#A4 INTEGERS 21 (2021)

THE SOUTH CAICOS FACTORING ALGORITHM

Michael O. Rubinstein[1]
Pure Mathematics, University of Waterloo, Waterloo, Ontario, Canada

Received: 6/13/20, Revised: 10/5/20, Accepted: 12/18/20, Published: 1/4/21

Abstract

Let $N = UV$, where U, V are integers, with $1 < U, V < N$, and $\gcd(U, V) = 1$. We describe a probabilistic algorithm for factoring N using $O(\max(U, V)^{1/2+\epsilon})$ bit operations.

1. Preliminaries

Let $N = UV$, where U, V are integers, with $1 < U, V < N$, and $\gcd(U, V) = 1$.

Let a be an integer, $1 < a < N$. By the division algorithm, write

$$\begin{aligned} U &= u_1 a + u_0, \quad \text{with } 0 < u_0 < a \\ V &= v_1 a + v_0, \quad \text{with } 0 < v_0 < a. \end{aligned} \tag{1}$$

If, for a given a, we can determine u_0, u_1, v_0, v_1 then we have found U and V. We have assumed that u_0 and v_0 are non-zero. Otherwise, $a|N$ and we easily extract a non-trivial factor of N.

Previously, the author developed a factoring algorithm (called 'Hide and Seek') requiring $O(N^{1/3+\epsilon})$ bit operations which involves studying (1) with large a, of size $N^{1/3}$. Details are provided in [1].

In this paper, we describe an alternative method for finding u_0, v_0, u_1 and v_1, requiring $O(\max(U, V)^{1/2+\epsilon})$ bit operations. Thus, in the case, for example, that both U and V are $O(N^{1/2})$, the algorithm has complexity $O(N^{1/4+\epsilon})$.

Let a be prime. We also let $a > \max(U, V)^{1/2}$, so that $u_1, v_1 < a$. Furthermore, u_0 and v_0 are invertible modulo a, because a is prime and $0 < u_0, v_0 < a$.

Our starting point is the formula

$$N = (u_1 a + u_0)(v_1 a + v_0) = u_1 v_1 a^2 + (v_0 u_1 + u_0 v_1) a + u_0 v_0 \tag{2}$$

with $0 < u_0, v_0 < a$, and $u_1, v_1 < a$. Thus, subtracting $u_0 v_0$, dividing by a, and reducing modulo a, we have:

$$(N - u_0 v_0)/a = v_0 u_1 + u_0 v_1 \mod a. \tag{3}$$

[1]Support for work on this paper was provided by an NSERC Discovery Grant

We will determine u_0, v_0, u_1, v_1 by considering this equation.

2. Model Case

We first examine the rare situation that $v_0 = u_0 \mod a$, i.e., that $a|V - U$. After explaining the method, we will relax this assumption.

Now, from (2), $u_0 v_0 = N \mod a$, hence, under the assumption $v_0 = u_0 \mod a$,

$$u_0^2 = N \mod a. \tag{4}$$

Since a is assumed prime, given N and a, we can use the Tonelli-Shanks algorithm [2] to determine the two possible solutions to the above equation.

The Tonelli-Shanks algorithm requires $O(\log a + r^2)$ multiplications modulo a, where r is the power of 2 dividing $a-1$. The average value of r, as one averages over primes a, is equal to 2 (see the appendix). Thus, on average, over primes a, Tonelli-Shanks requires $O(\log a)$ multiplications modulo a to determine the two possible values of u_0. And, because we are assuming $v_0 = u_0 \mod a$, v_0 is determined by u_0.

For each of the two possible solutions $0 < u_0 < a$ to (4), we multiply (3) by $u_0^{-1} \mod a$. We get, assuming $v_0 = u_0 \mod a$,

$$u_0^{-1}((N - u_0 v_0)/a) = u_1 + v_1 \mod a. \tag{5}$$

But $u_1 + v_1 < 2a$ (because $u_1, v_1 < a$), i.e., either $0 \leq u_1 + v_1 < a$, or $a \leq u_1 + v_1 < 2a$. Therefore, given the left-hand side of (5), i.e., given N, a, u_0, v_0, there are at most two possible values for $u_1 + v_1$, which we denote by s. For each of the two possible values of s (and given N, a, u_0, v_0), we substitute $v_1 = s - u_1$ into (2), and solve the resulting quadratic equation in u_1, yielding two possible values of u_1, which then also determines $v_1 = s - u_1$. We then test whether the u_0, v_0, u_1, v_1 thus obtained gives a correct integer factorization of N.

