New substitution bases for complexity classes

Stefano Mazzanti

Dipartimento di Culture del Progetto

Università Iuav di Venezia

Fondamenta delle Terese 2206, 30123 Venezia, Italy

email: mazzanti@iuav.it

Abstract

The set $AC^0(F)$, the AC^0 closure of F, is the closure with respect to substitution and concatenation recursion on notation of a set of basic functions comprehending the set F. By improving earlier work, we show that $AC^0(F)$ is the substitution closure of a simple function set and characterize well-known function complexity classes as the substitution closure of finite sets of simple functions.

1 Introduction

A finite function set F is a *substitution basis* for a function class G (and G is the *substitution closure* of F) when G can be defined using only the functions in F, the projection functions and the substitution operator. Several function classes like partial recursive functions, Grzegorczyk classes $\mathcal{E}_n$ for $n \geq 2$ and polynomial time computable functions have a substitution basis, see [6] for a list of references. But such bases may contain awkward functions.

A nice example of basis for a non-trivial function class was given in [7, 8] where the set $\{x+y, x \dot{-} y, x \wedge y, \lfloor x/y \rfloor, 2^{|x|^2}\}$ was shown to be a basis for the class TC^0 of functions computable by polysize, constant depth threshold circuits.[1]

Subsequently, the existence of plain bases was considered for the set $AC^0(F)$, the closure with respect to substitution and concatenation recursion on notation (CRN) of a set of basic functions comprehending the set F.[2] In [5], it was shown that $AC^0(F)$ admits a basis, provided that it contains integer division. From this result, the above mentioned basis for TC^0 was obtained. Later, the existence of a basis for $AC^0(F)$ was stated without assuming any hypothesis and a basis for AC^0 was introduced [6].

However, the basis for TC^0 depends on the fact that integer division is in TC^0, which is a hard result to show [4], and the basis for AC^0 contains some

[1] $x \wedge y$ is the *bitwise and* of x and y. Names AC^0, TC^0, NC^1 are usually intended to denote language classes. However, in this paper they will always denote function classes, since no misunderstanding is possible.

[2] $AC^0(F)$ is an obvious extension of Clote's characterization of AC^0 functions ([2]) obtained by adding the functions in F to the set of basic functions. For example, in [3] the set of TC^0 functions has been defined as $AC^0(mult)$ where $mult$ is the multiplication operation.

non-standard arithmetical functions which handle their arguments as sequence encodings.

This paper tries to eliminate these drawbacks and improves the results of [5] and [6]. New bases for AC^0, TC^0 and other complexity classes are obtained in a new, uniform and division-independent way by exploiting elementary properties of geometric series.

In the Preliminaries, the basic definitions and the main results of [5] and [6] are recalled.

Section 3 introduces a basis for $AC^0(F)$ that depends on a function parameter.

Then, by setting such function in two different ways, in Section 4 we obtain a new basis for $AC^0(F)$ which yields immediately a new basis for AC^0, and in Section 5 we obtain two new bases for TC^0.

Finally, following [6], we derive new bases for NC^1, L, P and $PSPACE$ computable functions.

Even if the results of this paper may seem just aesthetic improvements, they shed some light on the difference between AC^0 and TC^0 and could possibly lead to a new, algebraic proof that $AC^0 \neq TC^0$. Indeed, both AC^0 and TC^0 have a basis of six functions which differ for one function only.

2 Preliminaries

In this paper, we will only consider functions with finite arity on the set $\mathbb{N} = \{0, 1, \dots\}$ of natural numbers.

From now on, we agree that $x, y, z, u, v, w, i, j, k, l, n, m, r$ range over $\mathbb{N}$, that a, b, c range over positive integers, that $\mathbf{x}, \mathbf{y}$ range over sequences (of fixed length) of natural numbers, that p, q range over integer polynomials with non negative values and that f, g, h range over functions.

A function f is a *polynomial growth function* iff there is a polynomial p *majorizing (the length of) f*, i.e. such that $|f(\mathbf{x})| \leq p(|\mathbf{x}|)$ or, equivalently, $f(\mathbf{x}) < 2^{p(|\mathbf{x}|)}$ for any $\mathbf{x}$, where $|x_1, \dots, x_n| = |x_1|, \dots, |x_n|$ and $|x| = \lceil \log_2(x + 1) \rceil$ is the number of bits of the binary representation of x.

We will use the following unary functions: the *binary successor* functions $s_0 : x \longmapsto 2x$ and $s_1 : x \longmapsto 2x{+}1$; the *constant* functions $C_y : x \longmapsto y$; the *signum* function $sg : x \longmapsto min(x, 1)$; the *cosignum* function $cosg : x \longmapsto 1 - sg(x)$; the *quadratum* function $quad : x \longmapsto x^2$; *length* function $len : x \longmapsto |x|$; the *unary smash* function $us : x \longmapsto 2^{|x|^2}$; the *next power of two* function $pow : x \longmapsto 2^{|x|}$.

We will also use the following functions: the *addition* function $add : x, y \longmapsto x + y$; the *multiplication* function $mult : x, y \longmapsto xy$; the *modified subtraction* function $sub : x, y \longmapsto x \dot{-} y = max(x{-}y, 0)$; the *division* function $quot : x, y \longmapsto \lfloor x/y \rfloor$; the *remainder* function $rem : x, y \longmapsto x - y \lfloor x/y \rfloor$; the *conditional* function

$$cond(x, y, z) = \begin{cases} y & \text{if } x = 0 \\ z & \text{otherwise} \end{cases} ;$$

the *bit* function $bit : x, y \longmapsto rem(\lfloor x/2^y \rfloor, 2)$; the *multiplication by a power* function $multp : x, y \longmapsto x2^{|y|}$; the *concatenation* function $conc : x, y \longmapsto x * y = x2^{|y|} + y$; the *smash* function $smash : x, y \longmapsto x \# y = 2^{|x| \cdot |y|}$; the *most significant part* function $MSP : x, y \longmapsto \lfloor x/2^y \rfloor$; the *log most* *significant part* function $msp : x, y \longmapsto \lfloor x/2^{|y|} \rfloor$; the *least significant part* function $LSP : x, y \longmapsto rem(x, 2^y)$; the *log least significant part* function $lsp : x, y \longmapsto rem(x, 2^{|y|})$. A fundamental role will be played by the *bitwise and* function $and : x, y \longmapsto x \wedge y$ such that $bit(x \wedge y, i) = bit(x, i) \cdot bit(y, i)$ for any i.

