Int. J. Communications, Network and System Sciences, **2011, 4, 609-615**
doi:10.4236/ijcns.2011.410073 Published Online October 2011 (http://www.SciRP.org/journal/ijcns)

Integer Factorization of Semi-Primes Based on Analysis of a Sequence of Modular Elliptic Equations*

Boris S. Verkhovsky
Computer Science Department, New Jersey Institute of Technology, Newark, USA
E-mail: *verb73@gmail.com*

Received September 4, 2011; *revised* October 3, 2011; *accepted October* 11, 2011

Abstract

In this paper is demonstrated a method for reduction of integer factorization problem to an analysis of a sequence of modular elliptic equations. As a result, the paper provides a non-deterministic algorithm that computes a factor of a semi-prime integer *n*=*pq*, where prime factors *p* and *q* are unknown. The proposed algorithm is based on counting points on a sequence of at least four elliptic curves $y^2 = x\left(x^2 + b^2\right)\left(\bmod n\right)$, where *b* is a control parameter. Although in the worst case, for some *n* the number of required values of parameter *b* that must be considered (the number of basic steps of the algorithm) substantially exceeds *four*, hundreds of computer experiments indicate that the average number of the basic steps does not exceed six. These experiments also confirm all important facts discussed in this paper.

Keywords: Integer Factorization, Factorization of Semi-Primes, Non-Deterministic Algorithm, Elliptic Curves, Counting Points on Elliptic Curves, Crypto-Immunity, Dual Elliptic Curves

1. Introduction and Problem Statement

Security of information transmission via communication networks is provided by various cryptographic protocols. Crypto-immunity of these protocols is mostly based on hardness of either the integer factorization or the discrete logarithm problem.

There are several algorithms that factorize a semi-prime *n*=*pq*, where *n* is known, but its integer factors *p* and *q* are not. Fermat, Euler and other mathematicians/computer scientists introduced various algorithms for integer factorization. A survey of methods for factoring is provided in [1], and modern factoring algorithms are described in [2]. Various special methods are considered in [3-5]; an application of cubic forms for factorization, as one of these special methods, is provided in [6]. A comparison and analysis of factoring algorithms with exponential time complexity is provided in [7]. Algorithms based on the quadratic sieve (QS) are discussed in [8,9] while integer factoring via the number field sieve (NFS) is provided in [10]. Both the QS and NFS are the algorithms with sub-exponential time complexity. The application of special devises for factoring is described in [11,12]. A pioneering paper on application of quantum computing for integer factorization is discussed in [13].

A new factoring algorithm proposed in this paper is based on the analysis of several modular elliptic equations {called elliptic curves} and counting how many integer points {integer pairs (*x*,*y*)} satisfy these curves. The application of elliptic curves for factoring is described in [14-17]. Methods of counting points on elliptic curves are considered in [18,19] and more generally on modular equations with several variables in [20,21]. A relationship between integer factorization and constrained discrete logarithm problems is analyzed in [22].

Consider *n*=*pq*, where both *p* and *q* are multi-digit long primes. There are three special cases: where
1) each factor is congruent to 1 modulo4:

$$p = q = 1\left(\bmod 4\right); \tag{1.1}$$

2)
$$\left(p+q\right)\left(\bmod 4\right) = 0; \tag{1.2}$$

3)
$$p = q = 3\left(\bmod 4\right).$$

In this paper we discuss the factorization algorithm for (1.1) and (1.2) cases only.

Consider a sequence of elliptic curves (EC) modulo *n*:

$$E(n,b): y^2 = x\left(x^2 + b^2\right)\left(\bmod n\right). \tag{1.3}$$

*Dedicated to the memory of my mother Alla L. Verkhovsky.

Here $b \geq 1$ is an integer *control parameter*.

Let $P(n,b)$ denote the number of points on the EC (1.3).