3. Generalizing the Model Case

The model case, $v_0 = u_0 \mod a$, occurs rarely, but similar cases can be considered. For example, say

$$\beta v_0 = \alpha u_0 \mod a. \tag{6}$$

Assume further that:

$$\begin{aligned} &\alpha, \beta \text{ are invertible modulo } a, \\ &\gcd(\alpha, \beta) = 1, \\ &1 \le \alpha \le \beta_{\max}/2, \\ &-\beta_{\max} \le \beta \le \beta_{\max}/2, \end{aligned} \tag{7}$$

for some positive $\beta_{\max}$.

Equation (6) can be equivalently written as

$$a|\beta V - \alpha U. \tag{8}$$

Now, $u_0 v_0 = N \mod a$, hence, by (6),

$$u_0^2 = \alpha^{-1}\beta N \mod a. \tag{9}$$

Thus, given N, α, β, and prime a, we can again use the Tonelli-Shanks algorithm to determine the two possible values of $u_0 \mod a$.

Hence, multiplying (3) by $\beta u_0^{-1} \mod a$, we get

$$\beta u_0^{-1}((N - u_0 v_0)/a) = \alpha u_1 + \beta v_1 \mod a. \tag{10}$$

But, because of our assumed bounds on α and β, we have

$$-\beta_{\max} a < \alpha u_1 + \beta v_1 < \beta_{\max} a. \tag{11}$$

Hence, given the left-hand side of (10), there are at most $2\beta_{\max}$ possibilities for

$$s = \alpha u_1 + \beta v_1, \tag{12}$$

i.e., one per interval of length a.

For each of the possible values of s (and given $N, a, u_0, v_0, \alpha, \beta$), we substitute $v_1 = (s - \alpha u_1)/\beta$ into (2), and solve the resulting quadratic equation in u_1, yielding two possible values of u_1, from which we also determine $v_1 = (s - \alpha u_1)/\beta$. We then test whether the u_0, v_0, u_1, v_1 thus obtained gives a correct integer factorization of $N = (u_1 a + u_0)(v_1 a + v_0)$.

Note that if u_0 leads to a positive integer factorization of $N = UV$, then the other solution $-u_0 \mod a$ to (9) produces the factorization $N = (-U)(-V)$.

4. The South Caicos Algorithm

We are now ready to describe our South Caicos factoring algorithm.

Initially, assume that $\max(U, V) < (2N)^{1/2}$. In Section 6, we will remove this assumption.

This condition holds, for example, if $U < V < 2U$, since then $V^2 < 2UV = 2N$. But because the method of the previous section does not distinguish $U < V$, we prefer to state the condition as we have.

The idea is to loop through a small number of values of α and β, as determined by $\beta_{\max} = 2$, say, and primes, $(2N)^{1/4} < a < 2(2N)^{1/4}$, and apply the method of Section 3.

If, for given (α, β), we encounter a prime $(2N)^{1/4} < a < 2(2N)^{1/4}$ such that $a|\beta V - \alpha U$, then, for that choice of α, β, a, the method of Section 3 quickly uncovers u_0, v_0, u_1, v_1, and hence U and V.

However, if, for our given set of (α, β)'s, no such $(2N)^{1/4} < a < 2(2N)^{1/4}$ is encountered, then we can repeat the process with the same set of primes a, but with $\beta_{\max}$ replaced, say, with $\beta_{\max} + 2$, taking care to exclude (α, β)'s already tested.

Heuristically, as $\beta_{\max}$ grows, we quickly expect to find (α, β), and a prime $(2N)^{1/4} < a < 2(2N)^{1/4}$, such that (8) holds. A complexity analysis follows after the pseudo code below.

Algorithm 4.1 (South Caicos)**.** Let $N = UV$, with $U, V > 1$ positive integers to be determined satisfying $\gcd(U, V) = 1$, satisfying $\max(U, V) < (2N)^{1/2}$.

1 Let $\beta_{\max} = 2$, and let $S(\text{old})$ be the empty set.

2 Let

$$S(\beta_{\max}) = \{(\alpha, \beta) \in \mathbb{Z}^2 : \gcd(\alpha, \beta) = 1, \alpha \in [1, \beta_{\max}/2], \beta \in [-\beta_{\max}, \beta_{\max}/2], \beta \neq 0\}.$$

3 Let a to be the first prime $> (2N)^{1/4}$.

4 Use the Euclidean algorithm to compute $d = \gcd(N, a)$. If $d > 1$ then we have determined a non-trivial factor of N and quit.

