

Notices

of the American Mathematical Society

Recent Advances in Primality Testing
by Robert Rumely

Reprinted from the *Notices*, August 1983
©1983 American Mathematical Society

Recent Advances in Primality Testing

by Robert Rumely

Prime numbers are one topic in mathematics the public can relate to. People seem fascinated by the RSA "trap door" coding scheme, and by records for large primes. (Currently the largest known prime is the Mersenne number $2^{86243} - 1$.) Likewise, the primality test recently developed by Adleman, Pomerance, and Rumely [1] has received a great deal of attention in the press. That test, as significantly improved by Cohen and Lenstra [2], is the main subject of this article.

Recent developments in commerce and security aside, the problems of testing numbers for primality and factoring them have serious algorithmic interest. It may be a surprise to some that much better methods are available for both than trial division up to \sqrt{n} . Further, the two problems are distinct: it is possible to determine whether a number is prime or composite without attempting to factor it. In fact, it is very easy to decide as a practical matter whether a number is likely to be prime or composite. The idea is to apply a pseudo-primality test: to check a property (such as Fermat's congruence) which all primes share, but most composites do not. A number which is free of small prime divisors and which passes even a single pseudo-primality test is almost certain to be prime.

However, finding a rigorous proof of primality or compositeness is more difficult. Some numbers of special form, such as Mersenne numbers ($2^p - 1$, with p prime) can actually be proved prime by checking a single appropriately chosen pseudo-primality test. It is in this way that "record primes" are found. For a number n of general form, the classical method of proving primality was to show the existence of a generator for the multiplicative group $(\mathbb{Z}/n)^\times$, making use of a factorization of $n - 1$. (The school of D. H. Lehmer extended this idea significantly.) The breakthrough in the APR test was the discovery of a collection of pseudo-primality tests, such that the possible divisors of a number passing them all are limited to a small, computable set.

The Cohen-Lenstra version of the APR algorithm is the fastest general primality test known; its running time is bounded by

$$(1) \quad (\log n)^{C \log \log \log n} \\ = \exp(C \log \log n \log \log \log n)$$

bit operations. For comparison, the asymptotically fastest factoring algorithms have expected

running time

$$(2) \quad \exp(C \sqrt{\log n \log \log n}).$$

In complexity theory, a problem is considered "tractable" if it can be solved by an algorithm which runs in a polynomial number of steps in the length of the input: $(\log n)^C$ if the input is n . Thus, the APR algorithm just misses being polynomial. Miller [6] has given a primality algorithm which runs in polynomial time, but depends on the Extended Riemann Hypothesis for its correctness. Interestingly, because of the constants in the running time bounds, the Cohen-Lenstra algorithm is faster than Miller's for numbers of some tens to several thousands of digits.

Most modern primality tests have their roots in Fermat's congruence: if n is prime, and $(q, n) = 1$, then

$$(3) \quad q^{n-1} \equiv 1 \pmod{n}.$$

The left side of this can be calculated very quickly, by forming the powers $q^{2^k} \pmod{n}$ by repeated squarings, and then multiplying together appropriate powers as indicated by the binary expansion of $n - 1$. Given a fixed $q \neq 1$, for most composite n , (3) is false. However, there is a sparse set of composites known as Carmichael numbers (including 561 and 1729) for which (3) holds for all q with $(q, n) = 1$. Thus, to obtain a proof of primality, it is necessary to strengthen Fermat's congruence.

Solovay and Strassen [11] (and independently, Shanks, and Lehmer [3]) suggested using the congruence defining the quadratic residue symbol as a pseudo-primality test: if n is an odd prime, and $(q, n) = 1$, then

$$(4) \quad q^{(n-1)/2} \equiv \left(\frac{q}{n}\right) \pmod{n} = \pm 1$$

This time, if n is composite, (4) fails for at least half the numbers $q < n$. Hence, a number passing it for k randomly chosen q 's can be asserted prime with "probability of error" less than $1/2^k$.

