MATHEMATICS OF COMPUTATION
VOLUME 50, NUMBER 181
JANUARY 1988, PAGES 235-250

Polynomial Factorization and Nonrandomness of Bits of Algebraic and Some Transcendental Numbers

By R. Kannan, A. K. Lenstra, and L. Lovász

Abstract. We show that the binary expansions of algebraic numbers do not form secure pseudorandom sequences; given sufficiently many initial bits of an algebraic number, its minimal polynomial can be reconstructed, and therefore the further bits of the algebraic number can be computed. This also enables us to devise a simple algorithm to factor polynomials with rational coefficients. All algorithms work in polynomial time.

Introduction. Manuel Blum raised the following question: Suppose we are given an approximate root of an unknown polynomial with integral coefficients and a bound on the degree and size of the coefficients of the polynomial. Is it possible to infer the polynomial? We answer his question in the affirmative. We show that if a complex number α satisfies an irreducible polynomial $h(X)$ of degree d with integral coefficients in absolute value at most H, then given $O(d^2 + d \cdot \log H)$ bits of the binary expansion of the real and complex parts of α, we can find $h(X)$ in deterministic polynomial time (and then compute in polynomial time any further bits of α). Using the concept of secure pseudorandom sequences formulated by Shamir [23], Blum and Micali [3] and Yao [25], we then show that the binary (or m-ary for any m) expansions of algebraic numbers do not form secure sequences in a certain well-defined sense.

We are able to extend our results with the same techniques to transcendental numbers of the form $\log(\alpha), \cos^{-1}(\alpha)$, etc., where α is algebraic.

The technique is based on the lattice basis reduction algorithm from [16]. Our answer to Blum's question enables us to devise a simple polynomial-time algorithm to factor polynomials with rational coefficients: We find an approximate root of the polynomial and use our algorithm to find the irreducible polynomial satisfied by the exact root, which must then be a factor of the given polynomial. This is repeated until all the factors are found. This algorithm was found independently by Schönhage [22], and was already suggested in [16].

The technique of the paper also provides a natural, efficient method to compute with algebraic numbers.

This paper is the final journal version of [13], which contains essentially the entire contents of this paper.

1. A Polynomial-Time Algorithm for Blum's Question. Throughout this paper, **Z** denotes the set of the integers, **Q** the set of the rationals, **R** the set of the reals, and **C** the set of the complex numbers. The ring of polynomials with integral

Received December 23, 1985; revised October 13, 1986 and April 6, 1987.
1980 *Mathematics Subject Classification* (1985 *Revision*). Primary 68Q15, 68Q25, 68Q40.

(complex) coefficients will be denoted $\mathbf{Z}[X]$ ($\mathbf{C}[X]$). The *content* of a polynomial $p(X)$ in $\mathbf{Z}[X]$ is the greatest common divisor (abbreviated gcd) of its coefficients. A polynomial in $\mathbf{Z}[X]$ is *primitive* if its content is 1. A polynomial $p(X)$ *has degree d* if $p(X) = \sum_{i=0}^{d} p_i X^i$ with $p_d \neq 0$. We write $\deg(p) = d$. The *length* $|p|$ of $p(X) = \sum_{i=0}^{d} p_i X^i$ is the Euclidean length of the vector $(p_0, p_1, \ldots, p_d)$; the *height* $|p|_\infty$ of $p(X)$ is the L_∞-norm of the vector $(p_0, p_1, \ldots, p_d)$, so $|p|_\infty = \max_{0 \leq i \leq d} |p_i|$. An *algebraic number* is a root of a polynomial with integral coefficients. The *minimal polynomial* of an algebraic number α is the irreducible polynomial in $\mathbf{Z}[X]$ satisfied by α. The minimal polynomial is unique up to units in $\mathbf{Z}$ (see, for example, [11]). The *degree* and *height* of an algebraic number are the degree and height, respectively, of its minimal polynomial. The real and complex parts of a complex number z will be denoted $\mathrm{Re}(z)$ and $\mathrm{Im}(z)$ respectively.

A *lattice* in $\mathbf{R}^n$ is a set of the form

$$\left\{ \sum_{i=1}^{k} \lambda_i b_i \colon \lambda_i \in \mathbf{Z} \right\},$$

where $b_1, b_2, \ldots, b_k$ are linearly independent vectors in $\mathbf{R}^n$. The lattice is said to be generated by the vectors $b_1, b_2, \ldots, b_k$, which form a *basis* for the lattice. The lattice is denoted $L(b_1, b_2, \ldots, b_k)$. An important result we need is the basis reduction algorithm from [16, Section 1]. We will only state the consequence of this algorithm used in this paper. Denote by $|\cdot|$ the ordinary Euclidean length on $\mathbf{R}^n$.

(1.1) THEOREM (cf. [16, Propositions (1.11) and (1.26)]). *Let*

$$L = L(b_1, b_2, \ldots, b_k)$$

be a lattice in $\mathbf{Z}^n$ and let $B \in \mathbf{R}$, $B \geq 2$, be such that $|b_i|^2 \leq B$ for $1 \leq i \leq k$. It takes $O(n \cdot k^3 \cdot \log B)$ arithmetic operations (additions, subtractions, multiplications, and divisions) on integers having $O(k \cdot \log B)$ binary bits to transform the basis $b_1, b_2, \ldots, b_k$ by means of the basis reduction algorithm into a reduced basis $v_1, v_2, \ldots, v_k$ for L. The first vector v_1 in the reduced basis has length at most $2^{(k-1)/2} \cdot \Lambda_1(L)$, where $\Lambda_1(L)$ is the length of a shortest nonzero vector in L.

Now we are ready to describe the idea behind our main result. Suppose upper bounds d and H on the degree and height, respectively, of an algebraic number α are known. Then we show that a sufficiently close rational approximation $\bar{\alpha}$ to α enables us to determine the minimal polynomial $h(X)$ of α.

Given $\bar{\alpha}$, we compute rational approximations $\bar{\alpha}_i$ to the powers α^i of α. For a polynomial $g = \sum_i g_i X^i \in \mathbf{C}[X]$ we introduce the following notation for the approximated evaluation of g at α:

$$g_{\bar{\alpha}} = \sum_i g_i \bar{\alpha}_i. \tag{1.2}$$

Suppose the degree of $h(X)$ is n, $n \leq d$. We try the values of $n = 1, 2, \ldots, d$ in order. With n fixed, we define for each positive integer s the lattice L_s in $\mathbf{R}^{n+3}$ generated by $b_0, b_1, \ldots, b_n$, which are the rows (in order) of the following $(n+1) \times (n+3)$

matrix:

$$(1.3)\qquad \begin{bmatrix} 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 & 2^s\cdot\mathrm{Re}(\bar{\alpha}_0) & 2^s\cdot\mathrm{Im}(\bar{\alpha}_0) \\ 0 & 1 & 0 & \cdot & \cdot & \cdot & 0 & 2^s\cdot\mathrm{Re}(\bar{\alpha}_1) & 2^s\cdot\mathrm{Im}(\bar{\alpha}_1) \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 & 2^s\cdot\mathrm{Re}(\bar{\alpha}_2) & 2^s\cdot\mathrm{Im}(\bar{\alpha}_2) \\ \cdot & \cdot & \cdot & \cdot & & & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & & & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & & & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & 1 & 2^s\cdot\mathrm{Re}(\bar{\alpha}_n) & 2^s\cdot\mathrm{Im}(\bar{\alpha}_n) \end{bmatrix}.$$