5 For $(\alpha, \beta) \in S(\beta_{\max}) - S(\text{old})$:
Carry out the procedure described in Section 3 for given N, a, α, β.
If this results in a non-trivial integer factorization of N, then quit.

6 Replace a by the next prime, and, if $a < 2(2N)^{1/4}$, repeat from Step 4.

7 If $\beta_{\max} + 2 < (2N)^{1/4}$, replace $S(\text{old})$ by $S(\beta_{\max})$, $\beta_{\max}$ by $\beta_{\max} + 2$, and repeat from Step 2, but, henceforth, skipping over Step 4. Otherwise exit.

Note that we do not invoke the invertibility condition of (7) in our definition of $S(\beta_{\max})$. Instead, we assume that $\beta_{\max} < (2N)^{1/4}$, and also $\beta \neq 0$. Because

$a > (2N)^{1/4}$ is prime, this guarantees α, β are invertible $\mod a$. We expect the algorithm to produce a factorization of N well before the exit condition is reached. See the discussion below.

Analysis: The success and efficiency of the method hinges on encountering a prime $(2N)^{1/4} < a < 2(2N)^{1/4}$, and relatively small integers α, β, such that $a | \beta V - \alpha U$. Heuristically, for U, V much larger than, and relatively prime to a, and $\gcd(U, V) = 1$, we expect $\beta V - \alpha U$ to be divisible by a, on average over $S(\beta_{\max})$, $1/a$ of the time.

More precisely, letting $X = (2N)^{1/4}$, we expect, as $X \to \infty$ and $|S(\beta_{\max})|/\log X \to \infty$ (but also with $\beta_{\max} < X$), the number of triples α, β, a, with $a | \beta V - \alpha U$, $X < a < 2X$, and $(\alpha, \beta) \in S(\beta_{\max})$, to satisfy

$$\sum_{\substack{X<a<2X \\ a \text{ prime}}} \sum_{\substack{(\alpha,\beta)\in S(\beta_{\max}) \\ a|\beta V-\alpha U}} 1 \sim |S(\beta_{\max})| \sum_{\substack{X<a<2X \\ a \text{ prime}}} 1/a \sim |S(\beta_{\max})| \log(2)/\log(X). \quad (13)$$

The last step follows from the Prime Number Theorem and a summation by parts, or else using the elementary estimate $\sum_{\substack{a<Y \\ a \text{ prime}}} 1/a \sim \log\log(Y) + b + O(1/\log(Y))$, where b is a constant, and noting that $\log\log(2X) - \log\log(X) = \log((\log(2) + \log(X))/\log(X)) \sim \log(2)/\log(X)$.

However, from the definition of $S(\beta_{\max})$,

$$|S(\beta_{\max})| \sim \frac{6}{\pi^2}\frac{3}{4}\beta_{\max}^2, \quad (14)$$

with the factor $6/\pi^2$ to account for the condition $\gcd(\alpha, \beta) = 1$. Thus, by (13) and (14), as $\beta_{\max}/\log(N)^{1/2}$ grows, we expect to encounter at least one $(\alpha, \beta) \in S(\beta_{\max})$, and a prime $X < a < 2X$, with $X = (2N)^{1/4}$, such that $a | \beta V - \alpha U$, and hence such that the method of Section 3 with succeed in finding non-trivial factors U, V of N. We also note that this should occur long before we trigger the exit condition of Step 7, since $\log(N)^{1/2}$ grows much slower than $(2N)^{1/4}$.

The bulk of the work, per (α, β, a), involves one application of the Tonelli-Shanks algorithm in Equation (9), followed by the extraction of the roots of $2\beta_{\max}$ quadratic equations, one per each value of s from (12).

For each candidate $X < a < 2X$, primality testing of a can be done in polynomial time. Alternatively, one can sieve for all primes in the interval using the sieve of Eratosthenes, at a cost of $O(X^{1/2}/\log X)$, i.e., $O(N^{1/8}/\log N)$ bits of storage, needed to keep track of multiples of the primes $< (2X)^{1/2}$ as we carry out the sieve in short intervals. A table of primes $< (2X)^{1/2}$ needed to carry out the sieve can also be tabulated using the sieve of Eratosthenes.

Overall, we expect this algorithm to successfully factor N in $O(N^{1/4+\epsilon})$ bit operations. With this stated efficiency, the method is probabilistic, since it relies on finding a prime $X < a < 2X$, and small α, β, i.e., of order N^ϵ, such that $a | \beta V - \alpha U$.

