has the form $\exists z < 2^{p(|x|, |y|, |\vec{x'}|)} R_0(x, y, \vec{x'}, z)$ where R_0 is PV and set $p'(n) = p(n, 2n, n, \ldots, n)$. Using Lemma 14, let C_0 be a code of a circuit on variables $x_0, \ldots, x_{n-1}, y_0, \ldots, y_{2n-1}, z_0, \ldots, z_{p'(n)}$ that outputs 1 exactly when $R_0(x, y, z)$ holds (here, $\mathrm{Bit}(i, x) = x_i$, etc.). In more detail, C_0 is obtained from the circuit that computes R_0 with n bits for input x, $2n$ bits for input y, the bits for the parameters $\vec{x'}$ fixed to the bits of $\vec{b}$ (all of which have length $\leq n$), and the corresponding number of bits for z. Let $q(n)$ be a polynomial bound on the length of C_0. 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 $2n + p'(n)$. 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:

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

where $\mathrm{cond}(a,c,d)$ is either c or d as per whether $a=0$ or $a=1$. 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 $\mathit{Output}(\bar{G}(i),u,x,y,w)$. 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 $S < 2^{2^r n}$,

$$\begin{aligned}
&\exists e < \mathrm{SqBd}(n, 2^{2^{r-i}}) \exists w < \mathrm{SqBd}(i(2n+p'(n)), 2^{2^{r-i}}) \\
&\qquad \forall u < 2^i \forall v < 2^{r-i} G_i(u, (e)_v, \hat{\beta}(2^i v + u, n, S), (w)_v).
\end{aligned}$$

Since $r = ||z||$ and $i \leq r$, this predicate is Σ^{b}_1. This is a surjectivity claim about how we are iterating our partial function and is best explained by an example. Take $i=3$. The n-bit blocks of S are identified by numbers of the form $2^3 v + u$ for some $v < 2^{r-3}$ and $u < 2^3$. The claim says that there is a sequence e of 2^{r-3} n-bit blocks such that for each v and u, if we start with $(e)_v$, apply f, take the left- or right-hand side as per $\mathrm{Bit}(2,u)$, apply f again and take a side as per $\mathrm{Bit}(1,u)$, apply f again and take a side as per $\mathrm{Bit}(0,u)$, we obtain the $(2^3 v + u)$-th n-bit block of S. The sequence w captures all of the intermediate witnesses needed for the graph of f. In particular, taking $i=r$ we have that

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

(2)

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

In words, our iteration of f results in a partial function.

(3) The size of G_i is $O(iq(n))$.

The difficult claim is (1), which we prove here by length-induction on $i \leq r$. For $i=0$, take $(e)_v = \hat{\beta}(v,n,S)$. Suppose the claim is true for i. Let e'

and w' be the sequences given by the induction hypothesis. Since we have a (partial) surjection, for each $v < 2^{r-i}$ there are $(e)_v$ and $(w^*)_v$ such that $C_0((e)_v, (e')_{2v}(e')_{2v+1}, (w^*)_v)$. Set $(w)_v = (w')_v(w^*)_v(e')_{2v}(e')_{2v+1}$. e and w are definable by Σ^{b}_1-replacement. Fix $u < 2^{i+1}$ and $v < 2^{r-i-1}$. Set $u' = \mathrm{LSP}(u,i) < 2^i$ and $v' = 2v + \mathrm{Bit}(i,u) < 2^{r-i}$. By the induction hypothesis we have that $G_i(u', (e')_{v'}, \hat{\beta}(2^iv'+u', n, S), (w')_v)$. Since $2^iv'+u' = 2^iv+u$, we really have that $G_i(u', (e')_{v'}, \hat{\beta}(2^iv+u, n, S), (w')_v)$. Suppose that $\mathrm{Bit}(i,u) = 0$. To show the claim, we must show that $C_0((e)_v, (e')_{2v}(e')_{2v+1}, (w^*)_v)$, which we have by assumption, and $G_i(u', (w)_v[n..2n-1], \hat{\beta}(2^iv+u, n, S), \mathrm{MSP}((w)_v, 2n+p'(n)))$. Chasing the definition of w, this is the same as $G_i(u', (e')_{2v}, \hat{\beta}(2^iv+u, n, S), (w')_v)$. Since $v' = 2v + \mathrm{Bit}(i,u) = 2v$ in this case, this is the same as showing $G_i(u', (e')_{v'}, \hat{\beta}(2^iv+u, n, S), (w')_v)$, which is just the induction hypothesis. The case when $\mathrm{Bit}(i,u) = 1$ is similar.