For $l, n > 0$, let $\langle x_n, \ldots, x_1; l \rangle = \sum_{i<n} x_{i+1} 2^{li}$; if $x_n, \ldots, x_1 < 2^l$ then $x_n, \ldots, x_1$ are the base 2^l digits of $\langle x_n, \ldots, x_1; l \rangle$. Then, we will also use the functions $arl, ar2l, repl, convl$ such that

$$arl(l) = \sum_{i<|l|} i 2^{|l|i},$$
$$ar2l(l) = \sum_{i<|l|} i^2 2^{|l|i},$$
$$repl(x, l, n) = \sum_{i<|n|} rem(x, 2^{|l|}) 2^{|l|i}$$
$$convl(x, l, r, n) = \sum_{i<|n|} rem(x_{i+1}, 2^{|r|}) 2^{|r|i}$$

where $x_{|n|}, \ldots, x_1$ are the $|n|$ least significant base $2^{|l|}$ digits of x. All the functions above return 0 when one of l, r, n is 0.

Note that

$$repl(x, l, n) = \left\langle \overbrace{x, \ldots, x}^{|n|-times}; |l| \right\rangle$$

for $x < 2^{|l|}$ and

$$convl(\langle \mathbf{x}; |l| \rangle, l, r, n) = \langle \mathbf{x}; |r| \rangle$$

where $\mathbf{x} = x_{|n|}, \ldots, x_1$ with $x_i < 2^{min(|l|,|r|)}$ for $1 \leq i \leq n$.

As usual, the characteristic function of a predicate Q on natural numbers is the function $f(\mathbf{x})$ returning 1 if $Q(\mathbf{x})$ is true, 0 otherwise. We say that a predicate is in a class F of functions, meaning that its characteristic function is in F.

Let

$$rp(x, l, n) = \begin{cases} x \cdot \sum_{i<|n|} 2^{|l|i} & \text{if } AC^0\_SUM(x, l, n) \\ 0 & \text{otherwise} \end{cases}$$

where

$$AC^0\_SUM(x, l, n) \;\Leftrightarrow\; (ln > 0) \wedge (\bigvee_{i=1}^{3} P_i(x, l, n))$$

and P_1, P_2, and P_3 are respectively the following predicates

$$P_1(x, l, n) \;\Leftrightarrow x = \left\langle \overbrace{1, \ldots, 1}^{|l|-times}; |l| \right\rangle \wedge 1 < l,$$
$$P_2(x, l, n) \;\Leftrightarrow x = \langle |l| - 1, \ldots, 1, 0; |l| \rangle \wedge 1 < l,$$
$$P_3(x, l, n) \;\Leftrightarrow x < 2^{|l||n|} \;\wedge\; \forall_{i<j<|n|} \, (x 2^{|l|i} \wedge x 2^{|l|j} = 0).$$

As we will see in Section 4, the predicate AC^0_SUM guarantees that the function rp is in AC^0, even if $x \cdot \sum_{i<|n|} 2^{|l|i}$ is in $TC^0 - AC^0$.

Finally, we will use the following operators on functions:

- the *substitution* operator $SUBST(g_1, \ldots, g_b, h)$ transforming functions $g_1, \ldots, g_b : \mathbb{N}^a \to \mathbb{N}$ and function $h : \mathbb{N}^b \to \mathbb{N}$ into the function $f : \mathbb{N}^a \to \mathbb{N}$ such that $f(\mathbf{x}) = h(g_1\mathbf{x}, \ldots, g_b\mathbf{x})$;

- the *concatenation recursion on notation* operator $CRN(g, h_0, h_1)$ transforming functions $h_0 : \mathbb{N}^{a+1} \to \mathbb{N}$ and $h_1 : \mathbb{N}^{a+1} \to \mathbb{N}$ with values in $\{0, 1\}$ and function $g : \mathbb{N}^a \to \mathbb{N}$ into the function $f : \mathbb{N}^{a+1} \to \mathbb{N}$ such that

$$\begin{aligned} f(0, \mathbf{y}) &= g(\mathbf{y}) \,, \\ f(s_i(x), \mathbf{y}) &= s_{h_i(x,\mathbf{y})}(f(x, \mathbf{y})) \end{aligned}$$

  where in the second equation $i \in \{0, 1\}$ and $x > 0$ when $i = 0$.

For any set F of functions, let $\text{clos}_{SUBST}(F)$ be the closure under substitution of $F \cup I$ where I is the set of the projection functions

$$I^a[i] : \quad x_1, \ldots, x_a \longmapsto x_i \quad (1 \le i \le a)$$

with any arity a. In the following, we will abuse the notation above as usual and we will write $\text{clos}_{SUBST}(f_1, \ldots, f_n, G)$ when $F = \{f_1, \ldots, f_n\} \cup G$ (the sequence $f_1, \ldots, f_n$ may be empty and the set G may be omitted).

For any set F of polynomial growth functions, we define $AC^0(F)$, the AC^0 *closure of* F, as the closure under substitution and CRN of $\{C_0, s_0, s_1, smash, len, bit\} \cup F \cup I$ [3].

The class AC^0 of functions computable by polysize, constant depth, unbounded fan-in Boolean circuits, the class TC^0 of functions computable by polysize, constant depth, unbounded fan-in threshold circuits, the class NC^1 of functions computable by polysize, logarithmic depth, bounded fan-in Boolean circuits have been characterized using substitution and CRN [1, 2, 3]:

$$AC^0 = AC^0(\emptyset), \; TC^0 = AC^0(mult), \; NC^1 = AC^0(tree)$$

where $tree$ is a unary function taking values in $\{0, 1\}$ such that $tree(x)$ is the value of the and/or tree with or gate at the root represented by x when $|x| = 4^n + 1 > 1$. E.g. for $x = 10110_2$ we have $tree(x) = 0 = (0 \wedge 1) \vee (1 \wedge 0)$. For a definition of $tree$ see [1] or the Appendix of [5].