2. Integer Factorization Algorithm

```
Input: n is a semi-prime;
Output: integer factors p and q of n;
1: if nmod4=3, then for a randomly chosen b compute $P(n,b)$;
2: {Assign}: $p := \gcd\left[P(n,b),n\right]$; $q := n/p$; {end of algorithm};   (2.1)
3: if nmod4=1, then compute $P(n,1)$;
4: if $P(n,1) = n$, then p=q=3(mod4): the algorithm is not applicable;
   else for b=2, 3, 5, 7, 11,···co-prime with n compute $P(n,b)$ until four distinct integers A, Q, R, U are found;   (2.2)
   else, if b that divides n is found, then p:=b; $q := n/p$; {end of algorithm};   (2.3)
5: Let Q:=max(A, Q, R, U); U:=min(A, Q, R, U);   (2.4)
6: Compute $S := (Q - U - |A - R|)/4$;   (2.5)
7: {Assign}: $p := \gcd(n,S)$; $q := n/p$; {end of algorithm}.   (2.6)
```

Remark 2.1: For sake of reference, the (2.1)-(2.6) algorithm is called the *SQUAR*-algorithm.

Remark 2.2: For large *n*, computation of *A*, *Q*, *R* and *U* can be performed in parallel.

Remark 2.3: If *b* is not co-prime with *n*, then *b* divides *n*, *i.e.*, either *p* or *q* is equal *b*.

Remark 2.4: Notice that $S \ll$min (*A*, *Q*, *R*, *U*). This observation allows us to simplify the computations of *S* for large *n* {see the example with "larger" *n* in the Appendix}.

3. Numeric Illustrations

Table 3.1. Number of points $P(n,b_k)$ on four elliptic curves *E*(*n*, *b*).

n	$P(n,b_{k_1})$; $k_1=1$	$P(n,b_{k_2})$; k_2	$P(n,b_{k_3})$; k_3	$P(n,b_{k_4})$; k_4	max k
24869	**37981;1**	**13993;2**	**34713;3**	**12789;4**	4
3813809	**3850233;1**	**3774993;2**	**3674789;3**	**3955221;11**	11
3858521	**3996001;1**	**3652173;3**	**3717945;4**	**4067965;17**	17
4549289	**4255713;1**	**4558669;3**	**4852633;4**	**4530141;7**	7

Table 3.2. Major steps of factorization algorithm.

n	*A*	*Q*	*R*	*U*	*S*	*p*; *q*
24869	37981	34713	13993	12789	**1118**	**13; 1913**
3813809	3955221	3850233	3774993	3674789	**51298**	**1973; 1933**
3858521	4067965	3996001	3717945	3652173	**34434**	**1913; 2017**
4549289	4852633	4558669	4530141	4255713	**142098**	$p = \gcd(n,S)$

We leave to readers the computation of *p* and *q* for the last semi-prime in **Table 3.2**.

RSA Factoring Challenge

Consider a product of two integers: *n*:=*pq*, where
p=34905295108476509491478496199038\
98133417764638493387843990820577; and
q=32769132993266709549961988190834\
461413177642967992942539798288533.

This *n* is called a RSA-129 Challenge [23]: given *n*, it was necessary to find its factors *p* and *q*. Since in the RSA-129 *n*mod4=1 and $p = q = 1(\bmod 4)$, therefore the proposed algorithm (2.1)-(2.6) is applicable to solve this problem.

4. Algorithm Validation

Definition 4.1: A non-zero integer $\underline{a}$ is called a *quadratic residue* (*QR*) modulo *p* if there exists an integer *z* such that

$$z^2 = a(\bmod p); \tag{4.1}$$

otherwise $\underline{a}$ is called a quadratic non-residue (*QNR*) modulo *p*.

By the Euler criterion [1], if *p* is a prime, then $\underline{a}$ is a *QR* if and only if

$$a^{(p-1)/2} \bmod p = 1.$$

The algorithm (2.1)-(2.6) is based on the following proposition.

