Proceedings of Symposia in Applied Mathematics
Volume 42, 1990

Primality Testing

ARJEN K. LENSTRA

1. Introduction

How does one decide whether an integer $n > 1$ is composite or prime? It will be seen that for composite n a proof of compositeness can usually quite easily be found. If several attempts to find such a proof of compositeness have failed, there are good reasons to suspect that the number in question is actually prime. It then remains to *prove* that n is prime. Providing such a proof is the object of *primality testing*.

Many of the older primality testing algorithms reduce the problem of proving the primality of n to the problem of finding sufficiently many factors of related numbers like $n \pm 1$. But factoring integers seems to be a hard problem, at least in general. For that reason, those older primality testing algorithms cannot be called *general purpose methods*; they only work if n is lucky, i.e., if factoring $n \pm 1$ turns out to be easy. In the early 1980s scores of primes of less than 100 digits could not be proved prime. Nevertheless, quite impressive results still can be obtained with these older methods. For instance, the 65087-digit number $391581 \times 2^{216193} - 1$ (the largest prime currently† known), and $(10^{1031} - 1)/9$, a number consisting of 1031 ones, have been proved in this way to be prime.

More recent methods for primality testing do not have the disadvantage of depending so heavily on factoring. While primality proofs of arbitrary 65,087-digit primes are still out of reach, for primes well beyond 500 digits, and without special properties, primality proofs can now be given. The first primality test that could routinely handle primes of several hundred digits was the *Jacobi sum test*: 100-(200)-digit primes take an average of 7 (68) seconds on a single Cray X-MP processor. The Jacobi sum test runs in time $(\log n)^{O(\log\log\log n)}$, which makes it the fastest deterministic primality

1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 11A41, 11A51, 11-02.

† 1990.

Most of the work on this note was done while the author was visiting the Department of Computer Science of The University of Chicago. The author would like to thank this institute for its hospitality and support.

test. Other primality tests that do not rely exclusively on factoring are based on elliptic curves. They are the *random curve test*, its practical variant the *complex multiplication test*, and finally the *abelian variety test*, the latter test being the only primality test that can be proved to run in expected polynomial time.

The two most powerful practical primality tests that exist nowadays are an improved version of the Jacobi sum test and the complex multiplication test. It is not unlikely that both these methods will be able to tackle 1000-digit primes. The Jacobi sum test can easily be broken up into almost any number of smaller subtasks. For the complex multiplication test, parallelization is much harder to achieve. On the other hand, in case of primality, the Jacobi sum test only gives the result *yes indeed, that number is prime*, without any other way to verify the computation than redoing it, whereas the complex multiplication test yields a *certificate* of primality that can be checked much faster than it can be found.

In this note an attempt will be made to give a rough impression of those two primality tests. For more information on primality testing the reader is referred to the various survey articles [**6, 10, 13, 14, 17, 23, 27, 28, 29, 33**], and to the original papers [**1, 2, 5, 8, 9, 11, 21**].

2. Classical Methods

The first question to be addressed in this section is: how do we get rid of most composites quickly? Let $n > 1$ be the integer to be examined. In practice, the first thing to do is to check whether n is divisible by (or even equal to) some small prime, for all primes up to a certain trial division bound, and to draw the appropriate conclusion if that turns out to be the case. Clearly, if the trial division bound is $\geq \sqrt{n}$, then this is the end of the story. In general, however, trial division up to $\sqrt{n}$ is out of the question and nobody will consider doing that. Instead the bound will be set to some small number, say 20, so that this step only serves as a quick way to cast out trivial composites.

Now suppose that n survives this first primitive attack. If n is prime, then any integer a should satisfy $a^n \equiv a \bmod n$, due to Fermat's little theorem. And for any a and n this identity can be checked efficiently by means of the repeated squaring method. Consequently, to prove that n is composite, it suffices to find an integer a for which $a^n \not\equiv a \bmod n$, a so-called *witness* to the compositeness of n.

A witness might be difficult or even impossible to find, however: there exist composite numbers, the so-called *Carmichael numbers*, for which no witnesses exist. This unfortunate situation can be remedied by casting the test in a slightly different form, a formulation essentially due to Gary Miller [**20**]: if n is prime and $n - 1 = r \cdot 2^k$ with r odd, then any integer $a \in \{1, 2, \ldots, n-1\}$ satisfies

$$(2.1) \quad a^r \equiv 1 \bmod n \quad \text{or} \quad a^{r \cdot 2^i} \equiv -1 \bmod n \quad \text{for some } i \text{ with } 0 \leq i < k.$$

If (2.1) holds for n and a, then n is said to *pass the test* for that a. If n does not pass the test for a certain a, then a is again called a witness to the compositeness of n.

