Computación y Sistemas Vol. 4 No.3 pp. 242 - 248
© 2001, CIC - IPN. ISSN 1405-5546 Impreso en México

Quantum Cryptoanalysis of Elliptic Curve Systems

Juan Manuel Garcia Garcia *
Department of Computer Systems
Instituto Tecnologico de Morelia
Morelia, Mexico
E-mail: jmgarcia@sekureit.com

Rolando Menchaca Garcia
Center for Computing Research
Instituto Politecnico Nacional
Mexico City, Mexico
E-mail: menchaca@cic.ipn.mx

Article received on May 11, 1999; accepted on February 7, 2001

Abstract

The security of elliptic curve cryptosystems is based in the intractability of the elliptic curve discrete logarithm problem. The best classical algorithm known until date to solve this problem is fully exponential in time. This is the reason why the elliptic curve public key cryptography is considered the most secure known until date. We present in this paper an algorithm that running on a quantum computer, can solve in polynomial time the elliptic curve discrete logarithm problem. Then, if a functional quantum computer is ever build, all the elliptic curve cryptosystems would become insecure.

Keywords: Quantum algorithms, elliptic curve cryptosystems, public key cryptography

1 Introduction

The security of modern public key cryptographic systems is based in the difficulty to solve efficiently some kind of mathematical problems. Since the invention of the public key cryptography by Diffie and Hellman in 1976 [Diffie,1976], many public key cryptographic systems have been proposed, of these some have been broken and others have been demonstrated to be impractical. Today, only three type of systems are considered enough secure and efficient. Such systems are based in one of the following mathematical problems:

1. Integer factorization problem (IFP).
2. Discrete logarithm problem (DLP).
3. Elliptic curve discrete logarithm problem (ECDLP).

Although none of these problems have been proven to be intractable, are considered as intractable because years of study has failed to yield efficient algorithms to solving them.

The *integer factorization problem* (IFP) consist of the following: given a composite number n that is the product of two large primes p and q, find p and q. Rivest, Shamir and Adleman [Rivest,1978] developed the RSA public-key cryptosystem based on the difficulty of the IFP. Another public-key cryptosystem whose security is based in the intractability of the IFP was developed by Rabin and Williams [Rabin,1979],[Williams,1980].

The *discrete logarithm problem* (DLP) is the following: given a prime p, a generator α of $\mathbb{Z}_p$, and a non zero element $\beta \in \mathbb{Z}_p$, find the unique integer l, $0 \leq l \leq (p-2)$, such that

$$\beta \equiv \alpha^l (mod\ p) \tag{1}$$

The integer l is called the *discrete logarithm* of β to the base α.

Diffie and Hellman proposed the well known Diffie-Hellman key agreement scheme [Diffie,1976] based on the difficulty of the DLP. Since then, many other cryptographic systems whose security is based on the DLP

*This work was supported in part by a ANUIES-SUPERA fellowship.

have been proposed, such as: the U.S. government digital signature algorithm (DSA) [Johnson,1997], the ElGamal encryption and signature schemes [ElGamal,1985], the Schnorr signature scheme [Schnorr,1991], and the Nyberg-Rueppel signature scheme[Nyberg,1996].

The *elliptic curve discrete logarithm problem* (ECDLP) can be defined as follow: If q is a prime power, then F_q denotes the finite field containing q elements. In applications, q is typically a power of 2 (2^m) or an odd prime number p. Given an elliptic curve E defined over F_q, a point $P \in E(F_q)$ of order n , and a point $Q \in E(F_q)$, determine the integer k, between 0 and $n-1$, such that $Q = kP$, provided that such an integer exists.

Based on the intractability of this problem, Neal Koblitz [Koblitz,1987] and Victor Miller [Miller,1986] independently proposed using the group of points on a elliptic curve defined over a finite field to implement the various discrete log cryptosystems. Elliptic curves have been applied to modify public-key cryptographic system, such as the DSA [Williams,1980]. Currently there are underway initiatives for the standarization of elliptic curve cryptography [ANSI X9.62], [ANSI X9.63],[FIPS 186],[ISO/IEC].

With each of the three problems, there are special-purpose classical[1] algorithms that solve the problem in polynomial time for certain special instances. For integer factorization, there is a polynomial algorithm in the case that the integer has small prime factors [Lenstra,1987]. Similarly, for the discrete logarithm problem modulo p, there is a polynomial algorithm provided $p-1$ only has small prime factors. And the ECDLP can be solved relatively easy for a small class of elliptic curves, known as supersingular elliptic curves [Menezes, Okamoto,1993] and also for certain anomalous elliptic curves. However, in each case, the special instances of the problem are easily identified, so an implementation merely checks that the specific instance selected is not one of the class of easy problems. This approach avoids attacks employing these special purpose algorithms.