Corresponding to a polynomial $g = \sum_{i=0}^{n} g_i X^i$ in $\mathbf{Z}[X]$ of degree at most n (where some of the g_i are possibly zero), we have a vector $\tilde{g}$ in the lattice L_s defined by

$$(1.4)\qquad \tilde{g} = \sum_{i=0}^{n} g_i b_i.$$

Clearly,

$$|\tilde{g}|^2 = g_0^2 + g_1^2 + \cdots + g_n^2 + 2^{2s}\left(\mathrm{Re}\left(\sum_{i=0}^{n} g_i\bar{\alpha}_i\right)\right)^2 + 2^{2s}\left(\mathrm{Im}\left(\sum_{i=0}^{n} g_i\bar{\alpha}_i\right)\right)^2$$
$$= |g|^2 + 2^{2s}|g_{\bar{\alpha}}|^2.$$

This correspondence between polynomials in $\mathbf{Z}[X]$ of degree at most n and vectors in the lattice L_s is easily seen to be 1-1 onto and readily invertible. We will strongly separate the minimal polynomial $h(X)$ of α from all other polynomials $g(X)$ of degree n or less with $g(\alpha) \neq 0$ by showing that for a suitable choice of s and small enough $|\alpha^i - \bar{\alpha}_i|$,

$$|\tilde{g}|^2 > 2^n|\tilde{h}|^2.$$

We run the basis reduction algorithm on $b_0, b_1, \ldots, b_n$ to get a reduced basis. Suppose $\tilde{v}$ is the first vector of this basis, and $v(X)$ the corresponding polynomial. Because the degree of h was supposed to be equal to n, we have that $\tilde{h}$ is contained in L_s, so that $\Lambda_1(L_s) \leq |\tilde{h}|$. Theorem (1.1) now yields $|\tilde{v}|^2 \leq 2^n|\tilde{h}|^2$, and therefore $v(\alpha) = 0$ by the strong separation. This implies that h is a factor of v. Combining this with $\deg(v) \leq \deg(h)$, we see that v and h are associates; further, the fact that $\tilde{v}$ belongs to a basis for L_s implies that $v = \pm h$.

The s needed will be bounded by a polynomial function of d and $\log H$. Here is a short intuitive description of how the strong separation is proved. If the powers of α are sufficiently close to the $\bar{\alpha}_i$, clearly $h_{\bar{\alpha}}$ is close to $h(\alpha) = 0$ (quantified in Lemma (1.5)). Thus $|\tilde{h}|^2 = |h|^2$+(a small term) and can be bounded above. To show that $|\tilde{g}|^2$ is large for other g, we consider two cases: If $|g|$ is large, then of course $|\tilde{g}|$ is large. If $|g|$ is small, then we show that $|g(\alpha)|$ has to be bounded from below (Proposition (1.6)). Again, $|g_{\bar{\alpha}}|$ being close to $|g(\alpha)|$, we are able to bound it from below and hence bound also $|\tilde{g}|^2$ from below.

(1.5) LEMMA. *If α and $\bar{\alpha}_i$ for $0 \leq i \leq n$ are complex numbers such that $\bar{\alpha}_0 = 1$, and $|\alpha^i - \bar{\alpha}_i| \leq \varepsilon$ for $1 \leq i \leq n$ and f is a polynomial of degree at most n in* $\mathbf{C}[X]$, *then*

$$|f(\alpha) - f_{\bar{\alpha}}| \leq \varepsilon \cdot n \cdot |f|_\infty.$$

Proof. Immediate.

(1.6) PROPOSITION. *Let h and g be nonzero polynomials in $\mathbf{Z}[X]$ of degrees n and m, respectively, and let $\alpha \in \mathbf{C}$ be a zero of h with $|\alpha| \leq 1$. If h is irreducible and $g(\alpha) \neq 0$ then*

$$|g(\alpha)| \geq n^{-1} \cdot |h|^{-m} \cdot |g|^{-n+1}.$$

Proof. Because h is nonzero and α is a zero of h we have that $n \geq 1$. If $m = 0$, then $g(\alpha) = |g|$, so that the result follows. Now assume that $m \neq 0$. Define the $(n+m) \times (n+m)$ matrix M as the matrix having ith column $X^{i-1} \cdot h$ for $1 \leq i \leq m$, and $X^{i-m-1} \cdot g$ for $m+1 \leq i \leq n+m$, where the polynomials $X^{i-1} \cdot h$ and $X^{i-m-1} \cdot g$ are regarded as $(n+m)$-dimensional vectors. By R we denote the absolute value of the determinant of M, the so-called *resultant* of h and g.

We prove that this resultant R is nonzero. Suppose on the contrary that the determinant of M is zero. This implies that a linear combination of the columns of M is zero, so that there exist polynomials $a, b \in \mathbf{Z}[X]$ with degree$(a) < m$ and degree$(b) < n$ such that $a \cdot h + b \cdot g = 0$. Because h is irreducible, any nontrivial common factor of h and g must have α as a zero, so that with $g(\alpha) \neq 0$ we have that $\gcd(h, g) = 1$. Therefore, we have that h divides b, so that with degree$(b) < n$, we find $b = 0$, and also $a = 0$. This proves that the columns of M are linearly independent, so that $R \neq 0$. Because the entries of M are integral, we even have $R \geq 1$.

We add, for $2 \leq i \leq n + m$, the ith row of M times T^{i-1} to the first row of M. The first row of M then becomes $(h(T), T \cdot h(T), \ldots, T^{m-1} \cdot h(T), g(T), T \cdot g(T), \ldots, T^{n-1} \cdot g(T))$. Expanding the determinant of M with respect to the first row, we find that

$$R = |h(T) \cdot (a_0 + a_1 \cdot T + \cdots + a_{m-1} \cdot T^{m-1}) + g(T) \cdot (b_0 + b_1 \cdot T + \cdots + b_{n-1} \cdot T^{n-1})|,$$

where the a_i and b_j are determinants of $(n+m-1) \times (n+m-1)$ submatrices of M. Evaluating the above identity for $T = \alpha$ yields

$$R = |g(\alpha)| \cdot |b_0 + b_1 \cdot \alpha + \cdots + b_{n-1} \cdot \alpha^{n-1}|, \tag{1.7}$$

because $h(\alpha) = 0$. From Hadamard's inequality it follows that $|b_j| \leq |h|^m \cdot |g|^{n-1}$. Combining this with $|\alpha| \leq 1$ we get

$$|b_0 + b_1 \cdot \alpha + \cdots + b_{n-1} \cdot \alpha^{n-1}| \leq n \cdot |h|^m \cdot |g|^{n-1},$$

so that (1.6) follows from (1.7) and $R \geq 1$:

$$|g(\alpha)| \geq n^{-1} \cdot |h|^{-m} \cdot |g|^{-n+1}.$$

This proves Proposition (1.6).