This test is often referred to as a *probabilistic compositeness test*, because if n is an odd composite number, then more than $\frac{3}{4}$ of the integers a in $\{2, 3, \ldots, n-1\}$ are witnesses to the compositeness of n (cf. **[26]**). In other words, the probability that an odd composite number n passes the test for a randomly chosen $a \in \{2, 3, \ldots, n-1\}$ is less than $\frac{1}{4}$. This means that the probability of proving the compositeness of an odd composite number n by checking (2.1) for l independent random choices of a from $\{2, 3, \ldots, n-1\}$ is more than $1 - 4^{-l}$.

In practice, this means that composite numbers are immediately recognized by checking (2.1) for only a few a's. An odd number passing several tests, say 10, is called a *probable prime*. It should be clear that a probable prime is not *proved* to be prime; it is a number for which the compositeness could not be proved, and which is therefore suspected to be prime. It remains to prove that such a number is indeed prime.

Now that most composites have been dealt with, how do we prove the primality of a probable prime? For,odd numbers $n < 25 \cdot 10^9$, and unequal to $3{,}125{,}031{,}751 = 151 \cdot 751 \cdot 28{,}351$, the proof can easily be done by checking that n passes the test for $a = 2, 3, 5$, and 7: if n is odd, composite, and unequal to 3,125,031,751, then at least one of 2, 3, 5, and 7 is a witness to the compositeness of n (cf. **[25]**).

For larger numbers the proof would not be hard either, in theory at least, if the generalized Riemann hypothesis were known to be true. In that case it would suffice to verify (2.1) for the a in $\{2, 3, \ldots, [2(\log n)^2]\}$, because that interval would contain a witness if n were composite (cf. **[3, 20]**). A proof of the generalized Riemann hypothesis therefore would lead to a deterministic polynomial-time primality test, but in practice both the Jacobi sum test and the complex multiplication test are expected to be faster for numbers up to several thousands of digits.

Primality proofs that do not depend on any unproved hypotheses are in general harder to obtain. If the factorization of, for instance, $n-1$ is known, then a proof can quite easily be given. This can be seen as follows. If the positive integers $< n$ are all relatively prime to n, then n is prime. So, to prove the primality of n it suffices to find an element a of order $n-1$ in the multiplicative group $(\mathbf{Z}/n\mathbf{Z})^*$; i.e., an integer $a \in \{1, 2, \ldots, n-1\}$ for which

$$(2.2) \qquad \begin{cases} a^{n-1} \equiv 1 \bmod n \text{ and} \\ a^{(n-1)/q} \not\equiv 1 \bmod n \quad \text{for all primes } q \text{ dividing } n-1. \end{cases}$$

Finding a can be done by randomly selecting integers until one is found satisfying (2.2); if a cannot be found then it is unlikely that n is prime, and n should be subjected to more probabilistic compositeness tests.

It is not necessary to know all prime divisors of $n-1$. According to the following theorem it suffices to have the prime factors of a factor $> \sqrt{n}-1$ of n.

(2.3) POCKLINGTON'S THEOREM (cf. [22]). *Let n be an integer >1, and let s be a positive divisor of $n-1$. Suppose there is an integer a satisfying*

$$a^{n-1} \equiv 1 \bmod n,$$
$$\gcd(a^{(n-1)/q} - 1, n) = 1 \quad \textit{for each prime } q \textit{ dividing } s.$$

Then every prime dividing n is congruent to 1 modulo s, and if $s > \sqrt{n}-1$ then n is prime.

PROOF. Let p be a prime dividing n and let $b = a^{(n-1)/s} \bmod p \in \mathbf{F}_p$, where $\mathbf{F}_p$ denotes the finite field containing p elements. From $a^{n-1} \equiv 1 \bmod n$ and p divides n, it follows that $b^s = 1$, so the order of b divides s. Let q be a prime dividing s. From $\gcd(a^{(n-1)/q} - 1, n) = 1$ it follows that $b^{s/q} \equiv a^{(n-1)/q} \bmod p \neq 1$ in $\mathbf{F}_p$, so that the order of b is not a divisor of s/q, for any prime q dividing s. The order of b therefore equals s. But since the order of b also divides $p-1$, it follows that s divides $p-1$. Every prime dividing n is therefore congruent to 1 modulo s. The statement of the theorem now follows immediately.

This theorem is applied just as (2.2): randomly select a's until one is found satisfying the conditions in (2.3); if that does not work, then n is probably not prime.

Not only factors of $n-1$ can be utilized in primality proofs; factors of $n+1$ are equally useful, and they can be combined with the factors of $n-1$ into the so-called *combined theorem* (cf. [**6**]). This latter theorem leads to a primality proving strategy called DOWNRUN: if during the simultaneous factorization attempt of $n-1$ and $n+1$ the unfactored part of either $n-1$ or $n+1$ is found to be a probable prime, then apply the strategy recursively to prove the primality of this newly found probable prime (cf. [**6**]). In this way a chain of primes $n = n_0, n_1, \ldots, n_t$ is built, such that n_i divides $n_{i-1}+1$ or $n_{i-1}-1$ and such that the primality of n_i implies the primality of n_{i-1}. It will be seen that the primality tests based on elliptic curves build similar chains of primes.

