

Dual VP Classes

Rutgers University has made this article freely available. Please share how this access benefits you.
Your story matters. [\[https://rucore.libraries.rutgers.edu/rutgers-lib/47424/story/\]](https://rucore.libraries.rutgers.edu/rutgers-lib/47424/story/)

This work is an **ACCEPTED MANUSCRIPT (AM)**

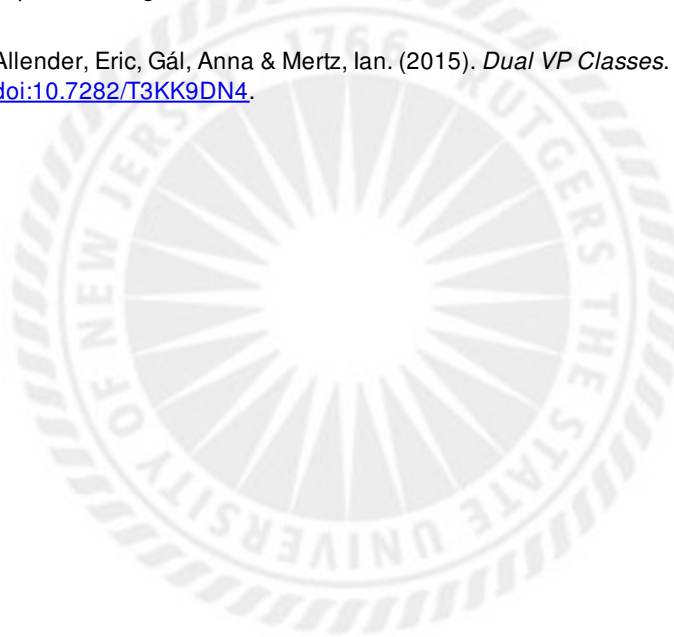
This is the author's manuscript for a work that has been accepted for publication. Changes resulting from the publishing process, such as copyediting, final layout, and pagination, may not be reflected in this document. The publisher takes permanent responsibility for the work. Content and layout follow publisher's submission requirements.

Citation for this version and the definitive version are shown below.

Citation to Publisher Allender, Eric, Gál, Anna & Mertz, Ian. (2015). *Dual VP Classes*. Milan (Italy).

Version: <http://dx.doi.org/10.1007/978-3-662-48054-0>.

Citation to *this* Version: Allender, Eric, Gál, Anna & Mertz, Ian. (2015). *Dual VP Classes*. Milan (Italy). Retrieved from [doi:10.7282/T3KK9DN4](https://doi.org/10.7282/T3KK9DN4).



Terms of Use: Copyright for scholarly resources published in RUcore is retained by the copyright holder. By virtue of its appearance in this open access medium, you are free to use this resource, with proper attribution, in educational and other non-commercial settings. Other uses, such as reproduction or republication, may require the permission of the copyright holder.

Article begins on next page

Dual VP Classes

Eric Allender¹, Anna Gál², and Ian Mertz¹

¹ Department of Computer Science, Rutgers University, Piscataway, NJ, USA
allender@cs.rutgers.edu, iwmerz@gmail.com

² Department of Computer Science, University of Texas, Austin, TX, USA
panni@cs.utexas.edu

Abstract. We consider the complexity class ACC^1 and related families of arithmetic circuits. We prove a variety of collapse results, showing several settings in which no loss of computational power results if fan-in of gates is severely restricted, as well as presenting a natural class of arithmetic circuits in which no expressive power is lost by severely restricting the algebraic degree of the circuits. These results tend to support a conjecture regarding the computational power of the complexity class VP over finite algebras, and they also highlight the significance of a class of arithmetic circuits that is in some sense dual to VP .

1 Introduction

Most of the well-studied subclasses of P are defined in terms of Boolean or arithmetic circuits. The question of the relative power of NC^1 , LogCFL , and AC^1 , or of $\#\text{NC}^1$ and $\#\text{LogCFL}$ boils down to the question of how the computational power of a (log-depth, polynomial-size) circuit model depends on the *fan-in* of gates in the model.

Our main contribution is to present several settings where fan-in can be severely restricted for log-depth, polynomial-size circuits, with *no* loss of computational power.

1.1 ACC^1 and TC^1

There is a large literature exploring the connections between Boolean and arithmetic circuit complexity; see [18]. For instance, the Boolean class TC^1 (log-depth MAJORITY circuits) corresponds to $\#\text{AC}^1(\mathbb{F}_{p_n})$ (log-depth *unbounded* fan-in arithmetic circuits where the circuits for inputs of size n operate over the field \mathbb{F}_{p_n} , where p_n is the n -th prime [12]). We show here that $\text{ACC}^1 = \bigcup_p \#\text{AC}^1(\mathbb{F}_p)$. Is unbounded fan-in necessary for these characterizations?

No, it is not! The *semiunbounded* fan-in model, where the $+$ gates have fan-in two, also yields ACC^1 (Corollary 1), and the same is true (modulo logspace-Turing reductions) for TC^1 (Theorem 6).

The usual definition of ACC^1 is in terms of *unbounded* fan-in AND and OR gates, along with MOD_m gates for different m , and we observe here that TC^1 has an analogous characterization with MOD_{p_n} gates. Here, too, the fan-in of the

AND and OR gates can be restricted, to *constant* fan-in for ACC^1 (Theorem 9) (while AND and OR gates are not needed at all for the TC^1 characterization (Theorem 6)).