Conjecture 4.1: If *p*=*q*=1(mod4), then there exist two positive integers *c*<*p* and *d*<*q*, and four sets S_1, S_2, S_3 and S_4 such that

$$S_1 = \{b : P(n,b) = U := (p-c)(q-d)\}; \tag{4.2}$$

$$S_2 = \{b : P(n,b) = A := (p-c)(q+d)\}; \tag{4.3}$$

$$S_3 = \{b : P(n,b) = R := (p+c)(q-d)\}; \tag{4.4}$$

$$S_4 = \{b : P(n,b) = Q := (p+c)(q+d)\}; \quad (4.5)$$

and where for every $i \neq j$ $S_i \cap S_j = \text{O}$; and

$S_1 \cup S_2 \cup S_3 \cup S_4 = \{1,2,3,5,7,11,13,\cdots\}$; {set of all primes} (4.6)

***Example* 4.1**: For semi-prime n=1352513 the sets S_1, S_2, S_3 and S_4 are as follows:

$$S_1 = \{b = 1,2,7,41,\cdots; U = 1267905\};$$

$$S_2 = \{b = 5,13,17,43,61,67,71,79,\cdots; A = 1313517\};$$

$$S_3 = \{b = 3,11,23,29,37,47,59,73,\cdots; R = 1389325\};$$

$$S_4 = \{b = 19,31,53,\cdots; Q = 1439305\}.$$

A priori it is not obvious how this conjecture can help to factorize n, but, as it is shown below, this is the case. It is assumed in this paper that there exists an efficient algorithm that computes A, Q, R and U {see (4.2)-(4.5)}.

From the definitions (4.2)-(4.5), it is easy to see that Q>max(A, R) and U<min(A, R),

i.e., that $n + (pd + qc) + cd > \max(A,R)$; and $\min(A,R) > n - (pd + qc) + cd$.

On the other hand, $R \neq A$, otherwise it implies that pd=qc. Since both p and q are primes, the latter equation holds only if c=p and d=q, which is impossible by the conjecture.

Consider the smallest product U and the largest product Q {see (4.2) and (4.5)}.

As a result, we derive a system of three equations with four integer unknowns p, q, c and d:

$$pq=n; \quad (p+c)(q+d) = Q; \quad (p-c)(q-d) = U; \quad (4.7)$$

$$i.e., \quad n + (pd + qd) + cd = Q; \quad (4.8)$$

$$\text{and} \quad n - (pd + qc) + cd = U. \quad (4.9)$$

Therefore, (4.8) and (4.9) imply that

$$pd + qc = (Q-U)/2. \quad (4.10)$$

Now consider another system of three equations with the same four unknowns:

$$pq=n; \quad (p-c)(q+d) = A \text{ and } (p+c)(q-d) = R; \quad (4.11)$$

$$i.e., \quad n + (pd - qc) - cd = A; \quad (4.12)$$

$$\text{and} \quad n - (pd - qc) - cd = R. \quad (4.13)$$

Then (4.12) and (4.13) imply that

$$pd\text{–}qc = (A-R)/2; \quad (4.14)$$

and, as a result, from (4.10) and (4.14)

$$qc = [(Q-U)/2 - (A-R)/2]/4. \quad (4.15)$$

Finally, from $qc \neq n$ it follows that

$$q = \gcd(n, qc), \text{ and } p = n/q. \quad (4.16)$$

5. Alternative Computation of Factors

It is easy to see that one of the factors is an average arithmetic of two greatest common divisors

$$p = [\gcd(A,Q) + \gcd(R,U)]/2, \text{ and } q = n/p; \quad (5.1).$$

However, computation of (5.1) is twice more complicated than the previously described procedure. Indeed, in (5.1) we must compute the two greatest common divisors and one addition, while in (2.6) it is necessary to compute only one greatest common divisor and three subtractions.

6. Generalized Factorization Algorithm

The factorization procedure described in (2.1)-(2.6) {*SQUAR*-algorithm} can be generalized and based on one of two following conjectures.

Consider a set of modular elliptic curves

$$E(n,a): \quad y^2 = x(x^2 + a)(\text{mod}\, n), \quad (6.1)$$

where $a \neq 0$, n=pq, p=q=1(mod4) and P(n, a) denotes the number of points on E(n, a).