The base case for (2) is trivial. For the induction step, if $G_{i+1}(u,e,y,w)$ and $G_{i+1}(u,e,y',w')$, then by definition $C_0(e, w[0..2n-1], w[2n..2n+p'(n)-1])$ and $C_0(e, w'[0..2n-1], w'[2n..2n+p'(n)-1])$. Thus there are z and z' such that $R_0(e, w[0..2n-1], z)$ and $R_0(e, w'[0..2n-1], z')$, which implies that $R(e, w[0..2n-1])$ and $R(e, w'[0..2n-1])$. But since R is the graph of a partial *function* it must be the case that $w[0..2n-1] = w'[0..2n-1]$. The induction hypothesis now applies to conclude that $y = y'$ and $\mathrm{MSP}(w, 2n+p'(n)) = \mathrm{MSP}(w', 2n+p'(n))$ and hence that $w = w'$.

Now fix a constant k and let $r = \left|2kn^{k-1}\right| = (k-1)\,|n|+1$, so that $2^rn \geq 2n^k$. Then as we just showed, for each $S < 2^{2^rn}$ there are (provably in S^1_2) e_S and w_S such that $G_r(\cdot, e_S, \cdot, w_S)$ n-block-recognizes S. For each S and r such that $S < 2^{2^rn}$ let $C^S_r(i,s) = G_r(i, e_S, s, w_S)$. For convenience, take ℓ such that $q(n) \leq n^\ell$ (we can assume $n > 1$). The size of C^S_r is then $\leq c((k-1)\,|n|+1)n^\ell \leq c'kn^{\ell+1}$ for some c and c'. Furthermore, any circuit of size m can be given a code of length $\leq 2m(|m|+1)$. Thus, if we take k large enough so that

$$n^k \geq 4(c')^2k^2n^{2\ell+2} \geq 2c'kn^{\ell+1}(\left|c'kn^{\ell+1}\right|+1),$$

then for any $S < 2^{2n^k}$ we have that $C^S_{(k-1)|n|+1} < 2^{n^k}$ is the code of a circuit that n-block recognizes S.

Let *HardString*(n,k) abbreviate

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

To summarize, Proposition 12 and Theorem 15 yield the following:

Theorem 16 *The following inclusions of theories holds:*

$$\begin{aligned}&S^1_2 + ps\,WPHP(\Pi^{\mathsf{b}}_1) \supseteq \\ &\qquad\qquad S^1_2 + \{\forall n \in Log\ HardString(n,k)\}_{k\geq 0} \supseteq \\ &\qquad\qquad\qquad\qquad S^1_2 + ps\,WPHP(\Sigma^{\mathsf{b}}_1).\end{aligned}$$

The surjective pigeonhole principle

Obtaining a similar result for the surjective pigeonhole principle is more problematic than the partial surjective one. On the one hand, the proof of Theorem 15 carries through if one assumes that one has a surjection from 2^n onto 2^{2n} that has a Σ^{b}_1 graph. However, if we wish to have a *total* function that maps circuits C to the string that C recognizes if there is one and, say, 0 otherwise, we end up applying $s\,WPHP$ to the (provable in S^1_2) claim

$$\begin{aligned}&\forall C < 2^{n^k} \exists! S < 2^{2n^k} \Big[BlockRec(C,S,2n^k,n) \vee \\ &\qquad \Big((\neg Fits(C,S,2n^k,n) \vee \exists i < 2n^{k-1} \neg\exists! s < 2^n C(i,n,s)) \wedge S = 0\Big)\Big].\end{aligned}$$

Since $BlockRec(C,S,2n^k,n)$ is Π^{b}_1 and $\exists i < 2n^{k-1} \neg\exists! s < 2^n C(i,n,S)$ can be rewritten as a disjunction of a Π^{b}_1 and Σ^{b}_1 formula, the predicate in brackets belongs to $B(\Sigma^{\mathsf{b}}_1)$. Summarizing this discussion, we have:

Theorem 17 *The following inclusions of theories holds:*

$$\begin{aligned}&S^1_2 + s\,WPHP(B(\Sigma^{\mathsf{b}}_1)) \supseteq \\ &\qquad\qquad S^1_2 + \{\forall n \in Log\ HardString(n,k)\}_{k\geq 0} \supseteq \\ &\qquad\qquad\qquad\qquad S^1_2 + s\,WPHP(\Sigma^{\mathsf{b}}_1).\end{aligned}$$

