Electronic Colloquium on Computational Complexity, Report No. 10 (1999)

A Lower Bound for Primality

ERIC ALLENDER*
Dept. of Computer Science
Rutgers University
Piscataway, NJ, USA
allender@cs.rutgers.edu

MICHAEL SAKS†
Mathematics Dept.
Rutgers University
Piscataway, NJ, USA
saks@math.rutgers.edu

IGOR SHPARLINSKI‡
School of Mathematics, Physics, Computing and Electronics
Macquarie University
NSW 2109, Australia
igor@mpce.mq.edu.au

Abstract

Recent work by Bernasconi, Damm and Shparlinski proved lower bounds on the circuit complexity of the square-free numbers, and raised as an open question if similar (or stronger) lower bounds could be proved for the set of prime numbers. In this short note, we answer this question affirmatively, by showing that the set of prime numbers (represented in the usual binary notation) is not contained in $AC^0[p]$ for any prime p. Similar lower bounds are presented for the set of square-free numbers, and for the problem of computing the greatest common divisor of two numbers.

1 Introduction

What is the computational complexity of the set of prime numbers? There is a large body of work presenting important upper bounds on the complexity of the set of primes (including [AH87, APR83, Mil76, R80, SS77]), but –

*Supported in part by NSF grant CCR-9734918.
†Supported in part by NSF grant CCR-9700239.
‡Supported in part by ARC grant A69700294.

ISSN 1433-8092

as was pointed out recently in [BDS98a, BDS98b, BS99, Shp98], other than the work of [Med91, Man92] almost nothing has been published regarding *lower bounds* on the complexity of this set. To be sure, in the context of space-bounded computation, it was shown in [HS68] that at least logarithmic space is required, in order to determine if a number is prime. This was improved in [HB76] to show that the same bound holds even if the number is presented in *unary* (and logarithmic space is sufficient in that case). However, these bounds do not address circuit complexity at all; note for instance that the *unary* encoding of prime numbers has trivial circuit complexity. Prior to the current work, it was not even known if primality testing could be accomplished by constant-depth, polynomial-size circuits of AND and OR gates. That is, it was not known if PRIMES was in AC^0.

In this note, we resolve this question, and in fact give a lower bound for primality that is essentially as strong as is known for any problem in NP. More precisely, we show that, for any prime p, PRIMES is not in $\mathrm{AC}^0[p]$. Our technique actually yields an exponential lower bound on circuit size for this class of circuits. In order to simplify the exposition, we present only the superpolynomial lower bound in this note.

Our proof applies equally well to other number-theoretic problems, such as SQUARE-FREE and GCD. It follows from the results of [BDS98a, BDS98b, BS99] that these problems do not belong to AC^0. Here we extend their results to the more powerful complexity classes $\mathrm{AC}^0[p]$. After presenting our definitions and the proofs of our main results, we close the paper with a discussion of related open problems.

2 Preliminaries

A *circuit family* is a set $\{C_n : n \in \mathbb{N}\}$ where each C_n is an acyclic circuit with n Boolean inputs $x_1, \ldots, x_n$ (as well as the constants 0 and 1 allowed as inputs) and some number of output gates $y_1, \ldots, y_r$. $\{C_n\}$ has *size* $s(n)$ if each circuit C_n has at most $s(n)$ gates; it has *depth* $d(n)$ if the length of the longest path from input to output in C_n is at most $d(n)$.

A function f is said to be in AC^0 if there is a circuit family $\{C_n\}$ of size $n^{O(1)}$ and depth $O(1)$ consisting of unbounded fan-in AND and OR and NOT gates such that for each input x of length n, the output of C_n on input x is $f(x)$.

It has been known since [Aj83, FSS84] that the parity function is not in AC^0. This led researchers to consider the power of AC^0 circuits that were

augmented with parity gates, and more generally with MOD_m gates.

Definition 1 *Let $m \in \mathbb{N}$. The Boolean MOD_m function is defined as*

$$\text{MOD}_m(x) = \begin{cases} 1 & \text{if } \sum_i x_i \equiv 0 \pmod m \\ 0 & \text{otherwise.} \end{cases}$$

where the input string x consists of the bits $x_n x_{n-1} \ldots x_1 x_0$.