Of the three problems, the IFP and the DLP both have general-purpose classical algorithms that run in subexponential time [Gordon,1993], [Lenstra,1993],[Pomerance,1985]. These subexponential time algorithms mean that the problem should still be considered hard, but not as hard as those problems which admit only fully exponential time algorithms. Precisely, the running time for the best general classical algorithm known for both of these problems is:

$$O(exp((c+o(1))(ln\ n)^{1/3}(ln\ ln\ n)^{2/3})) \qquad (2)$$

for a constant c. On the other hand, the best general classical algorithm for the ECDLP is fully exponential in time [Pollard,1978]. Its running time is:

$$O(\sqrt{q}) \qquad (3)$$

This is the reason why the ECDLP is considered to be harder than either the IFP or the DLP. With a 160 bit modulus, an elliptic curve systems offer the same level of security as DSA or RSA with 1024 bit moduli [Menezes,1993].

In 1994, Peter Shor found quantum algorithms to solve the IFP and DLP in polynomial time[Shor,1994]. Another polynomial quantum algorithm for the same problems was formulated by Kitaev [Kitaev,1995]. Both approaches are based in the assumption of the feasibility of the quantum computer. These results implies that if a fully functional quantum computer is ever build, all the cryptographic methods based on IFP or DLP would become insecure. Although it was showed before [Boneh,1995] that quantum computers could be used also to solve the ECDLP, in this paper we will show in a explicit way how the Shor algorithm can be easily extended to solve also the ECDLP, so we can conclude that all the current classical public key cryptographic systems would be insecure once we have the first fully functional quantum computer.

2 Background on Elliptic Curves

For simplicity, we shall restrict our discussion to elliptic curves over $\mathbb{Z}_p$ where p is a prime, although elliptic curves can be defined more generally over any finite field. In particular, the *characteristic two finite fields* F_2^m are of special interest because they give us the most efficient implementations of elliptic curve arithmetic.

An *elliptic curve* E over $\mathbb{Z}_p$ is defined by an equation of the form

$$y^2 = x^3 + ax + b \qquad (4)$$

where $a, b \in \mathbb{Z}_p$, such that $4a^3 + 27b^2 \neq 0$ in $\mathbb{Z}_p$. The set $E(\mathbb{Z}_p)$ consists in all points $(x, y) \in \mathbb{Z}_p \times \mathbb{Z}_p$, which satisfy the defining equation, together with a special point O, called the *point at the infinity*.

$E(\mathbb{Z}_p)$ forms an abelian group with the addition operation defined as follow:

1. $O + O = O$
2. $(x, y) + O = (x, y)$, O is the identity
3. $(x, y) + (x, -y) = O$. The inverse of one element is obtained changing the sign of the second component.
4. To add two different elements, which are not one inverse of the other, we apply the following rule:

$$(x_1, y_1) + (x_2, y_2) \quad (x_3, y_3)$$

where

[1] For purposes, of this paper, a *classical* algorithm is designed to run in a classical computer, instead of a quantum computer

$$x_3 = \lambda^2 - x_1 - x_2,$$
$$y_3 = \lambda(x_1 - x_3) - y_1,$$
$$\lambda = (y_2 - y_1)/(x_2 - x_1)$$

5. To add a point with himself, we apply the rule

$$2(x_1, y_1) = (x_3, y_3)$$

where:

$$x_3 = \lambda^2 - 2x_1,$$
$$y_3 = \lambda(x_1 - x_3) - y_1$$
$$\lambda = (3x_1^2 + a)/(2y_1).$$

The last two operations have a straight geometric interpretation. As it is depicted in the figure 1, if $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ are two distinct points over the elliptic curve, then the *sum* of P and Q, denoted as $R = (x_3, y_3)$, is constructed in the following way: the line which pass through P and Q, intersects the elliptic curve in a third point. Taking the reflection of this point over the x axis, we obtain the point R.

