Int. J. Math. And Appl., 6(2–A)(2018), 107–111
ISSN: 2347-1557
Available Online: http://ijmaa.in/

International Journal of *Mathematics And its Applications*

An Integer Factorization Method Equivalent to Fermat Factorization

Porkodi Chinniah[1,*] and Arumuganathan Ramalingam[1]

1 Department of Mathematics, PSG College of Technology, Coimbatore, Tamilnadu, India.

Abstract: Integer factorization problem had a great impact on security of many public key cryptosystems. Advancement in factoring increases the need to develop innovative public key cryptosystems. In this paper, a special purpose factorization method to find the factors of a composite number, which is the product of two distinct primes, is proposed. Fermat's method is theoretically important and many modern factorization algorithms like the quadratic sieve, multiple polynomial quadratic sieve, and number field sieves are based upon this method. The proposed scheme is equivalent to Fermat's method, and hence it can be used in such popular modern factorization methods. The steps involved in the algorithm are proved theoretically. Algorithm is illustrated numerically using Mathematica.

MSC: 11Y05.

Keywords: Integer factorization problem, Division algorithm, Fermat's method, Greatest Common Divisor.

 Accepted on: 16.04.2018

1. Introduction

Primality testing and Factorization are considered by many number theorists including Fermat and Gauss. Integer factorization problem, the apparent difficulty of factoring large integers is the basis of some modern public key cryptographic algorithms. The security of the popular cryptosystems like, RSA encryption algorithm by Rivest [1], Rabin Cryptosystem by M.O.Rabin [2] and the pseudo random number generator Blum Blum Shub cryptographic pseudorandom number generator by Blum et al [3] depends on the intractability of computational hard integer factorization problem. In the probabilistic encryption proposed by Blum [4] and Goldwasser [5], integer factorization problem plays a vital role in the security. Factorization algorithms are classified in to two categories: special purpose algorithms and general purpose algorithms.

(i). In special purpose algorithm, the integer being factored is of a particular form; for example an algorithm which factorizes a composite integer that is not a prime power. Examples include trial division, Pollard's $p-1$ algorithm, Pollard rho, Number field Sieve algorithm proposed by J.M.Pollard [6–8] and Elliptic curve algorithm proposed by HW.Lenstra Jr [9].

(ii). In general purpose algorithm, the integer being factored is not in any of the special structure. Examples include Quadratic Sieve introduced by C.Pomerance [10, 11] and General number field Sieve.

In trial division algorithm, one simply checks whether $s|N$ for $s = 2, 3, ..., \lfloor\sqrt{N}\rfloor$, then s and $t = \frac{N}{s}$ are the factors of N. Fermat [12] proposed a factorization method in which the composite number N is expressed as the difference of squares

* *E-mail: porkodi_c2003@yahoo.co.in*

$N = x^2 - y^2$, then $(x + y)$ and $(x - y)$ are factors of N. Solving $N = x^2 - y^2$ is equivalent to finding x and y such that $x^2 = y^2(mod\ N)$ and if x is not congruent to $\mp y(mod\ N)$ then $GCD\,(x - y, N)$ or $GCD(x + y, N)$ must be a non-trivial factor of N. Arjen K.Lenstra [13], HW. Lenstra Jr [14], Richard P. Brent [15] and N.M.Stephens [16] had done related work on integer factorization. In this paper, a special purpose factorization algorithm equivalent to Fermat method is proposed. The proposed algorithm is justified theoretically.

Notations: Throughout this paper, it is assumed that $N = pq$, where p and q are prime numbers such that $p < q$. Thus $p < N_0$ and $q > N_0$, where $N_0 = \lfloor\sqrt{N}\rfloor$; $\lceil N\rceil$-Ceiling of N; $\lfloor N\rfloor$-Floor of N; $GCD(a, b)$-Greatest common divisor of a and b; $a|b$-a divides b.

2. Proposed Algorithm

Input: A Composite number N, a product of two distinct primes p and q with $p < q$.

Output: Prime factors p and q.

Step 2.1: Compute $N_0 = \lfloor\sqrt{N}\rfloor$.

Step 2.2: Divide N by N_0, compute the quotient q_0 and remainder r_0.

Step 2.3: If $r_0 = 0$ then $p = N_0$ and $q = q_0$ are the factors of N.

Step 2.4: If $r_0 \neq 0$ and $GCD(N_0, r_0) \neq 1$, then $p = GCD(N_0, r_0)$ and $q = N/p$.

