Unprovability of Lower Bounds on Circuit Size in Certain Fragments of Bounded Arithmetic

Alexander A. Razborov*
School of Mathematics
Institute for Advanced Study
Princeton, NJ 08540
and
Steklov Mathematical Institute
Vavilova 42, 117966, GSP–1
Moscow, RUSSIA

To appear in *Izvestiya of the RAN*

Abstract

We show that if strong pseudorandom generators exist then the statement "α encodes a circuit of size $n^{(\log^* n)}$ for SATISFIABILITY" is not refutable in $S^2_2(\alpha)$. For refutation in $S^1_2(\alpha)$, this is proven under the weaker assumption of the existence of generators secure against the attack by small depth circuits, and for another system which is strong enough to prove exponential lower bounds for constant-depth circuits, this is shown without using any unproven hardness assumptions.

These results can be also viewed as direct corollaries of interpolation-like theorems for certain "split versions" of classical systems of Bounded Arithmetic introduced in this paper.

*Supported by the grant # 93-6-6 of the Alfred P. Sloan Foundation and by the grant # 93-011-16015 of the Russian Foundation for Fundamental Research

1. Introduction

Proving lower bounds on the complexity of explicitly given Boolean functions is one of the most challenging tasks in the computational complexity. This theory met with a remarkable success at least twice: in the 60's (see e.g. [35, 30, 31, 36, 37]) and in more recent time ([11, 1, 27, 12, 32, 33, 28, 2, 25, 29, 34, 22, 4, 15, 17]). Both times, however, the period of enthusiasm was followed by understanding that it is not quite clear to which extent the methods developed so far can be useful for attacking central open problems in Boolean complexity.

A logical analysis of this situation should start with understanding what is the right "minimal" fragment of ZFC which is really needed for formalizing all these methods, and this question was raised in [19]. It was argued there that the conceivable answer is the second order theory of Bounded Arithmetic V^1_1, and no example of a lower bound for explicit function not provable in V^1_1 has been found since that. The next goal is to develop machinery for understanding whether V^1_1 can prove superpolynomial lower bounds on the size of unrestricted circuits or not.

In this paper we present first partial results in this direction. Namely, we show that the existence of a pseudorandom generator secure against the attack by circuits of size 2^{n^ϵ} (for some fixed $\epsilon > 0$) implies that for any explicit Boolean function f_n and any integer-valued $t(n)$ such that $t(n) \geq n^{\omega(1)}$, the theory $S^2_2(\alpha)$ can not refute that α encodes a Boolean circuit of size $t(n)$ for f_n. For the theory $S^1_2(\alpha)$ the same statement holds under the weaker assumption of the existence of a generator secure against n^ϵ-depth circuits.

A few remarks concerning these results should be made immediately.

- Following [19], we work in the strongest possible framework in which α includes encodings of *truth-tables* of all Boolean functions appearing in the circuit as intermediate results.

- We do *not* require that Bounded Arithmetic would *prove* $t(n) \geq n^{\omega(1)}$, we only need this to be true on integers. Thus, our results are still applicable to e.g. $t(n) = n^{\log^* n}$.

- Since we are mostly interested in the provability in V^1_1, this is also natural to consider the hierarchy of its subtheories and wonder whether we can do better for them. The strongest theory in this hierarchy to which our method applies is $IE_1(f)$ (see [26] for the definition of IE_1), and for this theory we indeed can prove a slightly stronger result. Namely, we may replace $t(n)$ by n^k for a *fixed* constant $k > 0$ depending only on the quality of the generator. This improvement, however, is really marginal, so we prefer to work all the time in the language L_2 containing the smash function $\#$.

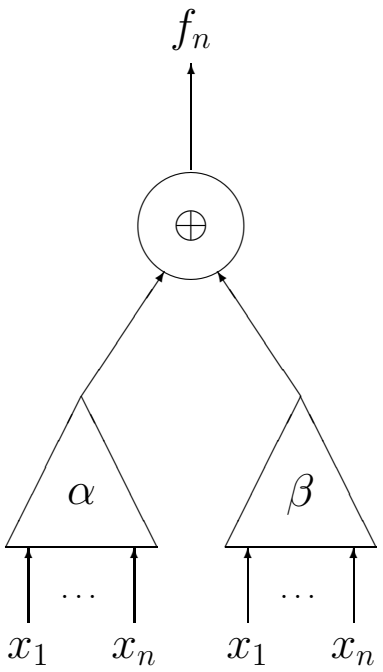

Figure 1: The framework for split versions

For proving these results we define the *split* version $\mathcal{S}(S_2)$ of S_2 as the theory in the language $L_2(\alpha, \beta)$ which allows induction on arbitrary bounded formulae in $L_2(\alpha)$ and arbitrary bounded formulae in $L_2(\beta)$. We consider the pair (α, β) as an encoding of a Boolean circuit with the PARITY gate at the top so that α encodes the left-hand side of the rest, and β encodes the right-hand side (see Figure 1).

$\mathcal{S}(S_2)$ proves in this framework exponential lower bounds on the size of constant-depth circuits over the standard basis. We show that on the other hand it can not prove super-polynomial lower bounds for depth-3 circuits with PARITY gates. We derive the above-mentioned results about $S^1_2(\alpha)$ and $S^2_2(\alpha)$ as direct consequences of similar statements concerning $\mathcal{S}(S_2)$ appended with the corresponding induction schemes.

The proofs consist of several fairly independent pieces. One of essential ingredients is the characterization of the circuit depth by a communication game [15], and a characterization of the circuit size in these terms based upon local search problems (Theorem 3.1 of this paper). These characterizations are non-uniform in their very nature, and this suggests

that our results might be extended to stronger theories allowing more computational power for both players.

To this end we define the split version $\mathcal{S}(V_2)$ of the second order theory V_2 in the same fashion as $\mathcal{S}(S_2)$, and extend our three results to this theory (appended with the appropriate induction scheme for the first two). These extensions follow from general interpolation-like theorems, and this is a close indication that $\mathcal{S}(V_2)$ and its extensions *exactly* capture Karchmer-Wigderson game and its analogue for the circuit size. Unfortunately, these second order versions are somewhat technical. Thus, for the convenience of the reader interested only in classical fragments of Bounded Arithmetic, we start with the simpler first order case.

The paper is organized as follows. In Section 2 we recall necessary definitions from Complexity Theory. In Section 3 we present the new characterization of the circuit size (Theorem 3.1). In Section 4 we briefly survey results from Bounded Arithmetic needed for our purposes. In Section 5 we recall the framework from [19] and introduce its split variant. In Section 6 we present first order versions of our main results, and in Section 7 show that they can be actually derived as corollaries of interpolation-like theorems for split versions of second order theories. The paper is concluded by some remarks and open problems in Section 8.

2. Background from Complexity Theory

In this section we recall necessary definitions and facts from Complexity Theory.

2.1. Boolean Complexity

We address the reader to [5] for an excellent treatment of the subject; the sole purpose of this section is to agree upon notation.

We denote by F_n the set of all Boolean functions in n variables $x_1, \ldots, x_n$. Let $x_i^1 \rightleftharpoons x_i$ and $x_i^0 \rightleftharpoons (\neg x_i)$. Most of the time, it will be convenient to think of $f_n \in F_n$ as of a binary string of length 2^n called the *truth-table* of f_n. We will denote by $S(f_n)$ the circuit size of f_n (over the standard basis $\{\wedge, \vee, \neg\}$ with negations appearing only at variables; all computational nodes must have fan-in 2). $D(f_n)$ is the minimal depth needed for computing f_n in the same model. $S_{mon}(f_n)$ and $D_{mon}(f_n)$ are, respectively, the monotone circuit size and the monotone depth of a monotone f_n. $S_d(f_n)$ is the circuit size with respect to depth-d (unbounded fan-in) circuits. $S_d^{\oplus}(f_n)$ is the same as $S_d(f_n)$, only now we additionally allow PARITY gates.

$SIZE(t(n))$ is the complexity class consisting of all functions $\{f_n\}$ for which $S(f_n) \leq O(t(n))$; $DEPTH(d(n))$ has a similar meaning. The notation $DEPTH, SIZE(d(n), t(n))$ and $DEPTH, SIZE^{\oplus}(d(n), t(n))$ corresponds to unbounded fan-in circuits with simultaneous restrictions $d(n)$ on their depth and $O(t(n))$ on their size. $DEPTH, SIZE(O(1), n^{O(1)})$ is the (non-uniform) class AC^0, $DEPTH, SIZE^{\oplus}(O(1), n^{O(1)})$ will be denoted by $AC^0[2]$, and $DEPTH, SIZE^{\oplus}(d, n^{O(1)})$ will be denoted by $AC^{0,d}[2]$.

All these complexity measures can be in a natural way extended to the case of *partial* Boolean functions $f_n : \{0,1\}^n \longrightarrow \{0,1,*\}$ ($*$ stands for "undefined"). E.g. $S(f_n)$ for a partial f_n is the minimum of $S(\bar{f}_n)$ taken over all total extensions $\bar{f}_n$ of f_n etc.

2.2. Karchmer-Wigderson game

This game was introduced in [15].

Let U, V, I be finite sets, and $R \subseteq U \times V \times I$ be a ternary relation such that

$$\forall u \in U \ \forall v \in V \ \exists i \in I \ ((u, v, i) \in R). \tag{1}$$

Assume that we have two players with unlimited computational power. Let player I receive $u \in U$, and player II receive $v \in V$. Their common task is to find some $i \in I$ such that $(u, v, i) \in R$ exchanging messages between each other. The minimal number of bits (taken over all possible protocols achieving this goal) to be exchanged in the worst case is called the *communication complexity* of R and denoted by $C(R)$.

Now, for a (possibly, partial) Boolean function f_n in n variables consider the relation $R_{f_n} \subseteq f_n^{-1}(0) \times f_n^{-1}(1) \times [n]$ given by $R_{f_n} \rightleftharpoons \{(u, v, i) \mid u_i \neq v_i\}$. If f_n is monotone (that is, has at least one total monotone extension in F_n), define also its monotone analogue $R_{f_n}^{mon}$ by $R_{f_n}^{mon} \rightleftharpoons \{(u, v, i) \mid u_i = 0, v_i = 1\}$.

Proposition 2.1 ([15]). **a)** *For every (partial) Boolean function f, $C(R_f) = D(f)$,*

b) *For every (partial) monotone Boolean function f, $C(R_f^{mon}) = D_{mon}(f)$.*

Denote by $C_d(R)$ the modification of $C(R)$ in which only d rounds are allowed. The following is a slightly refined version of the result implicitly contained in [14, Definition 3.5.2]:

Proposition 2.2. *For every (partial) Boolean function f and every $d > 0$,*

$$2^{\left(\frac{C_d(R_f)}{d} - 1\right)} \leq S_d(f) \leq 2^{C_d(R_f)}.$$

2.3. Polynomial local search problems

This concept was originally considered in [13]. We reproduce here the variant of the definition given in [8].

Definition 2.3. A *local search problem* L consists of a set $F_L(x) \subseteq \mathbf{N}$ of *solutions* for every instance $x \in \mathbf{N}$, an integer-valued *cost function* $c_L(s,x)$ and a *neighborhood function* $N_L(s,x)$ such that:

a) $0 \in F_L(x)$;

b) for all $s \in F_L(x)$, $N_L(s,x) \in F_L(x)$;

c) for all $s \in F_L(x)$, if $N_L(s,x) \neq s$ then $c_L(s,x) < c_L(N_L(s,x),x)$.