1.2 Algebraic Degree

In the previous section’s discussion, the arithmetic circuit families that characterize ACC^1 and TC^1 have algebraic degree $n^{O(\log n)}$. Much more has been written about poly-size arithmetic circuits with degree $n^{O(1)}$: VP . Similar to our characterization of ACC^1 in terms of semiunbounded circuits, VP also corresponds to semiunbounded circuits, but with the (more common) restriction of having the \times gates have fan-in two [16,2]. VP is usually studied as a class of *polynomials*, but it is also common to study the *Boolean part* of VP over a given semiring R , where (following [5]), the Boolean part of an arithmetic circuit class is the class of languages whose characteristic functions are computed by circuits in the class. Especially over finite fields, there is little to distinguish VP from its Boolean part.

Immerman and Landau conjectured that computing the determinant of integer matrices is complete for TC^1 [11]. This would have several consequences, including providing a characterization of TC^1 in terms of $\text{VP}(\mathbb{Q})$. Buhrman et al. [7] have argued that the Immerman-Landau conjecture is unlikely, in that this would imply that arbitrary polynomials having degree $n^{O(\log n)}$ and polynomial-size arithmetic circuits mod p_n could be simulated by arithmetic circuits of *much lower degree* over \mathbb{Q} . This raises the question: When can high-degree polynomials over one algebra be simulated by low-degree polynomials over another?

Our degree-reduction theorem (Corollary 7) gives one natural class of polynomials of degree $n^{O(\log n)}$ over one algebra (\mathbb{F}_2) that *can* be simulated by polynomials having much smaller degree. We show that restricting the fan-in of \times gates in $\#\text{AC}^1(\mathbb{F}_2)$ circuits to be logarithmic results in *no loss of expressive power*; the restricted class (whose polynomials have algebraic degree only $n^{O(\log \log n)}$) represents the same class of functions as the unrestricted class (with degree $n^{O(\log n)}$). We believe that this weakens the arguments against the Immerman-Landau conjecture that were raised in [7], and we suspect that there are other such examples, where restricting the fan-in of \times gates causes no loss of power. We also see no reason why degree $n^{O(\log \log n)}$ should be optimal. Lowering the degree to $n^{O(1)}$ would imply $\#\text{AC}^1(\mathbb{F}_2) = \text{AC}^1[2] = \text{VP}(\mathbb{F}_2)$. (We omit “Boolean part” if it causes no confusion.)

1.3 Duality

We have mentioned that VP corresponds to semiunbounded arithmetic circuits with bounded-fan-in \times gates. Over the Boolean semiring, logarithmic depth polynomial-size semiunbounded fan-in circuits (with bounded fan-in AND gates and unbounded fan-in OR gates, with NOT gates only at the input level) characterize the complexity class LogCFL , also known as SAC^1 , which has been the subject of numerous investigations [9,13].

Because LogCFL is closed under complement [6], it can be characterized in terms of semiunbounded fan-in circuits by restricting either the AND gates or the OR gates to have bounded fan-in. It is unknown if there is any other algebraic structure for which a similar phenomenon occurs. In particular, it is not known how the complexity of functions in $\text{VP}(\mathbb{F}_p)$ compares to that of the functions in the classes defined by logarithmic depth polynomial-size semiunbounded fan-in circuits with bounded fan-in $+$ gates and unbounded fan-in \times gates.

A large part of the motivation for this paper is to understand the computational power of these semiunbounded fan-in circuit classes, which are in some sense dual to Valiant's classes $\text{VP}(\mathbb{F}_p)$. We use the notation $\mathcal{AP}(\mathbb{F}_p)$ to refer to the class of problems characterized by logarithmic depth polynomial-size semiunbounded fan-in circuits with bounded fan-in addition gates and unbounded fan-in multiplication gates. Formal definitions appear in Section 2. We show that each class $\mathcal{AP}(\mathbb{F}_p)$ corresponds exactly to a particular subclass of ACC^1 , and that the union over all p of $\mathcal{AP}(\mathbb{F}_p)$ is exactly equal to ACC^1 (Corollary 1).

We conjecture that ACC^1 is precisely the class of languages logspace-Turing reducible to $\bigcup_m \text{VP}(\mathbb{Z}_m)$. If the conjecture is true, then ACC^1 can be defined using either kind of semiunbounded circuits, with bounded fan-in $+$ or bounded fan-in \times .

2 Preliminaries, and Definitions of \mathcal{A} -classes

We assume that the reader is familiar with Boolean circuit complexity classes such as AC^0 and ACC^0 ; a good source for this background material is the excellent text by Vollmer [18]. The following notation is used by Vollmer, and we follow those conventions here:

Definition 1. – AC^i is the class of languages accepted by Dlogtime-uniform circuit families of size $n^{O(1)}$ and depth $O(\log^i n)$, with NOT gates, and unbounded fan-in (AND, OR).