Conjecture 6.1: If $\underline{a}$ is a quadratic *residue* modulo n, then there exist two positive integers c<p and d<q, and four sets S_1, S_2, S_3 and S_4 such that

$$S_1 = \{a : P(n,a) = u_1 := (p-c)(q-d)\}; \quad (6.2)$$

$$S_2 = \{a : P(n,a) = u_2 := (p-c)(q+d)\}; \quad (6.3)$$

$$S_3 = \{a : P(n,a) = u_3 := (p+c)(q-d)\}; \quad (6.4)$$

$$S_4 = \{a : P(n,a) = u_4 := (p+c)(q+d)\}; \quad (6.5)$$

and where for every $i \neq j$ $S_i \cap S_j = \text{O}$;

and $S_1 \cup S_2 \cup S_3 \cup S_4 = \{\text{set of all } QR \text{ modulo } n\}$. (6.6)

Conjecture 6.2: If $\underline{a}$ is a quadratic *non-residue* modulo n, then there exist two positive integers g<p and h<q, and four sets T_1, T_2, T_3 and T_4 such that

$$T_1 = \{a : P(n,a) = w_1 := (p+g)(q+h)\}; \quad (6.7)$$

$$T_2 = \{a : P(n,a) = w_2 := (p-g)(q+h)\}; \quad (6.8)$$

$$T_3 = \{a : P(n,a) = w_3 := (p+g)(q-h)\}; \quad (6.9)$$

$$T_4 = \{a : P(n,a) = w_4 := (p-g)(q-h)\}; \quad (6.10)$$

and where for every $i \neq j$ $T_i \cap T_j = \text{O}$; and

$$T_1 \cup T_2 \cup T_3 \cup T_4 = \{\text{set of all } QNR \text{ modulo } n\}. \quad (6.11)$$

7. Algorithm Acceleration

The numeric experiments provided in **Tables 3.1** and **7.1** show that for n=3813809 it is necessary to compute $P(n,b)$ *six* times {for b=1,2,3,5,7 and 11} and for *n*=3858521 *eight* times {for *b*=1,2,3,5,7,11,13 and 17} **until** four distinct integers *A*, *Q*, *R* and *U* are found.

These numbers can be decreased if the following property holds.

Let *p*=*q*=1(mod4); *n*=*pq* and *M*(*n*, *b*) denote the number of points on a *dual* EC.

$$y^2 = x(x^2 - b^2)(\bmod n)\,; \tag{7.1}$$

Conjecture 7.1: If primes *p* and *q* are randomly selected, then *with probability* 3/4

$$P(n,1) \neq M(n,1)\,; \tag{7.2}$$

and, as a result, for every integer *b*

$$P(n,b) \neq M(n,b)\,; \tag{7.3}$$

otherwise for every integer *b*

$$P(n,b) = M(n,b)\,. \tag{7.4}$$

Proposition 7.2: If the factors *p* and *q* are congruent to 1 modulo *n*=*pq*, then the following identities hold for every positive integer *m*:

$$P\left(n,2^{m+1}\right) = M\left(n,2^{m-1}\right) \text{ and } M\left(n,2^{m+1}\right) = P\left(n,2^{m-1}\right). \tag{7.5}$$

A proof is provided in the Appendix.

Proposition 7.3: If the factors *p* and *q* are congruent to 1 modulo *n*=*pq*, and $b_1 \neq b_2$

and $$P(n,b_1) = M(n,b_2)\,,$$

then $$M(n,b_1) = P(n,b_2)\,. \tag{7.6}$$

***Example* 7.1**: Compute *P*(3813809,1)=38500233 and *M*(3813809,1)=3774993.

Hence, (7.5) implies that there is no reason to compute *P*(3813809,2).

Instead, compute *P*(*n*,3)=3674789. Since *P*(*n*,3) is the third distinct value, thus, (7.2) and (7.3) imply that *M*(*n*,3) is the *forth* distinct value. Indeed, *M*(*n*,3)=3955221. Therefore, the algorithm requires only *four* basic steps instead of *six*. However, this acceleration does not work in 25% of the cases. For instance, it does not works for *n*=3858521, since neither (7.2) nor (7.3) hold. Indeed, because

$$P(3858521,1)=M(3858521,1),$$

the computation of *M*(3858521,3) provides no acceleration.