A function f is said to be in $\text{AC}^0[p]$ if there is a circuit family $\{C_n\}$ of size $n^{O(1)}$ and depth $O(1)$ consisting of unbounded fan-in AND, OR, MOD_p, and NOT gates such that for each input x of length n, the output of C_n on input x is $f(x)$.

A language (i.e., a subset of $\{0,1\}^*$) is said to be in AC^0 (or $\text{AC}^0[p]$) if its characteristic function is in AC^0 ($\text{AC}^0[p]$, respectively).

2.1 Some Known Lower Bounds

Our lower bounds follow from the following result of Smolensky [Smo87], which builds on an earlier result of Razborov [Raz87].

Theorem 1 *[Smolensky] Let p be a prime, and let m not be a power of p. Then MOD_m is not in $AC^0[p]$.*

(In fact, [Smo87] provides an exponential lower bound on the size of $\text{AC}^0[p]$ circuits computing MOD_m.)

It is a curious fact that the proof given in [Smo87] relies heavily on the fact that the modulus p is prime. Amazingly, it remains unknown if there is any language in $\text{NTIME}(2^n)$ whose characteristic function cannot be computed by linear-size, depth three circuits of MOD_6 gates.

A problem that is very closely related to MOD_m is the problem of determining if a number (represented in the usual binary notation) is a multiple of m. Stated another way:

Definition 2

$$\text{MULT}_m(x) = \begin{cases} 1 & \text{if } \sum_i x_i 2^i \equiv 0 \pmod m \\ 0 & \text{otherwise.} \end{cases}$$

where the input string x consists of the bits $x_n x_{n-1} \ldots x_1 x_0$.

The Smolensky lower bound can be used to obtain a lower bound on the MULT_m function. This argument is presented most easily by making use of the notion of circuit-based reducibility between problems.

2.2 Reducibility

A language A_1 is $\leq_m^{AC^0}$ *reducible* to a language A_2, if there is a function f in AC^0 such that, for all x, $x \in A_1$ if and only if $f(x) \in A_2$.

A_1 is $\leq_T^{AC^0}$ *reducible* to A_2, if A_1 is recognized by a family of circuits of polynomial size and constant depth, consisting of NOT gates, unbounded fan-in AND and OR gates, and oracle gates for A_2. (An oracle gate for A_2 takes m inputs $x_1, \ldots, x_m$ and outputs 1 if $x_1 \ldots x_m$ is in A_2, and outputs 0 otherwise.)

The following proposition is well-known, and provides the motivation for considering these reducibilities.

Proposition 2 *$AC^0[p]$ is closed under $\leq_{\mathrm{T}}^{AC^0}$ reducibility. That is, if*

$$A \leq_{\mathrm{T}}^{AC^0} B \qquad \text{and} \qquad B \in AC^0[p]$$

then $A \in AC^0[p]$.

Note that $\leq_{\mathrm{m}}^{\mathrm{AC}^0}$ reducibility is more restrictive than $\leq_{\mathrm{T}}^{\mathrm{AC}^0}$ reducibility. The motivation for studying $\leq_{\mathrm{m}}^{\mathrm{AC}^0}$ reducibility stems from the fact that most computational problems that arise in practice turn out to be complete for some well-known complexity class under $\leq_{\mathrm{m}}^{\mathrm{AC}^0}$ reducibility. Number-theoretic problems such as the ones considered in this paper run counter to this trend. Almost no number-theoretic problems are known to be complete for natural complexity classes.

The following fact is a slight generalization of an observation of Boppana and Lagarias [BL87].

Theorem 3 *Let $m \in \mathbb{N}$ be odd. Then $\textsc{Mod}_m \leq_{\mathrm{m}}^{AC^0} \textsc{Mult}_m$.*

Proof: Note that it is important that m be odd. If $m = 2$, the conclusion is easily seen to fail.