- $\text{AC}^i[m]$ is defined as AC^i , but in addition unbounded fan-in MOD_m gates are allowed, which output 1 iff the number of input wires carrying a value of 1 is a multiple of m .
- For any finite set $S \subset \mathbb{N}$, $\text{AC}^i[S]$ is defined analogously to $\text{AC}^i[m]$, but now the circuit families are allowed to use MOD_r gates for any $r \in S$. It is known that, for any $m \in \mathbb{N}$, $\text{AC}^i[m] = \text{AC}^i[\text{Supp}(m)]$, where – following the notation of [8] – $\text{Supp}(m) = \{p : p \text{ is prime and } p \text{ divides } m\}$ [14]. Thus, in particular $\text{AC}^i[6] = \text{AC}^i[2, 3]$ and $\text{AC}^i = \text{AC}^i[\emptyset]$. (We omit unnecessary brackets, writing for instance $\text{AC}^i[2, 3]$ instead of $\text{AC}^i[\{2, 3\}]$.)
- $\text{ACC}^i = \bigcup_m \text{AC}^i[m]$.
- TC^i is the class of languages accepted by Dlogtime-uniform circuit families of size $n^{O(1)}$ and depth $O(\log^i n)$, consisting of unbounded fan-in MAJORITY gates, and NOT gates.
- SAC^i is the class of languages accepted by Dlogtime-uniform circuit families of polynomial size and depth $O(\log^i n)$, consisting of unbounded fan-in OR

gates and bounded fan-in AND gates, along with NOT gates at (some of) the leaves.

Note that the restriction that NOT gates appear only at the leaves in SAC^i circuits is essential; if NOT gates could appear everywhere, then these classes would coincide with AC^i . Similarly, note that we do not bother to define $\text{SAC}^i[m]$, since a MOD_m gate with a single input is equivalent to a NOT gate, and thus $\text{SAC}^i[m]$ would be the same as $\text{AC}^i[m]$.

The algebraic complexity classes $\text{VP}(R)$ for various algebraic structures R were originally defined [15] as classes of families of n -variate polynomials of degree $n^{O(1)}$ that can be represented by polynomial-size (*nonuniform*) arithmetic circuits over R . Here, we let $\text{VP}(R)$ denote the corresponding *uniform* class, and recall that the $\log^2 n$ depth bound of [16] can be made logarithmic:

Theorem 1. [2] *For any commutative semiring R , $\text{VP}(R)$ coincides with the class of families of polynomials over R represented by logspace-uniform circuit families of polynomial size and logarithmic depth with unbounded fan-in $+$ gates, and fan-in $2 \times$ gates.*

Note that over \mathbb{F}_p , many different polynomials yield the same function. For example, since $x^3 = x$ in \mathbb{F}_3 , every function on n variables has a polynomial of degree at most $2n$. Very likely there are functions represented by polynomials in $\text{VP}(\mathbb{F}_3)$ of degree, say, n^5 , but not by any VP polynomial of degree $2n$. On the other hand, there is a case to be made for focusing on the *functions* in these classes, rather than focusing on the *polynomials* that represent those functions. For instance, if the Immerman-Landau conjecture is true, and TC^1 is reducible to problems in $\text{VP}(\mathbb{Q})$, it would suffice for every *function* in $\text{TC}^1 = \#\text{AC}^1(\mathbb{F}_{p_n})$ to have a representation in $\text{VP}(\mathbb{Q})$, even though the *polynomials* represented by $\#\text{AC}^1(\mathbb{F}_{p_n})$ circuits have large degree, and thus cannot be in any VP class.

In the literature on VP classes, one standard way to focus on the *functions* represented by polynomials in VP is to consider what is called the *Boolean Part* of $\text{VP}(R)$, which is the set of *languages* $A \subseteq \{0, 1\}^*$ such that, for some sequence of polynomials (Q_n) , for $x \in A$ we have $Q_{|x|}(x) = 1$, and for $x \in \{0, 1\}^*$ such that $x \notin A$ we have $Q_{|x|}(x) = 0$.

When the algebra R is a finite field, considering the Boolean part of $\text{VP}(R)$ captures the relevant complexity aspects, since the computation of any function represented by a polynomial in $\text{VP}(R)$ (with inputs and outputs coming from R) is logspace-Turing reducible to some language in the Boolean Part of $\text{VP}(R)$.

In this paper, we are concerned exclusively with the “Boolean Part” of arithmetic classes. For notational convenience, we refer to these classes using the “VP” notation, rather than constantly repeating the phrase “Boolean Part”.

Following the naming conventions of [18], for any Boolean circuit complexity class \mathcal{C} defined in terms of circuits with AND and OR gates, we define $\#\mathcal{C}(R)$ to be the class of functions represented by arithmetic circuits defined over the algebra R , where AND is replaced by \times , and OR is replaced by $+$ (and NOT gates at the leaves are applied to the $\{0, 1\}$ inputs). (The classes $\#\text{P}$, $\#\text{L}$, and $\#\text{LogCFL}$ also fit into this naming scheme, using established connections between

Turing machines and circuits.) In particular, we will be concerned with the following two classes:

Definition 2. *Let R be any suitable semiring.³ Then*

- $\#\text{AC}^1(R)$ is the class of functions $f : \{0, 1\}^* \rightarrow R$ given by families of logspace-uniform circuits of unbounded fan-in $+$ and \times gates having depth $O(\log n)$ and size $n^{O(1)}$.
- $\#\text{SAC}^1(R)$ is the class of functions $f : \{0, 1\}^* \rightarrow R$ represented by families of logspace-uniform circuits of unbounded fan-in $+$ gates and \times gates of fan-in two, having depth $O(\log n)$ and polynomial size.

Input variables may be negated. Where no confusion will result, the notation $\#\mathcal{C}(R)$ will also be used to refer to the related class of languages.

Hence from Theorem 1 we see that $\text{VP}(\mathbb{F}_p) = \#\text{SAC}^1(\mathbb{F}_p)$ for any prime p .