It was Adleman's insight that the Solovay-Strassen test gives more information than just pass-or-fail: it also tells about the structure of divisors of n . Loosely, if n passes the Solovay-Strassen test for each q in a collection \mathcal{Q} of prime numbers then, for any divisor r of n , the order $(\pmod{2})$ of r in one of the multiplicative groups

$(\mathbb{Z}/q)^\times$ determines its order (mod 2) in all of them. This is proved using the quadratic reciprocity law and something called the "extraction lemma". Rumely found a collection of pseudo-primality tests, performed in cyclotomic fields $\mathbb{Q}(\zeta_p)$, which gave a similar result for odd primes p , linking the indices (mod p) of r in the groups $(\mathbb{Z}/q)^\times$. (For any m , ζ_m denotes a primitive m th root of unity.) Adleman suggested a scheme whereby these results could be used to reconstruct the divisors of n . Starting with a set I of "initial primes" p , let \mathcal{Q} be the set of all primes of the form $q = 1 + \prod_{p \in S} p$, $S \subset I$ (called "Euclidean primes", after Euclid's proof of the infinitude of the primes). Put

$$(5) \quad Q = \prod_{q \in \mathcal{Q}} q.$$

Suppose n passes all the necessary pseudo-primality tests. By guessing the index (mod p) of r for each $p \in I$, a possible divisor r of n can be located mod q for each $q \in \mathcal{Q}$. This is because of the linkage between the indices (mod p) of r in the various $(\mathbb{Z}/q)^\times$, and because each of the numbers $q - 1$ is square-free, with its prime factors in I . By the Chinese Remainder Theorem, r can be found mod $Q (= \prod q)$; if $Q > n$, it can be determined, period. The number of possible divisors to be tested is

$$(6) \quad P = \prod_{p \in I} p.$$

Clearly there is a tension between the numbers P and Q : P should be kept small, so the number of divisors to be tested will be small; but it must be large enough that $Q > n$. Pomerance and Odlyzko showed that the initial primes can be chosen so that

$$(7) \quad P \approx (\log n)^{C \log \log \log n}.$$

This is a best possible estimate, apart from the constant C ; it accounts for the running time of the algorithm. Heuristically, C can be any number greater than $1/\log 2$.

Lenstra [5] simplified and recast the APR test. He saw that the set of possible divisors of n was simply the set of powers of n (mod Q). To show this, he focused on the characters of $(\mathbb{Z}/Q)^\times$. The pseudo-primality tests he used were congruences satisfied by Gauss sums formed with generators for the character group. In the proofs, he replaced the deep power reciprocity laws, used by APR, with elementary properties of Gauss sums. One version of the test is short enough to present here in its entirety.

Lenstra's pseudo-primality tests are based on an easy computation. Let p and q be prime numbers with $p|q-1$, and let χ be a character of $(\mathbb{Z}/q)^\times$ having order p . Define the Gauss sum

$$(8) \quad \tau(\chi) = \sum_{a=1}^{q-1} \chi(a) \zeta_q^a \quad (\in \mathbb{Z}[\zeta_{pq}]).$$

It is elementary to show that if n is prime, then

$$(9) \quad \tau(\chi)^{n^{p-1}-1} \equiv \chi(n) \pmod{n\mathbb{Z}[\zeta_{pq}]}.$$

This congruence can of course be checked whether n is prime or composite. Under a mild technical condition, if (9) holds, then for any prime divisor r of n , there is a number b (mod p) independent of χ and q , such that

$$(10) \quad \chi(n)^b = \chi(r).$$

The technical condition is that for some character χ of order p , and some q , (9) should hold with $\chi(n) \neq 1$. It implies that the power of p dividing $\tau^{p-1} - 1$ is greater than or equal to the power of p dividing $n^{p-1} - 1$, so there exist integers $c \equiv 1 \pmod{p}$, and b , with $b(n^{p-1} - 1) = c(\tau^{p-1} - 1)$. To derive (10), note that (9) certainly holds with r replacing n . Therefore, since r divides n ,

$$(11) \quad \begin{aligned} \chi(n)^b &\equiv \tau(\chi)^{b(n^{p-1}-1)} \\ &\equiv \tau(\chi)^{c(\tau^{p-1}-1)} \equiv \chi(r)^c = \chi(r) \end{aligned}$$

mod $r\mathbb{Z}[\zeta_{pq}]$. Assuming $(n, pq) = 1$, this yields (10).