Step 2.5: If $r_0 \neq 0$ and $GCD(q_0, r_0) \neq 1$, then $p = GCD(q_0, r_0)$ and $q = N/p$.

Step 2.6: If $r_0 \neq 0$, $GCD(N_0, r_0) = 1$ and $GCD(q_0, r_0) = 1$,

Case 1: If $q_0 = N_0$ For $k = 1 : \lceil\sqrt{N - r_0}\rceil$. Compute $r_k = r_0 + k^2$, $N_0 + k$, $N_0 - k$, $GCD(N_0 + k, R_k)$, and $GCD(N_0 - k, R_k)$.

Step 2.6.1: If $GCD(N_0 + k, R_k) \neq 1$ then $p = GCD(N_0 + k, R_k)$ and $q = N/p$. If $GCD(N_0 - k, R_k) \neq 1$ then $p = GCD(N_0 - k, R_k)$ and $q = N/p$.

Case 2: If $q_0 = N_0 + 1$. For $k = 1 : \lceil\frac{1+\sqrt{1+4N-4r_0}}{2}\rceil$ compute $r_k = r_0 + k(k - 1)$, $N_0 - k + 1$, $N_0 + k$, $GCD(N_0 + k, R_k)$, and $GCD(N_0 - k + 1, R_k)$.

Step 2.6.2: If $GCD(N_0 + k, R_k) \neq 1$ then $p = GCD(N_0 + k, R_k)$ and $q = N/p$. If $GCD(N_0 - k + 1, R_k) \neq 1$ then $p = GCD(N_0 - k + 1, R_k)$ and $q = N/p$.

3. Mathematical Back Ground of the Algorithm

In this section the results related to the algorithm are mathematically proved.

Theorem 3.1. *On dividing any integer N by $N_0 = \lfloor\sqrt{N}\rfloor$, the quotient is either N_0 or $(N_0 + 1)$.*

Proof. On dividing N by $N_0 = \lfloor\sqrt{N}\rfloor$, let the quotient be q_0 and remainder be r_0. By division algorithm, $N = N_0q_0 + r_0$, $0 < r_0 < N_0$. Clearly, $N_0^2 < N < (N_0 + 1)^2$.

Case (1): Suppose, $q_0 > (N_0 + 1)$, say $(N_0 + 2)$. Then

$$N = N_0(n_0 + 2) + r_0, \quad 0 < r_0 < N_0$$

$$N = N_0^2 + 2N_0 + r_0$$

$$N = (N_0 + 1)^2 + \text{a positive integer}$$

$$N > (N_0 + 1)^2 \quad \text{a contradiction.}$$

Case (2): Suppose, $q_0 < N_0$, say $(N_0 - 1)$. Then

$$N = N_0(N_0 - 1) + r_0, \quad 0 < r_0 < N_0$$
$$N = N_0^2 - N_0 + r_0$$

As $N > N_0^2$, $N_0^2 - N_0 + r_0 > N_0^2 \Rightarrow r_0 - N_0 > 0 \Rightarrow r_0 > N_0$. It is a contradiction. Hence the quotient q_0 is either N_0 or $(N_0 + 1)$. □

Theorem 3.2.

(1). Any integer $N = N_0^2 + r_0$ with $N_0 = \lfloor \sqrt{N} \rfloor$ is expressed as $N = (N_0 + k)(N_0 - k) + r_k$, where $R_k = r_0 + k^2$, $k = 0, 1, 2, \ldots, N_0$.

(2). Any integer $N = N_0(N_0 + 1) + r_0$ with $N_0 = \lfloor \sqrt{N} \rfloor$ is expressed as $N = (N_0 + k)(N_0 - k + 1) + R_k$, where $R_k = r_0 + k(k - 1)$, $k = 0, 1, 2, \ldots, N_0 + 1$.

Theorem 3.3.

(1). If $N = (N_0 + k)(N_0 - k) + R_k$, $R_k = r_0 + k^2$ for $k = 0, 1, 2, \ldots, N_0$ and $u = GCD(N_0 + k, R_k) \neq 1$ or $u = GCD(N_0 - k, R_k) \neq 1$, then u is the smallest prime factor p of N.

(2). If $N = (N_0 + k)(N_0 - k + 1) + R_k$, $R_k = r_0 + k(k - 1)$ for $k = 0, 1, \ldots, N_0 + 1$ and if $u = GCD(N_0 + k, R_k) \neq 1$ or $u = GCD(N_0 - k + 1, R_k) \neq 1$, then u is the smallest prime factor p of N.

Proof. **Claim 1:** u is a factor of N.