A *local optimum* for the problem L on x is an s such that $s \in F_L(x)$ and $N_L(s,x) = s$. A local search problem L is *polynomial* if the binary predicate $s \in F_L(x)$ and the functions $c_L(s,x), N_L(s,x)$ are polynomially time computable, and also there exists a polynomial $p_L(n)$ such that $|s| \leq p_L(|x|)$ for all $s \in F_L(x)$.

Note that the concept of a polynomial local search (PLS) problem can be relativized in a standard way.

2.4. Natural proofs

This concept was introduced in [21].

Let Γ and Λ be complexity classes. Slightly altering the notation from [21], we call a sequence $\{C_n \mid n \in \omega\}$ of subsets $C_n \subseteq F_n$ a Γ-*natural combinatorial property useful against* Λ if it satisfies the following three conditions:

Constructivity: The predicate $f_n \stackrel{?}{\in} C_n$ is computable in Γ (note that the bit size of an input to this problem is 2^n which will be denoted further on by N),

Largeness: $|C_n| \geq 2^{-O(n)} \cdot |F_n|$,

Usefulness: For any sequence of functions f_n, where the event $f_n \in C_n$ happens infinitely often, $\{f_n\} \notin \Lambda$

(our C_n corresponds to C_n^* from [21]). A lower bound proof that some explicit function is not in Λ is called Γ-*natural against* Λ if it leads to a Γ-natural combinatorial property which is useful against Λ.

For a pseudo-random generator $G_n : \{0,1\}^n \longrightarrow \{0,1\}^{2n}$ define its *hardness* $H(G_n)$ as the minimal S for which there exists a circuit C of size $\leq S$ with the property

$$|\mathbf{P}[C(G_n(\boldsymbol{x})) = 1] - \mathbf{P}[C(\boldsymbol{y}) = 1]| \geq \frac{1}{S}. \tag{2}$$

Here $\boldsymbol{x}$ is taken at random from $\{0,1\}^n$, and $\boldsymbol{y}$ is taken at random from $\{0,1\}^{2n}$.

The following is a minor improvement on [21, Theorem 4.1] which is proved in the same way:

Proposition 2.4. *Assume that there exists a* $SIZE\left(2^{(\log N)^{O(1)}}\right)$*-natural combinatorial property which is useful against* $P/poly$ $(= SIZE(n^{O(1)}))$*. Then for every polynomial time computable* $G_k : \{0,1\}^k \longrightarrow \{0,1\}^{2k}$, $H(G_k) \leq 2^{k^{o(1)}}$.

We define *depth hardness* $DH(G_n)$ of G_n as the minimal S for which there exists a circuit C of depth $\leq \log_2 S$ such that (2) holds. The following is analogous to Proposition 2.4:

Proposition 2.5. *Assume that there exists a* $DEPTH\left((\log N)^{O(1)}\right)$*-natural combinatorial property which is useful against* $P/poly$*. Then for every polynomial time computable* $G_k : \{0,1\}^k \longrightarrow \{0,1\}^{2k}$, $DH(G_k) \leq 2^{k^{o(1)}}$.

Note that the classes $SIZE\left(2^{(\log N)^{O(1)}}\right)$, $DEPTH\left((\log N)^{O(1)}\right)$ appearing in the above two propositions are simply non-uniform analogues of quasipolynomial time and POLYLOGSPACE, respectively.

Finally, we improve along the same lines upon [21, Theorem 4.3]:

Proposition 2.6. *There is no* $DEPTH, SIZE\left(O(1), 2^{(\log N)^{O(1)}}\right)$*-natural combinatorial property useful against* $AC^{0,3}[2]$.

3. A new characterization of circuit size

Let U, V, I be finite sets, and $R \subseteq U \times V \times I$ be a ternary relation such that (1) holds. We will be considering those local search problems whose instances x are (encodings of) pairs (u, v); $u \in U, v \in V$.

For any such problem $L =< F_L, c_L, N_L >$, let $C(F_L, c_L)$ be the communication complexity of computing simultaneously the predicate $s \in F_L(u, v)$ and the function $c_L(s, u, v)$ in the model when the first player gets (s, u), and the second gets (s, v) (thus, s is in the public domain). $C(N_L)$ is defined similarly. The *size* of L, by definition, is

$$\left| \bigcup_{\substack{u \in U \\ v \in V}} F_L(u, v) \right| \cdot 2^{2C(F_L, c_L) + C(N_L)}$$

(the meaning of the coefficient 2 in front of $C(F_L, c_L)$ will become clear from the proof of Theorem 3.1).

We say that R *reduces* to L if there exists a function $p : \mathbf{N} \longrightarrow I$ such that for any $(u, v) \in U \times V$ and any local optimum s for L on (u, v), we have $(u, v, p(s)) \in R$. We define $size(R)$ as

$$\min \{ size(L) \mid R \text{ reduces to } L \}.$$

Theorem 3.1. **a)** *For every partial Boolean function f, $size(R_f) = \theta(S(f))$,*

b) *For every monotone partial Boolean function f, $size(R_f^{mon}) = \theta(S_{mon}(f))$.*

Proof. Since the proofs of the two parts are practically identical, we prove only part a).

Let f be a partial Boolean function in n variables, let $t \rightleftharpoons S(f)$, and let C be a size-t circuit computing f. Denote $f^{-1}(0)$ by U, and $f^{-1}(1)$ by V. We want to reduce R_f to a local search problem L of size $O(t)$. Disregarding all inessential variables not appearing in C, we may assume w.l.o.g. that

$$t \geq n - 1. \tag{3}$$

We arrange nodes $w_1, \ldots, w_t$ of the circuit C in such a way that a wire can go from w_μ to w_ν only when $\mu < \nu$. Let f_ν be the function computed at w_ν. Note for the record that f_t is an extension of f that is $f_t(u) = 0, f_t(v) = 1$ for all $u \in U, v \in V$.

We construct L as follows. Encode nodes $w_1, \ldots, w_t$ by integers $n_1, \ldots, n_t$ so that $n_t = 0$ and $\{1, \ldots, n\} \cap \{n_1, \ldots, n_t\} = \emptyset$. Let

$$\begin{aligned}
F_L(u, v) &\rightleftharpoons \{ i \mid 1 \leq i \leq n \ \&\ u_i \neq v_i \} \cup \{ n_\nu \mid 1 \leq \nu \leq t \ \&\ f_\nu(u) = 0 \ \&\ f_\nu(v) = 1 \}, \\
c_L(i, u, v) &\rightleftharpoons 0 \text{ for } 1 \leq i \leq n, \\
N_L(i, u, v) &\rightleftharpoons i \text{ for } 1 \leq i \leq n, \\
c_L(n_\nu, u, v) &\rightleftharpoons \nu \text{ for } 1 \leq \nu \leq t.
\end{aligned}$$

$N_L(n_\nu, u, v)$ is defined as follows. If $n_\nu \notin F_L(u, v)$, let $N_L(n_\nu, u, v) \rightleftharpoons 0$. Otherwise, that is when $f_\nu(u) = 0$ and $f_\nu(v) = 1$, we choose one of the two sons of the node w_ν for which this property is preserved. If this son is a computational node w_μ, we let $N_L(n_\nu, u, v) \rightleftharpoons n_\mu$; if this is a leaf x_i^ϵ, we let $N_L(n_\nu, u, v) \rightleftharpoons i$.

It is straightforward to check that so defined L is a local search problem, and that R_f reduces to L. Also, $C(F_L, C_L) \leq 2$ and $C(N_L) \leq 3$. Hence $size(L) \leq O(n + t)$ which is $O(t)$ due to (3).

For another (non-trivial) direction, assume that R_f reduces via a function p to a local search problem L. Let $h_0 \rightleftharpoons 2^{C(F_L, c_L)}$ and $h_1 \rightleftharpoons 2^{C(N_L)}$. Then for every fixed $s \in \bigcup_{\substack{u \in U \\ v \in V}} F_L(u, v)$ we have a communication protocol P_s for computing the binary relation $s \in F_L(u, v)$ and the cost function $c_L(s, u, v)$ which has at most h_0 different histories. These histories define a partition of $U \times V$ into rectangles $U_{s,1} \times V_{s,1}; \ldots; U_{s,h_0} \times V_{s,h_0}$ such that F_L, c_L are fully determined on $U_{s,i} \times V_{s,i}$. That is to say, for some predicates $\alpha_s \subseteq [h_0]$ and some functions $\eta_s : [h_0] \longrightarrow \mathbf{N}$ the following is true for all $i \in [h_0]$ and for all $(u, v) \in U_{s,i} \times V_{s,i}$:

$$s \in F_L(u, v) \text{ iff } i \in \alpha_s$$

and

$$c_L(s, u, v) = \eta_s(i).$$

We call those rectangles $U_{s,i} \times V_{s,i}$ for which $i \in \alpha_s$ *good*. We call $\eta_s(i)$ the *cost* of rectangle $U_{s,i} \times V_{s,i}$. We order all good rectangles in such a way that their costs are non-decreasing:

$$U^1 \times V^1; \ldots; U^{H_0} \times V^{H_0}.$$

Here $H_0 \leq \left| \bigcup_{\substack{u \in U \\ v \in V}} F_L(u, v) \right| \cdot h_0$.

We construct by induction on $\nu \leq H_0$ a circuit C_ν which has the following property. For every $\mu \leq \nu$ there exists a node w_μ of C_ν computing a function f_μ such that $f_\mu|_{U^\mu} \equiv 0$ and $f_\mu|_{V^\mu} \equiv 1$. Assume that we already have $C_{\nu-1}$. C_ν will be obtained from it by adding at most $h_0 h_1$ new nodes for computing a f_ν with required properties from already available $f_1, \ldots, f_{\nu-1}$.

Let $U^\nu \times V^\nu = U_{s,i} \times V_{s,i}$. Consider the following communication protocol P_s^* of complexity at most $C(F_L, c_L) + C(N_L)$. First we run the optimal protocol for computing $N_L(s, u, v)$. Let $s' \rightleftharpoons N_L(s, u, v)$ be its outcome. Then we run $P_{s'}$.

We introduce Boolean variables $y_1, \ldots, y_H$ *for those histories of* P_s^* *which actually correspond to at least one instance* $(u, v) \in U_{s,i} \times V_{s,i}$. For every $u \in U_{s,i}$ let $\bar{u}$ be the assignment on $\{0, 1\}^H$ defined by letting $\bar{u}_h$ be 0 if there exists $v \in V_{s,i}$ such that the computation of

P_s^* on (u,v) develops according to the history h, and 1 otherwise. Dually, $\bar{v}_h = 1$ iff there exists $u \in U_{s,i}$ so that the pair (u,v) leads to the history h. For every pair $(u,v) \in U_{s,i} \times V_{s,i}$ we have $\bar{u}_h = 0, \bar{v}_h = 1$, where h is the history of P_s^* corresponding to this pair. Hence, the partial Boolean function $\widehat{f}_\nu(y_1,\ldots,y_H)$ outputting 0 on $\{\,\bar{u} \mid u \in U_{s,i}\,\}$, outputting 1 on $\{\,\bar{v} \mid v \in V_{s,i}\,\}$ and undefined elsewhere, is monotone, and, moreover, the protocol P_s^* finds a solution to $R^{mon}_{\widehat{f}_\nu}$. Hence, by Proposition 2.1 b), $D_{mon}(\widehat{f}_\nu) \leq C(F_L, c_L) + C(N_L)$, and the same bound holds for some total monotone extension $\bar{f}_\nu$ of $\widehat{f}_\nu$. Note for the record that this implies $S_{mon}(\bar{f}_\nu) \leq h_0 h_1$.