(1.8) *Remark.* Proposition (1.6) implies that two algebraic numbers that are not conjugates (conjugates are roots of the same irreducible polynomial in $\mathbf{Z}[X]$) cannot get very close. More precisely, suppose α and β satisfy distinct irreducible primitive polynomials $h(X)$ and $g(X)$, respectively, in $\mathbf{Z}[X]$, each of degree at most n. Without loss of generality suppose that $|\beta| < |\alpha| \leq 1$, and let $|\alpha - \beta|$ be γ. It is easy to see that $|g(\alpha) - g(\beta)| \leq \gamma \cdot |g|_\infty \cdot n(n-1)/2$. Now a lower bound on γ follows from Proposition (1.6). This kind of separation result also holds if α and β are conjugates (see for instance [21, Section 20]).

(1.9) LEMMA. *Suppose α is a complex number with $|\alpha| \leq 1$ and with minimal polynomial h of degree at most $d \geq 1$ and height at most H, and suppose $\bar{\alpha}_i$ satisfies $\bar{\alpha}_0 = 1$ and $|\alpha^i - \bar{\alpha}_i| \leq 2^{-s}$ for $1 \leq i \leq d$. Let g be a polynomial with integral coefficients of degree at most d such that $g(\alpha) \neq 0$. Then with the notation introduced in* (1.4), *the following inequalities hold*:

$$|\tilde{h}| \leq (d+1) \cdot H, \tag{1.10}$$

$$|\tilde{g}| > 2^{d/2} \cdot (d+1) \cdot H, \tag{1.11}$$

provided

$$2^s \geq 2^{d^2/2} \cdot (d+1)^{(3d+4)/2} \cdot H^{2d}. \tag{1.12}$$

Proof. First notice that

$$|f|^2 \leq (d+1) \cdot |f|_\infty^2 \tag{1.13}$$

holds for any polynomial f of degree at most d. To prove (1.10), we combine $|\tilde{h}|^2 = |h|^2 + 2^{2s}|h_{\bar{\alpha}}|^2$ and $|h_{\bar{\alpha}}| = |h(\alpha) - h_{\bar{\alpha}}| \leq 2^{-s} \cdot d \cdot H$ (Lemma (1.5)):

$$\begin{aligned} |\tilde{h}|^2 &\leq |h|^2 + d^2 \cdot H^2 \\ &\leq (d+1) \cdot H^2 + d^2 \cdot H^2 \quad \text{(cf. (1.13))} \\ &< (d+1)^2 \cdot H^2. \end{aligned}$$

This proves (1.10). We now prove (1.11). Clearly, if $|g| > 2^{d/2} \cdot (d+1) \cdot H$, we are done because $|\tilde{g}|^2 = |g|^2 + 2^{2s}|g_{\bar{\alpha}}|^2$. So assume $|g| \leq 2^{d/2} \cdot (d+1) \cdot H$. By Proposition (1.6) and (1.13),

$$\begin{aligned} g(\alpha) &\geq d^{-1} \cdot ((d+1) \cdot H^2)^{-d/2} \cdot (2^{d/2} \cdot (d+1) \cdot H)^{-d+1} \\ &> 2^{-d(d-1)/2} \cdot (d+1)^{-3d/2} \cdot H^{-2d+1}, \end{aligned}$$

so that, with Lemma (1.5) and $|\alpha^i - \bar{\alpha}_i| \leq 2^{-s}$:

$$\begin{aligned} |\tilde{g}| &\geq 2^s \cdot |g_{\bar{\alpha}}| \\ &\geq 2^s \cdot \left(2^{-d(d-1)/2} \cdot (d+1)^{-3d/2} \cdot H^{-2d+1} - 2^{-s} \cdot d \cdot |g|_\infty\right) \\ &= 2^s \cdot 2^{-d(d-1)/2} \cdot (d+1)^{-3d/2} \cdot H^{-2d+1} - d \cdot |g|_\infty. \end{aligned} \tag{1.14}$$

From (1.12) and $|2^{d/2} \cdot (d+1) \cdot H| \geq |g| \geq |g|_\infty$ we get

$$\begin{aligned} &2^s \cdot 2^{-d(d-1)/2} \cdot (d+1)^{-3d/2} \cdot H^{-2d+1} \\ &\quad \geq 2^{d/2} \cdot (d+1)^2 \cdot H = (d \cdot (d+1) + (d+1)) \cdot 2^{d/2} \cdot H \\ &\quad \geq d \cdot |g|_\infty + 2^{d/2} \cdot (d+1) \cdot H, \end{aligned}$$

which, combined with (1.14), yields (1.11). This proves Lemma (1.9).

(1.15) THEOREM. *Let $\alpha, h(X), d, H$, and $\bar{\alpha}_i \in 2^{-s}\mathbf{Z}\left[\sqrt{-1}\right]$, for $0 \leq i \leq d$, satisfy the hypothesis of Lemma* (1.9), *where s is such that* (1.12) *holds. Let n be an integer satisfying $1 \leq n \leq d$, and suppose that the basis reduction algorithm on input $b_0, b_1, \ldots, b_n$ defined in* (1.3) *yields a reduced basis with $\tilde{v} = \sum_{i=0}^{n} v_i b_i$ as the first vector. Then the following three assertions are equivalent*:

(i) $|\tilde{v}| \leq 2^{d/2} \cdot (d+1) \cdot H$;

(ii) *α satisfies the polynomial $v(X) = \sum_{i=0}^{n} v_i X^i$*;

(iii) *the degree of α is at most n.*

Furthermore, if n equals the degree of α, then $h(X) = \pm v(X)$.

Proof. First notice that the lattice $L_s = L(b_0, b_1, \ldots, b_n)$ is contained in $\mathbf{Z}^{n+3}$, so that Theorem (1.1) can be applied to L_s, and that the conditions for Lemma (1.9) are satisfied.

Assume (i). From Lemma (1.9) we get $v(\alpha) = 0$, which is (ii).

Next, assume (ii). Then α satisfies a polynomial of degree at most n, which is (iii).

Finally, assume (iii). This implies that h has degree at most n, so that $\tilde{h}$ is a well-defined vector in L_s. Lemma (1.9) yields $|\tilde{h}| \leq (d+1) \cdot H$, so that in the notation of Theorem (1.1) we have $\Lambda_1(L_s) \leq (d+1) \cdot H$. It then follows from Theorem (1.1) that $|\tilde{v}| \leq 2^{d/2} \cdot (d+1) \cdot H$, which is (i). This proves the equivalence of (i), (ii), and (iii).

If n equals the degree of α, then (iii) is satisfied, so that α satisfies $v(X)$ (from (ii)). Because $\deg(h) = n, \deg(v) \leq n$, and h is irreducible, we then have that v is an integral multiple of h. It follows that $h = \pm v$ because both $\tilde{h}$ and $\tilde{v}$ are contained in L_s, and because $\tilde{v}$ belongs to a basis for L_s. This proves Theorem (1.15).