Since m is odd, there is some integer exponent $t > 0$ such that $2^t \equiv 1 \pmod{m}$. Our AC^0 reduction from $\textsc{Mod}_m$ to $\textsc{Mult}_m$ makes use of this constant t. On input $x = x_n \ldots x_0$, the reduction builds the string $f(x) = x_n 0^{t-1} x_{n-1} 0^{t-1} \ldots 0^{t-1} x_1 0^{t-1} x_0$. It follows easily from the observation

$$\sum_i x_i \equiv \sum_i x_i 2^{ti} \pmod{m}$$

that this is the desired $\leq_{\mathrm{m}}^{\mathrm{AC}^0}$ reduction. □

Corollary 4 *Let p be prime, and let m be an odd number that is not a power of p. Then* $\textsc{Mult}_m \notin AC^0[p]$.

3 Main Results

Our theorems concern the following three languages:

Definition 3 PRIMES *is the set of all strings $x \in \{0,1\}^*$ such that $\sum_i x_i 2^i$ is a prime number.*

SQUARE-FREE *is the set of all strings $x \in \{0,1\}^*$ such that $\sum_i x_i 2^i$ is a number that is not a multiple of any perfect square greater than 1.*

GCD *is the set of all triples (x, y, i) such that the i^{th} bit of the greatest common divisor of x and y is 1. (This just one of many equivalent ways of defining a language whose complexity is equivalent to the complexity of computing the greatest common divisor.)*

Theorem 5 *Let p be a prime number. Then* $\textsc{Mult}_p \leq_{\text{T}}^{AC^0}$ PRIMES.

Proof: We will need one fact from number theory, regarding the distribution of primes. Let $\pi(2^n, p, l)$ denote the number of primes q having binary representation of at most n bits, such that $q \equiv l \pmod{p}$, where $1 \leq l \leq p-1$. It is known (see Theorem 7.5 of Chapter 4 of [P57] for example) that

$$\pi(2^n, p, l) \sim \frac{2^n}{n(p-1)\ln 2} \qquad n \to \infty \tag{1}$$

for any fixed prime p. Thus there is a constant c such that for all large n, for any $1 \leq l \leq p-1$, at least $2^n/cn$ of the numbers having at most n bits that appear in the sequence $l, l+p, l+2p, \ldots$ are prime.

This gives rise to the following probabilistic test to see if an n-bit number x is a multiple of p. Given x, pick a random n-bit number y, and (using an oracle gate for PRIMES) check if $x + py$ is prime. If x is a multiple of p, the oracle gate says "no" with probability 1. If x is not a multiple of p, the oracle gate says "yes" with probability at least $1/cn$.

Now consider a circuit that performs cn^3 independent trials of this test in parallel. If x is a multiple of p, all of the tests say "no", whereas if x is not a multiple of p, then with probability at least $1 - 1/2^{2n}$, at least one of the tests will return "yes". Now, as in the standard argument of [Adl78], there must be at least one sequence of probabilistic inputs for the circuit

having the property that, for all n-bit inputs x, the OR of the cn^3 tests is equal to $\neg\textsc{Mult}_p(x)$.

This can be seen to be an $\leq_{\mathrm{T}}^{\mathrm{AC}^0}$ reduction from $\textsc{Mult}_p$ to $\textsc{Primes}$, since addition can be computed in AC^0. □

Corollary 6 *For any prime p,* $\textsc{Primes}$ *is not in $AC^0[p]$.*

Proof: Combining Theorems 3, and 5, we see that both $\textsc{Mod}_3$ and $\textsc{Mod}_5$ are $\leq_{\mathrm{T}}^{\mathrm{AC}^0}$-reducible to $\textsc{Primes}$. For any prime $p \neq 3$, the fact that $\textsc{Mod}_3 \leq_{\mathrm{T}}^{\mathrm{AC}^0} \textsc{Primes}$, combined with Smolensky's bound, shows that $\textsc{Primes} \notin \mathrm{AC}^0[p]$, while the fact that $\textsc{Mod}_5 \leq_{\mathrm{T}}^{\mathrm{AC}^0} \textsc{Primes}$ takes care of the case $p = 3$. □

A much simpler argument suffices to show that GCD is not in $\mathrm{AC}^0[p]$.

Theorem 7 *Let p be prime. Then* $\textsc{Mult}_p \leq_{\mathrm{T}}^{AC^0} \mathrm{GCD}$.

Proof: It suffices to observe that a number x is a multiple of p if and only if the greatest common divisor of x and p is p (which in turn is equivalent to the greatest common divisor not being 1). □

Corollary 8 *For any prime p,* GCD *is not in $AC^0[p]$.*

Finally we turn our attention to the square-free numbers.