Consider now a particular history of P_s^*, h. Let (s',j) be the corresponding output (here s' is the output of computing N_L, and j is the subhistory corresponding to the subprotocol $P_{s'}$). By Definition 2.3 b), the rectangle $U_{s',j} \times V_{s',j}$ is good. By part c) of this definition, either $s' = s$ or the cost of $U_{s',j} \times V_{s',j}$ is strictly less than the cost of $U_{s,i} \times V_{s,i}$.

In the first case s is a local optimum for L on every $(u,v) \in U_{s,i} \times V_{s,i}$ belonging to the non-empty rectangle which corresponds to h. Since R_f reduces to L, this means that $u_{p(s)} \neq v_{p(s)}$ for every such pair, and this implies that actually $u_{p(s)} = \epsilon, v_{p(s)} = (\neg\epsilon)$ for some fixed $\epsilon \in \{0,1\}$. Let $y'_h \rightleftharpoons x^{(\neg\epsilon)}_{p(s)}$.

In the second case $U_{s',j} \times V_{s',j} = U^\mu \times V^\mu$ for some $\mu < \nu$. Let $y'_h \rightleftharpoons f_\mu$.

Finally, let $f_\nu \rightleftharpoons \bar{f}_\nu\,(y'_1,\ldots,y'_H)$. f_ν can be computed by appending to $C_{\nu-1}$ at most $h_0 h_1$ new nodes.

Since for every $u \in U^\nu$, $\bar{f}_\nu\,(\bar{u}_1,\ldots,\bar{u}_H) = 0$, and $\bar{f}_\nu$ is monotone, in order to check that $f_\nu(u) = 0$ for $u \in U^\nu$, we only have to check that $y'_h(u) \leq \bar{u}_h$ for any history h. For doing this simply note that if $\bar{u}_h = 0$, then for some $v \in V^\nu$ the computation on (u,v) proceeds along h, which, due to our choice of y'_h, implies $y'_h(u) = 0$. By the dual argument, $f_\nu(v) = 1$ for all $v \in V^\nu$.

This completes the construction of C_ν.

Now, C_{H_0} has size at most $H_0 h_0 h_1$. Also, due to Definition 2.3 a), all rectangles $U_{0,i} \times V_{0,i}$ are good. Thus, applying the same argument as above and adding to C_{H_0} at most h_0 new nodes, we finally compute f by a circuit of size $O(size(L))$. This completes the proof of Theorem 3.1.■

4. Background from Bounded Arithmetic

We assume the familiarity with [6] and use the now-standard notation for denoting various hierarchies and fragments of Bounded Arithmetic from that book. We denote by L_2 Buss's first order language which consists of the constant 0, function symbols $S, +, \cdot, \lfloor \frac{1}{2}x \rfloor, |x|, x\#y$

and of the predicate symbol $\leq$. $BASIC_2$ is the set of 32 open axioms in the language L_2 from [6, §2.2] describing basic properties of its symbols. $\Sigma^b \rightleftharpoons \bigcup_{i\geq 0} \Sigma^b_i$ is the set of all (first-order) bounded formulae of L_2.

In [19] a convenient technical notion of a regular theory was introduced. The meaning of this notion is that many proofs in Bounded Arithmetic which do not involve the smash function # can be generalized to arbitrary regular theories. In this paper we need a stronger notion which is good also for #-involving proofs.

Definition 4.1. A first order theory R in a language $L \supseteq L_2$ is *strongly regular* if it possesses the following properties:

a) $BASIC_2 \subseteq R$,

b) R can be axiomatized by Σ^b_0-formulae,

c) every function symbol (and hence every term) of the language L can be bounded from above in the theory R by a term of the language L_2.

For a strongly regular theory R in a language L we denote by S^i_R the theory $R+\Sigma^b_i(L)-PIND$, and by T^i_R the theory $R+\Sigma^b_i(L)-IND$. Let also $S_R \rightleftharpoons \bigcup_{i=0} S^i_R$; this is the same theory as $T_R \rightleftharpoons \bigcup_{i=0} T^i_R$.

If $L = L_2$ and $R = BASIC_2$ then S^i_R is simply S^i_2, and T^i_R is T^i_2. Another important example is $L = L_2(\gamma)$, $R = BASIC_2$ (γ is a new predicate variable). In this case S^i_R and T^i_R coincide with ordinary theories $S^i_2(\gamma)$ and $T^i_2(\gamma)$. A less trivial example is provided by $L = L_{PV}$, $R =$ "$BASIC_2 + \Pi^b_1$-defining axioms for PV-symbols" (see [6, §6.2]), where PV is Cook's equational system [10]. In this case S^1_R is the theory $S^1_2(L_{PV})$ as defined in [6]. One more example of this sort will be given in Section 6.

As we already mentioned, the meaning of this definition is that many (if not all) results proven for S^i_2, T^i_2 relativize to arbitrary strongly regular theories R. For example, the (weaker form of) the main theorem from [6] in this setting looks like this:

Proposition 4.2. *Let R be a strongly regular theory in a language L extending L_2. Suppose $S^1_R \vdash \exists y\ A(\vec{a}, y)$, where $A(\vec{a}, b)$ is a $\Sigma^b_1(L)$-formula with all its free variables displayed. Then there is a polynomial time oracle Turing machine M allowed to ask queries of the form $\vec{n} \stackrel{?}{\in} P$ or $f(\vec{n}) =?$, where P is a predicate symbol of $L \setminus L_2$, and f is a function symbol of $L \setminus L_2$, such that the following holds.*

For every model $(\mathbf{N}, \Omega)$ of the theory R expanding the standard model of $BASIC_2$ and every tuple $\vec{n} \in \mathbf{N}$,

$$(\mathbf{N}, \Omega) \models A\left(\vec{n}, M^{\Omega}(\vec{n})\right).$$

Here Ω is the interpretation of symbols from $L \setminus L_2$, and $M^{\Omega}(\vec{n})$ is the result of the computation of M on $\vec{n}$ when M is fed with the oracle Ω.

We also need the following conservation result from [7]:

Proposition 4.3. *For any strongly regular theory R in a language $L \supseteq L_2$, S^2_R is $\Sigma^b_2(L)$-conservative over T^1_R.*

Finally, we recall the characterization of Σ^b_1-defined in T^1_2 functions in terms of PLS-problems [8]. Once again, we present the relativized version.

Proposition 4.4. *Let R be a strongly regular theory in a language $L \supseteq L_2$. Suppose $T^1_R \vdash \exists y\, A(a, y)$, where $A(a, b)$ is a $\Sigma^b_1(L)$-formula with all its free variables displayed. Then there is an oracle PLS-problem K, where the associated oracle computations of F_K, c_K, N_K are allowed to ask queries of the form $\vec{n} \stackrel{?}{\in} P$ or $f(\vec{n}) =?$; P, f being symbols of $L \setminus L_2$, and a (polynomial-time computable) function $p(s)$ such that the following holds.*

For every model $(\mathbf{N}, \Omega)$ of the theory R expanding the standard model of $BASIC_2$, every $x \in \mathbf{N}$, and every local optimum s for K^{Ω} on x,

$$(\mathbf{N}, \Omega) \models A(x, p(s)).$$

5. Boolean Complexity and Bounded Arithmetic: split framework

In our formalization of problems studied in Boolean complexity within the framework provided by Bounded Arithmetic we follow [19, Appendix A]. Namely, let $Circuit(t, N, \gamma)$ be a $\Sigma^b(\gamma)$-formula asserting that γ encodes the protocol of computation by a circuit of size t in $|N|$ variables. Similarly, for a fixed $d > 0$, let $Circuit_d(t, N, \gamma)$ and $Circuit^{\oplus}_d(t, N, \gamma)$ assert that $Circuit(t, N, \gamma)$ and, moreover, γ is a depth-d circuit or depth-d circuit with PARITY gates, respectively. Let $Output(t, N, x, \gamma)$ be a $\Sigma^b(\gamma)$-formula which represents the output of γ (viewed as a circuit of size t in $|N|$ variables) on a Boolean string x. The exact details of these encodings are unimportant; the only extra property which we require (and which is shared by all reasonable schemes) is that we can easily combine in this framework two circuits to compute PARITY of their outputs as shown on Figure 1. More precisely, we require that there exists a $\Delta^b_1(\alpha, \beta)$ (with respect to $S^1_2(\alpha, \beta)$) abstract

$PARITY(t, N, \alpha, \beta)$ such that

$$\left.\begin{array}{l} S^1_2(\alpha,\beta) \vdash \Big(Circuit(\lfloor (t\dot{-}3)/4\rfloor, N, \alpha) \wedge Circuit(\lfloor (t\dot{-}3)/4\rfloor, N, \beta)\Big) \supset \\ \quad \Big(Circuit(t, N, PARITY(t, N, \alpha, \beta)) \wedge \forall x \in \{0,1\}^{|N|} \\ \qquad (Output(\lfloor (t\dot{-}3)/4\rfloor, N, x, \alpha) \oplus Output(\lfloor (t\dot{-}3)/4\rfloor, N, x, \beta) \ \equiv \\ \qquad\quad Output(t, N, x, PARITY(t, N, \alpha, \beta)))\Big). \end{array}\right\} \quad (4)$$

Like in [19], we are mostly interested in the provability of the formula

$$Circuit(t(N), N, \gamma) \supset \exists x \in \{0,1\}^{|N|}(Output(t(N), N, x, \gamma) \not\equiv S(N, x)), \quad (5)$$

where $t(N)$ is a Σ^b-definable function such that $\mathbf{N} \models t(N) \geq (\log N)^{\omega(1)}$, and $S(N, a)$ is in Σ^b. (5) asserts that there is no circuit of size $t(N)$ (remember that $N \approx 2^n$) computing the Boolean function $\{x\}S(N, x)$; we denote this formula by $LB(t, S, \gamma)$. $LB_d(t, S, \gamma)$ and $LB^{\oplus}_d(t, S, \gamma)$ are obtained from $LB_d(t, S, \gamma)$ after replacing $Circuit(t, N, \gamma)$ by $Circuit_d(t, N, \gamma)$ and $Circuit^{\oplus}_d(t, N, \gamma)$, respectively.

One of the main results of this paper (Corollary 6.5) says that if sufficiently strong pseudorandom generators exist, then $S^2_2(\gamma) \nvdash LB(t, S, \gamma)$ for *any* choice of t, S with the above properties. We can, however, prove a stronger result at the same cost and better explain the mechanism of the proof if we split our circuit into two pieces as shown on Figure 1. The corresponding statement, denoted by $SLB(t, S, \alpha, \beta)$ is

$$\begin{array}{l} (Circuit(t(N), N, \alpha) \wedge Circuit(t(N), N, \beta)) \supset \\ \quad \exists x \in \{0,1\}^{|N|}(Output(t(N), N, x, \alpha) \oplus Output(t(N), N, x, \beta) \ \not\equiv \ S(N, x)). \end{array}$$

$SLB_d(t, S, \alpha, \beta)$ and $SLB^{\oplus}_d(t, S, \alpha, \beta)$ have the obvious meaning.