More complicated primality tests take this search for factors even further, combining factors of $n \pm 1$, n^2+1, and $n^2 \pm n + 1$; the reader is referred to [**6, 10, 14, 21, 27, 28, 31, 33**] for more information on these and similar methods. Using a test of this type, Williams and Dubner were able to prove the primality of $(10^{1031}-1)/9$ (cf. [**34**]), where they also made use of the following more recent result.

(2.4) THEOREM (cf. [**15**]). *Let r, s, and n be integers satisfying*

$$0 \leq r < s < n, \qquad s > \sqrt[3]{n}, \qquad \gcd(r, s) = 1.$$

Then there exist at most 11 *positive divisors of* n *that are congruent to* r *modulo* s, *and there is a polynomial algorithm for determining all these divisors.*

The algorithm referred to in (2.4) is not only polynomial time; according to [**34**] it is even efficient in practice. A consequence of this theorem is, for instance, that to apply Theorem (2.3) the lower bound $\sqrt{n} - 1$ on the factored part of $n-1$ can be relieved to $\sqrt[3]{n}$, at the cost of some extra work. Assuming that the factored part s of $n-1$ is $> \sqrt[3]{n}$, combining (2.3) and (2.4) yields the following primality test. First find an integer a satisfying both conditions of (2.3). This shows that the prime divisors of n are all congruent to 1 modulo s. Next apply the algorithm from (2.4) to find the at most 11 positive divisors of n that are congruent to 1 modulo s. If no nontrivial factor of n has been found in this way, then n is prime. Theorem (2.4) plays an important role in the Jacobi sum primality test (cf. [**5**]).

3. The Jacobi Sum Test

Theorem (2.3) appears to be a special case of a theorem that could be formulated thus: *if, for positive integers* n *and* s, *certain 'Fermat-like' tests hold for* n, s, *and the prime divisors* q *of* s, *then any prime divisor of* n *is congruent to a power of* n *modulo* s. Application of this theorem in a situation where $n \equiv 1 \bmod s$, i.e., s divides $n-1$ as in Theorem (2.3), leads to the conclusion that all divisors of n are 1 modulo s, just as in Theorem (2.3). Consequently, if $s > \sqrt{n} - 1$ then n is prime, and if s is only $> \sqrt[3]{n}$ then the possible divisors of n can be derived using (2.4), from which the primality of n will usually follow. The problem in this application is, of course, to satisfy one of those lower bounds for an integer s for which $n \equiv 1 \bmod s$.

But in this more general theorem s can be taken as *any* product $> \sqrt{n}$. Assuming that the 'Fermat-like' tests can be dealt with efficiently, the primality of n can be proved by verifying that none of the $n^i \bmod s$, for $i = 1, 2, \ldots$, is a nontrivial divisor of n. Notice that $s > \sqrt[3]{n}$ would also suffice, but then (2.4) should be applied to each of the $n^i \bmod s$ to verify that none of the possibilities leads to a nontrivial factor.

This will only be an efficient general purpose primality test if all different $n^i \bmod s$ can be generated in a reasonable amount of time. For the proper choice of s this is indeed possible. Assume that any prime occurs at most once in s, and that $\gcd(n, s) = 1$. It follows that $n^{q-1} \equiv 1 \bmod q$, for all primes q dividing s, so that $n^t \equiv 1 \bmod s$, where t is the least common multiple of the $q-1$'s. This implies that, in the application of the above theorem, it suffices to consider $n^i \bmod s$ for $i \in \{1, 2, \ldots, t-1\}$. To make this efficient, t should be kept small, which implies that s should be chosen as a product of q's such that the $q-1$'s have many factors in common. This suggests that it is a better idea not to select the q's and to compute the resulting t, but to do it the other way around: select t as a product of many

small primes, and see how many primes q can be found such that $q-1$ divides t.

For instance, for $t = 2^4 \cdot 3^2 \cdot 5 \cdot 7 = 5040$, the product of the 27 different q's for which $q-1$ divides t is more than 10^{48}; any prime of up to 96 digits can therefore be proved prime by carrying out the 'Fermat-like' tests and at most 5039 trial divisions.

Notice that the selection of t and s only depends on the size of n, and not on any divisibility properties of s and n (except that $\gcd(n, s)$ should be equal to 1). Odlyzko and Pomerance have shown that there is a positive constant c such that for every $n > e^e$ there exists an integer $t < (\log n)^{c \cdot \log\log\log n}$ such that the corresponding s is $> \sqrt{n}$ (cf. [2]). Because a similar lower bound on t can easily be derived, it follows that the trial division step of this primality test requires slightly more than polynomially many steps, namely $(\log n)^{O(\log\log\log n)}$.