Relativization

These results can be relativized in the following way. Expand the language L_2 with a new second-order predicate symbol $\alpha(\vec{x})$. For each class of formulas $\mathcal{C}$ define $\mathcal{C}(\alpha)$ to be the analogous class but where we allow atomic formulas of the form $\alpha(\vec{t})$ to occur. By allowing the appropriate form of induction now for $\Sigma^{\mathsf{b}}_i(\alpha)$-formulas, one can define the theories $R^i_2(\alpha)$, $S^i_2(\alpha)$, and $T^i_2(\alpha)$ (see Krajíček [17] for more details). For a function class $\mathcal{FC}$ defined from an initial set of functions and closure under composition as well as some kind of recursion, we denote by $\mathcal{FC}(\alpha)$ the class obtained by adding $\alpha(x)$ as a 0-1 valued function to the initial set of functions. One can also define circuits with new gates of type $A_{j_1,\ldots,j_n}$, in addition to AND, OR, and NOT that were used before. A gate of type $A_{j_1,\ldots,j_n}$ takes $\sum_{i=1}^n j_i$ inputs. To evaluate this gate with respect to a given setting of these input values and with respect to the second order variable α, one feeds into the k-th input slot of α the value x_k output from the j_k inputs starting from input $\sum_{i=1}^{k-1} j_i$. In $S^1_2(\alpha)$ one can define and reason about the predicates $\mathit{Circuit}_A(C, |n|)$ and $\mathit{Output}_A(C, \alpha, i)$ which now allow circuits with the new gate types, $A_{j_1,\ldots,j_n}$. Given the above definitions we can state a relativized versions of Jeřábek's result as:

Theorem 18 *Over S^1_2, the scheme $s\,\mathit{WPHP}(\mathit{FP}(\alpha))$ is equivalent to*

$$\forall n \in \mathit{Log} \exists S < 2^n \forall C < 2^{n-1} \exists i < n \Big(\mathit{Circuit}_A(C, |n|) \supset \mathit{Output}_A(C, \alpha, i) \neq \mathrm{Bit}(i, S)\Big)$$

The proof is essentially the same as in the unrelativized case. By defining relativized versions of our other formulas such as *BlockRec* and *Compute*, we can obtain by essentially the same proofs the following variants of our earlier results:

Theorem 19 *The following inclusions of the theories holds:*

$$\begin{aligned} &S^1_2(\alpha) + ps\,\mathit{WPHP}(\Pi^{\mathsf{b}}_1(\alpha)) \supseteq \\ &\qquad S^1_2(\alpha) + \{\forall n \in \mathit{Log}\, \mathit{HardString}_A(n, k, \alpha)\}_{k \geq 0} \supseteq \\ &\qquad\qquad S^1_2(\alpha) + ps\,\mathit{WPHP}(\Sigma^{\mathsf{b}}_1(\alpha)) \end{aligned}$$

and

$$\begin{aligned} &S^1_2(\alpha) + s\,\mathit{WPHP}(B(\Sigma^{\mathsf{b}}_1(\alpha))) \supseteq \\ &\qquad S^1_2(\alpha) + \{\forall n \in \mathit{Log}\, \mathit{HardString}_A(n, k, \alpha)\}_{k \geq 0} \supseteq \\ &\qquad\qquad S^1_2(\alpha) + s\,\mathit{WPHP}(\Sigma^{\mathsf{b}}_1(\alpha)). \end{aligned}$$

Krajíček [16] shows that $S^2_2(\alpha)$ does not prove $i\,\mathit{WPHP}(\alpha)$ and Riis [27] gives a general condition on formulas with undefined predicates symbols which implies $S^2_2(\alpha)$ does not prove $s\,\mathit{WPHP}(\alpha)$. Either result yields the following corollary:

Corollary 20 *The theory $S^2_2(\alpha)$ does not prove $\forall n \in \mathit{Log}\, \mathit{HardString}_A(n, k, \alpha)$ for all $k = 0, 1, \ldots$:*

Another use for developing relativized variants of the results of this paper is to extend some of these results up into higher levels of the bounded arithmetic hierarchy. Hájek and Pudlák [10, Thm. 4.18] show that for $i \geq 1$, there is a "universal" Σ^{b}_i formula U_i with the property that for any Σ^{b}_i-formula $A(x)$ there is a numeral e_A such that $S^1_2 \vdash A(x) \equiv U_i(e_A, x, 2^{|x|^{e_A}})$. It follows that S^{i+1}_2 is equivalent to $S^1_2(U_i)$. Thus as corollaries of Theorems 18 and 19 we get:

Corollary 21 *Over S_2^{i+1}, the scheme $sWPHP(FP(\Sigma_i^{\mathsf{b}}))$ is equivalent to the scheme*

$$\forall n \in Log \exists S < 2^n \forall C < 2^{n-1} \exists i < n\Big(\\ Circuit_A(C, |n|) \supset Output_A(C, U_i, i) \neq \mathrm{Bit}(i, S)\Big)$$

