MATHEMATICS OF COMPUTATION
VOLUME 36, NUMBER 153
JANUARY 1981

Asymptotically Fast Factorization of Integers

By John D. Dixon*

Abstract. The paper describes a "probabilistic algorithm" for finding a factor of any large composite integer n (the required input is the integer n together with an auxiliary sequence of random numbers). It is proved that the expected number of operations which will be required is $O(\exp\{\beta(\ln n \ln\ln n)^{1/2}\})$ for some constant $\beta > 0$. Asymptotically, this algorithm is much faster than any previously analyzed algorithm for factoring integers; earlier algorithms have all required $O(n^{\alpha})$ operations where $\alpha > 1/5$.

1. Introduction. Recently there have been several proposals for very fast "probabilistic" tests for primality (see [9] and [12]), and in [7] it is shown that a result, which so far has only been proved on the assumption of the extended Riemann hypothesis, would imply that there is an almost as fast strictly deterministic algorithm to decide primality. These all test an integer n for primality in $O(\ln n)^k$ time for some small integer k.

This has led to a situation in which it is relatively easy to decide when an integer is composite but very difficult to find any proper factor. Indeed, in spite of considerable recent progress, algorithms for finding a factor of an integer known to be composite are comparatively slow. The best that has been proved for any published algorithm is that it is possible to factor an integer n in time $O(n^{\alpha})$ with $\alpha > 1/5$ (see [5, §4.5.4], [11], [6] and [8]). In the present paper, we describe a class of algorithms for which it can be proved that "nearly all" will find a factor of an integer n in time $O(\exp\{\beta(\ln n \ln\ln n)^{1/2}\})$ for some constant $\beta > 0$. This may be interpreted as a proof of the effectiveness of a "probabilistic" algorithm for factoring large integers. As it stands, however, in contrast to the primality tests mentioned above, the method seems to be more of theoretical than practical interest.

2. The Algorithms. We shall consider a general family of algorithms which include (in a simplified form) the algorithm discussed in [5, pp. 351–353], where it is called "Factoring via continued fractions". Let n be an odd integer divisible by at least two primes (the prime power case is easily disposed of). Following Legendre, we know that there exist integers x and y which are relatively prime to n and such that $x^2 \equiv y^2 \pmod{n}$ but $x \not\equiv y$ or $-y$. For such integers $\mathrm{GCD}(n, x + y)$ is a proper factor of n. Our search for the integers x and y is carried out in two stages. First we look for squares z^2 which are congruent (mod n) to integers whose prime factors all lie in some set P. Then we use relations between the exponents in

Received September 5, 1979.
1980 *Mathematics Subject Classification.* Primary 10A25.
*Research supported in part by NSERC under Grant A7171.

0025-5718/81/0000-0022/$02.50

the factorization of these latter numbers to construct the desired x and y. This process is described more precisely as follows (v is a parameter which will be specified later).

ALGORITHM A_L.

Initialization. L is a list of integers in the range $[1, n]$, $P = \{p_1, \ldots, p_h\}$ is the list of the h primes $\leqslant v$, and B and Z are initially empty lists (Z will be indexed by B).

Step 1. If L is empty, then exit (algorithm unsuccessful); else let z be the first term in L, delete z from L, and go to Step 2.

Step 2. Let w be the least positive remainder of $z^2 \pmod n$. Factor $w = w' \prod_i p_i^{a_i}$, where w' has no factor in P. If $w' = 1$, then go to Step 3; else go to Step 1.

Step 3. Let $a \leftarrow (a_1, \ldots, a_h)$. Adjoin a to B and $z = z_a$ to Z. If the list B has at most h elements, then go to Step 1; else go to Step 4.

Step 4. Find the first vector c in B which is linearly dependent (mod 2) on earlier vectors in B. Delete c from B and z_c from Z. Compute coefficients $f_b = 0$ or 1 such that

$$c \equiv \sum_{b \in B} f_b b \pmod 2.$$

Let

$$d = (d_1, \ldots, d_n) \leftarrow \frac{1}{2}(c + \Sigma f_b b) \quad \text{(a vector of integers)}$$

and go to Step 5.