For a more complete picture of this primality test, it remains to say something about those 'Fermat-like' tests. It appears that for each q dividing s a certain test has to be carried out for each prime power $p^k > 1$ dividing $q-1$, with k maximal. The easiest formulation of these tests involves a huge exponentiation of so-called Gauss sums and can be found in [**9**], or in various survey articles on primality testing like [**10, 14, 28, 29**]. In practice this formulation would not lead to a fast test: for a pair (p^k, q) the Gauss sum to be considered is an element of $\mathbf{Z}[\zeta_{p^k}, \zeta_q]$, where ζ_m denotes a primitive mth root of unity. For $q = 2521$ and $p^k = 8$, one of the pairs that occurs for $t = 5040$, this would mean manipulation of elements of $\mathbf{Z}[\zeta_8, \zeta_{2521}]$, something that looks hardly appealing.

As explained in [**9**], the tests involving Gauss sums can be replaced by tests involving Jacobi sums. From a computational point of view Jacobi sums are much more attractive than Gauss sums, because they belong to $\mathbf{Z}[\zeta_{p^k}]$ instead of $\mathbf{Z}[\zeta_{p^k}, \zeta_q]$. Notice that p^k divides t and is therefore reasonably small. The resulting *Jacobi sum test* is quite efficient and achieves the results that were mentioned in the introduction (cf. [**8**]).

The implementation of the Jacobi sum test as described in [**8**] can be improved in several major ways. In the first place, as was already noted in [**9**], a Jacobi sum test can be carried out in an extension of degree order $(n \bmod p^k)$ instead of in $\mathbf{Z}[\zeta_{p^k}]/n\mathbf{Z}[\zeta_{p^k}]$. In practice the degree of this extension will therefore often turn out to be lower, although this of course cannot be guaranteed.

In the second place, various Jacobi sum tests can be combined into one test: a test in an extension of degree order $(n \bmod p_1^{k_1}) = u_1$ and a test in an extension of degree order $(n \bmod p_2^{k_2}) = u_2$ can be combined into one test in an extension of degree $\operatorname{lcm}(u_1, u_2)$ if at least $p_1 \neq p_2$. In practice this means that a test in an extension of some high degree u takes care of

various other tests that would have to be carried out in extensions of degrees dividing u (and with different p's).

In the third place, the implementation in [8] does not incorporate Theorem (2.4) but just selects $s > \sqrt{n}$. For large n, say around 1000 digits, it turns out to be faster to take s as small as possible, but of course still $> \sqrt[3]{n}$. Although per i the trial division step becomes more time consuming because of the application of (2.4), this is more than outweighed by the smaller value for t and the smaller number of exponentiations to be performed.

Finally, the Jacobi sum test can be combined with the classical tests to a much higher extent than was done in [8]. The practical consequence of this improvement is that any factor of $n^w - 1$, for small w, can be used to relieve the condition on s and thus make the test faster. The reader is referred to [5] for a detailed description of all these improvements.

Thus, the Jacobi sum test consists of various 'Fermat-like' tests, followed by a trial division stage. The tests can be carried out independently of each other, and the trial division interval $\{1, 2, \ldots, t-1\}$ can be split up into some big number of nonoverlapping intervals. This means that the Jacobi sum test can quite easily be run, for instance on a network of a few hundred machines.

As was mentioned in the introduction, the Jacobi sum test does not provide a certificate of primality. The only way to check the computation is to redo it.

4. Primality Testing Using Elliptic Curves

The main problem with the classical primality tests is that they are tied to groups that are fixed as soon as n is fixed. If those groups turn out not to have the favorable properties that are needed to complete the primality proof, then nothing can be done about it, since changing the group would change n and consequently change the problem. For instance, while applying Theorem (2.3) our success depends entirely on the order of the group $(\mathbf{Z}/n\mathbf{Z})^*$.

In this section it will be seen that primality proofs can be given that use groups that can be chosen in a more flexible way. If the group that has been chosen does not have the right properties, then another group will be chosen, and so on, until the group satisfies the requirements of the proof.

This possibility of choosing groups such that their relevant properties are randomized in the proper way is provided by elliptic curves. Let p be a prime unequal to 2 or 3 and let $\mathbf{F}_p$ denote the finite field containing p elements. An *elliptic curve* $E = E_{a,b}$ over $\mathbf{F}_p$ is a pair $a, b \in \mathbf{F}_p$ for which $4a^3 + 27b^2 \neq 0$, to be thought of as the coefficients in the Weierstrass equation

$$y^2 = x^3 + ax + b.$$

The *set of points* $E(\mathbf{F}_p)$ of an elliptic curve E over $\mathbf{F}_p$ is defined as

$$E(\mathbf{F}_p) = \{(x, y) \in \mathbf{F}_p^2 : y^2 = x^3 + ax + b\} \cup \{O\},$$

where O is called the *zero point*. The set of points $E(\mathbf{F}_p)$ has the structure of an abelian group. The group law, which will be written additively, can be found in Carl Pomerance's contribution to this short course [**24**], or for instance in [**13, 32**]; for the purpose of this note it suffices to know that for any $P, Q \in E(\mathbf{F}_p)$ the sum $P + Q \in E(\mathbf{F}_p)$ can be computed efficiently. It follows that for any $k \in \mathbf{Z}$ and $P \in E(\mathbf{F}_p)$ the element $k \cdot P \in E(\mathbf{F}_p)$ can be computed efficiently by repeated doublings and additions in $E(\mathbf{F}_p)$.