Corollary 22

$$S_2^{i+1} + ps\,WPHP(\Pi_{i+1}^{\mathsf{b}}) \supseteq \\ S_2^{i+1} + \{\forall n \in Log\; HardString_A(n, k, U_i)\}_{k \geq 0} \supseteq \\ S_2^{i+1} + ps\,WPHP(\Sigma_{i+1}^{\mathsf{b}})$$

and

$$S_2^{i+1} + s\,WPHP(B(\Sigma_{i+1}^{\mathsf{b}})) \supseteq \\ S_2^{i+1} + \{\forall n \in Log\; HardString_A(n, k, U_i)\}_{k \geq 0} \supseteq \\ S_2^{i+1} + s\,WPHP(\Sigma_{i+1}^{\mathsf{b}}).$$

5 The multifunction pigeonhole principle and iteration

In this section, we explore connections between the multifunction weak pigeonhole principle and hardness of circuit iteration principles. To begin our discussion we consider a way to define a class of formulas from an existing class of formula via iteration.

Definition 23 *Given a class $\mathcal{C}$ of formulas and a set τ of terms, $\mathsf{ITER}(\mathcal{C}, \tau)$ consists of formulas of the form*

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

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

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

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 26), 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 studied by Krajíček [15] 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. The second universal clause in *Comp* above is used to check at each step of the computation there is a unique next value for R.

Write $\{\|\mathrm{id}\|^{O(1)}\}$ 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 24

(1) The theory S^1_2 proves that $\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\})$ 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 S^1_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) Let $R^*(i, u, v, \vec{z})$ be the predicate

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

Let $B^*(\vec{z})$ be the term $\langle B(0, \vec{z}), \ldots, B(|h|, \vec{z})\rangle$ and $E^*(\vec{z})$ be $\langle E(0, \vec{z}), \ldots, E(|h|, \vec{z})\rangle$ (since these are computable in polynomial time from $\vec{z}$, they are terms in our language). The reverse direction of the claim is straightforward: for each $j \leq |h(\vec{z})|$, use the j-th "section" of the sequence given by the right-hand side. For the forward direction, assume the left-hand side holds. Then in particular

$$\forall j \leq |h(\vec{z})|\, \exists w \leq \mathrm{SqBd}(s, 2^{t+1}) \Big[\mathrm{Seq}(w) \wedge \mathit{Len}(w) = t + 2 \wedge \\ \forall i \leq t\Big(\beta(i, w) \leq s \wedge R(i, \beta(i, w), \beta(i + 1, w), \vec{z})\Big) \wedge \\ \beta(0, w) = B(\vec{z}) \wedge \beta(t + 1, w) = E(\vec{z})\Big].$$

Since t is sharply bounded the predicate in brackets is PV and so by PV-$REPL$ there is a sequence W such that for every $j \leq |h(\vec{z})|$ the predicate in brackets holds with w replaced by $\beta(j, W)$. Let W^* be the sequence defined by $\beta(i, W^*) = \langle \beta(i, \beta(0, W)), \ldots, \beta(i, \beta(|h(\vec{z})|, W))\rangle$. That W^* is a sequence of computations from $B^*(\vec{z})$ to $E^*(\vec{z})$ along R^* follows from the definition of W. The uniqueness criterion is proved by showing that for any $j \leq |h(\vec{z})|$, $\beta(j, W)$ is identical to the w given by the right-hand side; this is proved by induction on t using the uniqueness criterion for w.

Lemma 25 *S^1_2 proves $Uniq(\|t\|^m)$ for fixed m where $Uniq(a)$ is the formula*

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

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

PROOF. Suppose w_1 and w_2 are such that $\mathit{Comp}(R, B, E_1, w_1, \vec{z}, s, a) \wedge \mathit{Comp}(R, B, E_2, w_2, \vec{z'}, s, a)$. Then by the definition of Comp one has for each $i \leq a$ that $\beta(i, w_1) = \beta(i, w_2)$. From this condition, using PV-$LIND$ it is straightforward to get $w_1 = w_2$.

Definition 26

(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) For k a natural number, let $IterBlockRec(C, S, c, n, k, t)$ be

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

By Lemma 24, this is an iteration predicate. Note that Out depends only on the parameters b and C.

(3) Let $CompOutput(w, C, S, c, b, n, t)$ be

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

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

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

Theorem 27 *For $\|t\|^j$ in $\{\|\mathrm{id}\|^{O(1)}\}$, the theory S^1_2+$m\,WPHP(\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\}))$ proves the following principle for $k = 2, 3, 4, \ldots$*

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

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 S^1_2 and suppose that

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

Using Lemma 24, the expression in square brackets is equivalent in S^1_2 to an $\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\})$ predicate. Fix n. So by $m\,WPHP(\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\}))$ there are $S_1 \neq S_2 < 2^{2n^k}$, $C < 2^{n^k - 2n}$, $c < 2^{2n}$ such that

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

for $i = 1, 2$. Fix any $b < n^{k-1}$. By Lemma 25, 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_i), 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 28 *Let T be the theory S^1_2 extended by the axioms*

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

for each $k > 1$, $\|t\|^j$ in $\{\|\mathrm{id}\|^{O(1)}\}$. Then T proves $m\,WPHP(PV)$.