8. Dual Factorization Algorithm

It is assumed that *n*mod4=1 and $P(n,1) \neq n$, otherwise the algorithm is not applicable.

1) Compute $P(n,1)$ and $M(n,1)$ (7.1);

Table 7.1. Excerpts from Table 3.1.

N=3813809	3850233;1	3774993;2	3674789;3	3955221;**11**
N=3858521	3996001;1	3652173;3	3717945;4	4067965;**17**

2) **Case** $P(n,1) \neq M(n,1)$:

for *b*=3, 5, 7, 11,…co-prime with *n*

compute $P(n,b)$ **until** $P(n,b_*) \neq M(n,1)$ **and**

$$P(n,b_*) \neq P(n,1) \tag{8.1}$$

if *b* that divides *n* is found, **then** *p*:=*b*; *q*:=*n/p*; {**end** of algorithm}; (8.2)

else compute $M(n,b_*)$; {all four distinct integers *A*, *Q*, *R*, and *U* are found};

3) Let

$$U := \min\left[P(n,b_*), M(n,1), M(n,b_*), P(n,1)\right]; \tag{8.3}$$

$$Q := \max\left[P(n,b_*), M(n,1), M(n,b_*), P(n,1)\right]; \tag{8.4}$$

and let *A* and *R* be other two values,

$$\{i.e.,\ U < A < Q \text{ and } U<R<Q\}; \tag{8.5}$$

4) Compute $S := \left(Q - U - |A - R|\right)/4\,;$ (8.6)

5) {Factors}: $p := \gcd(n,S)$; *q*:=*n/p*; {**end** of algorithm}; (8.7)

6) **Case** $P(n,1) = M(n,1)$: **for** *b*=3,5,7,11,… co-prime with *n* compute $P(n,b)$ **until** four distinct integers *A*, *Q*, *R*, *U*, are found (2.2);

7) Let *U*:=min(*A*, *Q*, *R*, *U*); *Q*:=max(*A*, *Q*, *R*, *U*) (2.4);

8) Compute $S := \left(Q - U - |A - R|\right)/4$ (2.5);

9) {Factors}: $p := \gcd(n,S)$; *q*:=*n/p*; {**end** of algorithm} (2.6).

For the sake of simplicity of notations,

$$\text{let } P_k := P(n,b_k) \text{ and } M_k := M(n,b_k) \tag{8.8}$$

***Example* 8.1**: For semi-prime *n*=6525401 the sets S_1, S_2, S_3 and S_4 are as follows:

$$S_1 = \{b = 2,5,13,23,37,59,\cdots; U = 6055665\}\,;$$

$$S_2 = \{b = 3,19,47,\cdots;\ A = 6514053\}\,;$$

$$S_3 = \{b = \underline{\mathbf{43}},\ 53,67,83,\cdots; R = 6519205\}\,; \tag{8.9}$$

$$S_4 = \{b = 1,7,11,17,29,31,41,\cdots; Q = 7012681\}\,.$$

Therefore, the algorithm (2.1)-(2.6) requires at least *fifteen* basic steps, because 43 is the fourteenth prime (8.9). Yet, $P_1 \neq P_2$; and $P_2 = M_1$; imply that for every $k \geq 1$ $P_k \neq M_k$ (7.2)-(7.4).

Hence, instead of counting points

$$P_1, P_2, P_3, P_5, P_7, .., P_{43}$$

in fifteen elliptic curves, compute

$$M_3 = 6519205\,.$$

Thus, the algorithm (8.1-7) requires only *four* basic steps instead of *fifteen*.

Proposition 8.1: Let $P_1 \neq P_2$; and suppose

that there is *k*>2 for which $P_k \neq \{P_1, P_2, \cdots, P_{k-1}\}$ (8.10)

then $$M_k \neq \{P_1, P_2, .., P_{k-1}, P_k\} \tag{8.11}$$

Proof: Let assume that $M_k = P_j = P_i$,

where $3 \le j \le k-1$ and i=1 or i=2.