Suppose that an elliptic curve $E = E_{a,b}$ over $\mathbf{F}_p$ has been selected at random. What can be said about the order $\#E(\mathbf{F}_p)$ of the abelian group $E(\mathbf{F}_p)$? In the first place it is known that $\#E(\mathbf{F}_p) = p + 1 - t$ for some integer t with $|t| \leq 2\sqrt{p}$ (Hasse, 1934). Furthermore, for any set S of integers s for which $|s - (p+1)| < \sqrt{p}$, the probability that a random pair $a, b \in \mathbf{F}_p$ defines a curve E over $\mathbf{F}_p$ for which $\#E(\mathbf{F}_p) \in S$ is essentially equal to the probability that a random integer near $p + 1$ is in S. More precisely, this probability lies between

$$\frac{\#S - 2}{2[\sqrt{p}] + 1} \cdot c_1 (\log p)^{-1} \quad \text{and} \quad \frac{\#S}{2[\sqrt{p}] + 1} \cdot c_2 (\log p) \cdot (\log \log p)^2,$$

where c_1 and c_2 are positive constants independent of the choice of p (cf. [**16**]).

The order of $E(\mathbf{F}_p)$ can be computed by means of a deterministic method that is guaranteed to work if p is prime, the *division point method*, due to Schoof [**30**]. Although this method runs in polynomial time, its practical value seems to be questionable.

Now let n be a positive integer with $\gcd(n, 6) = 1$. An elliptic curve $E = E_{a,b}$ over $\mathbf{Z}/n\mathbf{Z}$ is a pair $a, b \in \mathbf{Z}/n\mathbf{Z}$ for which $4a^3 + 27b^2 \in (\mathbf{Z}/n\mathbf{Z})^*$. Notice that for any prime p dividing n the pair $a \bmod p$, $b \bmod p$ defines an elliptic curve over $\mathbf{F}_p$; the set of points of the latter curve will be denoted by $E(\mathbf{F}_p)$. The set of points $E(\mathbf{Z}/n\mathbf{Z})$ can be defined in a similar fashion as above, and on this set of points a "pseudoaddition" can be defined that has the following properties (cf. [**13, 21**]). When applied to $P, Q \in E(\mathbf{Z}/n\mathbf{Z})$ it either yields a nontrivial divisor of n, or it yields an element $R \in E(\mathbf{Z}/n\mathbf{Z})$ such that $R_p = P_p + Q_p$ in $E(\mathbf{F}_p)$ for any prime divisor p of n. Here $P_p \in E(\mathbf{F}_p)$ is obtained from $P \in E(\mathbf{Z}/n\mathbf{Z})$ by reducing its coordinates modulo p.

These definitions make it possible to replace in Theorem (2.3) the order of the fixed group $(\mathbf{Z}/n\mathbf{Z})^*$, i.e., $n - 1$ if n is prime, by the order of $E(\mathbf{Z}/n\mathbf{Z})$, which will be the order of $E(\mathbf{F}_n)$ if n is prime. For a randomly chosen curve E and prime n, this order will behave as a random integer near $n + 1$, and the choice of E can be repeated until $E(\mathbf{F}_n)$ has the favorable properties required. The following analogue of Theorem (2.3) can be formulated (cf. [**11, 13, 21**]).

(4.1) THEOREM. *Let $n > 1$ be an integer with $\gcd(n, 6) = 1$. Let E be an elliptic curve modulo n, and let m and s be positive integers with s dividing*

m. Suppose there is a point $P \in E(\mathbf{Z}/n\mathbf{Z})$ satisfying

$$m \cdot P = O,$$

$(m/q) \cdot P$ is defined and different from O, for each prime q dividing s.

Then $\#E(\mathbf{F}_p) \equiv 0 \bmod s$ for every prime p dividing n, and if $s > (n^{1/4}+1)^2$ then n is prime.

PROOF. Let p be a prime dividing n, and let $Q = (m/s) \cdot P_p \in E(\mathbf{F}_p)$. From $m \cdot P = O$ and p divides n it follows that $s \cdot Q = m \cdot P_p = (m \cdot P)_p = O_p$, so the order of Q divides s. Let q be a prime dividing s. From the fact that $(m/q) \cdot P$ is defined and different from O, it follows that $(s/q) \cdot Q = (m/q) \cdot P_p = ((m/q) \cdot P)_p \neq O_p$, so that the order of Q is not a divisor of s/q, for any prime q dividing s. The order of Q therefore equals s. It follows that $\#E(\mathbf{F}_p) \equiv 0 \bmod s$.