PROOF. Assume that $R(x, y, \vec{x'})$ is a PV-formula such that for the values $\vec{b'}$, $R(x, y, \vec{b'})$ is the graph of an injective multifunction from 2^{2n} into 2^n. By Proposition 5 we can assume $|b_i| \leq n$ for each i. Let r be some term we will describe in a moment and define $Amp'(S, j, ||r||, n, w, \vec{b'})$ to be the conjunction of the following statements:

(1) $S < 2^{2^{\|r\|}n}$;
(2) w is a sequence of length $j + 1$;
(3) For $0 \leq i \leq j$, $\beta(i, w)$ is a sequence of length $2^{\|r\|-i}$;
(4) For $0 \leq i \leq j$ and $0 \leq \ell < 2^{\|r\|-i}$, $|\beta(\ell, \beta(i, w))| \leq 2n$;
(5) For $0 \leq \ell < 2^{\|r\|}$, $\beta(\ell, \beta(0, w)) = \hat{\beta}(\ell, 2n, S)$;
(6) For $0 \leq i \leq j$ and $0 \leq \ell < 2^{\|r\|-i-1}$,

$$R(\beta(2\ell, \beta(i, w)), \mathrm{MSP}(\beta(\ell, \beta(i + 1, w)), n), \vec{b'});$$

(7) For $0 \leq i \leq j$ and $0 \leq \ell < 2^{\|r\|-i-1}$,

$$R(\beta(2\ell + 1, \beta(i, w)), \mathrm{LSP}(\beta(\ell, \beta(i + 1, w)), n), \vec{b'}).$$

In other words, w is a "trapezoid" with $j+1$ rows. The first row is the length-$2n$ blocks of S and the $(i + 1)$-st row is obtained by using R to "compress" each element of the i-th row to a length-n block and then joining each pair of adjacent blocks.

Let $Amp(S, j, \|r\|, n, \vec{b'})$ be the predicate

$$\exists w \leq \mathrm{SqBd}(\mathrm{SqBd}(2^{2n}, 2^{2^{\|r\|-1}}), 2^{\|r\|}) Amp'(S, j, \|r\|, n, w, \vec{b'}).$$

So Amp is (equivalent to) a Σ^{b}_1 formula over $BASIC$. By Σ^{b}_1-$LLIND$ on j one can show that $\forall j \leq \|r\|\, Amp(S, j, \|r\|, n, \vec{b'})$; for the induction step one just adds the next row of the trapezoid, which is obtained by PV-$REPL$.

Now let r be a term such that $\|r\| = (k-1)\,|n|+1$ so that $2^{\|r\|}n \geq 2n^k$, fix $S < 2^{2n^k}$, and let w be the trapezoid (now a "triangle") witnessing $Amp(S, \|r\|, \|r\|, n, \vec{b'})$. Let $c = \beta(0, \beta(\|r\|, w))$. Let $C(i, u, v, b)$ where $\vec{b'}$ has been hard-coded be the circuit that computes the predicate

$$R\Big(v, \mathrm{cond}\Big(\mathrm{Bit}((k - 1)\,|n| - i, b), \mathrm{MSP}(u, n), \mathrm{LSP}(u, n)\Big), \vec{b'}\Big).$$

Take any $b < n^{k-1}$ (the number of length-$2n$ blocks in S) and define a new sequence v by $\beta(i, v)) = \beta(\mathrm{MSP}(b, i), \beta(\|r\| - i, w))$. 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 length $\|r\|$ 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 $Amp(S, c, \|r\|, \|r\|, n, \vec{b'})$. Uniqueness of each step follows from the fact that R is injective. As in the proof of Theorem 15, take k large enough so that we can assume $C < 2^{n^k-2n}$; then by chasing definitions, we see that we have proved

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

completing the proof.

Theorem 29 *Let T be the theory S^1_2 extended by the axioms*

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

for each $k > 1$, $\|t\|^j$ in $\{\|\mathrm{id}\|^{O(1)}\}$. Then T proves $m\mathit{WPHP}(\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\}))$.