Suppose $u = GCD(N_0 + k, R_k)$ or $u = GCD(N_0 - k, R_k)$. Then, $u \,|(N_0 + k), u|\, R_k$ or $u|\,(N_0 - k), u|R_k$. In turn, $u\,|(N_0 + k)(N_0 - k), u|\,R_k$ or $u|\,(N_0 + k)(N_0 - k), u|R_k$, $u|[(N_0 + k)(N_0 - k) + R_k]$ i.e. $u|N$. Thus, u is a factor of N. If, $u = GCD(N_0 + k, R_k)$ or $u = GCD(N_0 - k + 1, R_k)$. Then, $u\,|(N_0 + k), u|\,R_k$ or $u|\,(N_0 - k + 1), u|R_k$. Thus, $u\,|(N_0 + k)(N_0 - k + 1), u|\,R_k$ or $u|\,(N_0 + k)(N_0 - k + 1), u|R_k$, $u|[(N_0 + k)(N_0 - k + 1) + R_k]$ i.e. $u|N$. Thus, u is a factor of N.

Claim 2: u is the smallest prime factor.

On contradiction assume, $u = q$, where q is the largest prime factor.

Case (i): $u = GCD(N_0 + k, R_k)$, $N = (N_0 + k)(N_0 - k) + R_k$, with $R_k = r_0 + k^2$, $k = 0, 1, 2, \ldots, N_0$; $u = GCD(N_0 + k, R_k)$ implies $N_0 + k = tq$ and $R_k = sq$, where $t > 0$, $s > 0$ Also, $tq = N_0 + k = 2N_0$, since k assumes any one of the values $0, 1, 2, \ldots, N_0$. Thus $q = \frac{2N_0}{t}$ and $q > N_0$ and implies $t = 1$. Now $t = 1$, implies

$$N = q(N_0 - k) + sq$$
$$pq = (N_0 - k)q + sq$$
$$p = N_0 - (k - s)$$

Now $p < N_0$, implies $(k - s) > 0$ and in turn $k > s$. Now $N = q(N_0 - k) + sq < qN_0$, a contradiction. Thus u is the smallest prime factor p.

Case (ii): $u = GCD(N_0 - k, R_k)$, $N = (N_0 + k)(N_0 - k) + R_k$, with $R_k = r_0 + k^2$, $k = 0, 1, 2, \ldots, N_0$. Suppose, $N_0 - k = tq$ and $R_k = sq$, where $t > 0$, $s > 0$. We get, $N = pq = (N_0 + k)tq + sq$. Thus, $p = (N_0 + k)t + s$. It is a contradiction to the fact that $p < N_0$. Thus u is the smallest prime factor p.

Case (iii): $u = GCD\,(N_0 + k, R_k)$, $N = (N_0 + k)\,(N_0 - k + 1) + R_k$, with $R_k = r_0 + k(k-1)$; $u = GCD\,(N_0 + k, R_k)$ implies $N_0 + k = tq$ and $R_k = sq$, where $t > 0$, $s > 0$ Also, $tq = n_0 + k = 2N_0$, since k assumes any one of the values $0, 1, 2, \ldots, N_0$. Thus $q = \frac{2N_0}{t}$ and $q > N_0$ and implies $t = 1$. Now $t = 1$, implies

$$N = q\,(N_0 - k + 1) + sq$$

$$pq = (N_0 - k + 1)\,q + sq$$

$$p = N_0 - (k - s - 1)$$

Now $p < N_0$, implies $(k - s - 1) > 0$ and in turn $k > s + 1$. Now $N = q\,(N_0 - k + 1) + sq < qN_0$, a contradiction. Thus u is the smallest prime factor p.

Case (iv): $u = GCD\,(N_0 - k + 1, R_k)$, $N = (N_0 + k)\,(N_0 - k + 1) + R_k$, with $R_k = R_0 + k(k-1)$. If $N_0 - k + 1 = tq$ and $R_k = sq$, where $t > 0$, $s > 0$. We get, $N = pq = (N_0 + k)\,tq + sq$. Thus, $p = (N_0 + k)\,t + s$ and $p > N_0$. It is a contradiction to the fact that $p < N_0$. □

Theorem 3.4. *The upper bound for k to get the factors of $N = pq$ are given by $k < \lceil \sqrt{N - r_0} \rceil$ when $N = (N_0 + k)\,(N_0 - k) + R_k$, $R_k = r_0 + k^2$ and $k < \left\lceil \frac{1+\sqrt{1+4N-4r_0}}{2} \right\rceil$ when $N = (N_0 + k)\,(N_0 - k + 1) + R_k$, $R_k = r_0 + k\,(k-1)$.*

Proof. The factors of $N = pq$ is obtained as p, when $GCD\,(N_0 + k, R_k) = p$ or $GCD\,(N_0 - k, R_k) = q$. Thus $R_k = N_0 + k$ or $R_k = N_0 - k$ and $R_k = N$. Thus $R_k = N_0 - k$.