From $\#E(\mathbf{F}_p) = p + 1 - t$ for some integer t with $|t| \leq 2\sqrt{p}$ it follows that $(p^{1/2}+1)^2 \geq \#E(\mathbf{F}_p)$. The terms $s > (n^{1/4}+1)^2$ and $\#E(\mathbf{F}_p) \equiv 0 \bmod s$ imply that $p > \sqrt{n}$, for any prime p dividing n, so that n must be prime. This proves the theorem.

Combined with the ideas mentioned above, this theorem leads to the following primality test.

(4.2) A PRIMALITY TEST BASED ON ELLIPTIC CURVES. Select an elliptic curve $E = E_{a,b}$ over $\mathbf{Z}/n\mathbf{Z}$ and an integer m such that

$m = \#E(\mathbf{F}_n)$ if n is prime, and
$m = k \cdot q$ for a small integer $k > 1$ and probable prime $q > (n^{1/4}+1)^2$.

Given the pair E, m, select a point $P \in E(\mathbf{Z}/n\mathbf{Z})$ satisfying the requirements of (4.1) with $s = q$ by performing steps (i), (ii), and (iii).

(i) Randomly select an x in $\mathbf{Z}/n\mathbf{Z}$ until $x^3 + ax + b$ is a square in $\mathbf{Z}/n\mathbf{Z}$. This can be done by checking whether $(x^3+ax+b)^{(n-1)/2} = 1$ because n is suspected to be a prime. Determine y as a zero of the polynomial $X^2 - (x^3 + ax + b) \in (\mathbf{Z}/n\mathbf{Z})[X]$ using for instance a probabilistic method for finding roots of polynomials over finite fields (cf. [**19**]); again, for this method to work, a proof of the primality of n is not needed. Put $P = (x, y)$ in $E(\mathbf{Z}/n\mathbf{Z})$.

(ii) Compute $(m/q) \cdot P = k \cdot P$. If $k \cdot P$ is undefined, a nontrivial divisor of n has been found, which is exceedingly unlikely. If $k \cdot P = O$, then go back to (i); this happens with probability $< \frac{1}{2}$ if n is prime. Otherwise, if $k \cdot P \neq O$, verify that $q \cdot (k \cdot P) = m \cdot P = O$, which must be the case if n is prime, because then $\#E(\mathbf{F}_n) = m$.

(iii) Prove the primality of q recursively using (4.2), unless the primality of q can be proved directly using some other method like DOWN-RUN (cf. §2).

Upon completion of the primality test in (4.2) a chain $(n = n_0, E_0, m_0, P_0)$, $(n_1, E_1, m_1, P_1), \ldots, (n_{t-1}, E_{t-1}, m_{t-1}, P_{t-1})$, n_t has been computed such that the n_i are primes, each $m_i = \#E_i(\mathbf{F}_{n_i})$ is a small multiple of n_{i+1}, and $(m_i/n_{i+1}) \cdot P_i$ is defined and different from O, where $P_i \in E_i(\mathbf{F}_{n_i})$. This chain is a certificate for the primality of n that can be verified much more easily than it can be constructed (cf. Theorem (4.1)).

It remains to explain how to select the pair E, m in (4.2). Goldwasser and Kilian, to whom the original idea of (4.2) is due, proposed to do this by performing the following three steps (cf. [**11**]).

> Randomly select $a, b \in \mathbf{Z}/n\mathbf{Z}$ until both $4a^3 + 27b^2 \neq 0$ and $\gcd(n, 4a^3 + 27b^2) = 1$. If n is prime, the probability of success per trial is $(n-1)/n$. Consider the elliptic curve $E = E_{a,b}$ over $\mathbf{Z}/n\mathbf{Z}$.
>
> Use Schoof's division point method to compute a number m that will be equal to $\#E(\mathbf{F}_n)$ if n is prime. If the division point method does not succeed in computing anything, then apparently n is not prime.
>
> If m is not of the form $k \cdot q$ for some small integer $k > 1$ and probable prime q, then return to the first step; otherwise the pair E, m has been computed successfully.

What about the expected running time of this *random curve test*, i.e., (4.2) combined with Goldwasser and Kilian's way of selecting E and m? Because $m = \#E(\mathbf{F}_n) \leq n + 1 + 2\sqrt{n}$ and $k > 1$, the next prime in the chain is $\leq (n + 1 + 2\sqrt{n})/2$, so that the length of the chain as produced by the random curve test is $O(\log n)$. If n is indeed prime, $E(\mathbf{F}_n)$ behaves approximately as a random number near n, so that $O(\log n)$ choices for E should suffice to find a pair E, m: the probability of hitting a small nontrivial multiple of a prime should be of the same order as the probability of hitting a prime near n, and is therefore of the order $(\log n)^{-1}$. Because all computations, including the applications of the division point method, can be carried out in (expected) polynomial time, this heuristic argument would lead to the conclusion that the expected running time of the random curve test is polynomial in $\log n$.