PROOF. We describe how to modify the proof of Theorem 28 to obtain this result. Let $Q := \mathit{Iter}(R, B, E, x, y, \vec{z}, s, \|t\|^m)$ be a predicate such that $\neg m\mathit{WPHP}(Q)$. 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 and that this is an injection for some setting $\vec{b'}$ of the remaining parameters. By Propositions 5 and 24(2) we may assume $|b'_i| \leq n$ for each i. We use the relation R to create a modified version of Amp, where we insert the iterations needed to compute Q between each row of the trapezoid. Set $\mathit{clen} = \|t\|^m + 3$ (recall that the length of the iteration sequence for Q is $\|t\|^m + 2$) and let $\mathit{Amp}'(S, j, \|r\|, n, w, \vec{b'})$ be the conjunction of the following statements:

(1) $S < 2^{2^{\|r\| n}}$;
(2) w is a sequence of length $j \cdot \mathit{clen} + 1$;
(3) For $0 \leq i \leq j$, $\beta(i \cdot \mathit{clen}, w)$ is a sequence of length $2^{\|r\|-i}$ and for $0 \leq \ell < 2^{\|r\|-i}$, $|\beta(\ell, \beta(i \cdot \mathit{clen}, w))| \leq 2n$.
(4) For $0 \leq \ell < 2^{\|r\|}$, $\beta(\ell, \beta(0, w)) = \hat{\beta}(\ell, 2n, S)$.
(5) For $0 \leq i \leq j$, $i' = i \cdot \mathit{clen}$, and $0 \leq a < \|t\|^m + 2$, $\beta(i' + a + 1, w)$ is a sequence w' of length $2^{\|r\|-i}$ and for $0 \leq \ell < 2^{\|r\|-i}$, $\beta(\ell, w') \leq s$;
(6) For $0 \leq i \leq j$, i', a, and ℓ as in the previous point, let $(w)_{a,\ell} = \beta(\ell, \beta(i' + a + 1, w))$. Then:
   (a) $R(a, \mathrm{MSP}((w)_{a,\ell}, n), \mathrm{MSP}((w)_{a+1,\ell}, n))$;
   (b) $\mathrm{LSP}((w)_{a,\ell}, n) = \mathrm{LSP}((w)_{a+1,\ell}, n)$.
(7) For $0 \leq i \leq j$, i', a, ℓ, and $(w)_{a,\ell}$ as in the previous point:
   (a) $(w)_{0,2\ell} = B(\beta(2\ell, \beta(i', w)), L, \vec{b'}) * L$, where $L = \mathrm{MSP}(\beta(\ell, \beta((i+1) \cdot \mathit{clen}, w)), n)$; and
   (b) $(w)_{\|t\|^m-1,2\ell} = E(\beta(2\ell, \beta(i', w)), L, \vec{b'}) * L$.
(8) For $0 \leq i \leq j$, i', a, ℓ, and $(w)_{a,\ell}$ as in the previous point:
   (a) $(w)_{0,2\ell+1} = B(\beta(2\ell+1, \beta(i', w)), R, \vec{b'}) * R$, where $R = \mathrm{LSP}(\beta(\ell, \beta((i- 1) \cdot \mathit{clen}, w)), n)$; and
   (b) $(w)_{\|t\|^m-1,2\ell+1} = E(\beta(2\ell+1, \beta(i', w)), R, \vec{b'}) * R$.

So this formula asserts that w is a "trapezoid of grids" with j grids. The i-th grid has $2^{\|r\|-i}$ columns and $\|t\|^m + 3$ rows. The first row corresponds to a row of the trapezoid from the proof of Theorem 28. The next row consists of blocks of the form $B(x, y, \vec{b'}) * y$ where $x < 2^{2n}$ is the value in the same column and previous row and $y < 2^n$ is the value x is mapped to by the multifunction with graph Q (we need the "extra" copy of y so that the circuit that we eventually construct can verify that a sequence represents a path through this trapezoid of grids while only examining adjacent elements of the sequence). 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 S^1_2, this Amp is still equivalent to a Σ^{b}_1-formula. So one can prove

$$\forall j \leq \|r\| \, Amp(S, j, \|r\|, n, \vec{b'})$$

by induction on j in S^1_2. The induction step is handled by using the fact that since $\neg m\,WPHP(Q)$, there is some unique sequence that makes Q an injective map from 2^{2n} into 2^n. So if w is the trapezoid so far and c is its last row, one can apply Q to the length-$2n$ blocks of c to obtain length-n blocks to get a $c' < 2^{2^{\|r\|-(j+1)}n}$. Adding to w the relevant rows from the sequence used to witness the existential of Q as well as this c' one can make a new w' that satisfies

$$Amp'(S, j+1, \|r\|, n, w', \vec{b'})$$

to complete the induction step.