If F is a class of functions such that $F = \text{clos}_{SUBST}(f_1, \ldots, f_a)$ then $\{f_1, \ldots, f_a\}$ is a (*substitution* or *superposition*) *basis* for F and F is the *substitution closure* of $\{f_1, \ldots, f_a\}$.

For any function $f : \mathbb{N}^a \longrightarrow \mathbb{N}$, we define the function $f^{\dagger_c}$, the *canonical dagger of* f, by setting $f^{\dagger_c}(x_1, \ldots, x_a, l, n) = 0$ if $l = 0$ or $n = 0$ or $x_i \ge 2^{|l||n|}$ for some $1 \le i \le a$ and

$$\begin{aligned} & f^{\dagger_c}(x_1, \ldots, x_a, l, n) \\ & \qquad = \left\langle rem(f(x_{1,|n|}, \ldots, x_{a,|n|}), 2^{|l|}), \ldots, rem(f(x_{1,1}, \ldots, x_{a,1}), 2^{|l|}); |l| \right\rangle \end{aligned}$$

[3] See [6] for a discussion about the relationship of our definition of $AC^0(F)$ and similar definitions given in the literature.

when $l, n > 0$, $x_1 = \langle x_{1,|n|}, \ldots, x_{1,1}; |l| \rangle, \ldots, x_a = \langle x_{a,|n|}, \ldots, x_{a,1}; |l| \rangle$ and $x_{i,j} < 2^{|l|}$ for $1 \leq i \leq a$ and $1 \leq j \leq |n|$. Note that the equation above reduces to

$$f^{\dagger_c}(x_1, \ldots, x_a, l, n) = \langle f(x_{1,|n|}, \ldots, x_{a,|n|}), \ldots, f(x_{1,1}, \ldots, x_{a,1}); |l| \rangle$$

if $f(x_{1,j}, \ldots, x_{a,j}) < 2^{|l|}$ for $1 \leq j \leq |n|$.[4]

In [5], the following result has been obtained.

Quotient Basis Theorem. *For any set F of polynomial growth functions, if $quot \in AC^0(F)$ then*

$$AC^0(F) = \mathrm{clos}_{SUBST}(add, sub, and, quot, us, F^{\dagger_c})$$

where $F^{\dagger_c} = \{f^{\dagger_c} | f \in F\}$.

The theorem above enabled us to show that $\{add, sub, and, quot, us\}$ is a basis for TC^0 by noting that $quad^{\dagger_c} \in \mathrm{clos}_{SUBST}(add, sub, and, quot, us)$ and to show that $\{add, sub, and, quot, us, tree^{\dagger_c}\}$ is a basis for NC^1.

Later, in [6], we improved the method of CRN elimination introduced in [5] and proved the following Quotient-free Basis Theorem which states that, for any finite set F of polynomial growth functions, the set $AC^0(F)$ has a basis. From the Quotient-free Basis Theorem we obtained a new basis for AC^0 by setting $F = \emptyset$.

Quotient-Free Basis Theorem. *For any set F of polynomial growth functions,*

$$AC^0(F) = \mathrm{clos}_{SUBST}(C_1, add, sub, and, conc, len, msp, ar2l, repl, convl, F^{\dagger_c}).$$

In this paper we will show the following improved version of the Quotient-free Basis Theorem.

Improved Quotient-free Basis Theorem. *For any set F of polynomial growth functions,*

$$AC^0(F) = \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, rp, F^{\dagger_c}) .$$

The Improved Quotient-free Basis Theorem yields immediately a new basis for AC^0.

Corollary 1. $AC^0 = \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, rp)$.

Moreover, we will state the following characterizations of TC^0.

Theorem 2.

$$\begin{aligned} TC^0 &= \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, x \cdot \sum_{i<|n|} 2^{|l|i}) \\ &= \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad). \end{aligned}$$

[4]See [6] for a full account about dagger operators.

3 A parametric Quotient-free Basis Theorem for $AC^0(F)$

Let *rpt* be any function such that

$$rpt(x,l,n) = rp(x,l,n) = x \cdot \sum_{i<|n|} 2^{|l|i}$$

when $AC^0\_SUM(x,l,n)$ is true. In this section, for any set F of polynomial growth functions, we will show that

$$AC^0(F) \subseteq \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, rpt, F^{\dagger_c})$$

and if $rpt \in AC^0(F)$ then

$$AC^0(F) = \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, rpt, F^{\dagger_c}).$$

The basic idea of the proof is that the functions $repl, arl, ar2l$ and $convl$ are special instances of $x \cdot \sum_{i<|n|} 2^{|l|i}$ and can be obtained from $rpt(x,l,n)$ by substituting a suitable AC^0 function for x.

A *normal* function class is a function class closed with respect to substitution which contains the function set $I \cup \{C_1, add, sub, and, msp, rpt\}$. Moreover, a function is *normal* iff it belongs to every normal class or, equivalently, iff it belongs to $clos_{SUBST}(C_1, add, sub, and, msp, rpt)$.

In the following, we will show that $repl, arl, ar2l$ and $convl$ are normal functions and so by the Quotient-free Basis Theorem, any normal function class contains $AC^0(F)$ if it contains $F^{\dagger_c}$.

Lemma 3. *If* $x < 2^{|l|}$ *then* $\forall_{i<j<|n|}\ (x2^{|l|i} \wedge x2^{|l|j} = 0)$ *and*

$$rpt(x,l,n) = \left\langle \overbrace{x, \ldots, x}^{|n|-times}; |l| \right\rangle .$$

Lemma 4. *The following functions are normal:* C_n *for any* n, $sg, cosg, s_0, s_1, pow, smash, 2^{|x| \dot{-} |y|}, multp, conc, lsp$, *and for any polynomial* p, $2^{p(|\mathbf{x}|)}$.