This theorem leads to the following algorithm for finding the minimal polynomial of α:

(1.16) ALGORITHM MINIMAL POLYNOMIAL. Suppose we get on input upper bounds d and H on the degree and height, respectively, of an algebraic number α with $|\alpha| \leq 1$ and a complex rational number $\bar{\alpha}$ approximating α such that $|\bar{\alpha}| \leq 1$ and $|\alpha - \bar{\alpha}| \leq 2^{-s}/(4d)$, where s is the smallest positive integer such that

$$2^s \geq 2^{d^2/2} \cdot (d+1)^{(3d+4)/2} \cdot H^{2d}.$$

First compute $\bar{\alpha}_i \in 2^{-s}\mathbf{Z}\left[\sqrt{-1}\right]$, for $0 \leq i \leq d$, such that $\bar{\alpha}_0 = 1$ and $|\bar{\alpha}^i - \bar{\alpha}_i| \leq 2^{-s-1/2}$ for $1 \leq i \leq d$. This can be done by rounding the powers of $\bar{\alpha}$ to s bits after the binary point. (It is easily verified that the $\bar{\alpha}_i$ satisfy the conditions in Theorem (1.15), see Explanation (1.17).)

For $n = 1, 2, \ldots, d$ in succession we do the following:

- Apply the basis reduction algorithm to the lattice $L_s = L(b_0, b_1, \ldots, b_n)$ as defined in (1.3).
- If the first basis vector $\tilde{v}$ in the reduced basis satisfies $|\tilde{v}| \leq 2^{d/2} \cdot (d+1) \cdot H$, then let $v(X)$ be the polynomial corresponding to $\tilde{v}$ by the relation defined in (1.4), return $v(X)$ as the minimal polynomial of α, and terminate the execution of Algorithm (1.16).

This finishes the description of Algorithm (1.16).

(1.17) *Explanation.* We show that the $\bar{\alpha}_i$ for $1 \le i \le d$ satisfy the conditions in Theorem (1.15), i.e., $|\alpha^i - \bar{\alpha}_i| \le 2^{-s}$:

$$\begin{aligned} |\alpha^i - \bar{\alpha}_i| &\le |\alpha^i - \bar{\alpha}^i| + |\bar{\alpha}^i - \bar{\alpha}_i| \\ &\le |\alpha - \bar{\alpha}| \cdot \sum_{j=1}^{i} |\alpha|^{i-j} \cdot |\bar{\alpha}|^{j-1} + 2^{-s-1/2} \quad \text{(due to the rounding)} \\ &\le \frac{d}{4d} \cdot 2^{-s} + 2^{-s-1/2} \\ &< 2^{-s}. \end{aligned}$$

(1.18) *Explanation.* It is no major restriction to consider α with $|\alpha| \le 1$ only. Namely, if $\alpha \ne 0$ satisfies the polynomial $h(X) = \sum_{i=0}^{d} h_i X^i$, then $1/\alpha$ satisfies $\sum_{i=0}^{d} h_{d-i} X^i$. Furthermore, an ε-approximation $\bar{\alpha}$ to α with $|\alpha| > 1$ easily yields a $3 \cdot \varepsilon$-approximation $\bar{\beta}$ to $\beta = 1/\alpha$. Let $|\alpha - \bar{\alpha}| \le \varepsilon$ with ε such that $0 < \varepsilon \le 1/2$. Determine $\bar{\beta}$ such that $|\bar{\beta} - 1/\bar{\alpha}| < \varepsilon$; then

$$\begin{aligned} |\beta - \bar{\beta}| \le \left|\beta - \frac{1}{\bar{\alpha}}\right| + \left|\bar{\beta} - \frac{1}{\bar{\alpha}}\right| &\le \left|\frac{\alpha - \bar{\alpha}}{\alpha \cdot \bar{\alpha}}\right| + \varepsilon \\ &\le \frac{\varepsilon}{|\alpha| \cdot |\bar{\alpha}|} + \varepsilon. \end{aligned}$$

Now $|\bar{\alpha}| \ge (1 - \varepsilon)|\alpha|$, so $|\bar{\alpha}| \ge |\alpha|/2 \ge 1/2$. So $|\beta - \bar{\beta}| \le \varepsilon[2 + 1] = 3 \cdot \varepsilon$.

(1.19) THEOREM. *Let α be an algebraic number and let d and H be upper bounds on the degree and height, respectively, of α. Suppose that we are given an approximation $\bar{\alpha}$ to α such that $|\alpha - \bar{\alpha}| \le 2^{-s}/(12d)$, where s is the smallest positive integer such that*

$$2^s \ge 2^{d^2/2} \cdot (d+1)^{(3d+4)/2} \cdot H^{2d}.$$

Then the minimal polynomial of α can be determined in $O(n_0 \cdot d^4 \cdot (d + \log H))$ arithmetic operations on integers having $O(d^2 \cdot (d + \log H))$ binary bits, where n_0 is the degree of α.

Proof. In order to be able to apply Algorithm (1.16), we replace α by $1/\alpha$ if necessary. It follows from Explanation (1.18) that $\bar{\alpha}$ then yields an approximation $\bar{\beta}$ to $\beta = 1/\alpha$ such that $|\beta - \bar{\beta}| \le 2^{-s}/(4d)$.

Now apply Algorithm (1.16). For a particular value of n the logarithm of the length of the vectors b_i in the initial basis for the lattice $L_s = L(b_0, b_1, \ldots, b_n)$ is $O(d^2 + d \cdot \log H)$ due to the choice of s. Application of the basis reduction algorithm to L_s can therefore be done in $O(n \cdot d^4 \cdot (d + \log H))$ arithmetic operations on integers having $O(d^2 \cdot (d + \log H))$ binary bits.

When going from n to $n + 1$ in Algorithm (1.16), we do not have to restart the basis reduction algorithm for the new lattice: We just add a new vector b_{n+1} and a new dimension in which all the old vectors have a zero component, whereas b_{n+1} has component 1. It follows from this observation and [16, (1.37)] that the applications of the basis reduction algorithm for all $n \le n_0$ together can be done in $O(n_0 \cdot d^4 \cdot (d + \log H))$ arithmetic operations on integers having $O(d^2 \cdot (d + \log H))$ binary bits.

This bound clearly also holds for the computation of the $\bar{\alpha}_i$, which proves Theorem (1.19).

(1.20) *Remark.* A. Schönhage [22] has shown that for the lattice and the basis in (1.3), the basis reduction algorithm only needs $O(n \cdot d^3 \cdot (d + \log H))$ arithmetic operations on integers having $O(d \cdot (d + \log H))$ binary bits. This implies that Algorithm (1.16) actually needs $O(n_0 \cdot d^3 \cdot (d + \log H))$ operations on $O(d \cdot (d + \log H))$-bit integers.

A further improvement of a factor d in the number of operations can be obtained by means of Schönhage's improved basis reduction algorithm [22]. The formulation of Algorithm (1.16) should however be modified slightly to incorporate this improvement, as the analogue of [16, (1.37)] does not hold for the improved basis reduction algorithm; for details we refer to [22]. For a more efficient algorithm for basis reduction see also a paper by Schnorr [20].

2. Ramifications. The algorithm of the preceding section can be interpreted as saying the following: Polynomially many bits of an algebraic number are sufficient to specify it completely (polynomially in the number of bits needed to write down its minimal polynomial). In a vague sense, then, the bits of algebraic numbers are not random, but are completely determined by the first polynomially many bits. We will not make this sense very precise here—the cryptography papers referred to below undertake this task, but we will attempt to provide an intuitive description of why the results of the previous section show that the bits of algebraic numbers are not '(secure) pseudorandom' bits in the terminology of cryptographers.