Now given $S < 2^{2n^k}$ and r such that $\|r\| = (k-1)|n| + 1$ we need a circuit $C(i, u, v, b)$ where $\vec{b'}$ has been hard-coded that recognizes a path through this "triangle of grids" that starts at the last row c and ends at block of the first row, $\hat{\beta}(b, 2n, S)$. From now on, we index rows starting at c. So row index i means the $(\|r\| - i)$-th row of the sequence. When $i = i' \cdot clen$, we are looking at a sequence of length-$2n$ blocks at the end of a grid; we verify that $\mathrm{LSP}(v, n)$ is the right- or left- half of u as per the $((k-1)|n| - i')$-th bit of b. This is why we need the L's and R's; without carrying them through the grid, we would not be able to perform this verification "locally." When $i = i' \cdot clen + a + 1$ for $0 \leq a < \|t\|^m + 2$ we transition according to R, so the circuit verifies that $R(\|t\|^m + 1 - a, \mathrm{MSP}(v, n), \mathrm{MSP}(u, n))$. When $i = (i' + 1) \cdot clen - 1$ we are transitioning from one grid to the next, so the circuit verifies that $u = B(v, \mathrm{LSP}(u, n), \vec{b'}) * \mathrm{LSP}(u, n)$.

The usual argument allows us to choose k large enough so that $C < 2^{n^k}$ and $IterBlockRec(C, S, \beta(0, \beta(\|r\| \cdot clen, w)), n, k, \|t\|^m)$ where w is the witness to $Amp(S, \|r\|, \|r\|, n, w, \vec{b'})$.

We do not know if $mWPHP(PV)$ implies $mWPHP(\mathsf{ITER}(PV, \{\|\mathrm{id}\|^{O(1)}\}))$ 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.

Relativization

Referring to the notation for relativizing these results at the end of the previous section, we have the analogous result for the multifunction principle:

Theorem 30 *Let T be the theory $S^1_2(\alpha)$ extended by the axioms*

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

for each $k > 1$, $\|t\|^j$ in $\{\|\mathrm{id}\|^{O(1)}\}$. Then T is equivalent to $S^1_2(\alpha)$ together with $mWPHP(\mathsf{ITER}(PV(\alpha), \{\|\mathrm{id}\|^{O(1)}\}))$.

The following corollary is again a direct consequence of the results of Krajíček [16] and Riis [27].

Corollary 31 *The theory $S^2_2(\alpha)$ does not prove the statement $\forall n \in Log \exists S < 2^{2n^k} \forall C < 2^{n^k-2n} \forall c < 2^{2n} \neg IterBlockRec(C, S, \alpha, c, n, k, \|t\|^j)$.*

Again using Hájek and Pudlák's universal formula U_i, we have

Corollary 32 *Let T be the theory S^{i+1}_2 extended by the axioms*

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

for each $k > 1$, $\|t\|^j$ in $\{\|\mathrm{id}\|^{O(1)}\}$. Then T is equivalent to S^{i+1}_2 together with $mWPHP(\mathsf{ITER}(PV(\Sigma^{\mathsf{b}}_i), \{\|\mathrm{id}\|^{O(1)}\}))$.

6 Iteration and RSA

In this section, the provability of our circuit and iteration principles in S^1_2 and S^2_2 is connected to the security of RSA. To state our results, we define the class PLS and recall the definition of RSA.

Definition 33 *A PLS problem consists of a polynomial time cost function c, a polynomial time neighborhood function N, and a polynomially bounded* *set of 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* PLS *problem is given by the relation $f(x) = y$ iff $F(x,y)$ and $c(x, N(x,y)) < c(x,y)$.*

The class PLS for polynomial search was defined by Johnson et al. [13] and was shown to contain several interesting optimization problems. Buss and Krajíček [7] showed that the Σ_1^{b} provably total multifunctions of T_2^1 can be characterized as the composition of a projection function with a PLS multifunction.

Recall what an instance of RSA is:

Definition 34 *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 35 *Let B_k denote*

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

and let $IB_{k,j}$ denote

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

(1) If for each $k > 1$, $j \geq 1$, S_2^1 proves $IB_{k,j}$ (similarly, B_k) then RSA is vulnerable to polynomial time based attacks.

(2) If for any $k > 1$, $j \geq 1$, S_2^2 proves either $IB_{k,j}$ (similarly B_k) then RSA is vulnerable to polynomial time in PLS *based attacks.*