Theorem 9 *Let p be a prime number. Then* $\textsc{Mult}_p \leq_{\mathrm{T}}^{AC^0} \textsc{Square-Free}$.

Proof: Again, we will need a fact from number theory about the distribution square-free numbers in residue classes. More precisely, we need an analogue of (1) for square-free numbers.

Accordingly we denote by $S(n,p,l)$ the number of square-free numbers s having binary representation of at most n bits, such that $s \equiv l \pmod{p^2}$, where $1 \leq l \leq p^2 - 1$. Let $T_d(n,p,l)$ be the number of integers m, $0 \leq m \leq 2^n - 1$ such that

$$m \equiv l \pmod{p^2} \qquad \text{and} \qquad m \equiv 0 \pmod{d^2}.$$

By applying the inclusion-exclusion principle we derive that

$$S(n,p,l) = \sum_{1 \leq d \leq 2^{n/2}} \mu(d) T_d(n,p,l),$$

where $\mu(d)$ is the Möbius function. We recall that $\mu(1) = 1$, $\mu(d) = 0$ if m is a perfect square and $\mu(d) = (-1)^{\nu(d)}$ otherwise, where $\nu(d)$ is the number of prime divisors of $d \geq 2$. Obviously, $T_d(n,p,l) = 0$ if $\gcd(d,p) > 1$ and

$$\left| T_d(n,p,l) - \frac{2^n}{p^2 d^2} \right| \leq 1$$

otherwise (because if $\gcd(d,p) = 1$ the above system of congruences defines the $m \pmod{p^2 d^2}$ uniquely). Therefore

$$\begin{aligned}
S(n,p,l) &= \sum_{\substack{1 \leq d \leq 2^{n/2} \\ \gcd(d,p)=1}} \mu(d) \left(\frac{2^n}{p^2 d^2} + O(1) \right) \\
&= \frac{2^n}{p^2} \sum_{\substack{1 \leq d \leq 2^{n/2} \\ \gcd(d,p)=1}} \frac{\mu(d)}{d^2} + O(2^{n/2}) \\
&= \frac{2^n}{p^2} \sum_{\substack{d=1 \\ \gcd(d,p)=1}}^{\infty} \frac{\mu(d)}{d^2} + O(2^{n/2}) \\
&= \frac{2^n}{p^2} \sum_{d=1}^{\infty} \frac{\mu(d)}{d^2} \\
&\quad - \frac{2^n}{p^2} \sum_{\substack{d=1 \\ \gcd(d,p)=p}}^{\infty} \frac{\mu(d)}{d^2} + O(2^{n/2}) \\
&= \frac{2^n}{p^2} \left(1 - \frac{1}{p^2} \right) \sum_{d=1}^{\infty} \frac{\mu(d)}{d^2} + O(2^{n/2}) \\
&= \frac{1}{\zeta(2) p^2} \left(1 - \frac{1}{p^2} \right) 2^n + O(2^{n/2}),
\end{aligned}$$

where $\zeta(s)$ is the Riemann zeta-function, see Theorem 4.4 of Chapter 3 of [P57]. Therefore

$$S(n,p,l) = \gamma(p) 2^n + O(2^{n/2}), \tag{2}$$

where

$$\gamma(p) = \frac{6}{\pi^2 p^2} \left(1 - \frac{1}{p^2} \right).$$

Thus (2) provides the desired analogue of (1).

This gives rise to the following probabilistic test to see if an n-bit number x is a multiple of p^2. Given x, pick a random n-bit number y, and (using an oracle gate for SQUARE-FREE) check if $x + p^2 y$ is square-free. If x is a multiple of p^2, the oracle gate says "no" with probability 1. If x is not a multiple of p^2, the oracle gate says "yes" with probability $\gamma(p) + o(1)$. The rest of the argument is entirely analogous to the proof of Theorem 5. □

Corollary 10 *For any prime p,* SQUARE-FREE *is not in $AC^0[p]$.*

4 Conclusions and Open Problems

It is irksome that our reduction from MOD_3 to PRIMES is *nonuniform.* (That is, we can provide no efficient procedure to *build* the AC^0 circuits that perform the reduction; we can show only that they exist, via a probabilistic argument.) Surely it is obvious that telling if a number if a multiple of 3 is no harder than telling if a number is composite! Is there a direct, *uniform* reduction that captures this intuition?