We are going to allow unlimited reasoning about each of the two halves α, β alone. In this and the next section we do as much as we can within the first order framework, and, with this restriction, we implement our idea as follows.

Denote by $\mathcal{S}(L_2)$ the language $L_2(\alpha, \beta)$ obtained from L_2 by appending to it two new unary predicate variables α and β, and define the *split hierarchy* $\mathcal{S}\Sigma^b_i$, $\mathcal{S}\Pi^b_i$ of bounded formulae in this language similarly to the ordinary hierarchy Σ^b_i, Π^b_i (see [6, §2.1]) with the exception of the base case. Namely, $\mathcal{S}\Sigma^b_0 = \mathcal{S}\Pi^b_0$ is the set of *all* bounded formula in the language $L_2(\alpha)$ *plus* the set of all bounded formulae in $L_2(\beta)$. The inductive definition of $\mathcal{S}\Sigma^b_{i+1}$, $\mathcal{S}\Pi^b_{i+1}$ is the same as for Σ^b_{i+1},Π^b_{i+1}. Note that $\mathcal{S}\Sigma^b_0$ is *not* closed under applying

the connectives $\wedge, \vee$ or sharply bounded quantifiers although all $\mathcal{S}\Sigma^b_i$, $\mathcal{S}\Pi^b_i$ for $i > 0$ are so closed.

Our "base" theory $\mathcal{S}(S_2)$ is the theory in the language $\mathcal{S}(L_2)$ with the set of axioms $BASIC_2 + \mathcal{S}\Sigma^b_0 - IND$. Another, more expressive description of $\mathcal{S}(S_2)$ (which also justifies the notation) is that it is axiomatized by $S_2(\alpha) + S_2(\beta)$.

We conclude this section by showing that $\mathcal{S}(S_2)$ is already capable of proving some non-trivial lower bounds.

Theorem 5.1. *For every fixed $d \geq 2$,*

$$\mathcal{S}(S_2) \vdash SLB_d(t, S, \alpha, \beta),$$

where $t(N) \rightleftharpoons \lfloor 2^{\frac{1}{50}|N|^{1/(2d-3)}} \rfloor$ and $S(N, x) \rightleftharpoons x_1 \oplus \cdots \oplus x_{|N|}$.

Proof. Arguing informally in $\mathcal{S}(S_2)$, let α and β be depth-d circuits of size at most $t(N)$. Since Hstad Switching Lemma is available in $S_2(\alpha)$ (see [19, Appendix E.4]), we can find a restriction ρ assigning at least $\frac{1}{5}|N|^{\frac{d-1}{2d-3}}$ stars and reducing the output of α to a constant. ρ, however, is coded by an integer, thus we can apply in $\mathcal{S}(S_2)$ the same argument to $\beta|_\rho$ and find an extension ρ' of ρ assigning at least two stars and reducing β to a constant as well. Now we take any two adjacent inputs compatible with ρ'; one of them will satisfy $Output(t(N), N, x, \alpha) \oplus Output(t(N), N, x, \beta) \not\equiv x_1 \oplus \cdots \oplus x_{|N|}$.■

6. Main results: first order versions

Throughout the rest of the paper, $t(N)$ will stand for a Σ^b-definable in S_2 function such that $\mathbf{N} \models t(N) \geq (\log N)^{\omega(1)}$, and $S(N, a)$ will stand for an arbitrary bounded formula.

We start with our base theory $\mathcal{S}(S_2)$ and show that it can not prove superpolynomial lower bounds for depth-3 circuits allowing PARITY gates. This, together with Theorem 5.1, provides some formal evidence toward the remark made in [21, Section 3.2] that [34, 22, 4] *had to require arguments from a stronger class than those of* [11, 27, 12].

Theorem 6.1. *For any $t(N), S(N, a)$ with the above properties,*

$$\mathcal{S}(S_2) \nvdash SLB^{\oplus}_3(t, S, \alpha, \beta).$$

The next theory of interest to us is $\mathcal{S}(S_2) + \mathcal{S}\Sigma^b_1 - PIND$.

Theorem 6.2. *Assume that there exists a polynomial time computable generator* $G_k : \{0,1\}^k \longrightarrow \{0,1\}^{2k}$ *with* $DH(G_k) \geq 2^{k^{\Omega(1)}}$. *Then for any* $t(N), S(N,a)$ *as above,*

$$\mathcal{S}(S_2) + \mathcal{S}\Sigma_1^b - PIND \not\vdash SLB(t, S, \alpha, \beta).$$

Corollary 6.3. *Under the same assumption as in Theorem* 6.2*,*

$$S_2^1(\alpha) \not\vdash LB(t, S, \alpha).$$

Proof of Corollary 6.3 from Theorem 6.2. Assume the contrary, that is $S_2^1(\alpha) \vdash LB(t, S, \alpha)$. Substitute in this proof the $\Delta_1^b(\alpha, \beta)$-abstract $PARITY(t(N), N, \alpha, \beta)$ for α. Then we will have $S_2^1(\alpha, \beta) \vdash SLB(t', S, \alpha, \beta)$, where $t'(N) \rightleftharpoons \lfloor (t(N) \dot{-} 3)/4 \rfloor$. This contradicts Theorem 6.2 (applied to $t := t'$) since $S_2^1(\alpha, \beta)$ is a subtheory of $\mathcal{S}(S_2) + \mathcal{S}\Sigma_1^b - PIND$.■

Our main result is similar to Theorem 6.2.

Theorem 6.4. *Assume that there exists a polynomial time computable generator* $G_k : \{0,1\}^k \longrightarrow \{0,1\}^{2k}$ *with* $H(G_k) \geq 2^{k^{\Omega(1)}}$. *Then for any* $t(N), S(N,a)$ *with the properties stated in the beginning of this section,*

$$\mathcal{S}(S_2) + \mathcal{S}\Sigma_2^b - PIND \not\vdash SLB(t, S, \alpha, \beta).$$

Corollary 6.5. *Under the same assumption as in Theorem* 6.4*,*

$$S_2^2(\alpha) \not\vdash LB(t, S, \alpha).$$

Proof is the same as that of Corollary 6.3.■

We begin proving these results with a straightforward definition of the *skolemization* $\widehat{S_2(\gamma)}$ of the theory $S_2(\gamma)$. Firstly, we define the language $\widehat{L_2(\gamma)}$ as the extension of $L_2(\gamma)$ obtained by recursively appending to it new function symbols $f_{A,t}(\vec{b})$ for every open formula $A(a, \vec{b})$ and term $t(\vec{b})$ of the language $\widehat{L_2(\gamma)}$; all occurrences of free variables in A, t are explicitly displayed.

$\widehat{S_2(\gamma)}$ is the open theory in the language $\widehat{L_2(\gamma)}$ axiomatized by $BASIC_2$ and the following *defining axioms* for $f_{A,t}$:

$$\begin{aligned}
&\forall \vec{y}\ f_{A,t}(\vec{y}) \leq t(\vec{y});\\
&\forall x, \vec{y}\,((x \leq t(\vec{y}) \wedge A(x, \vec{y})) \supset (A(f_{A,t}(\vec{y}), \vec{y}) \wedge f_{A,t}(\vec{y}) \leq x))\,;\\
&\forall \vec{y}\,(\neg A(f_{A,t}(\vec{y}), \vec{y}) \supset f_{A,t}(\vec{y}) = 0)\,.
\end{aligned}$$

Thus, the intended meaning of $f_{A,t}(\vec{b})$ is simply $\mu x \leq t(\vec{b})A(x,\vec{b})$. The following summarizes some easy properties of this theory:

Lemma 6.6. **a)** *For every $A \in \Sigma^b(\gamma)$ there exists $A' \in Open\left(\widehat{L_2(\gamma)}\right)$ such that $\widehat{S_2(\gamma)} \vdash A \equiv A'$, and vice versa;*

b) *$\widehat{S_2(\gamma)}$ is a strongly regular open extension of $S_2(\gamma)$ by definitions.*

We define the extension $\widehat{\mathcal{S}}(L_2)$ of $\mathcal{S}(L_2)$ as $\widehat{L_2(\alpha)} + \widehat{L_2(\beta)}$, where we assume, of course, that all non-logical symbols symbols in $\widehat{L_2(\alpha)}$ and $\widehat{L_2(\beta)}$ other than those of L_2 are pairwise distinct. Finally, let $\widehat{\mathcal{S}}(S_2)$ be the theory $\widehat{S_2(\alpha)}+\widehat{S_2(\beta)}$ in the language $\widehat{\mathcal{S}}(L_2)$. The following properties are inherited from Lemma 6.6:

Lemma 6.7. **a)** *For every $A \in \mathcal{S}\Sigma^b_0$ there exists $A' \in Open\left(\widehat{L_2(\alpha)}\right) \cup Open\left(\widehat{L_2(\beta)}\right)$ such that $\widehat{\mathcal{S}}(S_2) \vdash A \equiv A'$, and vice versa;*

b) *$\widehat{\mathcal{S}}(S_2)$ is a strongly regular open extension of $\mathcal{S}(S_2)$ by definitions. Thus, $\widehat{\mathcal{S}}(S_2)$ is conservative over $\mathcal{S}(S_2)$, and every model of $\mathcal{S}(S_2)$ has an unique extension to a model of $\widehat{\mathcal{S}}(S_2)$.*

The following observation provides a crucial link between the theory $\widehat{\mathcal{S}}(S_2)$ and the communication game from Section 2.2.

Lemma 6.8. *Let $s(a_1,\ldots,a_r,\alpha,\beta)$ be a term of the language $\widehat{\mathcal{S}}(L_2)$ with all its free variables displayed. Consider the following communication problem: player I receives $n_1,\ldots,n_r \in \mathbf{N}$ and a language $A \subseteq \mathbf{N}$; player II receives the same $n_1\ldots,n_r$ and $B \subseteq \mathbf{N}$, and they want to compute $s(n_1,\ldots,n_r,A,B)$ in the extension of the model $(\mathbf{N},A,B)$ of $\mathcal{S}(S_2)$ to a model of $\widehat{\mathcal{S}}(S_2)$. Then there exists a constant d depending only on the term s and a d-round communication protocol solving this problem whose complexity is polynomial in $|n_1|+\cdots+|n_r|$.*

Proof. Obvious induction on the logical depth of s (every function symbol of the language $\widehat{\mathcal{S}}(L_2)$ can be evaluated by one of the two players alone, and results of all intermediate evaluations are of polynomial length).■

Now we are ready to prove the results stated in the beginning of this section.