The question of when an (infinite) sequence of 'bits' (0's and 1's) is random has been raised for a long time, and various reasonable definitions have been provided. Since any such sequence may be considered to be the binary expansion of a real number between 0 and 1, a rewording of the question is: When are the bits of a real number random? (The phrase 'the bits of a real number' will mean the binary expansion of the fractional part of the number.) The classical definition was provided by Borel in 1909 [4]. The gist of it follows: Define a real number α to be *normal with respect to the base* 2 if for any natural number k, each of the 2^k possible 0-1 strings of length k occur with equal probability in the bits of α. A similar definition can be made for other bases. It was not difficult to show that most real numbers are normal. It was shown by Champernowne [7] in 1933 that the real number α_0 which equals the infinite decimal $.123456789101112\ldots$ (whose digits are obtained by juxtaposing the digits of the integers $1, 2, 3, 4, \ldots$) is normal to the base 10. Copeland and Erdős [6] generalized this to any basis and a class of reals including α_0 and $\alpha_1 = .2357111317\ldots$ whose digits are obtained by juxtaposing the digits of successive primes. An excellent discussion of the various classical definitions of when a sequence is random appears in [14, Section 3.5].

In several applications related to computer science one would like a notion of randomness that implies some kind of unpredictability. The importance of this for cryptography as well as complexity theory is discussed in [23], [3], and [25]. Some other relevant papers related to this discussion are [9] and [8]. Of course, the bits of the real number α_0 above are eminently predictable; thus intuitively, normalcy does not seem to be a good criterion for randomness in this setting. Besides this objection, there is another—we cannot really define randomness for one single real number and still have unpredictability. The model we have in mind is one where a player A presents a player B with some bits of a real number and B is trying to predict the next bit. If there is one fixed real, B can compute the bits as fast as A can, and all bits are clearly predictable. So we will have to consider a set of numbers. The simplest set is the set of rationals. Blum, Blum and Shub [2] have shown the following: If A announces that he is giving out the bits of a rational number with denominator at most H, then after seeing $2 \cdot \log_2 H$ bits of the rational number, B can figure out its fractional part and thus compute the other bits in polynomial time. Since A needed at least $\log_2 H$ bits to store the rational, he cannot get a pseudorandom sequence of length more than a constant (2) times the length of the 'seed'.

The main result of the preceding section may be restated as follows:

If A *announces that he is giving the bits of an algebraic number which is the root of an irreducible primitive polynomial of degree d or less with integral coefficients each of absolute value at most H, then after seeing $O(d^2 + d \cdot \log_2 H)$ bits,* B *can compute in deterministic polynomial time the polynomial and hence find for any n the nth bit of the algebraic number in time polynomial in the data and n (for the latter statement see also Section* 3*).*

Intuitively, our result can be interpreted as saying that the bits of algebraic numbers cannot form very long pseudorandom sequences, because after seeing a number of bits that is polynomial in the length of the seed (the seed in this case would be the polynomial held by A) the sequence can be easily and uniquely inferred. As mentioned earlier, the question of whether this can be done was first raised by M. Blum (private communication) who foresaw the importance of the notion of predictability.

Another ramification of the result of the preceding section is that computations involving algebraic numbers can be done in a natural way by representing algebraic numbers by suitable rational approximations. The traditional representation of algebraic numbers is by their minimal polynomials (see, for example, [24] or [17]). We now know an efficient method of converting the rational approximation representation to the minimal polynomial representation. (For the conversion in the other direction, see Section 3.) While it is not hard to see that computations in either representation can be changed to computations in the other without loss of efficiency (the running time will not change by more than a polynomial), the rational approximation method is closer to the intuitive notion of computation. For this reason we briefly sketch as an example a polynomial-time algorithm for finding a primitive element (see definitions below) of the rationals extended by two algebraics. Landau and Miller [15] gave in 1983 a polynomial-time algorithm for the same problem as part of their algorithm for testing solvability by radicals.

First we remark that if α and β are two algebraic numbers, then given sufficiently close approximations to both, we can find the minimal polynomial of β over $\mathbf{Q}(\alpha)$—the least-degree polynomial $p(X)$ with coefficients in $\mathbf{Q}(\alpha)$ satisfied by β. This is done as follows. Suppose the degree of α over $\mathbf{Q}$ is d; then clearly each coefficient of $p(X)$ can be taken to be a polynomial in α of degree at most $d-1$ with integral coefficients. Suppose the degree of β over $\mathbf{Q}(\alpha)$ is m (we try $m = 1, 2, \ldots$ in order). Then $p(X) = \sum_{i=0}^{m} \sum_{j=0}^{d-1} p_{ij} \alpha^j X^i$ for some $p_{ij} \in \mathbf{Z}$. We can turn the problem of finding the p_{ij} (i.e., the problem of finding the minimal integral dependence among the $\alpha^j \beta^i$ for $0 \le j \le d-1$ and $0 \le i \le m$) into a lattice problem in exactly the same way as we turned the problem of finding the minimal integral dependence among α^j for $0 \leq j \leq d$ into a lattice problem in the preceding section. In the interest of space we do not elaborate.

Suppose that α is algebraic over $\mathbf{Q}$ of degree d, and β is another algebraic number whose degree over $\mathbf{Q}(\alpha)$ is m, where d and m are determined as described above. The field $\mathbf{Q}(\alpha, \beta)$ obtained by adjoining α and β to the set of rationals is the set of all complex numbers expressible as polynomials in α and β with rational coefficients. It is known that this field has a primitive element γ, i.e., an element γ with the property that $\mathbf{Q}(\alpha, \beta) = \mathbf{Q}(\gamma)$, and indeed $\gamma = \alpha + l \cdot \beta$, where l is a nonnegative integer at most $d \cdot m$. It is also easy to see that if the degree of $\alpha + l \cdot \beta$ is $d \cdot m$ over $\mathbf{Q}$, then $\mathbf{Q}(\alpha + l \cdot \beta)$ must be equal to $\mathbf{Q}(\alpha, \beta)$. Thus we can use the algorithm of Section 1 to find the degree of $\alpha + l \cdot \beta$ over $\mathbf{Q}$ for $l = 0, 1, \ldots, d \cdot m$, given sufficiently close approximations to α and β, and thereby find the primitive element. It would be interesting to cast the entire algorithm for testing solvability by radicals into one that deals with explicit approximations to the algebraic numbers involved.

The idea of computing with algebraic numbers in this fashion needs to be explored further. While it is too early to say if the algorithms will be better in practice, they should yield good theoretical and/or empirical insights.

The method of finding the minimal polynomial of β over $\mathbf{Q}(\alpha)$ can be extended to finding algebraic dependence between any number of complex numbers. More exactly, let $\alpha_1, \alpha_2, \ldots, \alpha_t$ be (possibly transcendental) complex numbers given by sufficiently good approximations. Assume that we know an upper bound d on the degree and an upper bound H on the coefficients of a polynomial $f \in \mathbf{Z}[X_1, X_2, \ldots, X_t]$ with $f(\alpha_1, \alpha_2, \ldots, \alpha_t) = 0$. Then we can compute such a polynomial f in time polynomial in $\log H$ and $\binom{d+t-1}{d}$. (This latter number is polynomial in d for fixed t and in t for fixed d.) The precision to which the numbers α_i must be known is also a polynomial number of bits in $\log H$ and $\binom{d+t-1}{d}$.