Unfortunately, this cannot yet be proved rigorously. What can be proved is that, if there is a positive constant c such that for all $x \in \mathbf{R}_{\geq 2}$ the number of primes between x and $x + \sqrt{2x}$ is of the order $\sqrt{x}(\log x)^{-c}$, then the random curve test runs in expected time $O((\log n)^{9+c})$ (cf. [**11**]). For a further discussion of this point, the reader is referred to [**11**] and [**1**].

So, the main obstacle in proving the expected polynomial time behavior of the random curve test is that it cannot be proved that an interval of length $O(\sqrt{x})$ around x contains sufficiently many primes. For a slightly bigger interval, namely $O(x^{3/4})$, this can be proved (cf. [**12**]). This fact is used by Adleman and Huang in their *abelian variety test*, which can be guaranteed to run in expected polynomial time (cf. [**1**]). Like the random curve test, the abelian variety test builds a chain-like certificate of primality. A remarkable

feature of the resulting chain of primes is that initially the primes get bigger and bigger; i.e., the recursion goes in the wrong direction. It can be proved, however, that after a few iterations the chain is expected to hit upon a prime that can be proved to be prime using the random curve test, so that from that point on the primes in the chain shrink again. For a detailed description of the abelian variety test, see [**1**].

From a practical point of view the random curve test can be improved as well. Although the test is quite likely to run in polynomial time, it will not be very fast because it makes use of the division point method. Atkin therefore proposed to select the pair E, m in (4.2) in a different and considerably more complicated way. The practical performance of the resulting algorithm, however, the *complex multiplication test*, is quite impressive (cf. [**21**]).

The *complex multiplication field* of an elliptic curve E over a finite field $\mathbf{F}_p$ with $\#E(\mathbf{F}_p) = p + 1 - t$ is defined as the imaginary quadratic field $L = \mathbf{Q}(\sqrt{t^2 - 4p})$. The complex multiplication test is based on the following two observations:

(4.3) If the complex multiplication field L of an elliptic curve E over $\mathbf{F}_p$ is known, then $m = \#E(\mathbf{F}_p)$ can quite easily be computed. But even if only L and p are known and E is not known, then a small list of candidate m's can be computed for those elliptic curves over $\mathbf{F}_p$ that would have L as their complex multiplication field.

(4.4) Given some imaginary quadratic field $\mathbf{Q}(\sqrt{\Delta})$ and a prime p, a small list of elliptic curves over $\mathbf{F}_p$ having $\mathbf{Q}(\sqrt{\Delta})$ as their complex multiplication field can be constructed.

In the application of (4.3) and (4.4) in the complex multiplication test the role of p will be played by n; both (4.3) and (4.4) will work if n is prime, but they do not need a proof of the primality of n. The complex multiplication test combines (4.2) with the following way of selecting the pair E, m:

Select some imaginary quadratic field $L = \mathbf{Q}(\sqrt{\Delta})$ that has not yet been tried in the primality proof for this n.

Given L, compute a list of candidate m's for the elliptic curves having L as their multiplication field (cf. (4.3)).

If none of the m's on the list is of the form $k \cdot q$ with $k > 1$ and q a probable prime $> (n^{1/4} + 1)^2$, then go back to the first step.

Otherwise, let m have the proper form. Compute a small list of elliptic curves over $\mathbf{Z}/n\mathbf{Z}$ corresponding to L (cf. (4.4)), and select E from this list such that $\#E(\mathbf{F}_n) = m$, if n were prime. The selection of E can be done by selecting points on the curves (cf. (4.2)(i)) until only one pair P, E is left for which $m \cdot P = O$.

Thus, in the complex multiplication test the elliptic curve will only be constructed if the cardinality of its set of points satisfies the requirements of (4.2). The many details of the complex multiplication test will be left

untouched here; they can be found in [**21**]. A heuristic argument that the expected running time of the complex multiplication test is polynomial in $\log n$ (actually, $O((\log n)^{6+\varepsilon})$ for any $\varepsilon > 0$) can be found in [**13**]. Several people are working on improving this method, so it is not unlikely that more references will appear soon.

Other primality tests that make use of elliptic curves can be found in [**4**] and [**7**].