Proof. The proof is similar to that of Lemma 3 in [6]. Note first that

$$C_0(x) = C_1(x) \dot{-} C_1(x), C_{n+1}(x) = C_n(x) + C_1(x)$$

and

$$cosg(x) = C_1(x) \dot{-} x, sg(x) = cosg(cosg(x)) .$$

Then,

$$\begin{aligned}
s_0(x) &= x + x,\ s_1(x) = s_0(x) + C_1(x),\\
pow(x) &= 2^{|x|} = cosg(x) + (rpt(1, x, 2) \dot{-} 1),\\
smash(x, y) &= 2^{|x||y|} = rpt(2^{|x|} \dot{-} 1, x, y) + 1,\\
\max(x, y) &= (x \dot{-} y) + y,\\
x2^{|\max(x,y)|} &= rpt(x, \max(x, y), 2) \dot{-} x,\\
2^{|x| \dot{-} |y|} &= \left\lfloor 2^{|x|}/2^{|y|} \right\rfloor + sg(|y| \dot{-} |x|) = msp(2^{|x|}, 2^{|y|} - 1) + sg(2^{|x|} \dot{-} 2^{|y|})\\
multp(x, y) &= \left\lfloor x2^{|\max(x,y)|}/2^{|x| \dot{-} |y|} \right\rfloor = msp(x2^{|\max(x,y)|}, 2^{|x| \dot{-} |y|} - 1),\\
conc(x, y) &= multp(x, y) + y,\\
lsp(x, y) &= x \dot{-} multp(msp(x, y), y).
\end{aligned}$$

The function $2^{p(|\mathbf{x}|)}$ is normal for all polynomials p with non negative coefficients, because the function $2^{p(|\mathbf{x}|)}$ can be obtained from the normal functions 2^c and $2^{|x|}$ by a finite number of applications of the two equations $2^{p(|\mathbf{x}|)q(|\mathbf{y}|)} = (2^{p(|\mathbf{x}|)} - 1)\#(2^{q(|\mathbf{y}|)} - 1)$ and $2^{p(|\mathbf{x}|)+q(|\mathbf{y}|)} = 2^{|(2^{p(|\mathbf{x}|)}-1)*(2^{q(|\mathbf{y}|)}-1)|}$. Moreover, $2^{p(|\mathbf{x}|)}$ is normal even for any integer polynomial p with non negative values. Indeed, p can be expressed as the modified subtraction of two polynomials q, q' with non negative coefficients such that $q(|\mathbf{x}|) \geq q'(|\mathbf{x}|)$ and we obtain that $2^{p(|\mathbf{x}|)} = 2^{q(|\mathbf{x}|) \dot{-} q'(|\mathbf{x}|)} = msp(2^{q(|\mathbf{x}|)}, 2^{q'(|\mathbf{x}|)} - 1) + sg(2^{q(|\mathbf{x}|)} \dot{-} 2^{q'(|\mathbf{x}|)})$. □

Lemma 5. *Function cond is normal.*

Proof. By Lemma 4 the function $2^{|y|+|z|}$ is normal. Then, also the function

$$f(x, y, z) = rpt(sg(x), 1, 2^{|y|+|z|} - 1) = \begin{cases} 0 & \text{if } x = 0,\\ 2^{|y|+|z|} - 1 & \text{otherwise} \end{cases}$$

is normal. The lemma follows immediately by noting that

$$cond(x, y, z) = and(f(x, y, z), z) + and(f(cosg(x), y, z), y).$$

□

Any normal class is closed with respect to definition by cases and contains the predicates generated by the standard comparison predicates and the Boolean connectives.

Lemma 6. *Any normal class is closed with respect to definition by cases.*

Proof. Assume that C is a normal class and $f_1, \ldots, f_{a+1} \in C$. Let $g_1, \ldots, g_a \in C$ be the characteristic functions of $Q_1, \ldots, Q_a$, respectively. The lemma follows immediately from Lemma 5 because the function

$$f(\mathbf{x}) = \begin{cases} f_1(\mathbf{x}) & \text{if } Q_1(\mathbf{x}),\\ \ldots & \ldots\\ f_a(\mathbf{x}) & \text{if } Q_a(\mathbf{x}),\\ f_{a+1}(\mathbf{x}) & \text{otherwise} \end{cases}$$

can be defined as

$$f(\mathbf{x}) = cond(g_1(\mathbf{x}), cond(\ldots cond(g_a(\mathbf{x}), f_{a+1}(\mathbf{x}), f_a(\mathbf{x}))\ldots), f_1(\mathbf{x})).$$

□

Lemma 7. *The predicates of a normal class are closed with respect to conjunction, disjunction, and negation.*

Proof. Assume that C is a normal class and let $g_1 \in C$ and $g_2 \in C$ be the characteristic functions of predicates Q_1 and Q_2, respectively. Then, $cosg(g_1(x))$, $cond(g_1(x), C_0(x), cond(g_2(x), C_0(x), C_1(x)))$ and $cond(g_1(x), cond(g_2(x), C_0(x), C_1(x)), C_1(x))$ are the characteristic functions of $\neg Q_1$, $Q_1 \wedge Q_2$ and $Q_1 \vee Q_2$, respectively. The lemma follows immediately from Lemma 4 and Lemma 5. □

Lemma 8. *The comparison predicates $x < y, x \leq y, x > y, x \geq y, x = y, x \neq y$ are normal.*

Proof. Note that $x > y \Leftrightarrow sg(x \dot{-} y) = 1$ and $x = y \Leftrightarrow cosg((x \dot{-} y) + (y \dot{-} x)) = 1$. The remaing predicates can be defined using the Boolean operations and the lemma follows from Lemma 7. □

Lemma 9. *Function repl is normal.*

Proof. Since $lsp(x, l) = rem(x, 2^{|l|}) < 2^{|l|}$, by Lemma 3

$$\begin{aligned} repl(x, l, n) &= \left\langle \overbrace{rem(x, 2^{|l|}), \ldots, rem(x, 2^{|l|})}^{|n|-times}; |l| \right\rangle \\ &= \begin{cases} rpt(lsp(x, l), l, n) & \text{if } (l > 0) \ \wedge \ (n > 0), \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

The lemma follows immediately by Lemmata 6-8. □

Lemma 10. *If $\sum_{j=1}^{N} x_j < 2^L$ then*

$$\left\langle \sum_{j=N-1}^{N} x_j, \ldots, \sum_{j=2}^{N} x_j, \sum_{j=1}^{N} x_j, \ldots, \sum_{j=1}^{1} x_j; L \right\rangle = \langle x_N, \ldots, x_1; L \rangle \cdot \sum_{i<N} 2^{Li}.$$