Proof of Theorem 6.1. Assume the contrary, that is $\mathcal{S}(S_2) \vdash SLB^{\oplus}_3(t,S,\alpha,\beta)$. Then also $\widehat{\mathcal{S}}(S_2) \vdash SLB^{\oplus}_3(t,S,\alpha,\beta)$. But the theory $\widehat{\mathcal{S}}(S_2)$ is open, and, by Lemma 6.7

a), the formulae $Circuit_3^{\oplus}(t(N), N, \alpha)$, $Circuit_3^{\oplus}(t(N), N, \beta)$, $Output(t(N), N, x, \alpha)$ and $Output(t(N), N, x, \beta)$ are equivalent in $\widehat{\mathcal{S}}(S_2)$ to open formulae. Thus, by Herbrand's theorem, there exist terms $s_1(N, \alpha, \beta), \ldots, s_r(N, \alpha, \beta)$ of the language $\widehat{\mathcal{S}}(L_2)$ such that

$$\begin{aligned}\widehat{\mathcal{S}}(S_2) \vdash \Big(Circuit_3^{\oplus}(t(N), N, \alpha) \wedge Circuit_3^{\oplus}(t(N), N, \beta)\Big) \supset \\ \bigvee_{i=1}^{r} \Big(s_i(N, \alpha, \beta) \in \{0,1\}^{|N|} \wedge \\ (Output(t(N), N, s_i(N, \alpha, \beta), \alpha) \oplus Output(t(N), N, s_i(N, \alpha, \beta), \beta) \not\equiv \\ S(N, s_i(N, \alpha, \beta)))\Big).\end{aligned}$$

Let n be an integer, and $N \rightleftharpoons 2^n - 1$. By Lemma 6.8, there exists a communication protocol in which the first player receives n and a depth-3 size-$t(N)$ circuit C_1 in n variables allowing PARITY gates, the second player receives n and a circuit C_2 of the same kind, and they produce an input string x such that

$$C_1(x) \oplus C_2(x) \not\equiv S(N, x) \tag{6}$$

within $O(1)$ rounds and $n^{O(1)}$ bits exchanged. For doing this they simply compute

$$s_1(N, C_1, C_2), \ldots, s_r(N, C_1, C_2)$$

and find among this list some x satisfying (6).

But this protocol also gives raise to a similar protocol in which the players, instead of circuits, receive only Boolean functions $f_1, f_2 \in F_n$ such that $S_3^{\oplus}(f_1) \leq t(N)$ and $S_3^{\oplus}(f_2) \leq t(N)$. In fact, the players, using their unlimited power, simply reconstruct some C_1, C_2 computing f_1 and f_2, respectively, and then run the protocol above.

Let us now consider the partial Boolean function $\mathcal{F}_n$ in 2^n variables (we will call it a *functional*) which outputs a 1 on f if $S_3^{\oplus}(f) \leq t(N)$, outputs a 0 if $S_3^{\oplus}(f \oplus s_n) \leq t(N)$ (here $s_n(x) \rightleftharpoons S(N, x)$) and is undefined elsewhere. Then our protocol for every f_1, f_2 such that $\mathcal{F}_n(f_1) = 1$ and $\mathcal{F}_n(f_2) = 0$ finds a position x where $f_1(x) \neq f_2(x)$ (note that the second player should modify his f_2 to $f_2 \oplus s_n$ before entering the protocol from the previous paragraph). Hence, by Proposition 2.2, there exists $E_n \subseteq F_n$ in $DEPTH, SIZE\left(O(1), 2^{(\log N)^{O(1)}}\right)$ such that $\mathcal{F}_n^{-1}(1) \subseteq E_n$ and $\mathcal{F}_n^{-1}(0) \cap E_n = \emptyset$. If $|E_n| \geq \frac{1}{2}F_n$ then $E_n \oplus s_n$ makes a $DEPTH, SIZE\left(O(1), 2^{(\log N)^{O(1)}}\right)$-natural combinatorial property useful against $AC^{0,3}[2]$ since $t(N) \geq n^{\omega(1)}$ and for every $f_n \in E_n \oplus s_n$ we have the bound $S_3^{\oplus}(f_n) > t(N)$. Otherwise, $F_n \setminus E_n$ is such a property. We have arrived at a contradiction with Proposition 2.6.■

Proof of Theorem 6.2. Suppose $\mathcal{S}(S_2) + \mathcal{S}\Sigma^b_1 - PIND \vdash SLB(t, S, \alpha, \beta)$. By Lemma 6.7 a), the class of $\mathcal{S}\Sigma^b_1$-formulae is equivalent in $\widehat{\mathcal{S}}(S_2)$ to the class of $\Sigma^b_1\left(\widehat{\mathcal{S}}(L_2)\right)$-formulae. Denoting $\widehat{\mathcal{S}}(S_2)$ by R, we see that $\widehat{\mathcal{S}}(S_2) + \mathcal{S}\Sigma^b_1 - PIND$ is actually equivalent to S^1_R. In particular, $S^1_R \vdash SLB(t, S, \alpha, \beta)$. But R is strongly regular by Lemma 6.7 b), hence we can apply to it Proposition 4.2. We find a polynomial time (in n) oracle Turing machine M asking queries which depend either only on C_1 or only on C_2; C_1, C_2 being this time size-$t(N)$ circuits, and producing a length n string x with the property (6). But the two players, one holding (n, C_1) and another holding (n, C_2), can simulate M exchanging only $n^{O(1)}$ bits between each other. Now the proof is completed by the same argument as in the proof of Theorem 6.1 on the base of Propositions 2.1 a) and 2.5.■

Proof of Theorem 6.4. Suppose $\mathcal{S}(S_2) + \mathcal{S}\Sigma^b_2 - PIND \vdash SLB(t, S, \alpha, \beta)$. Let, once again, $R \rightleftharpoons \widehat{\mathcal{S}}(S_2)$. Then $\mathcal{S}(S_2) + \mathcal{S}\Sigma^b_2 - PIND$ is equivalent to S^2_R, and $S^2_R \vdash SLB(t, S, \alpha, \beta)$. By Proposition 4.3, $T^1_R \vdash SLB(t, S, \alpha, \beta)$. By Proposition 4.4, there is an oracle PLS-problem K and a function $p(s)$ such that for any two circuits C_1, C_2 of size at most $t(N)$, and any local optimum s for K^{C_1,C_2} on N, $p(s)$ is a binary string x of length n for which (6) holds.

Now we change our view and consider C_1, C_2 simply as extra inputs to K rather than as oracles, and let K_n be its subproblem obtained by fixing n to a particular value. Then the relation $R_{\mathcal{F}_n}$ corresponding to $\mathcal{F}_n$ ($\mathcal{F}_n$ is the functional defined as in the proof of Theorem 6.1) reduces to K_n if we encode a pair (f_1, f_2) by (C_1, C_2), where C_1 is a size-$t(N)$ circuit computing f_1, and C_2 is a size-$t(N)$ circuit computing $f_2 \oplus s_n$. Also, $size(K_n) \leq 2^{(\log N)^{O(1)}}$. Thus, by Theorem 3.1, $\mathcal{F}_n$ is computable by circuits of size $2^{(\log N)^{O(1)}}$, and we can apply Proposition 2.4 to complete the proof.■

7. Interpolation-like theorems in the second order setting

The proof of Proposition 2.1, as well as of Theorem 3.1 in the non-trivial direction involves a highly non-constructive step of deciding whether a rectangle is empty (cf. the sentence "those histories of P^*_s which actually correspond to at least one instance $(u, v) \in U_{s,i} \times V_{s,i}$" on page 9). This step seems to be intractable if we want to prove syntactic analogues of the results from the previous section within the framework provided by first order theories. In this section we briefly outline how to extend this framework to second order theories,

and present in this more general setting interpolation-like theorems which actually imply these results.

Let $\mathcal{L}_2$ be the second order extension of L_2 obtained by augmenting it with second order variables $\gamma_1, \gamma_2, \ldots$ (for simplicity we allow only unary variables). Let $\mathcal{S}(\mathcal{L}_2)$ be the second order language which has one sort for first order variables and two different sorts for second order variables. We will be denoting second order variables of the first sort by $\alpha_1, \alpha_2, \ldots$ (free variables) and $\phi_1, \phi_2, \ldots$ (bound variables); second order variables of the second sort will be denoted by $\beta_1, \beta_2, \ldots, \psi_1, \psi_2, \ldots$ We fix the notation $\mathcal{L}_2^\alpha$ $[\mathcal{L}_2^\beta]$ for the sublanguage of $\mathcal{S}(\mathcal{L}_2)$ (isomorphic to $\mathcal{L}_2$) which allows second order variables only of the first sort [of the second sort, respectively]. For a formula $A(\gamma_1, \ldots, \gamma_r)$ of $\mathcal{L}_2$ with all free second order variables displayed, we denote by $A^\alpha(\alpha_1, \ldots, \alpha_r)$ and $A^\beta(\beta_1, \ldots, \beta_r)$ its isomorphic copies in $\mathcal{L}_2^\alpha$ and $\mathcal{L}_2^\beta$, respectively.

We form the hierarchy $\Sigma_i^{w1,b}$ of second order bounded formulae similarly to the ordinary hierarchy $\Sigma_i^{1,b}$ (see [6, §9.1]) with the exception that the forming rule "if A is in $\Sigma_i^{1,b}$ then $(\forall x \leq t)A$ is in $\Sigma_i^{1,b}$" is weakened to "if A is in $\Sigma_i^{w1,b}$ then $(\forall x \leq |t|)A$ is in $\Sigma_i^{w1,b}$", and similarly for the dual case. In plain words, we allow sharply bounded first order quantifiers for free, whereas all other first order quantifiers are counted exactly as second order quantifiers.

We define the split versions $\mathcal{S}\Sigma_i^{w1,b}$ similarly to $\mathcal{S}\Sigma_i^b$. That is, $\mathcal{S}\Sigma_0^{w1,b} \rightleftharpoons \mathcal{S}\Pi_0^{w1,b} \rightleftharpoons \left(\Sigma^{1,b}\right)^\alpha \cup \left(\Sigma^{1,b}\right)^\beta$, and the inductive definition of $\mathcal{S}\Sigma_{i+1}^{w1,b}, \mathcal{S}\Pi_{i+1}^{w1,b}$ is the same as for $\Sigma_{i+1}^{w1,b}, \Pi_{i+1}^{w1,b}$ (the case $(\exists \eta)A$ gets split into two, depending on the sort of the second order bound variable η).

Definition 7.1. For a class Φ of bounded formulae in $\mathcal{L}_2$, we denote by $\Phi - SIM$ the following principle:

$$\bigwedge_{i=1}^{r} (\forall x(\alpha_i(x) \equiv \beta_i(x))) \supset \left(A^\alpha(\alpha_1, \ldots, \alpha_r) \equiv A^\beta(\beta_1, \ldots, \beta_r)\right),$$

where $A(\gamma_1, \ldots, \gamma_r)$ is in Φ.

Let Cl_2 be the class of bounded formulae without free second order variables. Note that $Cl_2 - SIM$ is simply $A^\alpha \equiv A^\beta$, where $A \in Cl_2$. This principle states that isomorphic internal computations run by the two parties (whatever complex) lead to the same result.

Our base theory, $\mathcal{S}(V_2)$ in the language $\mathcal{S}(\mathcal{L}_2)$ is, by definition, axiomatized by $(V_2)^\alpha + (V_2)^\beta + Cl_2 - SIM$.

For a class Φ of formulae in the language $\mathcal{L}_2$ we denote by Φ^+ the closure of Φ under the operation of substituting Cl_2-abstracts for second order variables.

Lemma 7.2. $\mathcal{S}(V_2) \vdash \left(\Sigma_0^{1,b}\right)^+ - SIM.$

Proof. Let $A(\gamma_1, \ldots, \gamma_r, V_1, \ldots, V_s) \in \left(\Sigma_0^{1,b}\right)^+$, where $A(\gamma_1, \ldots, \gamma_r, \gamma_{r+1}, \ldots, \gamma_{r+s})$ is in $\Sigma_0^{1,b}$, and $V_1, \ldots, V_s$ are Cl_2-abstracts. In order to show $A(\gamma_1, \ldots, \gamma_r, V_1, \ldots, V_s) - SIM$, we apply an obvious induction on the logical complexity of A; $Cl_2 - SIM$ takes care of the base case $A \equiv \gamma_i(t);\ r+1 \leq i \leq r+s$.■

Lemma 7.3. $\mathcal{S}(V_2) + \mathcal{S}\Sigma_1^{w1,b} - PIND \vdash \left(\Delta_1^{1,b}(U_2^1)\right)^+ - SIM$, *where* $\Delta_1^{1,b}(U_2^1)$ *is the set of formulae which are* $\Delta_1^{1,b}$ *with respect to* U_2^1.