Now we introduce classes that are dual to the $\#\text{SAC}^1(R)$ classes. Define $\#\text{SAC}^{1,*}(R)$ to be the class of functions $f : \{0, 1\}^* \rightarrow R$ represented by families of logspace-uniform circuits of unbounded fan-in \times gates and $+$ gates of fan-in two, having depth $O(\log n)$ and size $n^{O(1)}$. Because of the connection between VP and $\#\text{SAC}^1$, we use the convenient notation $\text{AP}(R)$ to denote the dual notation, rather than the more cumbersome $\#\text{SAC}^{1,*}(R)$.

Of course, the set of formal polynomials represented by AP circuits is not contained in any VP class, because AP contains polynomials of degree $n^{O(\log n)}$. However, as discussed in the previous section, we are considering the “Boolean Part” of these classes. More formally:

Definition 3. *Let p be a prime power. $\text{AP}(\mathbb{F}_p)$ is the class of all languages $A \subseteq \{0, 1\}^*$ with the property that there is a logspace-uniform family of circuits $\{C_n : n \in \mathbb{N}\}$, each of depth $O(\log n)$ consisting of input gates, $+$ gates of fan-in two, and \times gates of unbounded fan-in, such that for each string x of length n , x is in A if and only if $C_n(x)$ evaluates to 1, when the $+$ and \times gates are evaluated over \mathbb{F}_p . Furthermore, if $x \notin A$, then $C_n(x)$ evaluates to 0.*

Another way of relating arithmetic classes (such as VP and AP) to complexity classes of languages would be to consider the languages that are logspace-Turing reducible to the polynomials in $\text{VP}(R)$ or $\text{AP}(R)$, via a machine M with a polynomial p as an oracle, which obtains the value of $p(x_1, \dots, x_n)$ when M writes x_1, \dots, x_n on a query tape. It is worth mentioning that (the Boolean parts of) both $\text{VP}(\mathbb{F}_p)$ and $\text{AP}(\mathbb{F}_p)$ are closed under logspace-Turing reductions, although this is still open for classes over \mathbb{Z}_m when m is not prime.

³ Our primary focus in this paper is on *finite* semirings, as well as countable semirings such as \mathbb{Q} , where we use the standard binary representation of constants (say, as a numerator and denominator) when a logspace uniformity machine makes use of constants in the description of a circuit. It is not clear to us which definition would be most useful in describing a class such as $\#\text{AC}^1(\mathbb{R})$, and so for now we consider such semirings to be “unsuitable”.

Proposition 1. $AP(\mathbb{F}_p) = L^{AP(\mathbb{F}_p)}$ and $VP(\mathbb{F}_p) = L^{VP(\mathbb{F}_p)}$

(Proofs are omitted; see [3].) VP over fields of the same characteristic yield the same class of languages.

Proposition 2. Let p be a prime, and let $k \geq 1$. Then $VP(\mathbb{F}_p) = VP(\mathbb{F}_{p^k})$.

It is also appropriate to use the VP and AP notation when referring to the classes defined by Boolean semiunbounded fan-in circuits with negation gates allowed at the inputs. With this notation, $VP(B_2)$ corresponds to the Boolean class SAC^1 , and $AP(B_2)$ corresponds to the complement of SAC^1 (with bounded fan-in OR gates, and unbounded fan-in AND gates). It has been shown by [6] that SAC^1 is closed under complement. Thus we close this section with the equality that serves as a springboard for investigating the AP classes.

Theorem 2. [6] $VP(B_2) = AP(B_2) (= SAC^1 = LogCFL)$.

We believe $VP(\mathbb{F}_p) \neq AP(\mathbb{F}_p)$ for every prime p ; see Section 5.

3 Subclasses of ACC^1

In this section, we present our characterizations of ACC^1 in terms of $AP(\mathbb{F}_p)$.

Theorem 3. For any prime p and any $k \geq 1$, $AP(\mathbb{F}_{p^k}) = AC^1[Supp(p^k - 1)]$. (Recall that $Supp(m)$ is defined in Definition 1.)

Corollary 1. $ACC^1 = \bigcup_p AP(\mathbb{F}_p)$.

Note also that several of the $AP(\mathbb{F}_p)$ classes coincide. This is neither known nor believed to happen with the $VP(\mathbb{F}_p)$ classes. Augmenting the $AP(\mathbb{F}_p)$ classes with unbounded fan-in addition gates increases their computation power only by adding MOD_p gates, as the following theorem demonstrates.

Theorem 4. For each prime p and each $k \geq 1$, $\#AC^1(\mathbb{F}_{p^k}) = AC^1[\{p\} \cup Supp(p^k - 1)]$.

Corollary 2. $ACC^1 = \bigcup_p AP(\mathbb{F}_p) = \bigcup_p \#AC^1(\mathbb{F}_p) = \bigcup_m \#AC^1(\mathbb{Z}_m)$.

Corollary 3. For any prime p there is a prime q such that $\#AC^1(\mathbb{F}_p) \subseteq AP(\mathbb{F}_q)$.

$VP(\mathbb{F}_p)$ also has a simple characterization in terms of Boolean circuits. For this, we need a more general definition:

Definition 4. Let $m \in \mathbb{N}$, and let g be any function on \mathbb{N} . Define $g-AC^1[m]$ to be the class of languages with logspace-uniform circuits of polynomial size and depth $O(\log n)$, consisting of unbounded-fan-in MOD_m gates, along with AND gates of fan-in $O(g(n))$. Clearly $g-AC^1[m] \subseteq AC^1[m]$.

Observe that, since a MOD_m gate can simulate a NOT gate, $g-AC^1[m]$ remains the same if OR gates of fan-in $O(g)$ are also allowed.