This yields a factorization algorithm for multivariate polynomials: Given $f \in \mathbf{Z}[X_1, X_2, \ldots, X_t]$, substitute sufficiently large random numbers $s_2, s_3, \ldots, s_t$ for $X_2, X_3, \ldots, X_t$, compute an s_1 such that $f(s_1, s_2, \ldots, s_t) \approx 0$, and then find an algebraic dependence between $s_1, s_2, \ldots, s_t$. For $t = 2$, a slight variant of this idea is worked out in detail in [12].

Applications to Some Transcendental Numbers. The same technique can be applied to transcendental numbers of the form $\cos^{-1}(\alpha), \sin^{-1}(\alpha), \log(\alpha)$ etc., where α is an algebraic number. The number π is included in this class since it is the principal value (i.e., the value belonging to the interval $(0, \pi]$) of $\cos^{-1}(-1)$.

Suppose β is the principal value of $\cos^{-1}(\alpha)$ for some unknown α, which is, however, known to be algebraic of degree and height at most d and H, respectively. The question is: Can we infer (in deterministic polynomial time) the minimal polynomial of α from an approximation $\bar{\beta}$ to β? We show that if $|\beta - \bar{\beta}|$ is at most $\varepsilon = 2^{-s}/(24d)$, this can be done, where s is such that

$$2^s \geq 2^{d^2/2} \cdot (d+1)^{(3d+4)/2} \cdot H^{2d}$$

as usual. The argument is as follows. First we show that a good approximation to β gives us a good approximation to $\alpha = \cos(\beta)$:

$$|\cos(\beta) - \cos(\bar{\beta})| \leq \varepsilon \cdot \max\left\{\left[\left|\frac{d}{dx}\cos(x)\right|\right]_{x=y} : y \text{ between } \beta \text{ and } \bar{\beta}\right\} \leq \varepsilon.$$

This can be utilized if we can compute $\cos(\bar{\beta})$ at least approximately. To do this, we employ the Taylor series expansion of the cosine function and the argument that the tail of the series is small, once we consider several terms of the series. For all y with $0 \leq y < 2\pi$ we have

$$\cos(y) = 1 - y^2/2! + y^4/4! - y^6/6! + y^8/8! - \dots,$$

and further

$$\begin{aligned} &|\cos(y) - (1 - y^2/2! + y^4/4! - \dots + y^{4k}/(4k)!)| \\ &\leq \frac{|y|^{4k+1}}{(4k+1)!} \cdot \max\left\{\left[\left|\frac{d^{4k+1}}{dz^{4k+1}}\cos(z)\right|\right]_{z=x} : x \text{ between } 0 \text{ and } y\right\} \\ &\leq (2\pi)^{4k+1}/(4k+1)!. \end{aligned}$$

Let k equal the maximum of $\lceil -(\log\varepsilon)/4 \rceil$ and $\lceil \pi e^2/2 \rceil$. Then using Stirling's formula, we see that $(2\pi)^{4k+1}/(4k+1)! \leq \varepsilon$. Denoting

$$g(y) = 1 - y^2/2! + y^4/4! - \dots + y^{4k}/(4k)!,$$

we find that

$$|g(\bar{\beta}) - \cos(\beta)| \leq |g(\bar{\beta}) - \cos(\bar{\beta})| + |\cos(\bar{\beta}) - \cos(\beta)| \leq 2 \cdot \varepsilon.$$

Thus, in polynomial time we can compute from $\bar{\beta}$ an approximation $\bar{\alpha}$ to an unknown algebraic number α such that $|\alpha - \bar{\alpha}| \leq 2 \cdot \varepsilon = 2^{-s}/(12d)$, with s as above. Now Theorem (1.19) guarantees that we can find the minimal polynomial of α in polynomial time. This argument can be extended to the inverses of functions that satisfy the following two definitions.

(2.1) *Definition.* A complex-valued function f defined on a subset D of the complex numbers is *approximable* if there is a deterministic algorithm that, given a complex number x in D with rational real and imaginary parts and a natural number t, computes a complex number α satisfying $|\alpha - f(x)| \leq 2^{-t}$ in time bounded by a polynomial function of t and the number of bits of x.

(2.2) *Definition.* A complex-valued function f defined on a subset D of the complex numbers satisfies the *uniform Lipschitz condition* if there exist $\delta, M > 0$ such that $|f(x) - f(y)| \leq M \cdot |x - y|$ for any x, y in D with $|x - y| \leq \delta$.

(2.3) THEOREM. *Suppose a complex-valued function f defined on a subset D of the complex numbers is approximable and satisfies the uniform Lipschitz condition, for certain $\delta, M > 0$. There is an algorithm which, given a complex number $\bar{\beta}$ in D with rational real and imaginary parts and two natural numbers d and H, determines whether or not there is a complex number β in D satisfying*

(i) $|\beta - \bar{\beta}| \leq \varepsilon$, *with* $\varepsilon = \min((24d \cdot 2^{d^2/2} M (d+1)^{(3d+4)/2} H^{2d})^{-1}, \delta)$, *and*

(ii) $f(\beta)$ *is an algebraic number of degree at most d and height at most H. Further, if such a β exists, then $f(\beta)$ is unique, and the algorithm determines the* *minimal polynomial of $f(\beta)$. The algorithm works in time bounded by a polynomial function of $d, \log H$, and the number of bits of $\bar{\beta}$.*

Proof. First we show that if a β satisfying (i) and (ii) exists in D, then $f(\beta)$ is unique. Suppose not; then let β and γ satisfy (i) and (ii) and $f(\beta) \neq f(\gamma)$. Because $|\beta - \bar{\beta}| \leq \varepsilon$, we have that $|f(\beta) - f(\bar{\beta})| \leq \varepsilon \cdot M$ by the Lipschitz condition, and similarly $|f(\gamma) - f(\bar{\beta})| \leq \varepsilon \cdot M$. But then, $f(\beta)$ and $f(\gamma)$ are two algebraic numbers of degree at most d and height at most H with $|f(\beta) - f(\gamma)| \leq 2\varepsilon \cdot M$, contradicting the fact that distinct algebraic numbers cannot come too close (cf. Remark (1.8)). This proves the uniqueness of $f(\beta)$.

By the approximability of f we can compute $\bar{\alpha}$ such that $|\bar{\alpha} - f(\bar{\beta})| \leq \varepsilon \cdot M$. If a suitable β exists, then the Lipschitz condition gives $|f(\beta) - f(\bar{\beta})| \leq \varepsilon \cdot M$, so that $|f(\beta) - \bar{\alpha}| \leq 2\varepsilon \cdot M$. The proof now follows by Theorem (1.19).

The exponential function, sine function, hyperbolic sine and cosine functions, etc., when restricted to a finite interval (note that we need such a restriction for the exponential function), satisfy both definitions, and thus the theorem can be applied to them. At present, the only interesting consequence is the statement that the bits of reals of the form $\cos^{-1}(\alpha), \sin^{-1}(\alpha), \log(\alpha)$, where α is algebraic, do not form a pseudorandom sequence.