Lemma 11. *Functions arl and $|x|$ are normal.*

Proof. Set $L = |l|$ and consider the normal function $f(l) = rpt(rpt(1, l, l), l, l)$. By definition of rpt and Lemma 10

$$f(l) = \left\langle \overbrace{1, \ldots, 1}^{L-times}; L \right\rangle \cdot \sum_{i<L} 2^{Li} = \left\langle 1, \ldots, L \dot{-} 1, L, L \dot{-} 1, \ldots, 1; L \right\rangle.$$

Therefore, $arl(l) = lsp(f(l), 2^{|l|^2} - 1) \dot{-} rpt(1, l, l)$ because $lsp(f(l), 2^{|l|^2} - 1) = \langle L, \ldots, 1; L \rangle$ and arl is normal because $lsp(x, 2^{|l|^2} - 1)$ is normal by Lemma 4. Finally, $|x| = lsp(msp(f(x), 2^{|x|^2 \dot{-} |x|} - 1), x)$ and $|x|$ is normal because $lsp, 2^{|x|^2 \dot{-} |x|}$ and $2^{|x|}$ are normal by Lemma 4. □

Lemma 12. *Function ar2l is normal.*

Proof. Set $L = |l|$ and consider the function $f(l) = rpt(arl(l), l, l)$. By definition of *rpt* and Lemma 10

$$\begin{aligned} f(l) = \left\langle L\dot{-}1, \ldots, 1, 0; L\right\rangle \cdot \sum_{i<L} 2^{Li} \\ = \left\langle t_{L\dot{-}1}\dot{-}t_{L\dot{-}2}, \ldots, t_{L\dot{-}1}\dot{-}t_2, t_{L\dot{-}1}\dot{-}t_1, t_{L\dot{-}1}, \ldots, t_1, t_0; L\right\rangle \end{aligned}$$

where $t_n = \frac{n(n+1)}{2}$ is the n-th triangular number.

Now, since $2t_n - n = n^2$, we obtain $ar2l(l) = 2 \cdot lsp(f(l), 2^{|l|^2} - 1)\dot{-}arl(l)$ because $lsp(f(l), 2^{|l|^2} - 1) = \left\langle t_{L\dot{-}1}, \ldots, t_1, t_0; L\right\rangle$ and $ar2l$ is normal because arl and $lsp(x, 2^{|l|^2} - 1)$ are normal by the lemma above and Lemma 4. □

The function $incr(x, l, r, n) = rpt(x, 2^{|r|\dot{-}|l|}, n) \wedge rpt(2^{|l|} - 1, r, n)$ has been introduced in [8]. The following lemma is analogous to Lemma 2.10 of [5] and Statement 1.1.4.3 of [8].

Lemma 13. *If* $l, n > 0$, $|r| \geq (|n| + 1)|l|$ *and* $x_{|n|}, \ldots, x_1 < 2^{|l|}$ *then*

$$incr(\left\langle x_{|n|}, \ldots, x_1; |l|\right\rangle, l, r, n) = \left\langle x_{|n|}, \ldots, x_1; |r|\right\rangle .$$

Lemma 14. *Function incr is normal.*

Proof. The lemma follows immediately from Lemma 4. □

Lemma 15. *Function convl is normal.*

Proof. Set $L = |l|, R = |r|$ and $N = |n|$. We first define a function *decr* such that $decr(\langle x_N, \ldots, x_1; L\rangle, l, r, n) = \langle x_N, \ldots, x_1; R\rangle$ provided that $R < L$ and $x_N, \ldots, x_1 < 2^R$.

For $x = \langle x_N, \ldots, x_1; L\rangle$, we have

$$\begin{aligned} incr(x, l, 2^{L(N+1)+R} - 1, n) &= \langle x_N, \ldots, x_2, x_1; L(N+1) + R\rangle \\ &= \left\langle x_N 2^{R(N-1)}, \ldots, x_2 2^R, x_1; L(N+1)\right\rangle \end{aligned}$$

by Lemma 13 and Lemma 20 of [6].

Now, for $y = \left\langle x_N 2^{R(N-1)}, \ldots, x_2 2^R, x_1; L(N+1)\right\rangle$, we have

$$i < j < N \Rightarrow y 2^{L(N+1)i} \wedge y 2^{L(N+1)j} = 0$$

because

$$bit(y 2^{L(N+1)i}, k) = \begin{cases} 0 & \text{if } k < L(N+1)i \\ bit(x_q 2^{Rq}, s) & \text{otherwise} \end{cases}$$

where

$$q = \lfloor k - L(N+1)i)/L(N+1)\rfloor = \lfloor k/L(N+1)\rfloor - i$$

and

$$s = rem(k - L(N+1)i), L(N+1)) = rem(k, L(N+1)).$$

So, if

$$bit(y2^{L(N+1)i}, k) = bit(y2^{L(N+1)j}, k) = 1$$

then

$$bit(x_q 2^{Rq}, s) = bit(x_p 2^{Rp}, s) = 1$$

with $p < q$. But this means that $Rp \leq s < Rp + R$ and $Rq \leq s < Rq + R$ which implies $s < R(p+1) \leq Rq \leq s$, a contradiction.

Therefore, by Lemma 10, the function

$$f(x, l, r, n) = rpt(incr(x, l, 2^{L(N+1)+R} - 1, n), 2^{L(N+1)} - 1, n)$$

satisfies the following equations

$$\begin{aligned} f(\langle \mathbf{x}; L\rangle, l, r, n) &= \left\langle x_N 2^{R(N-1)}, \ldots, x_2 2^R, x_1; L(N+1) \right\rangle \cdot \sum_{i<N} n 2^{L(N+1)i} \\ &= \langle \left\langle x_N \overbrace{0, \ldots, 0}^{N-1}; R \right\rangle, \ldots, \langle x_N, \ldots, x_2, 0; R\rangle, \\ &\qquad \langle x_N, \ldots, x_1; R\rangle, \ldots, \left\langle \overbrace{0, \ldots, 0}^{N-1}, x_1; R \right\rangle; L(N+1)\rangle \end{aligned}$$

where $\mathbf{x} = x_N, \ldots, x_1$. Then, for

$$decr(x, l, r, n) = lsp(msp(f(x, l, n), 2^{L(N+1)(N-1)} - 1), 2^{RN} - 1)$$

we have $decr(\langle x_N, \ldots, x_1; L\rangle, l, r, n) = \langle x_N, \ldots, x_1; R\rangle$. Moreover, $decr$ is normal because $2^{L(N+1)(N-1)} - 1$ and $2^{RN} - 1$ are normal by Lemma 4 and f is normal by Lemma 14 and Lemma 4.