Corollary 4. For every prime p , $\text{VP}(\mathbb{F}_p) = 2\text{-AC}^1[p] \subseteq \text{AC}^1[p]$.

We remark that the same proof shows that, for any $m \in \mathbb{N}$, $\text{VP}(\mathbb{Z}_m) \subseteq 2\text{-AC}^1[m]$. However, the converse inclusion is not known, unless m is prime. We remark that the proofs of Theorems 3 and 4 carry over also for depths other than $\log n$. (Related results for constant-depth unbounded-fan-in circuits can be found already in [14,1].)

Corollary 5. For any prime p , $\#\text{SAC}^{i,*}(\mathbb{F}_p) = \text{AC}^i[\text{Supp}(p-1)]$ and $\#\text{AC}^i(\mathbb{F}_p) = \text{AC}^i[p \cup \text{Supp}(p-1)]$.

3.1 Comparing AP and VP .

How do the AP and VP classes compare to each other? As a consequence of Corollary 4 and Theorem 3, $\text{VP}(\mathbb{F}_p) \subseteq \text{AP}(\mathbb{F}_q)$ whenever p divides $q-1$. In particular, $\text{VP}(\mathbb{F}_2) \subseteq \text{AP}(\mathbb{F}_q)$ for any prime $q > 2$. No inclusion of any AP class in any VP class is known unconditionally, although $\text{AP}(B_2)(= \text{SAC}^1)$ is contained in every $\text{VP}(\mathbb{F}_p)$ class in the nonuniform setting [9,13], and this holds also in the uniform setting under a plausible derandomization hypothesis [4].

No $\text{AP}(\mathbb{F}_q)$ class can be contained in $\text{VP}(\mathbb{F}_p)$ unless $\text{AC}^1 \subseteq \text{VP}(\mathbb{F}_p)$, since $\text{AC}^1 = \text{AP}(\mathbb{F}_2) \subseteq \text{AP}(\mathbb{F}_3) \subseteq \text{AP}(\mathbb{F}_q)$ for every prime $q \geq 3$. AC^1 is not known to be contained in any VP class.

4 Threshold circuits and small degree

The inspiration for the results in this section comes from the following theorem of Reif and Tate [12] (as re-stated by Buhrman et al. [7]):

Theorem 5. $\text{TC}^1 = \#\text{AC}^1(\mathbb{F}_{p_n})$.

Here, the class $\#\text{AC}^1(\mathbb{F}_{p_n})$ consists of the languages whose (Boolean) characteristic functions are computed by logspace-uniform families of arithmetic circuits of logarithmic depth with unbounded fan-in $+$ and \times gates, where the arithmetic operations of the circuit C_n are interpreted over \mathbb{F}_{p_n} , where p_1, p_2, p_3, \dots is the sequence of all primes $2, 3, 5, \dots$. That is, circuits for inputs of length n use the n -th prime to define the algebraic structure.

This class is closed under logspace-Turing reductions – but when we consider *other* circuit complexity classes defined using \mathbb{F}_{p_n} , it is *not* clear that these other classes are closed.

As an important example, we mention $\text{VP}(\mathbb{F}_{p_n})$. As we show below, this class has an important connection to $\text{VP}(\mathbb{Q})$, which is perhaps the canonical example of a VP class. Vinay [17] proved that $\text{VP}(\mathbb{Q})$ has essentially the same computational power as $\#\text{LogCFL}$ (which counts among its complete problems the problem of determining how many distinct parse trees a string x has in a certain context-free language). Here, we mention one more alternative characterization of the computational power of $\text{VP}(\mathbb{Q})$.

Proposition 3. $\mathsf{LVP}(\mathbb{F}_{p_n}) = \mathsf{LVP}(\mathbb{Q}) = \mathsf{L}\#\mathsf{LogCFL}$.

With arithmetic circuits of superpolynomial algebraic degree (such as AP), evaluating the circuits over \mathbb{Z} produces output that needs a superpolynomial number of bits to express in binary. Thus, when we consider such classes, it will always be in the context of structures (such as \mathbb{F}_{p_n}) where the output can be represented in a polynomial number of bits.

Our first new result in this section, is to improve Theorem 5. (Recall the definition of $g\text{-AC}^1[m]$ from Definition 4.)

Theorem 6. $\mathsf{TC}^1 = \#\mathsf{AC}^1(\mathbb{F}_{p_n}) = \mathsf{LAP}(\mathbb{F}_{p_n}) = \mathsf{AC}^1[p_n] = 0\text{-AC}^1[p_n]$.

We also mention that Theorem 6 generalizes to other depths, in a way analogous to Corollary 5:

Corollary 6. $\mathsf{TC}^i = \#\mathsf{AC}^i(\mathbb{F}_{p_n}) = \mathsf{AC}^i[p_n] = 0\text{-AC}^i[p_n]$.

For $i \geq 1$ the equality $\mathsf{TC}^i = \mathsf{L}\#\mathsf{SAC}^{i,*}(\mathbb{F}_{p_n})$ also holds, but for $i = 0$ a more careful argument is needed, using AC^0 -Turing reducibility in place of logspace-Turing reducibility.

In order to set the context for the results of the next section, it is necessary to consider an extension of Theorem 6, involving arithmetic circuits over certain *rings*. Thus we require the following definition.