Therefore, $P_k = M_j = M_i = P_{3-i}$ (7.5)-(7.6). However, this contradicts with the assumption (8.10). Q.E.D.

9. Conclusions

An algorithm and its generalizations for the integer factorization are proposed. These algorithms are as computationally efficient as an algorithm that counts points on elliptic curves (1.3). Numerous computer experiments demonstrate that, if $P(n,b)$ is computed for sequential values of prime b, then on average there are *four* distinct values among the first *six* ones.

The *SQUAR*-algorithm (2.1)-(2.6) and its enhanced modification (8.1)-(8.7) described above is based on Conjecture 4.1 and its generalization {Conjecture 5.1}. Although an analogous algorithm can be designed on the basis of Conjecture 5.2, such an algorithm is computationally less efficient since it is a time-consuming procedure to find a *QNR* modulo n.

10. Acknowledgements

I express my appreciation to Professor W. Gruver, Professor I. V. Semushin and to R. Rubino for constructive suggestions that improved the style of this paper. For computer experiments, the counting of points on the EC is performed with applets created by S. Sadik and by my former student B. Saraswat. I am grateful to them, and to my former students Dr. Y. S. Polyakov and S. Medicherla for their assistance in running computer experiments and to the reviewers for their fruitful comments. I deeply appreciate advice of Dr. D. Kanevsky.

11. References

[1] R. Crandall and C. Pomerance, "Prime Numbers: A Computational Perspective," Springer, New York, 2001.

[2] H. Cohen, "A Course in Computational Algebraic Number Theory," Springer-Verlag, New York, 1996.

[3] D. Shanks, "Class Number, a Theory of Factorization and Genera," *Proceedings of Symposium of Pure Mathematics*, Vol. 20, 1969, pp. 415-440.

[4] S. Lehman, "Factoring Large Integers," *Mathematics of Computation*, Vol. 28, 1974, pp. 637-646. doi:10.1090/S0025-5718-1974-0340163-2

[5] J. Pollard, "Theorems on Factorization and Primality Testing," *Mathematical Proceedings of the Cambridge Philosophical Society*, Vol. 76, 1974, pp. 521-528. doi:10.1017/S0305004100049252

[6] J. Pollard, "Factoring with Cubic Integers," *The Development of the Number Field Sieve, Lecture Notes in Mathematics*, Vol. 1554, 1993, pp. 4-10. doi:10.1007/BFb0091536

[7] C. Pomerance, "Analysis and Comparison of Some Integer Factoring Algorithms," In: H. W. Lenstra and R. Tijdeman, Eds., *Computational Methods in Number Theory*, Math Centre Tracts—Part 1, Math Centrum, Amsterdam, 1982, pp. 89-139.

[8] C. Pomerance, "The Quadratic Sieve Factoring Algorithm," *Advances in Cryptology*, *Proceedings of Eurocrypt*'84, LNCS, Springer-Verlag, Berlin, 1985, 169-182.

[9] R. D. Silverman, "The Multiple Polynomial Quadratic Sieve," *Mathematics of Computation*, Vol. 48, 1987, pp. 329-339. doi:10.1090/S0025-5718-1987-0866119-8

[10] J. Buhler, H. W. Lenstra and C. Pomerance, "Factoring Integers with the Number Field Sieve," In: A. K. Lenstra and H. W. Lenstra, Eds., *The Development of the Number Field Sieve*, *Lecture Notes in Mathematics*, Springer-Verlag, Berlin, Vol. 1554, 1993, pp. 50-94. doi:10.1007/BFb0091539

[11] A. K. Lenstra and A. Shamir, "Analysis and Optimization of the TWINKLE Factoring Device," *Advances in Cryptology—EUROCRYPT* 2000, *Lecture Notes in Computer Science*, Springer-Verlag, New York, Vol. 1807, 2000, pp. 35-52.