Furthermore, define the function

$$trim(x, l, r, n) = \begin{cases} x \wedge rpt(2^R - 1, l, n) & \text{if } L \geq R > 0, \\ 0 & \text{otherwise.} \end{cases}$$

and note that for $R \leq L$ we have

$$trim(x, l, r, n) = \langle rem(x_N, 2^R), \ldots, rem(x_1, 2^R); L\rangle$$

where $x_N, \ldots, x_1$ are the N least significant base 2^L digits of x.

Finally, from Lemma 13 we have

$$convl(x, l, r, n) = \begin{cases} decr(incr(trim(x, l, l, n), l, 2^{R(N+1)} - 1, n), & \\ 2^{R(N+1)} - 1, r, n) & \text{if } R > L > 0, \\ decr(trim(x, l, r, n), l, r, n) & \text{if } L > R > 0, \\ trim(x, l, r, n) & \text{if } L = R > 0, \\ 0 & \text{otherwise.} \end{cases}$$

and the functions $trim$ and $convl$ are normal by Lemmata 6-8 because $incr$ and $decr$ are normal. □

Lemma 16. *For any set F of polynomial growth functions and any normal class C, if $F^{\dagger_c} \subseteq C$ then $AC^0(F) \subseteq C$.*

Proof. By lemmata 4, 11, 9, 12 and 15 we have $conc, len, repl, ar2l, convl \in C$ and so $\text{clos}_{SUBST}(C_1, add, sub, and, conc, len, msp, ar2l, repl, convl, F^{\dagger_c}) \subseteq C$. The lemma follows immediately from the Quotient-free Basis Theorem of [6]. □

Corollary 17. *For any set F of polynomial growth functions,*

$$AC^0(F) \subseteq \text{clos}_{SUBST}(C_1, add, sub, and, msp, rpt, F^{\dagger_c}).$$

Proof. By definition, $\text{clos}_{SUBST}(C_1, add, sub, and, msp, rpt, F^{\dagger_c})$ is a normal class which contains $F^{\dagger_c}$. □

Theorem 18 (Parametric Quotient-Free Basis Theorem)**.** *For any set F of polynomial growth functions, if $rpt \in AC^0(F)$ then*

$$AC^0(F) = \text{clos}_{SUBST}(C_1, add, sub, and, msp, rpt, F^{\dagger_c}).$$

Proof. Note that $C_1, add, sub, and, msp \in AC^0$, moreover $rpt \in AC^0(F)$ by hypothesis and $F^{\dagger_c} \subseteq AC^0(F)$ by Lemma 8 of [6]. □

4 A new basis for $AC^0(F)$

In this section we will prove the Improved Quotient-free Basis Theorem. In order to do so, we just need to show that $rp \in AC^0$ and to set $rpt = rp$ in Theorem 18.

Lemma 19. $rp(x, l, n) \in AC^0$.

Proof sketch. Recall that the predicate $AC^0\_SUM(x, l, n)$ introduced in the Preliminaries is defined as

$$AC^0\_SUM(x, l, n) \;\Leftrightarrow\; (ln > 0) \wedge (P_1(x, l, n) \vee P_2(x, l, n) \vee P_3(x, l, n))$$

are P_1, P_2 and P_3 are mutually disjoint AC^0 predicates. We show that rp can be defined by cases. Indeed, there are functions h_1, h_2 and h_3 in AC^0 such that $rp(x, l, n) = h_i(x, l, n)$ if $P_i(x, l, n) \wedge ln > 0$ is true. We assume that $ln > 0$ and set $L = |l|$ and $N = |n|$.

First, assume that $P_1(x, l, n)$ is true. Then $x = \left\langle \overbrace{1, \ldots, 1}^{L-times}; L \right\rangle \wedge (1 < l)$ and

$$\begin{aligned} rp(x, l, n) &= \langle 1, \ldots, L \dot{-} 1, L, L \dot{-} 1, \ldots, 1; L \rangle \\ &= \langle 1, \ldots, L \dot{-} 1, L; L \rangle \cdot 2^{L(L-1)} + \langle L \dot{-} 1, \ldots, 1; L \rangle \\ &= (repl(L, l, l) \dot{-} arl(l)) \cdot 2^{L(L-1)} + msp(arl(l), l). \end{aligned}$$

Assume that $P_2(x, l, n)$ is true and recall that $2t_m = m(m+1)$.
Then $x = \langle L-1, \ldots, 1, 0; L \rangle \wedge (1 < l)$ and

$$\begin{aligned} rp(x, l, n) &= \langle t_{L \dot{-} 1} \dot{-} t_{L \dot{-} 2}, \ldots, t_{L \dot{-} 1} \dot{-} t_2, t_{L \dot{-} 1} \dot{-} t_1, t_{L \dot{-} 1}, \ldots, t_1, t_0; L \rangle \\ &= \langle t_{L \dot{-} 1} \dot{-} t_{L \dot{-} 2}, \ldots, t_{L \dot{-} 1} \dot{-} t_2, t_{L \dot{-} 1} \dot{-} t_1; L \rangle \cdot 2^{L(L-1)} + \langle t_{L \dot{-} 1}, \ldots, t_1, t_0; L \rangle \\ &= (repl(t_{L \dot{-} 1}, l, \lfloor l/2 \rfloor) \dot{-} T(\lfloor l/2 \rfloor)) \cdot 2^{L(L-1)} + T(l) \end{aligned}$$

where $T(l) = \lfloor (ar2l(l) + arl(l))/2 \rfloor = \langle t_{L \dot{-} 1}, \ldots, t_0; L \rangle$ belongs to AC^0 by Lemma 10 and Lemma 19 of [6].