Proof. It is an immediate corollary of the main result in [19] that every $A(\vec{a}, \vec{\gamma})$ in $\Delta_1^{1,b}(U_2^1)$ is equivalent to the result of evaluating a $\Sigma_0^{1,b}$-definable circuit $\delta(\vec{a}, \vec{\gamma})$ of depth $|\vec{a}|^{O(1)}$. Thus, we only have to show in $\mathcal{S}(V_2) + \mathcal{S}\Sigma_1^{w1,b} - PIND$ that $\bigwedge_i \forall x(\alpha_i(x) \equiv \beta_i(x)) \supset \left(\delta^\alpha(\vec{a}, \vec{\alpha}, \vec{V}) \equiv \delta^\beta(\vec{a}, \vec{\beta}, \vec{V})\right)$ for any circuit δ of this kind and any abstracts $\vec{V}$ in Cl_2. This is done by $\mathcal{S}\Pi_1^{w1,b} - PIND$ on d applied to the formula "every node of δ at the dth level outputs the same value in $\delta^\alpha(\vec{a}, \vec{\alpha}, \vec{V})$ and in $\delta^\beta(\vec{a}, \vec{\beta}, \vec{V})$".■

The following is proved in exactly the same way.

Lemma 7.4. $\mathcal{S}(V_2) + \mathcal{S}\Sigma_1^{w1,b} - IND \vdash \left(\Delta_1^{1,b}(V_2^1)\right)^+ - SIM.$

Now we are in position to formulate and prove interpolation-like theorems generalizing the results of the previous section.

Theorem 7.5. *Let* $A(\vec{\gamma}), B(\vec{\gamma}'), C(a, \vec{\gamma}), D(a, \vec{\gamma}')$ *be* $\Sigma^{1,b}$*-formulae, where all occurences of* a *and of all free second order variables are explicitly displayed. Then* $\mathcal{S}(V_2)$ *proves the formula*

$$\forall\vec{\phi}\forall\vec{\psi}\left((A^\alpha(\vec{\phi}) \wedge B^\beta(\vec{\psi})) \supset \exists x(C^\alpha(x, \vec{\phi}) \not\equiv D^\beta(x, \vec{\psi}))\right) \tag{7}$$

if and only if there exists $E(\gamma) \in \left(\Sigma_0^{1,b}\right)^+$ *such that*

$$V_2 \vdash \forall\vec{\phi}(A(\vec{\phi}) \supset E(\{x\}C(x, \vec{\phi}))) \tag{8}$$

and

$$V_2 \vdash \forall\vec{\psi}(B(\vec{\psi}) \supset \neg E(\{x\}D(x, \vec{\psi}))). \tag{9}$$

Theorem 7.6. *Let A, B, C, D have the same meaning as in Theorem 7.5. Then the formula (7) is provable in $\mathcal{S}(V_2) + \mathcal{S}\Sigma_1^{w1,b} - PIND$ if and only if there exists $E(\gamma) \in \left(\Delta_1^{1,b}(U_2^1)\right)^+$ with the properties (8), (9).*

Theorem 7.7. *For the same A, B, C, D, $\mathcal{S}(V_2) + \mathcal{S}\Sigma_2^{w1,b} - PIND$ proves (7) if and only if there exists $E(\gamma) \in \left(\Delta_1^{1,b}(V_2^1)\right)^+$ satisfying (8), (9).*

These theorems, combined with the material from Section 2.4, indeed generalize the results of the previous section if we notice that $E(\gamma)$ with properties (8), (9) encodes a circuit from the class needed in each of the three cases separating functions $\{\{x\}C(x,\vec{\alpha}) \mid A(\alpha)\}$ from functions $\left\{\{x\}D(x,\vec{\beta}) \mid B(\beta)\right\}$. The output of this circuit corresponds to E_n in the proof of Theorem 6.1, and the Cl_2-abstracts provide non-uniformity.

The proofs of Theorems 7.5, 7.6, 7.7 in the easy direction are based on Lemmas 7.2, 7.3, 7.4, respectively. Namely, assume that we have (8), (9) for some $E(\gamma)$ from the class Φ prescribed in each of the three cases. We lift these proofs to $(V_2)^\alpha$ and $(V_2)^\beta$, and find that $\mathcal{S}(V_2) \vdash \forall\vec{\phi}\forall\vec{\psi}\left((A^\alpha(\vec{\phi}) \wedge B^\beta(\vec{\psi})) \supset (E(\{x\}C^\alpha(x,\vec{\phi})) \not\equiv E(\{x\}D^\beta(x,\vec{\psi})))\right)$. Now we only have to apply $\Phi - SIM$ to the formula $E(\gamma)$.

The proofs in another direction can be viewed as formalized analogues of Propositions 2.2, 2.1 and Theorem 3.1. In the rest of this section we briefly outline those aspects of this formalization which may appear less obvious.

Firstly, we, similarly to [18], treat V_2 simply as a two-sorted *first order* theory. This allows us to define a language $\widehat{\mathcal{L}_2}$ and the skolemization $\widehat{V_2}$ of V_2 in this language similarly to $\widehat{L_2(\gamma)}$, $\widehat{S_2(\gamma)}$. Namely, behind function symbols $f_{A,t}$ already known to us from the previous section, we introduce function symbols $\theta_A(\vec{b},\vec{\gamma})$ and $\pi_B(\vec{b},\vec{\gamma})$ taking values in the sort for second order variables with the intended meaning $\theta_A(\vec{b},\vec{\gamma}) \equiv \mu\phi A(\phi,\vec{b},\vec{\gamma})$ and $\pi_B(\vec{b},\vec{\gamma}) \equiv \{x\}B(x,\vec{b},\vec{\gamma})$. Here $A(\gamma_0,\vec{b},\vec{\gamma}), B(a,\vec{b},\vec{\gamma})$ are in $Open\left(\widehat{\mathcal{L}_2}\right)$, and the operator μ corresponds to the ordering of second order objects γ given by $\gamma \mapsto \sum 2^{-n}\gamma(n)$. The definition of θ_A makes sense in $\widehat{V_2}$ since there always exists a term $t_A(\vec{b})$ such that $\widehat{V_2} \vdash \forall x \leq t_A(\vec{b})(\gamma_0(x) \equiv \gamma_0'(x)) \supset (A(\gamma_0,\vec{b},\vec{\gamma}) \equiv A(\gamma_0',\vec{b},\vec{\gamma}))$. We omit the exact details.

Then we define $\widehat{\mathcal{S}}(\mathcal{L}_2)$ and $\widehat{\mathcal{S}}(V_2)$ analogously to $\widehat{\mathcal{S}}(L_2)$ and $\widehat{\mathcal{S}}(S_2)$. We will be denoting terms of $\widehat{\mathcal{S}}(\mathcal{L}_2)$ taking values in the second order variables of the first sort by $\mathcal{A}_1, \mathcal{A}_2, \ldots$, and terms taking values in the second order variables of the second sort by $\mathcal{B}_1, \mathcal{B}_2, \ldots$

Now, suppose $\mathcal{S}(V_2)$ proves (7). Then $\widehat{\mathcal{S}}(V_2)$ also proves this formula. Applying Herbrand's theorem (for the three-sorted case) as in the proof of Theorem 6.1, we find witnesses

$s_1(\vec{\alpha},\vec{\beta}),\ldots,s_r(\vec{\alpha},\vec{\beta})$ to this fact, and it is easy to see that actually they can be combined into one term $s(\vec{\alpha},\vec{\beta})$ such that

$$\widehat{\mathcal{S}}(V_2) \vdash (A^\alpha(\vec{\alpha}) \wedge B^\beta(\vec{\beta})) \supset (C^\alpha(s(\vec{\alpha},\vec{\beta}),\vec{\alpha}) \not\equiv D^\beta(s(\vec{\alpha},\vec{\beta}),\vec{\beta})). \tag{10}$$

Next, we make an easy observation that the term $s(\vec{\alpha},\vec{\beta})$ can be represented in an equivalent form $s'(\mathcal{A}(\vec{\alpha}),\mathcal{B}(\vec{\beta}))$, where all occurences of second order variables are explicitly displayed, and $s'(\alpha,\beta)$ is a term of $\widehat{\mathcal{S}}(L_2)$.

In order to find $E(\gamma) \in \left(\Sigma_0^{1,b}\right)^+$ with the required properties (8), (9), we apply induction on the logical complexity of s'.

Base case $s' \equiv a$**.** We have $\mathcal{S}(V_2) \vdash (A^\alpha(a,\vec{\alpha}) \wedge B^\beta(a,\vec{\beta})) \supset (C^\alpha(a,\vec{\alpha}) \not\equiv D^\beta(a,\vec{\beta}))$. Applying the sort-erasing interpretation, we find

$$V_2 \vdash (A(a,\vec{\alpha}) \wedge B(a,\vec{\beta})) \supset (C(a,\vec{\alpha}) \not\equiv D(a,\vec{\beta})).$$

The formula $E(a,\gamma)$ defined by

$$E(a,\gamma) \rightleftharpoons \begin{cases} \gamma(a) \equiv \exists\vec{\phi}(A(a,\vec{\phi}) \wedge C(a,\vec{\phi})) \text{ if } \exists\vec{\phi}A(a,\vec{\phi}) \wedge \exists\vec{\psi}B(a,\vec{\psi}) \\ \top \text{ if } \exists\vec{\phi}A(a,\vec{\phi}) \wedge \forall\vec{\psi}\neg B(a,\vec{\psi}) \\ \bot \text{ if } \forall\vec{\phi}\neg A(a,\vec{\phi}) \wedge \exists\vec{\psi}B(a,\vec{\psi}) \\ \text{ arbitrary if } \forall\vec{\phi}\neg A(a,\vec{\phi}) \wedge \forall\vec{\psi}\neg B(a,\vec{\psi}) \end{cases}$$

has the required properties. Note that the case analysis in the definition of $E(a,\gamma)$ is exactly the place where we use the power of our base theory not available in the first order setting.

Inductive step. $s'(\alpha,\beta) \equiv s''(f^\alpha(\alpha),\alpha,\beta)$, where $f(\gamma)$ is a function symbol of $\widehat{L_2(\gamma)}$, and we are guaranteed the existence of E with the desired properties anytime when (10) is true for the term $s''(a,\mathcal{A}(\vec{\alpha}),\mathcal{B}(\vec{\beta}))$ and *any* choice of A,B,C,D.