Definition 5. *Let (m_n) be any sequence of natural numbers (where each $m_n > 1$) such that the mapping $1^n \mapsto m_n$ is computable in logspace. We use the notation $\#\mathsf{AC}^1(\mathbb{Z}_{m_n})$ to denote the class of functions f with domain $\{0, 1\}^*$ such that there is a logspace-uniform family of arithmetic circuits $\{C_n\}$ of logarithmic depth with unbounded fan-in $+$ and \times gates, where the arithmetic operations of the circuit C_n are interpreted over \mathbb{Z}_{m_n} , and for any input x of length n , $f(x) = C_n(x)$. We use the notation $\#\mathsf{AC}^1(\mathbb{Z}_{\mathsf{L}})$ to denote the union, over all logspace-computable sequences of moduli (m_n) , of $\#\mathsf{AC}^1(\mathbb{Z}_{m_n})$.*

Since the sequence of primes (p_n) is logspace-computable, we have that TC^1 ($= \#\mathsf{AC}^1(\mathbb{F}_{p_n})$) is clearly contained in $\#\mathsf{AC}^1(\mathbb{Z}_{\mathsf{L}})$. Conversely, each function in $\#\mathsf{AC}^1(\mathbb{Z}_{\mathsf{L}})$ is in TC^1 . Thus, arithmetic circuits over the integers mod m_n for reasonable sequences of moduli m_n give yet another arithmetic characterization of TC^1 .

4.1 Degree Reduction

In this subsection, we introduce a class of circuits that is intermediate between the unbounded fan-in circuit model and the semiunbounded fan-in model, for the purposes of investigating when arithmetic circuits of superpolynomial algebraic degree can be simulated by arithmetic circuits (possibly over a different algebra) with much smaller algebraic degree.

The starting point for this subsection is Theorem 4.3 in [2], which states that every problem in AC^1 is reducible to a function computable by polynomial-size

arithmetic circuits of degree $n^{O(\log \log n)}$. In this section, we refine that result and put it in context with the theorems about TC^1 that were presented in the previous subsection. Those results show that TC^1 reduces to semiunbounded fan-in arithmetic circuits in the $\text{AP}(\mathbb{F}_{p_n})$ model, but leave open the question of whether TC^1 also reduces to semiunbounded fan-in arithmetic circuits in the $\text{VP}(\mathbb{F}_{p_n})$ model (which coincides with $\text{VP}(\mathbb{Q})$). We are unable to answer this question, but we show some interesting inclusions occur if we relax the VP model, by imposing a less-stringent restriction on the fan-in of the \times gates.

Definition 6. *Let (m_n) be any sequence of natural numbers (where each $m_n > 1$) such that the mapping $1^n \mapsto m_n$ is computable in L . $\#\text{WSAC}^1(\mathbb{Z}_{m_n})$ is the class of functions represented by logspace-uniform arithmetic circuits $\{C_n\}$, where C_n is interpreted over \mathbb{Z}_{m_n} , where each C_n has size polynomial in n , and depth $O(\log n)$, and where the $+$ gates have unbounded fan-in, and the \times gates have fan-in $O(\log n)$. Thus these circuits are not semiunbounded, but have a “weak” form of the semiunbounded fan-in restriction. We use the notation $\#\text{WSAC}^1(\mathbb{Z}_{\perp})$ to denote the union, over all logspace-computable sequences of moduli (m_n) , of $\#\text{WSAC}^1(\mathbb{Z}_{m_n})$. In the special case when $m_n = p$ for all n , we obtain the class $\#\text{WSAC}^1(\mathbb{F}_p)$.*

We refrain from defining a weakly semiunbounded analog of AP , because it would coincide with AP , since AC^0 circuits can add $O(\log n)$ numbers.

We improve on [2, Theorem 4.3] by showing that AC^1 is contained in the class $\#\text{WSAC}^1(\mathbb{F}_2)$; note that all polynomials in $\#\text{WSAC}^1(\mathbb{F}_p)$ have degree $n^{O(\log \log n)}$, and note also that the class of functions considered in [2] is not obviously even in TC^1 . In addition, we improve on [2] by reducing not merely AC^1 , but also $\text{AC}^1[p]$ for any prime p . Also, we obtain an equality.

Theorem 7. *Let p be any prime. Then $\text{AC}^1[p] = \#\text{WSAC}^1(\mathbb{F}_p)$.*

We especially call attention to the following corollary, which shows that, over \mathbb{F}_2 , polynomial size logarithmic depth arithmetic circuits of degree $n^{O(\log n)}$ and of degree $n^{O(\log \log n)}$ represent precisely the same functions!

Corollary 7. $\#\text{WSAC}^1(\mathbb{F}_2) = \#\text{AC}^1(\mathbb{F}_2) = \text{AC}^1[2] = \text{AP}(\mathbb{F}_3)$.

If we focus on the Boolean classes, rather than on the arithmetic classes, then we obtain a remarkable collapse.

Theorem 8. *Let $m \in \mathbb{N}$. Then $\text{AC}^1[m] = \text{log-AC}^1[m]$.*

It follows that arithmetic AC^1 circuits over *any* finite field \mathbb{F}_p can be simulated by Boolean circuits with MOD gates and small fan-in AND gates. It remains open whether this in turn leads to small-degree arithmetic circuits over \mathbb{F}_p when $p > 2$, and also whether the fan-in of the AND gates can be sublogarithmic, without loss of power.

When m is composite, Theorem 8 can be improved to obtain an even more striking collapse, by invoking the work of Hansen and Koucký [10].