Notice that complex numbers of the form $\log(\alpha)$, where α is an algebraic number ($\neq 0, 1$), cannot be algebraic. This follows from the famous theorem of A. Baker [1] (on log linear forms).

3. Factorization of Polynomials. In this section we describe an algorithm to factor primitive polynomials over the integers in polynomial time. The first polynomial-time algorithm for this was provided in [16]. As described in the introduction, our algorithm is conceptually simple—we find the roots of the given polynomial to a certain accuracy, and then find the minimal polynomials of the roots using the algorithm of Section 1. These must then be the irreducible factors of the given polynomial. Rabin [19, Section 3] first used such an idea to factor over finite fields, where it is possible to find the minimal polynomial of a root (which in general lies in an extension field) by solving a system of simultaneous linear equations. For polynomials with integral coefficients, an algorithm similar to ours is described in [5], without being polynomial-time, however.

Throughout this section, $f(X) \in \mathbf{Z}[X]$ is the given primitive polynomial to be factored, $\deg(f(X)) = d$. Let $H = \binom{d}{d/2} \cdot |f|$. In [18] it is shown that this H bounds the height of any factor in $\mathbf{Z}[X]$ of f (see also [14, Exercise 4.6.2.20]). The factoring algorithm now follows immediately from Algorithm (1.16).

(3.1) ALGORITHM FACTOR. Let f, d and H be as above. If $d \leq 1$, then return that f is irreducible and terminate the execution of the algorithm. Otherwise, do the following as long as $d \geq 2$:

- Let s be the smallest positive integer such that

$$2^s \geq 2^{d^2/2} \cdot (d+1)^{(3d+4)/2} \cdot H^{2d}$$

- Compute an approximation $\bar{\alpha}$ to a root α of f such that $|\alpha - \bar{\alpha}| \leq 2^{-s}/(12d)$ (this can be replaced by $2^{-s}/(4d)$ if $|\alpha| \leq 1$, cf. Explanation (1.18)), apply

Algorithm (1.16) to determine the minimal polynomial $h(X)$ of α, and return h as an irreducible factor of f.

- Replace d by $d - \deg(h)$, put $g(X) = f(X)/h(X)$ and return g as an irreducible factor of f if $d = 1$. Terminate the execution of the algorithm if $d \leq 1$; otherwise, replace f by g and go on.

This finishes the description of Algorithm (3.1).

It follows from Explanation (1.18), Theorem (1.19) and the definition of H that all application of Algorithm (1.16) together can be done in $O(d^5 \cdot (d + \log|f|))$ arithmetic operations on $O(d^2 \cdot (d + \log|f|))$-bit integers. A. Schönhage's observation (cf. Remark (1.20)) even brings this down to $O(d^4 \cdot (d + \log|f|))$ arithmetic operations on $O(d \cdot (d + \log|f|))$-bit integers.

It remains to analyze the cost of the computation of an approximation to a root of f. In [21] it is shown that the cost of computing approximations to all roots of f simultaneously, up to the precision needed in Algorithm (3.1), is dominated by the cost of the applications of Algorithm (1.16). This paper is however not yet published, and therefore we sketch how an approximation to a root of $f \in \mathbf{Z}[X]$ of degree d can be found in time polynomial in $d, \log|f|$ and the number of bits needed. The algorithm is due to A. Schönhage and is considerably slower than his method in [21]; we only include it to show that the problem can be solved in polynomial time. We need the following lemma, which follows from [10, Theorems 6.4b and 6.4e].

(3.2) LEMMA. *Let $g(X) = \sum_{i=0}^{d} g_i X^i \in \mathbf{C}[X]$, and let α be the root of g which is smallest in absolute value. If $R(g) = \min\{|g_0/g_m|^{1/m} : m \geq 1, g_m \neq 0\}$, then*

$$\frac{1}{2} \cdot R(g) \leq |\alpha| \leq d \cdot R(g).$$

Proof. If $g_0 = 0$, then $X = 0$ is a root, and the lemma is obviously true. So assume $g_0 \neq 0$. First, suppose that the lower bound on $|\alpha|$ is violated. Then

$$|\alpha| < \frac{1}{2}\left|\frac{g_0}{g_m}\right|^{1/m}$$

for all m with $g_m \neq 0$. So

$$\left|\sum_{m=1}^{d} g_m \alpha^m\right| \leq \sum_{m=1}^{d} |g_m||\alpha|^m < |g_0| \sum_{m=1}^{d} \frac{1}{2^m} < |g_0|.$$

This implies that we cannot have $\sum_{i=0}^{d} g_i \alpha^i = 0$, a contradiction.

Now suppose that $|\alpha| > d \cdot R(g)$. Let $\alpha = \alpha_1, \alpha_2, \ldots, \alpha_d$ be the roots of g. Then

$$g_m = g_d \cdot \sum_{i_1, i_2, \ldots, i_{d-m}} \alpha_{i_1} \cdot \alpha_{i_2} \cdot \cdots \cdot \alpha_{i_{d-m}}$$

for $m = 0, 1, \ldots, d-1$, and in particular

$$g_0 = g_d \cdot \prod_{i=1}^{d} \alpha_i.$$

So,

$$\frac{g_m}{g_0} = \sum_{i_1, i_2, \ldots, i_m} \frac{1}{a_{i_1}} \cdot \frac{1}{\alpha_{i_2}} \cdot \cdots \cdot \frac{1}{\alpha_{i_m}}$$

for $m = 0, 1, \ldots, d-1, d$. Since $|\alpha_i| > d \cdot R(g)$ for $i = 1, 2, \ldots, d$, we have

$$\left|\frac{g_m}{g_0}\right| < \binom{d}{m}\left(\frac{1}{d \cdot R(g)}\right)^m \le \frac{1}{R(g)^m}$$

for any m. It follows that

$$\left|\frac{g_0}{g_m}\right|^{1/m} > R(g)$$

for all m with $g_m \neq 0$. This is in contradiction with the definition of $R(g)$. This proves Lemma (3.2).

We now show how to approximate a root in polynomial time. We may assume that among the roots $\alpha_1, \alpha_2, \ldots, \alpha_d \in \mathbf{C}$ of f there is an α_i satisfying $|\alpha_i| \le 1$ (otherwise, replace $f(X)$ by $X^d \cdot f(1/X)$). Let $t \in \mathbf{Z}_{\ge 0}$ and $a_t \in 2^{-t}\mathbf{Z}\left[\sqrt{-1}\right]$ such that

$$\min_i |a_t - \alpha_i| \le 4d \cdot 2^{-t}. \tag{3.3}$$

Initially, this condition is satisfied for $t = 0$ and $a_0 = 0$. We show how to compute $a_{t+1} \in 2^{-(t+1)}\mathbf{Z}\left[\sqrt{-1}\right]$ such that (3.3) holds with t replaced by $t+1$.