We have seen that, for any odd prime p, the MOD_p problem is reducible to the set of primes, written in base two. A similar argument shows that, for any distinct primes p and q, the MOD_p problem is reducible to the set of primes, written in base q. However, the following question requires a different proof strategy: Is $\text{MOD}_2 \leq_{\text{T}}^{\text{AC}^0} \text{PRIMES}$?

The theory of many-one reducibility has been extremely useful in characterizing the complexity of many problems, although it has not turned out to be very useful for studying number-theoretic problems. For example, although we know that MOD_3 is AC^0-Turing reducible to PRIMES, we do not know if it is many-one reducible to PRIMES. Might it be possible to *prove* that there is no many-one reduction from MOD_3 (or PARITY) to PRIMES? This would show that PRIMES is not NP-complete (under $\leq_{\text{m}}^{\text{AC}^0}$ reductions), and in fact would show that it is not complete for any familiar complexity class. Although in general it is difficult to show that there *is* no $\leq_{\text{m}}^{\text{AC}^0}$ reduction from one problem to another (since, for example, the NP $\neq$ NC^1 question can be phrased this way), it is worth noting that a set A in NP is presented in [AAIPR97] such that there is no $\leq_{\text{m}}^{\text{AC}^0}$ reduction from PARITY to A.

If PRIMES were complete for NP (or for any other reasonable complexity class) under $\leq_{\text{m}}^{\text{AC}^0}$ reductions, the isomorphism theorems of [AAR98,

AAIPR97] show that PRIMES would be isomorphic to all of the other complete sets for that class, under isomorphisms computable and invertible by P-uniform depth-three AC^0 circuits. In particular, there would be an isomorphism of this sort between PRIMES and PRIMES $\times \{0,1\}^*$. Among other things, this would yield a fairly "dense" set of primes in P, by looking at the isomorphic image of $\{2\} \times \{0,1\}^*$. (Observe that it was shown only fairly recently that there is an infinite set of primes in P [PPS89].) Perhaps the existence of such an isomorphism would bestow PRIMES with some properties that it provably does not have. Perhaps such an isomorphism must involve multiplication (which cannot be computed by AC^0 circuits). That is, perhaps it is possible to prove that PRIMES is not complete for any familiar complexity class. Of course, in the foregoing discussion we are considering only *unconditional* proofs. It is well known, thanks to [Mil76], that PRIMES is in P under the Extended Riemann Hypothesis.

Additional observations and speculations of this sort pertaining to the factoring problem can be found in [All98].

We remark that several other natural number theoretic problems can be AC^0-reduced to MULT_q, and hence can be shown not to belong to $AC^0[p]$. For example, let us consider the problem of computing the parity of $\omega(x)$, which is the number of distinct prime divisors of $x \in \mathbb{N}$. We remark that

$$\text{MULT}_3(x) = 0 \quad \Longleftrightarrow \quad \omega(x) + 1 \equiv \omega(3x) \pmod 2.$$

Thus, this problem does not belong to $AC^0[p]$ for any prime $p \neq 3$. Considering $\text{MULT}_5(x)$ and $\omega(5x)$, we see that this problem is not in $AC^0[p]$ for any prime p.

It would be very interesting to obtain similar results for other number theoretic problems. For example, it is shown [Shp99] that deciding quadratic residuosity modulo a large prime q is not in AC^0. Note that this question is equivalent to computing the rightmost bit of the discrete logarithm modulo q. It would be very desirable to extend this lower bound to the classes $AC^0[p]$.

Acknowledgment. This paper was essentially written during a visit by the third author to Rutgers University, whose hospitality is gratefully acknowledged.

References

[Adl78] L. Adleman, *Two theorems on random polynomial time*, in "Proc. 19th IEEE Symposium on Foundations of Computer Science", 1978, 75–83.

[AH87] L. Adleman and M.-D. Huang, *Recognizing primes in random polynomial time*, in "Proc. 19th ACM Symposium on Theory of Computing", 1987, 462–469.