Assume that $P_3(x, l, n)$ is true and recall that this is equivalent to

$$x < 2^{|l||n|} \wedge \forall_{i<j<|n|} (x2^{|l|i} \wedge x2^{|l|j} = 0).$$

Then, for any $k < 2|l||n|$ there is at most one index $i < N$ such that $bit(x2^{|l|i}, k) = 1$ and so, no carry is generated in the computation of $\sum_{i<|n|} x2^{|l|i}$. Therefore,

$$rp(x, l, n) = x \cdot \sum_{i<|n|} 2^{|l|i} = \sum_{i<|n|} x2^{|l|i} = \bigvee_{i<|n|} x2^{|l|i}$$

where $\bigvee_{i<|n|} f(x, i)$ is defined as

$$bit(\bigvee_{i<|n|} f(x, i), j) = 1 \Leftrightarrow \exists_{i<|n|} bit(f(x, i), j) = 1$$

and belongs to AC^0 because AC^0 is closed with respect to sharply bounded quantifiers, see [2].

Concluding, rp is defined by cases from AC^0 functions and predicates and so it belongs to AC^0. □

We apply now Theorem 18 to obtain the Improved Quotient-free Basis Theorem.

Theorem 20. *For any set F of polynomial growth functions,*

$$AC^0(F) = \text{clos}_{SUBST}(C_1, add, sub, and, msp, rp, F^{\dagger_c}).$$

Proof. Set $rpt = rp$ in Theorem 18. The theorem follows immediately from Lemma 19. □

Corollary 21. $AC^0 = \text{clos}_{SUBST}(C_1, add, sub, and, msp, rp)$.

5 New bases for TC^0

In this section we show that both $\{C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad\}$ and $\{C_1, add, sub, and, msp, x \cdot \sum_{i<|n|} 2^{|l|i}\}$ are bases for TC^0. This result is independent from the striking result of [4], namely integer division is in TC^0, which was used in [7] to introduce the first basis for TC^0. The new bases are obtained as another application of Theorem 18.

Lemma 22.

$$\{xy, x \cdot \sum_{i<|n|} 2^{|l|i}\} \cup AC^0 \subseteq \text{clos}_{SUBST}(C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad).$$

Proof. Note first that $xy = (x + y)^2 - x^2 - y^2$. The lemma follows from Corollary 17 by setting $F = \emptyset$ and $rpt(x, l, n) = x \cdot \sum_{i<|n|} 2^{|l|i}$. □

Now, we show that

$$quad^{\dagger_c} \in \text{clos}_{SUBST}(C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad).$$

By Theorem 18, this implies that $\{C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad\}$ is a basis for TC^0.

Lemma 23 (Lemma 3.4 of [5]). $x^2 = \sum_{i<|x|,\, bit(x,i)=1}(2 \cdot 4^i \cdot \text{MSP}(x,i) - 4^i)$.

Proof. By induction on x, using the following definition of the quadratum function:

$$\begin{aligned} 0^2 &= 0 \\ (2y)^2 &= 4y^2 \\ (2y+1)^2 &= 4y^2 + 4y + 1\,. \end{aligned}$$

□

Lemma 24. $quad^{\dagger_c} \in \text{clos}_{SUBST}(C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad)$.

Proof. Set $R = |r|$ and $N = |n|$. Let $\mathbf{x} = x_N, \ldots, x_1$ and assume that $x_j^2 < 2^R$ for any $1 \le j \le N$. Consider the function

$$f(x,y) = \begin{cases} 2\text{MSP}(x,y)4^{\min(|x|,y)} - 4^{\min(|x|,y)} & \text{if } bit(x,y) = 1 \\ 0 & \text{otherwise} \end{cases}$$

and note that f is in AC^0 because $4^{\min(|x|,y)} = 2^{\min(|x*x|,2y)} \in AC^0$. Then, $f^{\dagger_c} \in AC^0$ by Lemma 8 of [6]. Furthermore, consider the functions

$$M(r,n) = convl(arl(r), r, 2^{RN}-1, r) \sum_{i<N} 2^{Ri}$$

and

$$g(x,r,n) = f^{\dagger_c}(repl(x, 2^{RN}-1, r), M(r,n), r, 2^{RN}-1)$$

belonging to AC^0 such that

$$M(r,n) = \left\langle \overbrace{R-1, \ldots, R-1}^{N-times}, \ldots, \overbrace{0, \ldots, 0}^{N-times}; R \right\rangle$$

and

$$g(\langle \mathbf{x}; R\rangle, r, n) = \langle \langle u_{R-1,N}, \ldots, u_{R-1,1}; R\rangle, \ldots, \langle u_{0,N}, \ldots, u_{0,1}; R\rangle; RN\rangle$$

where $u_{i,j} = \begin{cases} 2\text{MSP}(x_j,i)4^i - 4^i & \text{if } bit(x_j,i) = 1 \\ 0 & \text{otherwise} \end{cases}$.

So, for $\langle s_{2R-1}, \ldots, s_1; RN\rangle = g(\langle \mathbf{x}; R\rangle, r, n) \cdot \sum_{i<R} 2^{RNi}$ we have

$$s_R = \langle x_N^2, \ldots, x_1^2; R\rangle$$

because

$$s_R = \sum_{i<R} \langle u_{i,N}, \dots, u_{i,1}; R \rangle = \left\langle \sum_{i<R} u_{i,N}, \dots, \sum_{i<R} u_{i,1}; R \right\rangle$$

by Lemma 10 and, for any $1 \le j \le N$,

$$\sum_{i<R} u_{i,j} = \sum_{i<R, bit(x_j,i)=1} 2\mathrm{MSP}(x_j, i)4^i - 4^i = x_j^2$$

by Lemma 23. Therefore, for

$$q(x,r,n) = lsp(msp(g(x,r,n) \cdot \sum_{i<R} 2^{RNi}, 2^{(R-1)RN} - 1), 2^{RN} - 1)$$

we have $q(\langle \mathbf{x}; R \rangle, r, n) = \langle x_N^2, \dots, x_1^2; R \rangle$. The lemma follows by noting that

$$quad^{\dagger_c}(x,l,n) = convl(trim(q(convl(x,l,2^{2L}-1,n), 2^{2L}-1,n), 2^{2L}-1,l,n) \\ , 2^{2L}-1,l,n)$$

where $trim$ is the AC^0 function defined in Lemma 15. □

Now we obtain two new bases for TC^0.

Theorem 25. $TC^0 = \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad)$.