Step 5. Let $x \leftarrow z_c \prod_b z_b^{f_b}$ and $y \leftarrow \prod_i p_i^{d_i}$ (so $x^2 \equiv \prod_i p_i^{2d_i} = y^2 \pmod n$). If $x \equiv y$ or $-y \pmod n$, then go to Step 1; else return GCD$(n, x + y)$ (a proper factor of n) and exit (algorithm successful).

Now suppose that L has length N and consider the number of operations (comparable to arithmetic operations between pairs of integers the size of n) which are involved in the execution of A_L. If we let N_i denote the number of times which A_L executes Step i, then clearly $N + 1 \geqslant N_1 \geqslant N_2 \geqslant N_3$ and $N_4 = N_5 = \max(N_3 - h, 0)$. In Step 2, factorization of w requires $O(h \ln n)$ operations since each $a_i \leqslant \ln n$. In Step 4, the determination of c and the calculation of the coefficients f_b can be carried out by a Gaussian elimination and so require at most $O(h^3)$ operations. In Step 5, the GCD can be computed in $O(\ln n)$ operations; see [5].

Hence the number of operations carried out in the execution of A_L is

$$N_1 O(1) + N_2 O(h \ln n) + N_3 O(1) + N_4 O(h^3) + N_5 O(h + \ln n)$$
$$= O\big(N_1 h \ln n + N_4 h^3\big).$$

Our remaining problem is to choose the parameters v and N such that: (i) the algorithms A_L terminate successfully for nearly all lists L of length N, and (ii) the number of operations required is as small as possible.

Let $\mathfrak{L}(n, N)$ denote the set of all n^N lists of length N consisting of integers from $[1, n]$. Then our main result is as follows.

THEOREM. *Let n be an odd integer divisible by at least two distinct primes. Put $v = \exp\{(2 \ln n \ln\ln n)^{1/2}\}$ and put $N = [v^2 + 1]$. Then the average number of operations required in the execution of A_L ($L \in \mathfrak{L}(n, N)$) is $O(\exp\{3(2 \ln n \ln\ln n)^{1/2}\})$, and the proportion of algorithms A_L which fail to find a proper factor of n is $O(N^{-1/2})$ (uniformly in n).*

Remark. One can deduce from the theorem that, if L is an infinite random sequence of integers from the interval $[1, n]$, then with probability 1 A_L will factor n and it will find a factor within expected time $O(\exp\{3(2 \ln n \ln\ln n)^{1/2}\})$ (see [2] for a discussion of such "probabilistic" algorithms). One could simulate L using a quasi-random sequence, but it seems unlikely that this will give a method competitive with known methods within the present range of practical computation (Step 3 seems to require $O(h(\ln n)^2)$ storage). Algorithm E of [5, p. 352] is an algorithm of type A_L where L is chosen to (hopefully) minimize the number of times that Step 2 is executed. Making certain plausible assumptions about the regularity of primes, Richard Schroeppel has given a heuristic argument which suggests that Algorithm E will require only $O(\exp\{(2 \ln n \ln\ln n)^{1/2}\})$ operations. He has also proposed another related algorithm which he estimates will require only $O(\exp\{(\ln n \ln\ln n)^{1/2}\})$ operations. It is not clear to me how these heuristic arguments could be transformed into actual proofs of the performance of these algorithms, but the arguments certainly suggest that these methods should do better than the average behavior described in our theorem. [Schroeppel's results are unpublished, but are referred to in [10]. I am indebted to Professor R. L. Rivest who supplied me with Schroeppel's description of his algorithm and its analysis. Professor Rivest's helpful comments also improved the exposition of the present paper.]

3. Proof of the Theorem. We shall begin with some general lemmas. For any positive real numbers u, v, let $\Psi(u, v)$ denote the number of positive integers $k \leqslant u$ which have all prime divisors $\leqslant v$. Then de Bruijn [1] proves the following.

LEMMA 1. *$\Psi(v^k, v) \geqslant \binom{\pi(v)+k}{k}$ for each integer $k \geqslant 1$ where $\pi(v)$ is the number of primes $\leqslant v$.*

The proof is almost immediate since the binomial coefficient counts the number of ways to choose sets of up to k integers (permitting repetitions) from among the first $\pi(v)$ primes. Although better asymptotic estimates for $\Psi(u, v)$ are known (see [1] and [4]), they require conditions on u and v which cannot be verified to hold in our case.