[APR83] L. Adleman, C. Pomerance and R. S. Rumely, *On distinguishing prime numbers from composite numbers*, Annals Math. **117** (1987), 173–206.

[Aj83] M. Ajtai, Σ_1^1 *formulae on finite structures*, Annals of Pure and Applied Logic **24** (1983), 1–48.

[AAIPR97] M. Agrawal, E. Allender, R. Impagliazzo, T. Pitassi and S. Rudich, *Reducing the complexity of reductions*, in "Proc. 29th ACM Symposium on Theory of Computing", 1997, 730–738.

[AAR98] M. Agrawal, E. Allender and S. Rudich, *Reductions in circuit complexity: An isomorphism theorem and a gap theorem*, J. Comp. Sys. Sci. **57** (1998), 127–143.

[All98] E. Allender, *News from the isomorphism front*, Computational Complexity Column, Bulletin of the EATCS **66** (1998), 73–82.

[BDS98a] A. Bernasconi, C. Damm and I. E. Shparlinski, *Circuit and decision tree complexity of some number theoretic problems*, Tech. Report 98-21, Dept. of Math. and Comp. Sci., Univ. of Trier, 1998, 1–17.

[BDS98b] A. Bernasconi, C. Damm and I. E. Shparlinski, *On the average sensitivity of testing square-free numbers*, in "Proc. 5th Intern. Computing and Combin. Conf.", Lect. Notes in Comp. Sci., Springer-Verlag, Berlin, (to appear).

[BS99] A. Bernasconi and I. E. Shparlinski, *Circuit complexity of testing square-free numbers*, in "Proc. 16th Intern. Symp. on Theor. Aspects in Comp. Sci.", Lect. Notes in Comp. Sci., Springer-Verlag, Berlin, **1563** (1999), 47–56.

[BL87] R. Boppana and J. Lagarias, *One-way functions and circuit complexity*, Information and Computation **74** (1987), 226–240.

[FSS84] M. Furst, J. Saxe and M. Sipser, *Parity, circuits, and the polynomial-time hierarchy*, Math. Systems Theory **17** (1984), 13–27.

[HB76] J. Hartmanis and L. Berman, *On tape bounds for single letter alphabet language processing*, Theoretical Computer Science **3** (1976), 213–224.

[HS68] J. Hartmanis and H. Shank, *On the recognition of primes by automata*, J ACM **15** (1968), 382–389.

[Man92] S.-G. Mantzivis, *Circuits in bounded arithmetic, 1*, Ann. Math. Artificial Intelligence **6** (1992), 127–156.

[Med91] J. Meidânis, *Lower bounds for arithmetic problems*, Inform. Proc. Letters **38** (1991), 83–87.

[Mil76] G. Miller, *Riemann's hypothesis and tests for primality*, J. Comput. System Sci. **13** (1976), 300–317.

[PPS89] J. Pintz, W. Steiger and E. Szemerédi, *Two infinite sets of primes with fast primality tests*, Math. Comp. **53** (1989), 399–406.

[P57] K. Prachar, Primzahlverteilung, Springer-Verlag, 1957.

[R80] M. Rabin. *Probabilistic algorithm for primality testing*, Journal of Number Theory **12** (1980), 128–138.

[Raz87] A. A. Razborov, *Lower bounds on the size of bounded depth networks over a complete basis with logical addition*, Mathematicheskie Zametki, **41** (1987), 598–607, English translation in Mathematical Notes of the Academy of Sciences of the USSR **41** (1987), 333–338.

[Shp98] I. E. Shparlinski, *On polynomial representations of Boolean functions related to some number theoretic problems*, Electronic Colloq. on Comp. Compl., TR98-054, 1998, 1–13.

[Shp99] I. E. Shparlinski, *Number theoretic methods in cryptography: Complexity lower bounds*, Birkhäuser, 1999.

[Smo87] R. Smolensky, *Algebraic methods in the theory of lower bounds for Boolean circuit complexity*, in "Proc. 19th ACM Symposium on Theory of Computing", 1987, 77–82.

[SS77] R. Solovay and V. Strassen, *A fast Monte Carlo test for primality*, SIAM J. Comput. **6** (1977), 84–85, erratum **7** (1978), 118.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'