PROOF. Both (1) and (2) are proved in essentially the same way. By [4], S_2^2 is Σ_2^{b}-conservative over T_2^1. Let T be either S_2^2 or S_2^1. Then if T proves $IB_{k,j}$, then by Theorem 28, T proves $m\mathit{WPHP}(PV)$ so by Proposition 3 it also proves $i\mathit{WPHP}(PV)$ and thus $i\mathit{WPHP}(FP)$. Similarly, since the partial surjective principle is equivalent to the multifunction principle for Σ_1^{b}-formulas and the Σ_1^{b}-formulas contain the graphs of FP functions, a similar chain of implications shows that the principles B_k imply $i\mathit{WPHP}(FP)$. So if T proves B_k we also get T proves $i\mathit{WPHP}(FP)$. The schema $i\mathit{WPHP}(FP)$ 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_1^{b}-formulas. As we have just remarked if $T = S_2^1$ that T proves $i\,WPHP(FP)$. If, though, $T = S_2^2$, then this in turn is Σ_2^{b}-conservative T_2^1, so we will have T_2^1 proves $i\,WPHP(FP)$. Using the witnessing arguments used to show the characterizations of Σ_1^{b}-definability in S_2^1 and T_2^1 one can say the following: (1) for S_2^1, there is a 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 $f(x_1,c) = f(x_2,c)$; (2) for T_2^1, and hence S_2^2, g can be computed as a a projection of a PLS problem. By Krajíček and Pudlák [19] there is polynomial time algorithm using g as an oracle which solves RSA.

References

[1] M. Ajtai. The complexity of the pigeonhole principle. *Combinatorica*, 14 (4):417–433, 1994. ISSN 0209-9683.

[2] 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.

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

[4] 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.

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

[6] S. R. Buss and L. Hay. On truth-table reducibility to SAT. *Inform. and Comput.*, 91(1):86–102, 1991. ISSN 0890-5401.

[7] 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. ISSN 0024-6115.

[8] S. A. Cook and R. A. Reckhow. The relative efficiency of propositional proof systems. *J. Symbolic Logic*, 44(1):36–50, 1979. ISSN 0022-4812.

[9] F. Ferreira. What are the $\forall\Sigma_1^{\mathsf{b}}$-consequences of T_2^1 and T_2^2? *Ann. Pure Appl. Logic*, 75(1-2):79–88, 1995. ISSN 0168-0072. Proof theory, provability logic, and computation (Berne, 1994).

[10] P. Hájek and P. Pudlák. *Metamathematics of First-Order Arithmetic.* Perspectives in Mathematical Logic. Springer-Verlag, Berlin, 1993. ISBN 3-540-50632-2.

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

[12] E. Jeřábek. Dual weak pigeonhole principle, Boolean complexity, and derandomization. *Ann. Pure App. Logic*, 129(1–3):1–37, 2004. URL `doi:10.1016/j.apal.2003.12.003`.

[13] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *J. Comput. System Sci.*, 37(1):79–100, 1988. ISSN 0022-0000. 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. ISSN 0019-9958.

[15] J. Krajíček. Dual weak pigeonhole principle, pseudo-surjective functions, and provability of curcuit lower bounds. *J. Symbolic Logic*, 69(1):265–286, 2004. ISSN 0022-4812.

[16] J. Krajíček. No counterexample interpretation and interactive computation. In *Logic from Computer Science (Berkeley, CA, 1989)*, volume 21 of *Math. Sci. Res. Inst. Publ.*, pages 287–293. Springer, New York, 1992.

[17] J. Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory*, volume 60 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 1995. ISBN 0-521-45205-8.

[18] J. Krajíček and P. Pudlák. Quantified propositional calculi and fragments of bounded arithmetic. *Z. Math. Logik Grundlag. Math.*, 36(1):29–46, 1990. ISSN 0044-3050.

[19] J. Krajíček and P. Pudlák. Some consequences of cryptographical conjectures for S^1_2 and EF. *Inform. and Comput.*, 140(1):82–94, 1998. ISSN 0890-5401.

[20] J. Krajíček, P. Pudlák, and G. Takeuti. Bounded arithmetic and the polynomial hierarchy. *Ann. Pure Appl. Logic*, 52(1-2):143–153, 1991. ISSN 0168-0072. International Symposium on Mathematical Logic and its Applications (Nagoya, 1988).

[21] A. Maciel, T. Pitassi, and A. R. Woods. A new proof of the weak pigeonhole principle. *J. Comput. System Sci.*, 64(4):843–872, 2002. ISSN 0022-0000. Special issue on STOC 2000 (Portland, OR).

[22] 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.

[23] 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. ISSN 0022-4812.

[24] C. Pollett. Structure and definability in general bounded arithmetic theories. *Ann. Pure Appl. Logic*, 100(1-3):189–245, 1999. ISSN 0168-0072.

[25] C. Pollett and N. Danner. Circuit prinicples and weak pigeonhole variants. In M. Atkinson and F. Dehne, editors, *Computing: The Australasian Theory Symposium (Newcastle, Australia, 2005)*, volume 41 of *Conferences in Research and Practice in Information Technology*, pages 31–40. Australian Computer Society, 2005.

[26] 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.

[27] S. Riis. Making infinite structures finite in models of second order bounded arithmetic. In *Arithmetic, Proof theory, and Computational Complexity (Prague, 1991)*, volume 23 of *Oxford Logic Guides*, pages 289–319. Oxford Univ. Press, New York, 1993.

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