Theorem 9. *Let m not be a prime power. Then $\text{AC}^1[m] = 2\text{-AC}^1[m]$.*

Corollary 8. $\text{ACC}^1 = \bigcup_p \text{AP}(\mathbb{F}_p) = \bigcup_p \#\text{AC}^1(\mathbb{F}_p) = \bigcup_m \#\text{AC}^1(\mathbb{Z}_m) = \bigcup_m 2\text{-AC}^1[m]$.

It might be useful to have additional examples of algebras, where some degree reduction can be accomplished. Thus we also offer the following theorem:

Theorem 10. *Let p be any prime. Then $\text{AC}^1[p] \subseteq \text{L}\#\text{WSAC}^1(\mathbb{Z}_L)$.*

Using Theorems 3 and 4 we obtain the following.

Corollary 9. *If p is a Fermat prime, then $\text{AP}(\mathbb{F}_p) \subseteq \text{L}\#\text{WSAC}^1(\mathbb{Z}_L)$.*

5 Conclusions, Discussion, and Open Problems

We have introduced the complexity classes $\text{AP}(R)$ for various algebraic structures R , and have shown that they provide alternative characterizations of well-known complexity classes. Furthermore, we have shown that arithmetic circuit complexity classes corresponding to polynomials of degree $n^{O(\log \log n)}$ also yield new characterizations of complexity classes, such as the equality $\text{AC}^1[p] = \log\text{-AC}^1[p] = \#\text{WSAC}^1(\mathbb{F}_p)$. In the case when $p = 2$, we additionally obtain $\#\text{AC}^1(\mathbb{F}_2) = \text{AC}^1[2] = \log\text{-AC}^1[2] = \#\text{WSAC}^1(\mathbb{F}_2)$, showing that algebraic degree $n^{O(\log n)}$ and $n^{O(\log \log n)}$ have equivalent expressive power, in this setting.

We have obtained new characterizations of ACC^1 in terms of restricted fan-in: $\text{ACC}^1 = \bigcup_p \#\text{AC}^1(\mathbb{F}_p) = \bigcup_p \text{AP}(\mathbb{F}_p) = \bigcup_m 2\text{-AC}^1[m]$. That is, although ACC^1 corresponds to unbounded fan-in arithmetic circuits of logarithmic depth, and to unbounded fan-in Boolean circuits with modular counting gates, no power is lost if the addition gates have bounded fan-in (in the arithmetic case) or if only the modular counting gates have unbounded fan-in (in the Boolean case). It remains unknown if every problem in ACC^1 is reducible to a problem in $\bigcup_m \text{VP}(\mathbb{Z}_m)$, although we believe that our theorems suggest that this is likely. It would be highly interesting to see such a connection between ACC^1 and VP .

We believe that it is fairly likely that several of our theorems can be improved. For instance:

- * Perhaps Theorems 8 and 9 can be improved, to show that for all m , $\text{AC}^1[m] = 2\text{-AC}^1[m]$. Note that this is known to hold if m is not a prime power. By Corollary 4 this would show that $\text{VP}(\mathbb{F}_p) = \text{AC}^1[p]$ for all primes p . It would also show that $\#\text{AC}^1(\mathbb{F}_2) = \text{VP}(\mathbb{F}_2) = \text{AP}(\mathbb{F}_p)$ for every Fermat prime p . (We should point out that this would imply that $\text{AC}^1 \subseteq \text{VP}(\mathbb{F}_p)$ for every prime p , whereas even the weaker inclusion $\text{SAC}^1 \subseteq \text{VP}(\mathbb{F}_p)$ is only known to hold non-uniformly [9].)

- * Can Corollary 9 be improved to hold for all primes p , or even for $\text{AP}(\mathbb{F}_{p^n})$?

The latter improvement would show that $\text{TC}^1 \subseteq \text{L}\#\text{WSAC}^1(\mathbb{Z}_L)$.

- * Perhaps one can improve Theorem 10, to achieve a simulation of degree $n^{O(1)}$. Why should $n^{O(\log \log n)}$ be optimal? Perhaps this could also be improved to hold for composite moduli?

Note that if some combinations of the preceding improvements are possible, TC^1 would reduce to $\text{VP}(\mathbb{Q})$, which would be a significant step toward the Immerman-Landau conjecture.

It appears as if $\text{VP}(\mathbb{F}_p)$ and $\text{AP}(\mathbb{F}_p)$ are incomparable for every non-Fermat prime $p > 2$, since $\text{VP}(\mathbb{F}_p) = 2\text{-AC}^1[p]$ and $\text{AP}(\mathbb{F}_p) = 2\text{-AC}^1[\text{Supp}(p-1)]$, involving completely different sets of primes. For Fermat primes we have $\text{AP}(\mathbb{F}_p) = \text{log-AC}^1[2]$ and again the VP and AP classes seem incomparable. When $p = 2$, we have $\text{VP}(\mathbb{F}_2) = 2\text{-AC}^1[2]$ and $\text{AP}(\mathbb{F}_2) = \text{AC}^1$; if $\text{VP}(\mathbb{F}_2) = \text{AC}^1[2]$ (which may be possible), then it would appear that the VP class could be *more* powerful than the AP class. But based on current knowledge it also appears possible that the VP and AP classes are incomparable even for $p = 2$.