(10) implies

$$\begin{aligned} \widehat{\mathcal{S}}(V_2) \;\vdash\; & (A^\alpha(\vec{\alpha}) \wedge f^\alpha(\mathcal{A}(\vec{\alpha})) = a \wedge B^\beta(\vec{\beta})) \supset \\ & (C^\alpha(s''(a,\mathcal{A}(\vec{\alpha}),\mathcal{B}(\vec{\beta})),\vec{\alpha}) \not\equiv D^\beta(s''(a,\mathcal{A}(\vec{\alpha}),\mathcal{B}(\vec{\beta})),\vec{\beta})), \end{aligned}$$

and we can use our inductive assumption (with $A(a,\vec{\alpha}) := A(\vec{\alpha}) \wedge f(\mathcal{A}(\vec{\alpha})) = a$) to find $E'(a,\gamma) \in \left(\Sigma_0^{1,b}\right)^+$ such that

$$V_2 \vdash (A(\vec{\alpha}) \wedge f(\mathcal{A}(\vec{\alpha})) = a) \supset E'(a,\{x\}C(x,\vec{\alpha}))$$

and

$$V_2 \vdash B(\vec{\beta}) \supset \neg E'(a, \{x\}D(x, \vec{\beta})).$$

We simply set $E(\gamma) \rightleftharpoons \exists x \leq t\ E'(x, \gamma)$, where t is a term such that $V_2 \vdash f(\alpha) \leq t$. This completes the inductive step and the proof of Theorem 7.5.

Coming to Theorem 7.6, we notice that in the theory $\widehat{\mathcal{S}}(V_2) + \mathcal{S}\Sigma_1^b - PIND$ every $\mathcal{S}\Sigma_1^{w1,b}$-formula is equivalent to a $\Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right)$-formula of the form $\exists x \leq t\left(A^\alpha(x) \wedge B^\beta(x)\right)$, where $A(a), B(a) \in \Sigma^{1,b}$. Indeed, the class of such formulae is closed under applying second order quantifiers:

$$\exists\phi\exists x \leq t\left(A^\alpha(\phi, x) \wedge B^\beta(x)\right) \equiv \exists x \leq t\left(\exists\phi A^\alpha(\phi, x) \wedge B^\beta(x)\right),$$

and (in the presence of $\Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) - PIND$) under applying sharply bounded universal quantifiers[1]:

$$\begin{aligned}
&\forall y \leq |s|\exists x \leq t\left(A^\alpha(x, y) \wedge B^\beta(x, y)\right) \equiv \\
&\quad \exists w\Big(Seq(w) \wedge Len(w) \leq |s| + 1 \wedge Size(w) \leq t \wedge \forall y \leq |s|\ A^\alpha((w)_{y+1}, y) \wedge \\
&\qquad \forall y \leq |s|\ B^\beta((w)_{y+1}, y)\Big).
\end{aligned}$$

Thus, $\mathcal{S}(V_2) + \mathcal{S}\Sigma_1^{w1,b} - PIND$ is equivalent to $\widehat{\mathcal{S}}(V_2) + \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) - PIND$. But it is straightforward to establish for $\widehat{\mathcal{S}}(V_2) + \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) - PIND$ the cut elimination theorem and extend to it the syntactic version of Proposition 4.2; in fact, this theory more resembles the first order theory S_R^1 for what might be called "a many-sorted strongly regular theory R, where no quantifiers other than those on first order variables are allowed" than a second order theory. We skip the details.

The proof of Theorem 7.6 is completed by formalizing the standard proof of Proposition 2.1 in the same fashion as we did above with the proof of Proposition 2.2. We omit exact and somewhat tedious details.

The same ideas work for the weaker version of Theorem 7.7 in which $\mathcal{S}\Sigma_2^{w1,b} - PIND$ is replaced by $\mathcal{S}\Sigma_1^{w1,b} - IND$: extending the syntactic variant of Proposition 4.4 to this case and formalizing the proof of Theorem 3.1 is more or less straightforward.

The analogue of Proposition 4.3 is, however, much less straightforward since we in general can not eliminate second order quantifiers from $\mathcal{S}\Sigma_2^{w1,b}$-formulae. We circumvent this as follows.

[1]to avoid collision with another usage of β, we denote the ith member of a sequence w by $(w)_i$ rather than by $\beta(i, w)$

For $A(\vec{a},\vec{\alpha},\vec{\beta}) \in \mathcal{S}\Sigma_2^{w1,b}$ we introduce a *family* $\mathcal{W}_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}$ of witnessing formulae

$$Witness_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \in \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) \cup \Pi_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right)$$

rather than a single formula. All old cases in the standard definition of $Witness$ (see [7, Section 4]) are modified in an obvious way, e.g. we say "if A is $B \wedge C$ then $\mathcal{W}_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}$ consists of all formulae of the form $Witness_B^{2,\vec{a},\vec{\alpha},\vec{\beta}}((w)_1,\vec{a},\vec{\alpha},\vec{\beta}) \wedge Witness_C^{2,\vec{a},\vec{\alpha},\vec{\beta}}((w)_2,\vec{a},\vec{\alpha},\vec{\beta})$, where $Witness_B^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \in \mathcal{W}_B^{2,\vec{a},\vec{\alpha},\vec{\beta}}$, and $Witness_C^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \in \mathcal{W}_C^{2,\vec{a},\vec{\alpha},\vec{\beta}}$,".

The only case when the branching really occurs is the following new case:

(8) *If $A \notin \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) \cup \Pi_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right)$ and A is $\exists\phi\ B(\vec{a},\vec{\alpha},\phi,\vec{\beta})$ then $\mathcal{W}_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}$ consists of all formulae $Witness_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta})$ of the form*

$$Seq(w) \wedge Len(w) = 2 \wedge (w)_1 \leq t(\vec{a}) \wedge Witness_{B(\vec{a},\vec{\alpha},\alpha_0,\vec{\beta})}^{2,\vec{a},\vec{\alpha},\alpha_0,\vec{\beta}}((w)_2,\vec{a},\vec{\alpha},\mathcal{A}(\vec{a},(w)_1,\vec{\alpha},\vec{\beta}),\vec{\beta}),$$

where $t(\vec{a})$ and $\mathcal{A}(\vec{a},w,\vec{\alpha},\vec{\beta})$ run over all terms of the language $\widehat{\mathcal{S}}(\mathcal{L}_2)$, and

$$Witness_{B(\vec{a},\vec{\alpha},\alpha_0,\vec{\beta})}^{2,\vec{a},\vec{\alpha},\alpha_0,\vec{\beta}}(w,\vec{a},\vec{\alpha},\alpha_0,\vec{\beta}) \in \mathcal{W}_B^{2,\vec{a},\vec{\alpha},\alpha_0,\vec{\beta}}.$$

The case $A \equiv \exists\psi B(\psi)$ is treated in the same way.

For every $Witness_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \in \mathcal{W}_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}$,

$$\widehat{\mathcal{S}}(V_2) \vdash \exists w Witness_A^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \supset A(\vec{a},\vec{\alpha},\vec{\beta}).$$

Due to the very limited nature of witnessing second order variables, we can not hope to reverse this implication in any reasonable sense. But we actually do not need this. We simply show the straightforward analogue of [7, Theorem 17] in the following form:

if

$$\widehat{\mathcal{S}}(V_2) + \mathcal{S}\Sigma_2^{w1,b} - PIND \vdash G(\vec{a},\vec{\alpha},\vec{\beta}) \supset H(\vec{a},\vec{\alpha},\vec{\beta}),$$

where G, H are in $\mathcal{S}\Sigma_2^{w1,b}$ then for every $Witness_G^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \in \mathcal{W}_G^{2,\vec{a},\vec{\alpha},\vec{\beta}}$ there exist $Witness_H^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \in \mathcal{W}_H^{2,\vec{a},\vec{\alpha},\vec{\beta}}$ and a Q_2-defined function $f(w,\vec{a},\vec{\alpha},\vec{\beta})$ of $\widehat{\mathcal{S}}(V_2) + \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) - IND$ such that

$$\widehat{\mathcal{S}}(V_2) + \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) - IND \ \vdash \ Witness_G^{2,\vec{a},\vec{\alpha},\vec{\beta}}(w,\vec{a},\vec{\alpha},\vec{\beta}) \supset Witness_H^{2,\vec{a},\vec{\alpha},\vec{\beta}}(f(w,\vec{a},\vec{\alpha},\vec{\beta}),\vec{a},\vec{\alpha},\vec{\beta}).$$

This allows us to conclude that $\widehat{\mathcal{S}}(V_2) + \mathcal{S}\Sigma_2^{w1,b} - PIND$ is $\mathcal{S}\Sigma_2^{w1,b}$-conservative over $\widehat{\mathcal{S}}(V_2) + \Sigma_1^b\left(\widehat{\mathcal{S}}(\mathcal{L}_2)\right) - IND$ and complete the proof of Theorem 7.7.

8. Conclusion

Naturally, the most interesting question is to which extent the techniques developed in this paper can advance us toward the main goal of understanding the strength of V_1^1. Let us first point out that the hierarchy of second order theories introduced in the previous section collapses already at the next level. Indeed,

$$\widehat{\mathcal{S}}(V_2) + \mathcal{S}\Sigma_2^{w1,b} - IND \vdash \forall\phi\exists\psi\forall x \leq t(\phi(x) \equiv \psi(x)) \wedge \forall\psi\exists\phi\forall x \leq t(\phi(x) \equiv \psi(x)).$$

Thus, at least with respect to bounded formulae, $\widehat{\mathcal{S}}(V_2)+\mathcal{S}\Sigma_2^{w1,b}-IND$ is simply equivalent to V_2. So, we restrict our discussion to first order theories.

What we actually did in the proof of Theorem 6.4 (this is also a direct corollary of Theorem 7.7) was to show the following *separation* theorem. Whenever

$$S_R^2 \vdash (A(N,\alpha) \wedge B(N,\beta)) \supset \exists x(C(N,x,\alpha) \not\equiv D(N,x,\beta)), \tag{11}$$

where $R = \mathcal{S}(S_2)$, the sets $\{\{x\}C(N,x,\alpha) \mid A(N,\alpha)\}$ and $\{\{x\}D(N,x,\beta) \mid B(N,\beta)\}$ can be separated by a size-$2^{(\log N)^{O(1)}}$ circuit. An informal reformulation of this is that every two NP-sets which are provably disjoint in S_R^2 can actually be separated by a set computable in quasipolynomial time. Is it possible to improve this by replacing S_R^2 in (11) with a stronger theory like T_R^2, $S_2(\alpha,\beta)$, U_2^1 or V_2^1? This seems to be open even under any reasonable complexity assumption. Note for the comparison that even for the case of V_2^1, the affirmative answer to a similar question in which we are interested in separating $co-NP$ sets is a straightforward corollary of Proposition 4.2 and RSUV-isomorphism [23, 24, 18].

There are several examples showing that for NP-sets the situation may be different. A couple of them originated from a discussion with Steven Rudich are based upon the lower bound proof for voting polynomials [3] and one-way functions, respectively. In these examples, however, in order to prove the formula (11) one apparently needs at least the strength of U_1^1. Also, their impact on the future research in this direction is still to be understood. Thus, we confine ourselves here with a simpler combinatorial example which gives a new unexpected proof of a known result from [9] and raises several immediate open questions.

Example 1. The proof of the separation theorem works for the monotone case as well. That is to say, if

$$S_R^2 \vdash (A(N,\alpha) \wedge B(N,\beta)) \supset \exists x(C(N,x,\alpha) \wedge \neg D(N,x,\beta))$$

then there exists a *monotone* size-$2^{(\log N)^{O(1)}}$ circuit outputting 1 on all $\{x\}C(N,x,\alpha)$ with $A(N,\alpha)$, and outputting 0 on all $\{x\}D(N,x,\beta)$ with $B(N,\beta)$. We will show that this is no longer the case if we replace S^2_R with T^3_R.