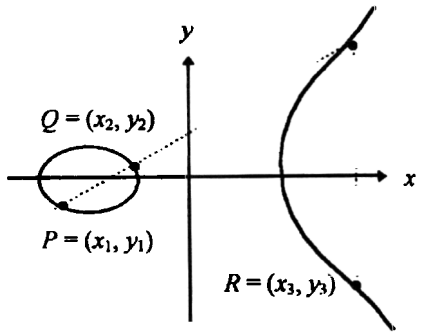

Figure 1: Geometric description of the addition of two elliptic curve points $P + Q = R$

As is depicted in the figure 2, if $P = (x_1, y_1)$ is a point over the elliptic curve, the tangent line to the elliptic curve at P, intersect the curve in a second point. The reflection of this point over the horizontal axis, is the *double* of P, denoted as $R = (x_3, y_3)$.

If $P \in E(\mathbf{Z}_p)$ is a elliptic curve point and $n \in \mathbf{Z}$, then we denote as nP the result of adding n times P with himself; $P + ... + P$. To multiply P by $-n$, means adding n times Q with himself, $Q + ... + Q$, where Q is the inverse of P. As usual, we denote the inverse of P as $-P$. We shall emphasize that all the operations defined on $E(\mathbf{Z}_p)$ could be reduced to operations in the underlying finite field $\mathbf{Z}_p$.

Example 1 Consider the elliptic curve defined by E : $y^2 = x^3 + x + 1$ over $\mathbf{Z}_{23}$. The points in $E(\mathbf{Z}_{23})$ are O and the following:

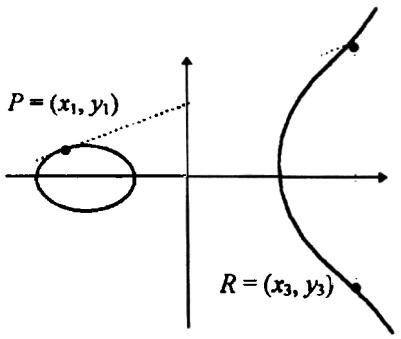

Figure 2: Geometric description of the doubling of an elliptic curve point $2P = R$

(0,1) (0,22)	(6,4) (6,19)	(13,7) (13,16)
(1,7) (1,16)	(7,11) (7,12)	(17,3) (17,20)
(3,10) (3,13)	(9,7) (9,16)	(18,3) (18,20)
(4,0)	(11,3) (11,20)	(19,5) (19,18)
(5,4) (5,19)	(12,4) (12,19)	

Note that each point is grouped with his inverse additive. (The point (4,0) is his own inverse). The number of points in $E(\mathbf{Z}_{23})$ is 28. For example, if $P_1 = (3, 10)$ and $P_2 = (9, 7)$ then $P_1 + P_2 = (17, 20)$ and $10P_1 = 2(2(2P_1) + P_1) = (6, 4)$.

One important result of the elliptic curve theory, is known as the theorem of Hasse, which states that:

$$p + 1 - 2\sqrt{p} \leq |E(\mathbf{Z}_p)| \leq p + 1 + 2\sqrt{p} \qquad (5)$$

where $|E(\mathbf{Z}_p)|$ is the number of points on an elliptic curves, and is called the *order* of the elliptic curve. In other words, the order of $|E(\mathbf{Z}_p)|$ is roughly equal to the size p of the underlying field. Schoof [Schoof,1985], found a polynomial time classical algorithm to count the number of points on an elliptic curve. Some improvements on practical aspects of the algorithm have been made recently [Lercier,1995].

If $P \in E(\mathbf{Z}_p)$ is a elliptic curve point, we define the *order* of P as the least positive n, such that $nP = O$, where O is the point at the infinity, as we introduced before.

The security of elliptic curve cryptosystems, such as the ECDSA [Johnson,1997], is based on the apparent intractability of the *elliptic curve discrete logarithm problem* (ECDLP) stated as follow: given an elliptic curve E defined over $\mathbf{Z}_p$, a point $P \in E(\mathbf{Z}_p)$ of order n, and a point $Q \in E(\mathbf{Z}_p)$, determine the integer k, $0 \leq k \leq n - 1$, such that $Q = kP$, provided that such an integer exists.

There is a classical algorithm, due to Pohlig and Hellman [Pohlig,1978] that reduces the determination of k to the determination of k modulo each of the prime factors of n. Hence, in order to achieve the maximum attainable security level, n should be a large prime.

The best classical algorithm known until now for the general ECDLP is the Pollard rho-method [Pollard,1978], which is of order:

$$O(\sqrt{\pi n/2}) \qquad (6)$$

Menezes, Okamoto and Vanstone [Menezes, Okamoto,1993] showed how the ECDLP can be reduced to the DLP in extension fields of $\mathbb{Z}_p$, for which subexponential time classical algorithms are known. However, this reduction algorithm is efficient only for a special class of curves known as supersingular curves. Such kind of curves can be easily detected, so can be avoided in practical implementations.