[12] A. Shamir and E. Tromer, "Factoring Large Numbers with the TWIRL Device," *Advances in Cryptology—CRYPTO* 2003, *Lecture Notes in Computer Science*, Springer-Verlag, New York, Vol. 2729, 2003, pp. 1-26.

[13] P. W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM Journal on Computing*, Vol. 26, No. 5, 1997, pp. 1484-1509. doi:10.1137/S0097539795293172

[14] R. P. Brent, "Some Integer Factorization Algorithms Using Elliptic Curves," *Proceedings of* 9*th Australian Computer Science Conference*, Canberra, January 1985.

[15] H. W. Lenstra Jr., "Factoring Integers with Elliptic Curves," *Annals of Mathematics*, Vol. 126, No. 2, 1987, pp. 649-673. doi:10.2307/1971363

[16] P. L. Montgomery, "A FFT Extension of the Elliptic Curve Method of Factorization," PhD Thesis, University of California, Los Angeles, 1992.

[17] R. Schoof, "Counting Points of Elliptic Curves over Finite Fields," *Journal de Théorie des Nombres de Bordeaux*, Vol. 7, No. 1, 1995, pp. 219-254. doi:10.5802/jtnb.142

[18] R. Lencier, D. Lubicz and F. Vercauteren, "Point Counting on Elliptic and Hyperelliptic Curves," In: H. Cohen and G. Frey, Eds., *Handbook of Elliptic and Hyperelliptic Curve Cryptography*, Chapman & Hall/CRC, Boca Raton, 2006, pp. 407-453.

[19] A. G. B. Lauder and D. Wan, "Counting Points on Varieties over Finite Fields of Small Characteristics," In: J. P. Buhler and P. Stevenhagen, Eds., *Algorithmic Number Theory*, Cambridge University Press, Cambridge, 2008, pp. 579-612.

[20] A. Weil, "Number of Solutions of Equations in Finite Fields," *Bulletin of American Mathematical Society*, Vol. 55, 1949, pp. 497-508. doi:10.1090/S0002-9904-1949-09219-4

[21] A. G. B. Lauder, "Counting Solutions to Equations in Many Variables over Finite Fields," *Foundation of Computational Mathematics*, Vol. 4, No. 3, 2004, pp. 221-267. doi:10.1007/s10208-003-0093-y

[22] Boris S. Verkhovsky, "Integer Factorization: Solution via Algorithm for Constrained Discrete Logarithm Problem," *Journal of Computer Science*, Vol. 5, No. 9, 2009, pp. 674-679. doi:10.3844/jcssp.2009.674.679

[23] "RSA Factoring Challenge," http://en.wikipedia.org/wiki/RSA_Factoring_Challenge

Appendix

A1: Integer factorization of "larger" *n*

Suppose that n = 5,912,473,983,049,810,121,582,491,435,559,753.

1. Compute four distinct values *A*, *Q*, *R*, and *U* {see (2.2) and **Table A.1**}, where $Q > \max(A, R) > \min(A,R) > U$;

2. Reduce $Q := Q \bmod 10^{28}$; $A := A \bmod 10^{28}$; $R := R \bmod 10^{28}$; $U := U \bmod 10^{28}$;

3. $S := (Q - U - |A - R|)/4$;

4. $p = \gcd(n,S)$ = 59,604,644,783,353,249;

5. $q = n/p$=99,194,853,094,755,497.

Table A.1. Values of *A, Q, R, U* and *S*.

Outputs	Number of Points on Elliptic Curves and *S*
A	5,912,473,961,382,574,071,288,527,255,437,165
Q	5,912,474,044,194,428,121,806,637,304,594,777
R	5,912,474,004,717,045,895,410,530,286,258,045
U	5,912,473,921,905,192,397,824,270,895,949,025
S	19,738,690,974,965,090,844,456,218

A2: Proof of Proposition 7.2

Consider two elliptic curves for an integer positive *m*:

$$ECP:\ \ y^2 = x\left(x^2 + 2^{m+1}\right)\left(\bmod p\right); \qquad \text{(A.1)}$$

$$ECN:\ \ Y^2 = X\left(X^2 - 2^{m-1}\right)\left(\bmod p\right). \qquad \text{(A.2)}$$