Indeed, denote by $WPHP(f)$ the *weak pigeon hole principle* taken in the following form:

$$\Big(a \geq 2 \wedge \forall x \leq a^2(f(x) < a)\Big) \supset \exists x_1, x_2 < a^2(f(x_1) = f(x_2) \wedge x_1 \neq x_2).$$

Note that, contrary to the common belief, it is open whether $T^2_2(f) \vdash WPHP(f)$. But the proof in [16] lets us conclude at least that $T^3_2(f) \vdash WPHP(f)$, and this (naturally) extends to showing that $T^3_R(f) \vdash WPHP(f)$ for every Σ^b_1-definable f.

Now, let $A(N, f_\alpha)$ say "f_α is an injective mapping from $[N^2]$ to $[N^4]$". Let $B(N, f_\beta)$ say "f_β is a mapping from $[N^4]$ to $[N]$". Then, applying $WPHP(f_\beta \circ f_\alpha)$ (available in T^3_R), we see that

$$T^3_R \vdash (N \geq 2 \wedge A(N, f_\alpha) \wedge B(N, f_\beta)) \supset \exists x_1 < x_2 < N^4 \, (x_1, x_2 \in \mathrm{im}(f_\alpha) \wedge f_\beta(x_1) = f_\beta(x_2)) \, .$$

But $\{x_1, x_2\}\,(x_1 < x_2 < N^4 \wedge x_1, x_2 \in \mathrm{im}(f_\alpha))$ taken over all possible injective $f_\alpha : [N^2] \longrightarrow [N^4]$ is simply the set of all N^2-cliques. $\{x_1, x_2\}\,(x_1 < x_2 < N^4 \wedge f_\beta(x_1) = f_\beta(x_2))$ is the set of all N-partite complete subgraphs. These two sets can not be separated by a subexponential size *monotone* circuit [2].

This example suggests several open questions. Is it true that $T^2_2(f) \vdash WPHP(f)$? Is it true that $T^2_R(f) \vdash WPHP(f_\beta \circ f_\alpha)$? Is the monotone version of the separation theorem true for T^2_R?

In connection with the last question the following observation made by J. Krajíček may turn out useful. Let the weaker principle $WPHP_1(f, g)$ state that f and g do not form two inverse *bijections* between $[a^2]$ and $[a]$, for $a \geq 2$. Then this principle *is* already provable in $T^2_2(f)$.

In general, we lack a decent characterization of Σ^b_1-theorems of T^2_2. In particular, it is still open whether $S_2(\alpha)$ is $\Sigma^b_1(\alpha)$-conservative over $T^2_2(\alpha)$ or not. Obtaining such a characterization and understanding its meaning in the context of split versions seems to be the most immediate accessible question. The first part of this question is undoubtedly interesting in its own right, irrespectively of the application to particular problems from Boolean complexity.

It is also worth noting that the reasoning in Example 1 can be reversed: since we have the monotone separation theorem for S^2_R, we also have the independence result $S^2_R \nvdash WPHP(f_\beta \circ f_\alpha)$. This implies the result from [9] that $S^2_2(f) \nvdash WPHP(f)$.

In the formal sense, Example 1 can not be used for refuting the separation theorem for nonmonotone circuits. Indeed, É. Tardos [25] noticed that the classes of graphs G with $\omega(G) \geq s$ and of graphs G with $\chi(G) < s$ *can* be separated by (non-monotone) polynomial size circuits. Still, her proof involves highly nontrivial combinatorial argument known as Lovasz lower bound for Shannon capacity, and it hardly can be expected that this argument would follow from a separation theorem in Bounded Arithmetic.

9. Added in proof

After this paper was submitted, the author has found a purely complexity characterization of pairs of NP-sets which are provably disjoint in certain fragments of Bounded Arithmetic, including $\mathcal{S}(S_2) + \mathcal{S}\Sigma_i^b - IND$, U_2^1 and V_2^1 [20]. This characterization, in particular, gives raise to another and, perhaps, more natural, proof of Theorem 6.4.

10. Acknowledgement

I am indebted to Sam Bass, Steven Cook, Mauricio Karchmer, Jan Krajíček, Steven Rudich, Avi Wigderson, and Andy Yao for their useful remarks concerning various aspects of this patchwork paper.

References

[1] M. Ajtai. Σ_1^1-formulae on finite structures. *Annals of Pure and Applied Logic*, 24:1–48, May 1983.

[2] N. Alon and R. Boppana. The monotone circuit complexity of Boolean functions. *Combinatorica*, 7(1):1–22, 1987.

[3] J. Aspnes, R. Beigel, M. Furst, and S. Rudich. The expressive power of voting polynomials. In *Proceedings of the* 23*rd ACM STOC*, pages 402–409, 1991. Journal version to appear in *Combinatorica*.

[4] D. A. Barrington. A note on a theorem of Razborov. Technical report, University of Massachusetts, 1986.

[5] R. B. Boppana and M. Sipser. The complexity of finite functions. In Jan van Leeuwen, editor, *Handbook of Theoretical Computer Science, vol. A* (*Algorithms and Complexity*), chapter 14, pages 757–804. Elsevier Science Publishers B.V. and The MIT Press, 1990.

[6] S. R. Buss. *Bounded Arithmetic*. Bibliopolis, Napoli, 1986.

[7] S. R. Buss. Axiomatizations and conservations results for fragments of Bounded Arithmetic. In *Logic and Computation, Contemporary Mathematics* 106, pages 57–84. American Math. Society, 1990.

[8] S. R. Buss and J. Krajíček. An application of Boolean complexity to separation problems in Bounded Arithmetic. To appear in *Proceedings of the London Mathematical Society*, 1992.

[9] M. Chiari and J. Krajíček. Witnessing functions in Bounded Arithmetic and search problems. Manuscript in preparation, 1994.

[10] S. Cook. Feasibly constructive proofs and the propositional calculus. In *Proceedings of the 7th Annual ACM Symposium on the Theory of Computing*, pages 83–97, 1975.

[11] M. Furst, J. B. Saxe, and M. Sipser. Parity, circuits and the polynomial time hierarchy. *Math. Syst. Theory*, 17:13–27, 1984.

[12] J. Håstad. *Computational limitations on Small Depth Circuits*. PhD thesis, Massachusetts Institute of Technology, 1986.

[13] D. S. Johnson, C. H. Papadimitriou, and M. Yannakakis. How easy is local search? *Journal of Computer and System Sciences*, 37:79–100, 1988.

[14] M. Karchmer. *Communication complexity: A new approach to circuit depth*. PhD thesis, Massachusetts Institute of Technology, 1989.

[15] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require superlogarithmic depth. *SIAM J. on Disc. Math.*, 3(2):255–265, May 1990.

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

[17] R. Raz and A. Wigderson. Monotone circuits for matching require linear depth. In *Proceedings of the 22th Ann. ACM Symposium on the Theory of Computing*, pages 287–292, 1990.

[18] A. Razborov. An equivalence between second order bounded domain bounded arithmetic and first order bounded arithmetic. In P. Clote and J. Krajíček, editors, *Arithmetic, Proof Theory and Computational Complexity*, pages 247–277. Oxford University Press, 1992.

[19] A. Razborov. Bounded Arithmetic and lower bounds in Boolean complexity. Submitted to the volume *Feasible Mathematics II*, 1993.

[20] A. Razborov. On provably disjoint **NP**-pairs. Technical Report RS-94-36, Basic Research in Computer Science Center, Aarhus, Denmark, 1994.

[21] A. Razborov and S. Rudich. Natural proofs. To appear in the 26th ACM STOC, 1994.

[22] R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *Proceedings of the 19th ACM Symposium on Theory of Computing*, pages 77–82, 1987.

[23] G. Takeuti. S_3^i and $\overset{\circ}{V}{}_2^i(BD)$. *Archive for Math. Logic*, 29:149–169, 1990.

[24] G. Takeuti. *RSUV* isomorphisms. In P. Clote and J. Krajíček, editors, *Arithmetic, Proof Theory and Computational Complexity*, pages 364–386. Oxford University Press, 1992.

[25] É. Tardos. The gap between monotone and nonmonotone circuit complexity is exponential. *Combinatorica*, 8:141–142, 1988.

[26] G. Wilmers. Bounded existential induction. *The Journal of Symbolic Logic*, 50(1):72–90, March 1985.

[27] A. Yao. Separating the polynomial-time hierarchy by oracles. In *Proceedings of the 26th IEEE FOCS*, pages 1–10, 1985.

[28] А.Е. Андреев. Об одном методе получения нижних оценок сложности индивидуальных монотонных функций. *ДАН СССР*, 282(5):1033–1037, 1985. A.E. Andreev, On a method for obtaining lower bounds for the complexity of individual monotone functions. *Soviet Math. Dokl.* 31(3):530-534, 1985.

[29] А.Е. Андреев. Об одном методе получения эффективных нижних оценок монотонной сложности. *Алгебра и логика*, 26(1):3–21, 1987. A.E. Andreev, On one method of obtaining effective lower bounds of monotone complexity. *Algebra i logika*, 26(1):3-21, 1987. In Russian.

[30] А. А. Марков. О минимальных контактно-вентильных двухполюсниках для монотонных симметрических функций. In *Проблемы кибернетики*, volume 8, pages 117–121. Наука, 1962. A. A. Markov, On minimal switching-and-rectifier networks for monotone symmetric functions, *Problems of Cybernetics*, vol. 8, 117-121 (1962).

[31] Э. И. Нечипорук. Об одной булевской функции. *ДАН СССР*, 169(4):765–766, 1966. E. I. Nečiporuk, On a Boolean function, *Soviet Mathematics Doklady* 7:4, pages 999-1000.

[32] А. А. Разборов. Нижние оценки монотонной сложности некоторых булевых функций. *ДАН СССР*, 281(4):798–801, 1985. A. A. Razborov, Lower bounds for the monotone complexity of some Boolean functions, *Soviet Math. Dokl.*, 31:354-357, 1985.

[33] А. А. Разборов. Нижние оценки монотонной сложности логического перманента. *Матем. Зам.*, 37(6):887–900, 1985. A. A. Razborov, Lower bounds of monotone complexity of the logical permanent function, *Mathem. Notes of the Academy of Sci. of the USSR*, 37:485-493, 1985.

[34] А. А. Разборов. Нижние оценки размера схем ограниченной глубины в полном базисе, содержащем функцию логического сложения. *Матем. Зам.*, 41(4):598–607, 1987. A. A. Razborov, Lower bounds on the size of bounded-depth networks over a complete basis with logical addition, *Mathem. Notes of the Academy of Sci. of the USSR*, 41(4):333-338, 1987.

[35] Б. А. Субботовская. О реализации линейных функций формулами в базисе $\&, \vee, -$. *ДАН СССР*, 136(3):553–555, 1961. B.A.Subbotovskaya, Realizations of linear functions by formulas using $+, *, -$, *Soviet Mathematics Doklady* 2(1961), 110-112.

[36] В. М. Храпченко. О сложности реализации линейной функции в классе П-схем. *Матем. заметки*, 9(1):35–40, 1971. V.M. Khrapchenko, Complexity of the

realization of a linear function in the class of π-circuits, *Math. Notes Acad. Sciences USSR* 9(1971), 21-23.

[37] В. М. Храпченко. Об одном методе получения нижних оценок сложности П-схем. *Матем. заметки*, 10(1):83–92, 1971. V.M. Khrapchenko, A method of determining lower bounds for the complexity of Π-schemes, *Math. Notes Acad. Sciences USSR* 10(1971), 474-479.