5. Example

For example, if $N = 23713634802068266491347$, the algorithm first uncovers the triple $a = 804901$, $\alpha = 1$, $\beta = 3$, with $u_0 = 523125$, $v_0 = 174375$, being a solution to $\beta v_0 = \alpha u_0 \mod a$, and $u_0 v_0 = N \mod a$, found by applying Tonelli-Shanks to (9). Then, following the method in Section 3, we obtain $u_1 = 235108$, $v_1 = 155684$ (with the value of s that succeeds in (12) being $s = 702160$) , giving a correct factorization of $N = UV$, with $U = u_1 a + u_0 = 189239187433$, $V = v_1 a + v_0 = 125310381659$.

In Table 1 we list additional triples a, α, β, with $\beta_{\max} = 16$, such that $a|\beta V - \alpha U$, and the corresponding values of u_0, v_0, s, u_1, v_1, U and V, produced by our method.

a	α	β	u_0	v_0	s	u_1	v_1	U	V
804901	1	3	523125	174375	702160	235108	155684	189239187433	125310381659
804901	3	1	174375	523125	702160	155684	235108	125310381659	189239187433
546671	1	-7	268355	274047	-2193938	229224	346166	125310381659	189239187433
601291	4	-5	282622	134677	216874	314721	208402	189239187433	125310381659
837043	3	-7	505993	22301	-369702	226080	149706	189239187433	125310381659
601291	5	-4	134677	282622	-216874	208402	314721	125310381659	189239187433
685099	6	-7	456554	293767	376970	276221	182908	189239187433	125310381659
546671	7	-1	274047	268355	2193938	346166	229224	189239187433	125310381659
644153	1	7	77804	563246	2250988	194535	293779	125310381659	189239187433
644153	7	1	563246	77804	2250988	293779	194535	189239187433	125310381659
685099	7	-6	293767	456554	-376970	182908	276221	125310381659	189239187433
837043	7	-3	22301	505993	369702	149706	226080	125310381659	189239187433
743161	7	-16	60161	670393	-2893914	168618	254640	125310381659	189239187433

Table 1: We list, for $N = 23713634802068266491347$ the values of prime a, $1 \le \alpha \le 8$, $-16 \le \beta \le 8$, such that the method of Section 3 produces values of u_0, v_0, u_1, v_1 that give a correct positive integer factorization of N. We also list those parameters, along with the corresponding value of s in (12), and the values of U and V.

6. Removing the Assumption $\max(U, V) < (2N)^{1/2}$

The assumption that $\max(U, V) < (2N)^{1/2}$ was made so that, with $a > (2N)^{1/4}$, one has, for given a, that $u_1, v_1 < a$. This is important in Equation (12) so that we only need to check $2\beta_{\max}$ possibilities for s.

However, we need not assume this bound on $\max(U, V)$.

Let $X = (2N)^{1/4}$. We run the algorithm of Section 4, but, at the j-th iteration of Step 3, we change it to read 'let a be the first prime $> 2^{j-1}X$, and in Step 6, replace '$2(2N)^{1/4}$' with '$2^j X$'. We also use, for given N, the value $\beta_{\max} = j \log N$, and eliminate $S(\text{old})$.

Thus, at the j-th iteration, we look at sets of ever larger primes $2^{j-1}X < a <$

$2^j X$. For j sufficiently large, we have $a > \max(U, V)^{1/2}$, and thus $u_1, v_1 < a$, as needed for the method of Section 3 to succeed.

The large value of $\beta_{\max}$ relative to $\log(N)^{1/2}$, and the analysis of Section 3, suggests that, with probability tending to 1, as $N \to \infty$, that we will thus succeed in factoring N using $O(\max(U, V)^{1/2+\epsilon})$ bit operations.

Algorithm 6.1 (South Caicos B)**.** Let $N = UV$, with $U, V > 1$ positive integers to be determined satisfying $\gcd(U, V) = 1$.

1 Let $\beta_{\max} = \log N$, $j = 1$, and $X = (2N)^{1/4}$.

2 Let

$$S(\beta_{\max}) = \{(\alpha, \beta) \in \mathbb{Z}^2 : \gcd(\alpha, \beta) = 1, \alpha \in [1, \beta_{\max}/2], \beta \in [-\beta_{\max}, \beta_{\max}/2], \beta \neq 0\}.$$

3 Let a to be the first prime $> 2^{j-1}X$.

4 Use the Euclidean algorithm to compute $d = \gcd(N, a)$. If $d > 1$ then we have determined a non-trivial factor of N and quit.

5 For $(\alpha, \beta) \in S(\beta_{\max})$:
Carry out the procedure described in Section 3 for given N, a, α, β.
If this results in a non-trivial integer factorization of N, then quit.