Let us show that there exist such integers *u* and *w* that substitutions

$$x{:=}uX \bmod p;\ \text{and}\ y{:=}wY \bmod p \qquad \text{(A.3)}$$

establish an one-to-one correspondence between points of *ECP* and *ECN* for every integer *m*. First of all, from (A.1) we derive

$$w^2Y^2 = uX\left(u^2X^2 + 2^{m+1}\right)\left(\bmod p\right). \qquad \text{(A.4)}$$

Let us select integers *u* and *w*, each co-prime with *p*, for which hold $w^2 = u^3\left(\bmod p\right)$; (A.5)

and $4u = -u^3\left(\bmod p\right)$; or $u^2 = -4\left(\bmod p\right)$. (A.6)

If integer solutions of (A.5) and (A.6) exist, then after cancellation of equal terms in both sides of (A.4) we derive (A.2).

Therefore, from (A.6) $\quad u = 2\sqrt{p-1}\left(\bmod p\right)$; (A.7)

and from (A.5) $\quad w = \sqrt{u}^{3}\left(\bmod p\right)$. (A.8)

Integer *u* exists if *p*mod4=1; since (*p*–1)/2 is *even*.

Therefore, $(-1)^{(p-1)/2} \bmod p = 1$, *i.e.*, *p*–1 is a QR.

On the other hand, integer *w* also exists, because *u* itself is QR modulo *p*.

Indeed, $$\left[2\sqrt{p-1}\right]^{(p-1)/2} \equiv 2^{(p-1)/2} \times \sqrt{p-1}^{(p-1)/2} \bmod p = 1; \tag{A.9}$$

since, as it shown in **Table A.2**, both 2 and $\sqrt{p-1}$ are simultaneously either QR or QNR modulo *p*.

Q.E.D.

Table A.2. Parity of quadratic residuosities of 2 and $\sqrt{p-1}$.

*p*mod8	*p*mod8=1	*p*mod8=5
2	QR	QNR
$\sqrt{p-1}$	QR	QNR

ExampleA1: Let *p*=13; find *u* and *w*, such that $w^2 = u^3 \pmod{13}$;

and $u^2 = 9 \pmod{13}$, *i.e.*, $u = 3$.

Then $w^2 = u^3 = 27 \pmod{13} = 1$.

Therefore *w*= ± 1 and *u*=3. Indeed, $1^2 \equiv 3^3 \pmod{13}$; $3^3 \equiv -4 \times 3 \pmod{13}$.

Therefore, $ECP \xleftrightarrow{x=3X;\, y=Y \pmod{13}} ECN$.

Table A.3 shows one-to-one correspondence between *ECP* and *ECN*

Table A.3. Correspondence between (*x*,*y*) and (*X*,*Y*).

ECP	(0,0)	(3,0)	(10,0)	(2,4)	(2,9)	(11,6)	(11,7)
ECN	(0,0)	(1,0)	(12,0)	(5,4)	(5,9)	(8,6)	(8,7)

ExampleA2: Let *p*=41; find *u* and *w*, such that $w^2 = u^3 \pmod{41}$;

and $u^2 = -4 \equiv 37 \pmod{41}$, *i.e.*, $u = \pm 18$.

Then $w^2 = u^3 = \pm 5832 \equiv \pm 10 \pmod{41}$,

where both 10 and 31 are QR modulo 41. Therefore *w*=16; and *u*=18. Indeed,

$16^2 \equiv 18^3 \pmod{41}$; $18^3 \equiv -4 \times 18 \pmod{41}$. Thus, $ECP \xleftrightarrow{x=18X;\, y=16Y \pmod{41}} ECN$.

Table A.4 shows one-to-one correspondence between points on *ECP* and *ECN* for several non-Blum primes.

Table A.4. (*u*,*w*) as function of *p*.

p	13	29	37	41
(*u*,*w*)	(3, ± 1)	(5, ± 3)	(12, ± 10)	(18, ± 16)