The Gauss sums primality test is as follows. First, one constructs the sets I and \mathcal{Q} of "initial" and "Euclidean" primes as in Adleman's schema, with $Q > n$, and checks that $(n, PQ) = 1$. For each pair p, q with $p \in I$, $q \in \mathcal{Q}$, and $p|q-1$, one finds a character χ of $(\mathbb{Z}/q)^\times$ having order p , and carries out the pseudo-primality test (9). If any of the tests fail, n is of course composite. Most likely, during the tests, for each p a character χ will have been found for which $\chi(n) \neq 1$, verifying the technical condition. If not, one checks (9) for other characters until one is found for which $\chi(n) \neq 1$, or n is shown composite. Finally, for each b , $1 \leq b \leq P$, one constructs the least positive residue of n^b mod Q , and checks whether it is a proper divisor of n . If no divisors are found, n is prime.

The test is correct, because if r is a prime dividing n , by the Chinese Remainder Theorem there is a number b in the range $1 \leq b \leq P$ such that (10) holds for all the characters χ above: that is, $\chi(n^b) = \chi(r)$. These characters generate the character group of $(\mathbb{Z}/Q)^\times$, so $n^b \equiv r \pmod{Q}$.

The test as presented here is what is called a "nondeterministic" algorithm, because of the possibility of trying a large number of characters χ in verifying the technical condition. However, there is a deterministic form with the running time (1).

The Gauss sums pseudo-primality tests involve calculations in the number ring $\mathbb{Z}[\zeta_{pq}]$. It may be worth remarking how these can be carried out. The minimal polynomial $\Phi(x)$ of ζ_{pq} is known explicitly and has degree $(p-1)(q-1)$. Elements in $\mathbb{Z}[\zeta_{pq}]$ can be represented by polynomials in $\mathbb{Z}[x]$ of degree less than $(p-1)(q-1)$. Addition and subtraction in $\mathbb{Z}[\zeta_{pq}]$ correspond to addition and subtraction of polynomials; multiplication

corresponds to multiplication of polynomials, followed by taking the remainder upon division by $\Phi(x)$. To check a congruence mod $nZ[\zeta_{pq}]$, one checks the corresponding congruence mod n for each of the coefficients.

The version of the test given above is simple theoretically, but is not practical for a computer. Cohen and Lenstra [2] have given another version which is computer-practical. They replace the Gauss sums pseudo-primality tests by tests in the smaller rings $Z[\zeta_p]$, using so-called "Jacobi sums". They permit the use of characters χ with prime power (rather than just prime) order; especially in the case $p = 2$, due to Cohen, this is nontrivial. Clearly the condition $Q > n$ can be replaced by $Q > n^{1/2}$; Lenstra showed that even $n^{1/3}$ is sufficient. In addition to many other improvements, they have taken great care to perform arithmetic operations efficiently. Their program, now running on a CDC Cyber 170-750 computer in the SARA computer center in Amsterdam, is able to deal with 100 digit numbers in about 30 seconds and 200 digit numbers in 8 minutes.