6 Replace a by the next prime, and, if $a < 2^j X$, repeat from Step 4.

7 Replace j by $j + 1$, $\beta_{\max}$ by $j \log N$, and repeat from Step 2.

Acknowledgement. The above algorithm was developed by the author in South Caicos while on vacation with his lovely girlfriend Lisa.

References

[1] M.O. Rubinstein, The distribution of solutions to $XY = N \mod a$ with an application to factoring integers, *Integers* **13** (2013), A12, 1–13.

[2] D. Shanks, Five number theoretic algorithms, *Proceedings of the Second Manitoba Conference on Numerical Mathematics*, Utilitas Mathematica, Winnipeg, 1973, 51-–70.

Appendix

We justify the assertion made in Section 2 regarding the average value of r that appears in the Tonelli-Shanks algorithm.

Lemma 1. *Let a be prime, and r the power of 2 dividing $a-1$. Then, the average value of r tends to 2, when averaged over primes $A < a \le 2A$, as $A \to \infty$.*

Proof. Let k be a positive integer. If $a = m \mod 2^k$, with m odd and $1 \le m < 2^k$, then the value of r, the power of 2 dividing $a-1$, is equal to

$$\begin{gathered} 1, \text{ if } m-1 = 2, 6, 10, 14, \ldots \\ 2, \text{ if } m-1 = 4, 12, 20, 28, \ldots \\ 3 \text{ if } m-1 = 8, 24, 40, 56, \ldots \\ \text{etc.} \end{gathered}$$

More precisely, if we write m as a k bit binary number (possibly with some leading zeros), then $r = 1$ if m ends in 11, $r = 2$ if m ends in 101, $r = 3$ if m ends in 1001, etc. In particular, 2^{k-2} of these m have $r = 1$, 2^{k-3} have $r = 2$, 2^{k-4} have $r = 3$, …, one has $r = k-1$ (namely $m = 2^{k-1}+1$). The residue class $m = 1$ requires more careful consideration. If $m = 1$, then the value of r is not precisely determined, but rather satisfies, for $a < 2A$,

$$k \le r \le \log(2A)/\log(2). \tag{15}$$

Now, the primes are equi-distributed amongst the odd residue classes mod 2^k. However, we require slightly more than just the main term of the Prime Number Theorem in arithmetic progressions. Specifically, let $c > 0$, and q a positive integer with $q \le \log(x)^c$. The Siegel-Walfisz Theorem implies that, if $\gcd(m, q) = 1$ then, $\pi(x; q, m)$, the number of primes less than or equal to x and congruent to $m \mod q$, satisfies

$$\pi(x; q, m) = \frac{1}{\phi(q)} \frac{x}{\log x}(1 + o(1)), \tag{16}$$

as $x \to \infty$, with the implied constant dependent on c, and ineffective. If we assume the GRH, then this holds with the implied constant effectively computable (and also a much stronger remainder term). Thus, for k satisfying, say,

$$\log(A)^2 < 2^k \le 2\log(A)^2, \tag{17}$$

we have, unconditionally,

$$\pi(2A, 2^k, m) - \pi(A, 2^k, m) = \frac{1}{2^{k-1}} \frac{A}{\log A}(1 + o(1)), \tag{18}$$

as $A \to \infty$.

Counting the contribution from each residue class $m \mod 2^k$, and taking into account (15) and (18), the average value of r, over primes $A < a \leq 2A$, is equal to:

$$\frac{1}{\pi(2A) - \pi(A)} \left(\sum_{r=1}^{k-1} r 2^{k-r-1} + O(\log A) \right) \frac{1}{2^{k-1}} \frac{A}{\log A} (1 + o(1)). \tag{19}$$

But the sum in parentheses is equal to $2^k - k - 1$, as can be verified inductively. Furthermore, $\pi(2A) - \pi(A) \sim A/\log A$. Thus, the above equals

$$\left(2 + O((\log A + k)/2^k)\right)(1 + o(1)). \tag{20}$$

But, by (17), $(\log(A) + k)/2^k \to 0$ as $A \to \infty$. Hence, the average value of r tends to 2 as $A \to \infty$. □

We note that condition (17) is used in two places. We need 2^k to grow faster than $\log(A)$ so as to get the limiting value of 2 in Equation (20). We also invoke the Siegel-Walfisz theorem in (16) which gives a uniform estimate for the Prime Number Theorem in arithmetic progressions, so long as the modulus 2^k grows slower than a power of $\log(A)$, hence the assumption that $2^k < 2\log(A)^2$.