Proof. Set $F = \{quad\}$ and $rpt(x,l,n) = x \cdot \sum_{i<|n|} 2^{|l|i}$. Then, by Theorem 18 and Lemma 24,

$$\begin{aligned} AC^0(quad) &= \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, x \cdot \sum_{i<|n|} 2^{|l|i}, quad^{\dagger_c}) \\ &\subseteq \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, \sum_{i<|n|} 2^{|l|i}, quad) \\ &\subseteq TC^0 \end{aligned}$$

and the theorem follows immediately because $AC^0(quad) = AC^0(mult) = TC^0$. □

Theorem 26. $TC^0 = \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, x \cdot \sum_{i<|n|} 2^{|l|i})$.

Proof. By Theorem 25 it suffices to show that

$$quad \in \mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, x \cdot \sum_{i<|n|} 2^{|l|i}).$$

First, consider the function $f_1(x,y) = convl(y, 2, 2^{|x|+|y|+1}, y)$ such that

$$f_1(x,y) = \langle y_{|y|-1}, \dots, y_0; |x| + |y| + 1 \rangle$$

where $y_i = bit(y,i)$ and the function $f_2(x,y) = repl(x, 2^{|x|+|y|+1}, y)$ such that

$$f_2(x,y) = \left\langle \overbrace{x, \dots, x}^{|y|-times}; |x| + |y| + 1 \right\rangle.$$

Then, for $f_3(x,y) = (2^{|x|}-1)\cdot f_1(x,y) \wedge f_2(x,y)$ we have by Lemma 20 of [6]

$$\begin{aligned} f_3(x,y) &= \langle xy_{|y|-1}, \ldots, xy_0; |x|+|y|+1\rangle \\ &= \left\langle xy_{|y|-1}2^{|y|-1}, \ldots, xy_0; |x|+|y| \right\rangle . \end{aligned}$$

Note that f_1, f_2 and f_3 belong to AC^0 and therefore to $\mathrm{clos}_{SUBST}(C_1, add, sub, and, msp, x \cdot \sum_{i<|n|} 2^{|l|i})$ by Corollary 17.

Finally, for $f_4(x,y) = f_3(x,y) \cdot \sum_{i<|y|} 2^{|2^{|x||y|}-1|i}$ we have

$$f_4(x,y) = \left\langle xy_{|y|-1}2^{|y|-1}, \ldots, xy_0; |x|+|y| \right\rangle \cdot \sum_{i<|y|} 2^{(|x|+|y|)i}$$

and the theorem follows by Lemma 10 because the $|y|-$th digit in base $2^{|x|+|y|}$ of $f_4(x,y)$ is $\sum_{i<|y|} xy_i 2^i = xy$. □

Remark. The difference between AC^0 and TC^0 seems to be very subtle. Indeed, the basis for AC^0 and the basis for TC^0 of Theorem 26 differ for one function only. Moreover, the former basis contains rp while the latter basis contains $x \cdot \sum_{i<|n|} 2^{|l|i}$, which is a sort of "extension" of rp. This result could be the starting point for a new, algebraic proof that $AC^0 \neq TC^0$.

6 Bases for complexity classes with complete problems

The new bases introduced in Section 4 and Section 5 can be used to obtain bases for complexity classes with complete problems. Indeed, in [6] it was shown that a function class F with complete decision problems under AC^0 reductions can be characterized as the AC^0 closure of the characteristic function of a suitable complete problem, provided that F is closed with respect to substitution and CRN. Then, the Improved Quotient-free Basis Theorem yields immediately a new basis for F. Here we state the new bases without proofs. The interested reader may refer to Section 3 of [6] for a full treatment of the subject.

Theorem 27.

$$NC^1 = AC^0(ch_{BFVP}) = clos_{SUBST}(C_1, add, sub, and, msp, rp, ch^{\dagger_c}_{BFVP}),$$

$$L = AC^0(ch_{1GAP}) = clos_{SUBST}(C_1, add, sub, and, msp, rp, ch^{\dagger_c}_{1GAP}),$$

$$P = AC^0(ch_{CVP}) = clos_{SUBST}(C_1, add, sub, and, msp, rp, ch^{\dagger_c}_{CVP}),$$

$$PSPACE = AC^0(ch_{QBF}) = clos_{SUBST}(C_1, add, sub, and, msp, rp, ch^{\dagger_c}_{QBF})$$

where BFVP is the Boolean Formula Value Problem, 1GAP is the Degree-One Graph Accessibility Problem, CVP is the Circuit Value Problem and QBF is the Quantified Boolean Formulas Problem.

Note that Theorem 27 also holds when rp is replaced by $\sum_{i<|n|} 2^{|l|i}$ and $quad$ or by $x \cdot \sum_{i<|n|} 2^{|l|i}$ (and $BFVP$, $1GAP$, CVP and QBF are possibly replaced by TC^0-complete problems for NC^1, L, P and $PSPACE$ respectively).

References

[1] P. Clote. *Sequential, machine-independent characterizations of the parallel complexity classes AlogTIME, AC^k, NC^k and NC*, pages 49–69. Birkhäuser Boston, Boston, MA, 1990.

[2] P. Clote. Computation models and function algebras. In Edward R. Griffor, editor, *Handbook of Computability Theory*, volume 140 of *Studies in Logic and the Foundations of Mathematics*, pages 589 – 681. Elsevier, 1999.

[3] P. Clote and G. Takeuti. *First Order Bounded Arithmetic and Small Boolean Circuit Complexity Classes*, pages 154–218. Birkhäuser Boston, Boston, MA, 1995.

[4] W. Hesse, E. Allender, and D. A. Mix Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *Journal of Computer and System Sciences*, 65(4):695 – 716, 2002. Special Issue on Complexity 2001.

[5] S. Mazzanti. CRN elimination and substitution bases for complexity classes. *Fundam. Inform.*, 120(1):29–58, 2012.

[6] S. Mazzanti. Bases for AC^0 and other complexity classes. *Fundam. Inform.*, 136(4):433–460, 2015.

[7] S. A. Volkov. Generating some classes of recursive functions by superpositions of simple arithmetic functions. *Doklady Mathematics*, 76(1):566–567, 2007.

[8] S. A. Volkov. Finite bases with respect to the superposition in classes of elementary recursive functions, dissertation. *CoRR*, abs/1611.04843, 2016.