The running time of deterministic versions of the algorithm, and the expected running time of nondeterministic versions, are polynomial in the number P . To shed light on the bound (7), we present a heuristic argument for the size of P , based on estimating Q when P is the product of an initial segment of primes. Exactly 2^t "Euclidean numbers" of the form $1 + \prod p_i$ can be built up from the first t primes. By the Prime Number Theorem, the "average size" of these numbers should be $e^{(1/2)t \log t}$. Assuming that these are no less likely to be prime than ordinary numbers, about $2^{t/2} \log t$ of them should be prime. Thus we expect

$$(12) \quad Q \approx \exp\left(\left(\frac{1}{2}t \log t\right) \frac{2^t}{(1/2t \log t)}\right) = e^{2^t}.$$

$Q = n$ and solving for t gives $t \approx \log \log n / \log 2$, and the Prime Number Theorem yields

$$(13) \quad P \approx e^{t \log t} \approx (\log n)^{\log \log n / \log 2}.$$

Thus, the triple log in the exponent is ultimately due to the logarithm factor in the Prime Number Theorem. Pomerance and Odlyzko's proof of (7) uses an entirely different idea; it is based on sieve methods, and follows an averaging argument, showing that most multiples of the product of a sufficiently long initial segment of primes can be taken for P .

It has only been possible to touch on the topics of primality testing and factoring here. For

those interested in an overview of the subject, we recommend the survey articles of Williams [13], Lenstra [4], and Pomerance [8], which contain further references. On primality testing, we note the original paper of Adleman, Pomerance, and Rumely [1], and the paper of Cohen-Lenstra [2], which contains a very complete description of their algorithm. On factorization, Brillhart-Morrison [7] present the continued fraction algorithm, and the papers of Pomerance [9], Schoof [10], and Voorhoeve [12] analyze a number of methods and their running times.

References

1. L. M. Adleman, C. Pomerance and R. Rumely, *On distinguishing prime numbers from composite numbers*, Annals of Mathematics (Series 2), volume 117, 1983, pages 173-206.
2. H. Cohen and H. W. Lenstra, Jr., *Primality testing and Jacobi sums*, Mathematics of Computation (to appear).
3. D. H. Lehmer, *Strong Carmichael numbers*, Journal of the Australian Mathematical Society Series A, volume 21, 1976, pages 508-510.
4. H. W. Lenstra, Jr., *Primality testing*, in Computational Methods in Number Theory (H. W. Lenstra, Jr. and R. Tijdeman, editors), Mathematical Centrum Tracts, Number 154, part I and Number 155, part II, 1983.
5. H. W. Lenstra, Jr., *Primality testing algorithms (after Adleman, Rumely and Williams)*, Séminaire Bourbaki (June 1981) Number 561.
6. G. L. Miller, *Riemann's hypothesis and tests for primality*, Journal of Computer and System Sciences, volume 13, 1976, pages 300-317.
7. M. A. Morrison and J. Brillhart, *A method of factoring and the factorization of F_7* , Mathematics of Computation, volume 29, 1975, pages 183-205.
8. C. Pomerance, *Recent developments in primality testing*, Mathematical Intelligencer, volume 3, 1981, pages 97-105.
9. C. Pomerance, *Analysis and comparison of some integer factoring algorithms*, in Computational Methods in Number Theory (H. W. Lenstra, Jr. and R. Tijdeman, editors), Mathematical Centrum Tracts, Number 154, part I and Number 155, part II, 1983.
10. R. J. Schoof, *Quadratic fields and factorization*, in Computational Methods in Number Theory (H. W. Lenstra, Jr. and R. Tijdeman, editors), Mathematical Centrum Tracts, Number 154, part I and Number 155, part II, 1983.
11. R. Solovay and V. Strassen, *A fast Monte-Carlo test for primality*, SIAM Journal on Computing, volume 6, 1977, pages 84-85; erratum, volume 7, 1978, page 118.
12. M. Voorhoeve, *Factorization algorithms of exponential order*, in Computational Methods in Number Theory (H. W. Lenstra, Jr. and R. Tijdeman, editors), Mathematical Centrum Tracts, Number 154, part I and Number 155, part II, 1983.
13. H. C. Williams, *Primality testing on a computer*, Ars Combinatoria, volume 5, 1978, pages 127-185.