We shall suppose that n (which is odd) has the canonical prime factorization

$$n = q_1^{l_1} q_2^{l_2} \cdots q_d^{l_d} \qquad (d \geqslant 2). \tag{1}$$

Let $T(v)$ denote the set of $\Psi(n, v)$ positive integers $\leqslant n$ which have all prime factors $\leqslant v$, and let Q be the integers relatively prime to n which are quadratic residues (mod n).

The following results about Q are well known; see, for example, [13, Chapter V, §4]. An integer k lies in Q if and only if k is a quadratic residue (mod $q_i^{l_i}$) for $i = 1, \ldots, d$. Moreover, if k and k' are quadratic residues (mod $q_i^{l_i}$), then kk' is

also a quadratic residue (mod $q_i^{l_i}$). Thus, to each integer k relatively prime to n, we can define a quadratic character $\chi(k) = (\varepsilon_1, \ldots, \varepsilon_d)$ with $\varepsilon_i = 1$ or -1 depending on whether k is or is not a quadratic residue (mod $q_i^{l_i}$). Then χ has the following properties: (i) $\chi(k) = (1, \ldots, 1)$ if and only if $k \in Q$; and (ii) $\chi(k) = \chi(k')$ if and only if $kk' \in Q$. Finally, if $w \in Q$, then there are exactly 2^d solutions to $z^2 \equiv w$ (mod n) with $z \in [1, n]$.

In Step 2 of the algorithm A_L, we are concerned with the set $M(v)$ consisting of all integers $z \in [1, n]$ such that $z^2 \equiv w$ (mod n) for some $w \in T(v)$. If no prime $p \leqslant v$ divides n, then $|M(v)| = 2^d|T(v) \cap Q|$.

LEMMA 2. *There exists a constant $c_0 > 0$ such that, for all positive integers n and r and real $v \geqslant n^{1/2r}$, the conditions*

(i) $c_0^{-1} \ln n \geqslant r \geqslant \ln \ln n$ *and*

(ii) *all prime factors of n are $> v$*

together imply that $|M(v)| \geqslant n(\ln n)^{-4r}$.

Remark. This bound is probably not very good. It seems reasonable to expect that $|M(v)|$ is approximately equal to $\Psi(n, v)$, which would imply a lower bound better than $n(\ln n)^{-2r}$. An improvement of this type in Lemma 2 would give improved constants in the Theorem but leave the general form of the result unchanged.

Proof. It follows from (ii) that the integers in $T(v)$ are relatively prime to n and so we can partition $T(v)$ into a union of disjoint subsets T_i $(i = 1, \ldots, 2^d)$ corresponding to the 2^d different possible values for χ. Let $\tau(t)$ denote the number of divisors of t and write S (respectively, S_i) for the sum of $\tau(t)^{-1}$ taken over all $t \leqslant \sqrt{n}$ with $t \in T(v)$ (respectively, $t \in T_i$). Similarly, let S' denote the sum of $\tau(t)$ over all $t \leqslant \sqrt{n}$ with $t \in T(v)$. Then the Cauchy-Schwarz inequality gives

$$\Psi(\sqrt{n}, v)^2 \leqslant SS' \tag{2}$$

and

$$S^2 \leqslant 2^d \sum_{i=1}^{2^d} S_i^2. \tag{3}$$

Also, it is well known (see [13, p. 52]) that

$$S' \leqslant \sum_{t \leqslant \sqrt{n}} \tau(t) \leqslant \sqrt{n} \ln \sqrt{n} + \sqrt{n} < \sqrt{n} \ln n \tag{4}$$

for all n sufficiently large (for example, if $c_0 \geqslant 2$ and (i) holds).