References

1. L. M. Adleman and M. A. Huang, *Recognizing primes in random polynomial time*, research report, Department of Computer Science, University of Southern California, 1988; extended abstract in Proc. of the 19th ACM Symp. on Theory of Computing 1987, 462–469.
2. L. M. Adleman, C. Pomerance, and R. S. Rumely, *On distinguishing prime numbers from composite numbers*, Ann. of Math. **117** (1983), 173–206.
3. E. Bach, *Analytic methods in the analysis and design of number-theoretic algorithms*, MIT Press, Cambridge, MA, 1985.
4. W. Bosma, *Primality testing using elliptic curves*, report 85–12, Mathematisch Instituut, Universiteit van Amsterdam, 1985.
5. W. Bosma and M.-P. van der Hulst, *Primality proving with cyclotomy*, Ph.D. thesis, Amsterdam, November 1990.
6. J. Brillhart, D. H. Lehmer, J. L. Selfridge, B. Tuckerman, and S. S. Wagstaff, Jr., *Factorizations of* $b^n \pm 1$, $b = 2, 3, 5, 6, 7, 10, 11, 12$ *up to high powers*, 2nd ed., Contemporary Mathematics, vol. 22, Amer. Math. Soc., Providence, RI, 1988.
7. D. V. Chudnovsky and G. V. Chudnovsky, *Sequences of numbers generated by addition in formal groups and new primality and factorization tests*, Adv. Appl. Math. **7** (1986), 187–237.
8. H. Cohen and A. K. Lenstra, *Implementation of a new primality test*, Math. Comp. **48** (1987), 103–121.
9. H. Cohen and H. W. Lenstra, Jr., *Primality testing and Jacobi sums*, Math. Comp. **42** (1984), 297–330.
10. J. D. Dixon, *Factorization and primality testing*, Amer. Math. Monthly **91** (1984), 333–352.
11. S. Goldwasser and J. Kilian, *Almost all primes can be quickly certified*, Proc. 18th Annual ACM Symp. on Theory of Computing, 1986, 316–329.
12. H. Iwaniec and M. Jutila, *Primes in short intervals*, Ark. Mat. **17** (1979), 167–176.
13. A. K. Lenstra and H. W. Lenstra, Jr., *Algorithms in number theory*, in Handbook of theoretical computer science (J. van Leeuwen ed.), Elsevier, Amsterdam, 1990, 673–715.
14. H. W. Lenstra, Jr., *Primality testing*, in [**18**].
15. ———, *Divisors in residue classes*, Math. Comp. **42** (1984), 331–340.
16. ———, *Factoring integers with elliptic curves*, Ann. of Math. **126** (1987), 649–673.
17. ———, *Elliptic curves and number-theoretic algorithms*, Proc. Internat. Congr. of Math. (Berkeley 1986), Amer. Math. Soc., Providence, RI, 1988, pp. 99–120.
18. H. W. Lenstra, Jr., and R. Tijdeman (eds.), *Computational methods in number theory*, Math. Centre Tracts **154/155**, Mathematisch Centrum, Amsterdam, 1982.
19. D. E. Knuth, *The art of computer programming*, vol. 2, Seminumerical algorithms, 2nd ed., Addison-Wesley, Reading, MA, 1981.
20. G. L. Miller, *Riemann's hypothesis and tests for primality*, J. Comput. System Sci. **13** (1976), 300–317.
21. F. Morain, *Implementation of the Goldwasser-Kilian-Atkin primality testing algorithm*, INRIA report 911, 1988.
22. H. C. Pocklington, *The determination of the prime and composite nature of large numbers by Fermat's theorem*, Proc. Cambridge Philos. Soc. **18** (1914–1916), 29–30.
23. C. Pomerance, *Recent developments in primality testing*, Math. Intelligencer **3** (1981), 97–105.

24. ____, *Factoring*, this volume.
25. C. Pomerance, J. L. Selfridge, and S. S. Wagstaff, Jr., *The pseudoprimes to* $25 \cdot 10^9$, Math. Comp. **35** (1980), 1003–1026.
26. M. O. Rabin, *Probabilistic algorithms for testing primality*, J. Number Theory **12** (1980), 128–138.
27. P. Ribenboim, *The book of prime number records*, Springer-Verlag, New York, 1988.
28. H. Riesel, *Prime numbers and computer methods for factorization*, Progr. Math., vol. 57, Birkhäuser, Boston, 1985.
29. R. Rumely, *Recent advances in primality testing*, Notices Amer. Math. Soc. **30** (1983), 475–477.
30. R. J. Schoof, *Elliptic curves over finite fields and the computation of square roots* mod p, Math. Comp. **44** (1985), 483–494.
31. J. L. Selfridge and M. C. Wunderlich, *An efficient algorithm for testing large numbers for primality*, Proc. Fourth Manitoba Conf. Numer. Math., University of Manitoba, Congressus Numerantium XII, Utilitas Math., Winnipeg, 1975, pp. 109–120.
32. J. H. Silverman, *The arithmetic of elliptic curves*, Graduate Texts in Math., vol. 106, Springer-Verlag, New York, 1986.
33. H. C. Williams, *Primality testing on a computer*, Ars Combin. **5** (1978), 127–185.
34. H. C. Williams and H. Dubner, *The primality of R1031*, Math. Comp. **47** (1986), 703–711.

ROOM 2Q334, BELL COMMUNICATIONS RESEARCH, 445 SOUTH STREET, MORRISTOWN, NJ 07960

E-mail address: LENSTRA@BELLCORE.COM