Case (i): $R_k = r_0 + k^2 \Rightarrow r_0 + k^2 = N \Rightarrow k = \sqrt{N - r_0}$.

Case (ii): $R_k = r_0 + k\,(k-1) \Rightarrow r_0 + k(k-1) = N \Rightarrow k = \frac{1 \mp \sqrt{1+4N-4r_0}}{2} \Rightarrow k = \left\lceil \frac{1+\sqrt{1+4N-4r_0}}{2} \right\rceil$. □

Example 3.5. *$N = 24961$; $N_0 = \lfloor \sqrt{N} \rfloor = 157$, $r_0 = N \bmod N_0 = 155$. For $k = 61$, $N_0 + k = 218$, $R_k = 3815$, $GCD\,(216, 3815) = 109$. Thus, the smallest prime factor of 24961 is 109 and other factor is 229.*

4. Conclusion

In this paper a factorization method equivalent to Fermat factorization method is proposed. Fermat's method is theoretically important in the factorization methods such as: the quadratic sieve, multiple polynomial quadratic sieve, and the special and the number field sieves are all based upon this method. As the proposed scheme is similar to Fermat's method, it can be used in such factorization methods. The steps involved in the algorithm are theoretically proved. The computational complexity of the method depends on the value of 'k', which is involved in the computation of R_k. The upper bound for k is provided theoretically. But the complexity of the proposed algorithm can be improved by computing the exact value of k which is an open problem. Our future enhancement is to find out the exact value of k.

References

[1] R.L. Rivest, A. Shamir and L. Adelman, *A method for obtaining digital signatures and public key cryptosystem*, Commun of ACM, 21(2()1978), 120-126.

[2] M.O. Rabin, *Digitized signatures and public key functions as intractable as factorization*, MIT Lab for Computer Science Technical report, (1979), LCS/TR-212.

[3] L. Blum, M. Blum and M. Shub, *A simple unpredictable random number generator*, SIAM Journal on Computing, 15(1986), 364-383.

[4] M. Blum and S. Goldwasser, *An efficient probabilistic public-key cryptosystem that hides all partial information*, Lecture notes in Computer science, 196(1985), 289-302.

[5] S. Goldwassern and Micali, *Probabilistic encryption*, Journal of Computer and Systems Science, 28(1984), 270-299.

[6] J.M. Pollard, *Theorems on Factorization and Primality Testing*, Proceedings of the Cambridge Philosophical Society, 76(1974), 521-528.

[7] J.M. Pollard, *A Monte Carlo method for Factorization*, BIT, 15(1975), 331-334.

[8] J.M. Pollard, *Factoring with cubic integers The development of the Number Field Sieve*, vol. 1554, Lecture Notes in Mathematics, Springer Verlag, (1993), 4-10.

[9] HW. Lenstra JR, *Factoring integers with elliptic curves*, Annals of Mathematics, 126(1987), 649-673.

[10] C. Pomerance, *Analysis and Comparison of some integer factoring algorithms*, Computational Methods in Number Theory, Part 1, Mathematics Centrum, (1982), 89-139.

[11] C. Pomerance, *The quadratic sieve factoring algorithm*, Advances in Cryptology-Proceedings of Eurocrypt 84 (LNCS 209)(1985), 169-182.

[12] J. Alfred Menezes, C. Paul Van Oorchot and Scott A. Vanstone, *Hand book of Applied Cryptography*, CRC Press, (1997).

[13] Arjen K. Lenstra, *Integer factoring, Designs, Codes and Cryptography*, 19, Boston Kluwer Academic Publishers, (2000), 101-128.

[14] HW.Lenstra JR and Carl Pormerance, *A rigorous time bound for factoring integers*, Journal of the American Mathematical Society, 5(3)(1992), 483-516.

[15] Richard P. Brent, *Parallel Algorithms for Integer Factorization, Number Theory and Cryptography*, Cambridge University Press, (1990), 26-37.

[16] N.M. Stephens, *Lenstra's Factorization Method based on elliptic curves*, in Advances in Cryptology, Proceedings of CRYPTO'85, (1985), 409-416.