On the other hand, $\sum S_i^2 = \sum_{t \leqslant n} c(t)$ where $c(t) = \sum \tau(s)^{-1}\tau(s')^{-1}$ where the latter sum is taken over all pairs (s, s') such that $ss' = t$, $s \leqslant \sqrt{n}$, $s' \leqslant \sqrt{n}$, and both s and s' lie in the same T_i. Since $s, s' \in T_i$ implies that $\chi(s) = \chi(s')$, it follows from the above that $c(t) \neq 0$ implies that $t \in Q$. Moreover, if $t \in Q$, then the inequality $\tau(s)\tau(s') \geqslant \tau(ss')$ shows that $c(t) \leqslant 1$. Thus, $\sum S_i^2$ is a lower bound on the number of $t \in Q$ with $t \in T(v)$. Hence,

$$|T(v) \cap Q| \geqslant \sum S_i^2. \tag{5}$$

Combining (2)–(5), we obtain (when $c_0 \geqslant 2$) that

$$|M(v)| = 2^d|T(v) \cap Q| \geqslant n^{-1}(\ln n)^{-2}\Psi(\sqrt{n}, v)^4. \tag{6}$$

Now condition (i) implies that $v \geqslant n^{1/2r} \geqslant e^{c_0/2}$ and $r \geqslant \ln c_0$. Therefore, if c_0 is chosen sufficiently large, then (i) implies that v and r will both be large and so

$$\pi(v) \geqslant v(2 \ln v)^{-1} \quad \text{and} \quad r! \leqslant (r/2)^r.$$

Then, by Lemma 1, we have

$$\Psi(\sqrt{n}\,, v) \geqslant \pi(v)^r/r! \geqslant v^r(r \ln v)^{-r} \geqslant \sqrt{n}\left(\tfrac{1}{2} \ln n\right)^{-r}.$$

Hence, by (6) and (i), we have

$$|M(v)| \geqslant 2^{4r} n(\ln n)^{-4r-2} \geqslant n(\ln n)^{-4r},$$

as required.

Now, for each $L \in \mathfrak{L}(n, N)$, we shall define $\sigma(L) \in \mathfrak{L}(n, N)$ as the list whose ith term w is given by $w \equiv z^2 \pmod n$ where z is the ith term of L. For each L_0, let $[L_0]$ denote the set of all $L \in \mathfrak{L}(n, N)$ such that $\sigma(L) = \sigma(L_0)$; these subsets form a partition of $\mathfrak{L}(n, N)$.

LEMMA 3. *Let n have the factorization* (1). *Then, for each integer k, the proportion of algorithms A_L ($L \in \mathfrak{L}(n, N)$), which execute Step 5 more than k times, is at most 2^{-k}.*

Proof. Let us say that L is 'bad' if A_L executes Step 5 more than k times. Then it is enough to show that if L_0 is bad then at most 2^{-k} of the lists in $[L_0]$ are bad. Write z_j for the value of z_c which occurs when A_{L_0} executes Step 5 for the jth time, and suppose that z_j originally occurred as the i_jth term in L_0. Then, for each $L \in [L_0]$, the term z in the i_jth position satisfies $z^2 \equiv z_j^2 \pmod n$. Since each element in Q has exactly 2^d square roots (mod n), $[L_0]$ is partitioned into 2^{dk} subsets of equal size where L and L' lie in the same subset if and only if they have the same terms at the positions $i_1, \ldots, i_k$, respectively. However, $L \in [L_0]$ is bad if and only if its terms at the positions $i_1, \ldots, i_k$ are $\pm z_1, \ldots, \pm z_k$ since the value for y in Step 5 only depends on $\sigma(L)$. Thus, the proportion of bad lists in $[L_0]$ is $2^k/2^{dk} \leqslant 2^{-k}$, as asserted.