For all $a \in 2^{-(t+1)}\mathbf{Z}\left[\sqrt{-1}\right]$ such that

$$|a - a_t| \le 4d \cdot 2^{-t} + 2^{-(t+1)} \tag{3.4}$$

we compute the coefficients of $g_a(X) = f(X + a)$ and an approximation $r(g_a)$ to $d \cdot R(g_a)$ such that

$$d \cdot R(g_a) \le r(g_a) \le 2d \cdot R(g_a), \tag{3.5}$$

where $R(g_a)$ is defined as in Lemma (3.2). Define a_{t+1} as the a for which $r(g_a)$ is minimal.

To prove that a_{t+1} satisfies (3.3) with t replaced by $t+1$, notice that the roots of $g_a(X)$ are the $\alpha_i - a$, and that it follows from (3.3) and (3.4) that there is an a' among the a such that $\min_i |a' - \alpha_i| \le 2^{-(t+1)}$. This yields:

$$\begin{aligned}
\min_i |a_{t+1} - \alpha_i| &\le r(g_{a_{t+1}}) \quad \text{(Lemma (3.2) and (3.5))} \\
&\le r(g_a) \quad \text{(choice of } \alpha_{t+1}) \\
&\le 2d \cdot R(g_a) \quad \text{(due to (3.5))} \\
&\le 4d \cdot \min_i |a' - \alpha_i| \quad \text{(Lemma (3.2))} \\
&\le 4d \cdot 2^{-(t+1)} \quad \text{(choice of } a').
\end{aligned}$$

It is clear that the computation of a_{t+1} can be done in time polynomial in d, t, and $\log|f|$. It follows that an approximation to a root of f can be found in time polynomial in $d, \log|f|$ and the number of bits needed.

We have shown the following theorem.

(3.6) THEOREM. *A primitive polynomial f of degree d in one variable with integral coefficients can be completely factored over the integers in time polynomial in d and* $\log |f|$.

Using A. Schönhage's observation mentioned in Remark (1.20) and his improved version of the polynomial-time root finding algorithm described above (cf. [21]), we get the following theorem.

(3.7) THEOREM. *A primitive polynomial f of degree d in one variable with integral coefficients can be completely factored over the integers in $O(d^4 \cdot (d + \log |f|))$ arithmetic operations on $O(d \cdot (d + \log |f|))$-bit integers.*

As mentioned in Remark (1.20), the number of operations can be reduced to $O(d^3 \cdot (d + \log |f|))$ if we use Schönhage's improved basis reduction algorithm. The description of the algorithm should in that case be slightly modified; we refer to [22] for details.

Acknowledgment. We gratefully acknowledge one referee for pointing out two errors in Section 2.

Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

Department of Computer Science
The University of Chicago
Chicago, Illinois 60637

Eőtvős Loránd University
Budapest, Hungary

1. A. BAKER, "Linear forms in the logarithms of algebraic numbers I, II, III, IV," *Mathematika*, v. 13, 1966, pp. 204–216; *ibid.*, v. 14, 1967, pp. 102–107; *ibid.*, v. 14, 1967, pp. 220–228; *ibid.*, v. 15, 1968, pp. 204–216.

2. L. BLUM, M. BLUM & M. SHUB, *A Simple Secure Pseudo Random Number Generator*, Proceedings of Crypto 82.

3. M. BLUM & S. MICALI, *How to Generate Cryptographically Strong Sequences of Pseudo Random Bits*, Proc. 23rd Annual Symposium on Foundations of Computer Science, 1982, pp. 112–117.

4. É. BOREL, *Leçons sur la Théorie des Fonctions*, 2nd ed., 1914, pp. 182–216.

5. A. J. BRENTJES, "Multi-dimensional continued fraction algorithms," in *Computational Methods in Number Theory* (H. W. Lenstra, Jr. and R. Tijdeman, eds.), Math. Centre Tracts 154, 155, Mathematisch Centrum, Amsterdam, 1982.

6. A. H. COPELAND & P. ERDŐS, "Note on normal numbers," *Bull. Amer. Math. Soc.*, v. 52, 1946, pp. 857–860.

7. D. G. CHAMPERNOWNE, "The construction of decimals normal in the scale of ten," *J. London Math. Soc.*, v. 8, 1933, pp. 254–260.

8. O. GOLDREICH, S. GOLDWASSER & S. MICALI, *How to Construct Random Functions*, Proc. 25th Annual Symposium on Foundations of Computer Science, 1984, pp. 464–479.

9. S. GOLDWASSER, S. MICALI & P. TONG, *Why and How to Establish a Private Code on a Public Network*, Proc. 23rd Annual Symposium on Foundations of Computer Science, 1982, pp. 134–144.

10. P. HENRICI, *Applied and Computational Complex Analysis*, vol. 1, Wiley, New York, 1974.

11. I. N. HERSTEIN, *Topics in Algebra*, 2nd ed., Xerox, 1976.

12. M.-P. VAN DER HULST & A. K. LENSTRA, *Polynomial Factorization by Transcendental Evaluation*, Proceedings Eurocal 85.

13. R. KANNAN, A. K. LENSTRA & L. LOVÁSZ, *Polynomial Factorization and Nonrandomness of Bits of Algebraic and Some Transcendental Numbers*, Proc. 16th Annual ACM Symposium on Theory of Computing, 1984, pp. 191–200.

14. D. E. KNUTH, *The Art of Computer Programming*, Vol. 2, 2nd ed., Addison-Wesley, Reading, Mass., 1981.

15. S. LANDAU & G. MILLER, *Solvability by Radicals is in Polynomial Time*, Proc. 15th Annual ACM Symposium on Theory of Computing, 1983, pp. 140–151.

→ A. K. LENSTRA, H. W. LENSTRA, JR. & L. LOVÁSZ, "Factoring polynomials with rational coefficients," *Math. Ann.*, v. 261, 1982, pp. 513–534.

17. R. LOOS, "Computing in algebraic extensions," *Computer Algebra* (B. Buchberger, G. Collins and R. Loos, eds.), Springer-Verlag, Berlin and New York, 1982, pp. 173–187.

18. M. MIGNOTTE, "An inequality about factors of polynomials," *Math. Comp.*, v. 28, 1974, pp. 1153–1157.

19. M. O. RABIN, "Probabilistic algorithms in finite fields," *SIAM J. Comput.*, v. 9, 1980, pp. 273–280.

20. C. P. SCHNORR, "A more efficient algorithm for lattice basis reduction," manuscript, 1985.

21. A. SCHÖNHAGE, *The Fundamental Theorem of Algebra in Terms of Computational Complexity*, Preliminary report, Math. Inst. Univ. Tübingen, 1982.

22. A. SCHÖNHAGE, *Factorization of Univariate Integer Polynomials by Diophantine Approximation and an Improved Basis Reduction Algorithm*, Proc. 11th International Colloquium on Automata, Languages, and Programming, 1984, LNCS 172, 1984, pp. 436–447.

23. A. SHAMIR, *On the Generation of Cryptographically Strong Pseudo-Random Sequences*, Proc. 8th International Colloquium on Automata, Languages, and Programming, 1981.

24. B. TRAGER, *Algebraic Factoring and Rational Function Integration*, Proc. SYMSAC 76, pp. 219–226.

25. A. YAO, *Theory and Applications of Trapdoor Functions*, Proc. 23rd Annual Symposium on Foundations of Computer Science, 1982, pp. 80–91.