Some of our theorems overcome various hurdles that would appear to stand in the way of a proof of our conjecture that $\text{ACC}^1 = \bigcup_m \text{LVP}(\mathbb{F}_{Z_m})$. First, recall that $\text{VP}(\mathbb{Z}_m) \subseteq 2\text{-AC}^1[m]$ (Corollary 4). Thus, if the conjecture is correct, then *unbounded* fan-in AND and OR gates would have to be simulated efficiently with *bounded* fan-in gates. But this is true in this context: $\text{AC}^1[m] = 2\text{-AC}^1[m]$, if m is not a prime power (Theorem 9). If m is a prime power, then the fan-in can be reduced to $\log n$ (Theorem 8). If the fan-in can be reduced to $O(1)$ also in the case of prime power moduli, or if ACC^1 circuits with *bounded* fan-in AND and OR (which have the full power of ACC^1 , by Corollary 8) can be simulated by $\text{VP}(\mathbb{Z}_m)$ circuits, then the conjecture holds. (The latter simulation is possible if the MOD gates in the ACC^1 circuits are for a prime modulus; see Corollary 4.)

A second objection that might be raised against the conjecture deals with algebraic degree. ACC^1 corresponds precisely to polynomial-size logarithmic depth unbounded fan-in arithmetic circuits over finite fields (Corollary 2). Such circuits represent polynomials of degree $n^{O(\log n)}$, whereas VP circuits represent polynomials of degree only $n^{O(1)}$. One might assume that there are languages represented by polynomial-size log-depth arithmetic circuits of degree $n^{O(\log n)}$ that actually *require* such large degree in order to be represented by arithmetic circuits of small size and depth.

Our degree-reduction theorem (Corollary 7) shows that this assumption is incorrect. Every Boolean function that can be represented by an arithmetic AC^1 circuit over \mathbb{F}_2 (with algebraic degree $n^{O(\log n)}$) can be represented by an arithmetic AC^1 circuit over \mathbb{F}_2 where the multiplication gates have fan-in $O(\log n)$ (and thus the arithmetic circuit has algebraic degree $n^{O(\log \log n)}$).

References

1. M. Agrawal, E. Allender, and S. Datta. On TC^0 , AC^0 , and arithmetic circuits. *Journal of Computer and System Sciences*, 60:395–421, 2000.
2. Allender, J. Jiao, M. Mahajan, and V. Vinay. Non-commutative arithmetic circuits: Depth reduction and size lower bounds. *Theoret. Comp. Sci.*, 209:47–86, 1998.
3. E. Allender, A. Gál, and I. Mertz. Dual VP classes. *ECCC*, 2014. TR14-122.
4. E. Allender, K. Reinhardt, and S. Zhou. Isolation, matching, and counting: Uniform and nonuniform upper bounds. *Journal of Computer and System Sciences*, 59(2):164–181, 1999.

5. L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation*. Springer, 1998.
6. A. Borodin, S. A. Cook, P. W. Dymond, W. L. Ruzzo, and M. Tompa. Two applications of inductive counting for complementation problems. *SIAM Journal on Computing*, 18:559–578, 1989. See Erratum in *SIAM J. Comput.* 18, 1283.
7. H. Buhrman, R. Cleve, M. Koucký, B. Loff, and F. Speelman. Computing with a full memory: catalytic space. In *STOC*, pages 857–866, 2014.
8. C. Corrales-Rodríguez and R. Schoof. The support problem and its elliptic analogue. *Journal of Number Theory*, 64(2):276–290, 1997.
9. A. Gál and A. Wigderson. Boolean complexity classes vs. their arithmetic analogs. *Random Struct. Algorithms*, 9(1-2):99–111, 1996.
10. K. Arnsfelt Hansen and M. Koucký. A new characterization of ACC^0 and probabilistic CC^0 . *Computational Complexity*, 19(2):211–234, 2010.
11. N. Immerman and S. Landau. The complexity of iterated multiplication. *Information and Computation*, 116:103–116, 1995.
12. J. Reif and S. Tate. On threshold circuits and polynomial computation. *SIAM Journal on Computing*, 21:896–908, 1992.
13. K. Reinhardt and E. Allender. Making nondeterminism unambiguous. *SIAM Journal on Computing*, 29:1118–1131, 2000.
14. R. Smolensky. Algebraic methods in the theory of lower bounds for Boolean circuit complexity. In *STOC*, pages 77–82, 1987.
15. L.G. Valiant. Completeness classes in algebra. In *Proc. 11th ACM STOC*, pages 249–261, 1979.
16. L.G. Valiant, S. Skyum, S. Berkowitz, and C. Rackoff. Fast parallel computation of polynomials using few processors. *SIAM J. Comput.*, 12(4):641–644, 1983.
17. V Vinay. Counting auxiliary pushdown automata and semi-unbounded arithmetic circuits. In *Proceedings of 6th Structure in Complexity Theory Conference*, pages 270–284, 1991.
18. H. Vollmer. *Intro. to Circuit Complexity: A Uniform Approach*. Springer, 1999.

Acknowledgments

The first and third authors acknowledge the support of NSF grants CCF-0832787 and CCF-1064785. The second author was supported in part by NSF grant CCF-1018060. We also acknowledge stimulating conversations with Meena Mahajan, which occurred at the 2014 Dagstuhl Workshop on the Complexity of Discrete Problems (Dagstuhl Seminar 14121), and illuminating conversations with Stephen Fenner and Michal Koucký, which occurred at the 2014 Dagstuhl Workshop on Algebra in Computational Complexity (Dagstuhl Seminar 14391). We also thank Igor Shparlinski and our Rutgers colleagues Richard Bumby, John Miller and Steve Miller, for helpful pointers to the literature, as well as helpful feedback from Pascal Koiran and Russell Impagliazzo.