Now consider the main theorem. We shall begin by proving that the second assertion of the theorem holds under the slightly weaker hypothesis $n \geqslant N \geqslant 4hv$. Denote by X_L the number of w in $\sigma(L)$ lying in $T(v)$ and observe that X_L, as a random variable on the space $\mathfrak{L}(n, N)$, has a binomial distribution with mean λN and variance $\lambda(1 - \lambda)N$ where λ (the probability of the event $w \in T(v)$) satisfies $\lambda \geqslant (\ln n)^{-4r}$ for n large enough by Lemma 2; see [3, Chapter 9]. By our choice of v, this shows that $\lambda \geqslant v^{-1}$. By Chebyshev's inequality, the proportion of $L \in \mathfrak{L}(n, N)$, for which $X_L \leqslant \lambda N - c$, is at most $c^{-2}\lambda(1 - \lambda)N$ for any $c > 0$. In particular, taking $c = \frac{1}{2}\lambda N$, we find that the proportion of algorithms A_L, for which $X_L \leqslant c$, is smaller than $2c^{-1}$. But, by the choice of N, we have $c \geqslant \frac{1}{2}v^{-1}4hv = 2h$ if n is large enough. Hence, if $X_L > c$ and A_L fails to find a factor of n, then A_L must execute Step 3 at least $2h$ times and Steps 4 and 5 at least h times. But then Lemma 3 shows that the proportion of A_L ($L \in \mathfrak{L}(n, N)$), which are unsuccessful and have $X_L > c$, is at most 2^{-h}. Thus, the proportion of all A_L ($L \in \mathfrak{L}(n, N)$), which fail to find a factor of n, is at most

$$2c^{-1} + 2^{-h} = O(vN^{-1}) + O(n^{-1}) = O(N^{-1/2}).$$

This proves the second assertion of the theorem. To complete the proof of the theorem we recall that in Section 2 we showed that the number of operations required in the execution of A_L is $O(N_1 h \ln n + N_4 h^3)$. From what we have just proved, all but $O(v^{-1})$ of the A_L will have found a proper factor of n with no more than $4hv + 2$ executions of Step 1. Thus, the average value of $N_1 h \ln n$ is

$$O(v^{-1}(N + 1)h \ln n + (4hv + 2)n \ln n) = O(vh^2 \ln n) = O(v^3)$$

since $h = O(v/\ln v)$. On the other hand, the average value of $N_4 h^3$ is $O(h^3)$ by Lemma 3. Thus, the average number of operations required by A_L is $O(v^3)$, as asserted.

Department of Mathematics
Carleton University
Ottawa, Ontario, Canada K1S 5B6

1. N. G. DE BRUIJN, "On the number of positive integers $\leq x$ and free of prime factors $> y$. II," *Indag. Math.*, v. 28, 1966, pp 239–247.

2. G. J. CHAITIN & J. T. SCHWARTZ, "A note on Monte Carlo primality tests and algorithmic information theory," *Comm. Pure Appl. Math.*, v. 31, 1978, pp. 521–527.

3. W. FELLER, *Probability Theory and its Applications*. Vol. 1, Wiley, New York, 1950.

4. H. HALBERSTAM, "On integers all of whose prime factors are small," *Proc. London Math. Soc.*, (3), v. 21, 1970, pp. 102–107.

5. D. KNUTH, *The Art of Computer Programming*. Vol. 2, Addison-Wesley, New York, 1971.

6. R. S. LEHMAN, "Factoring large integers," *Math. Comp.*, v. 28, 1974, pp. 637–646.

7. G. L. MILLER, "Riemann's hypothesis and tests for primality," *J. Comput. System Sci.*, v. 13, 1976, pp. 300–317.

8. J. M. POLLARD, "Theorems on factorization and primality testing," *Proc. Cambridge Philos. Soc.*, v. 76, 1974, pp. 521–528.

9. M. O. RABIN, "Probabilistic algorithms," *Algorithms and Complexity–New Directions and Recent Results* (J. F. Traub, Ed.), Academic Press, New York, 1976.

10. R. L. RIVEST, A. SHAMIR & L. ADLEMAN, "A method for obtaining digital signatures and public-key cryptosystems," *Comm. ACM*, v. 21, 1978, pp. 120–128.

11. D. SHANKS, *Class Number, a Theory of Factorization, and Genera*, Proc. Sympos. Pure Math., vol. 20, Amer. Math. Soc., Providence, R. I., 1970, pp. 415–440.

12. R. SOLOVAY & V. STRASSEN, "A fast Monte-Carlo test for primality," *SIAM J. Comput.*, v. 6, 1977, pp. 84–85; Errata, *ibid.*, v. 7, 1978, p. 118.

13. I. M. VINOGRADOV, *Elements of Number Theory*, 5th rev. ed., Dover, New York, 1954.