As an example of the intractability of the ECDLP, if 10,000 classical[2] computers each rated at 1,000 MIPS are available, and $n \approx 2^{160}$ then an elliptic curve discrete logarithm can be computed in 96,000 years [Certicom,1997].

3 Quantum Implementation of Elliptic Curves

In order to solve the ECDLP on a quantum computer, we must show first that elliptic curve arithmetic could be efficiently implemented on this quantum computer. First of all, we must note that all the operations defined over $E(\mathbb{Z}_p)$ can be reduced to operations in the underlying field $\mathbb{Z}_p$. These operations can be implemented by quantum gates as was showed by Beckman et.al. [Beckman,1996], so we can expect that the elliptic curve points operations can be reduced efficiently to elementary quantum gates.

If $P = (x, y)$ is a point over the elliptic curve $E(\mathbb{Z}_p)$, we can represent such point using two registers of our quantum computer in the states:

$$|x, y\rangle \tag{7}$$

but for sake of clarity we represent these states as only one:

$$|P\rangle \tag{8}$$

We must reserve a special pair of states to represent the point at the infinity. In all the operations we must check when it is involved as operand, or gives as result, this special point, so all the elliptic curve operations are well defined in terms of operations in $\mathbb{Z}_p$.

Since we can build unitary operators who implements all the operations in $\mathbb{Z}_p$, using the representation mentioned before, we can have unitary operators for each one of the operations defined on $E(\mathbb{Z}_p)$. Then, if $a \in \mathbb{Z}$, and $P \in E(\mathbb{Z}_p)$, we have the unitary operator:

$$U_{\otimes} : |a, P\rangle \rightarrow |aP\rangle \tag{9}$$

which give us the multiple of the elliptic curve point P. Similarly, we have the unitary operator to add two elliptic curve points:

$$U_{\oplus} : |P, Q\rangle \rightarrow |P + Q\rangle \tag{10}$$

[2]Classical computer as opposed to quantum computer

In this way, we can always define a unitary operator to compute the linear combination of a pair of elliptic curve points, as a transform of the kind:

$$|a, b, P, Q\rangle \rightarrow |aP - bQ\rangle \tag{11}$$

This particular operator will be used in our quantum algorithm to solve the ECDLP. Also we will need to make use of the quantum Fourier transform, in a very similar way as the main quantum algorithms formulated until now [Josza,1997]. If we have a number a with $0 \leq a < q$, for some q, the Fourier transform converts the state $|a\rangle$ into

$$\frac{1}{\sqrt{q}} \sum_{c=0}^{q-1} |c\rangle exp(2\pi iac/q) \tag{12}$$

Although in this transformation q is of exponential size, actually it can be done in polynomial time if q is a smooth number, that is, it has only small prime factors [Ekert,1996]. Matter of fact, it can be implemented using only one bit-gates and measurements of single bits [Griffiths,1996].

4 Algorithm

Given a point $P \in E(\mathbb{Z}_p)$, of order n, and a point $Q \in E(\mathbb{Z}_p)$, we want to find $k \in \mathbb{Z}_n$ such that $Q = kP$.

We must consider first the case when n, the order of P, is a smooth number, i.e. have only small prime factors, and once we have explained our algorithm, we will show how can be extended to the general case in a similar way to Shor algorithm for the DLP [Shor,1994].

In the first step of our algorithm we put the first two registers in the quantum computer in the uniform superposition of all states,

$$\frac{1}{n} \sum_{a=0}^{n-1} \sum_{b=0}^{n-1} |a, b\rangle \tag{13}$$

We can do this by application of the Fourier transform to the registers in the ground state. Next, using the unitary operator of the linear combination of two elliptic curve points, such as we have seen in the preceding section, we compute $aP - bQ$ and then, the state of our quantum computer will be:

$$\frac{1}{n} \sum_{a=0}^{n-1} \sum_{b=0}^{n-1} |a, b, aP - bQ\rangle \tag{14}$$

Then we use the Fourier transform to convert

$$|a, b\rangle \longrightarrow \frac{1}{n} \sum_{c=0}^{n-1} \sum_{d=0}^{n-1} exp\left(\frac{2\pi i}{n}(ac + bd)\right) |c, d\rangle \tag{15}$$

Therefore the state of the quantum computer is

$$\frac{1}{n^2}\sum_{a,b=0}^{n-1}\sum_{c,d=0}^{n-1} exp\left(\frac{2\pi i}{n}(ac+bd)\right)|c,d,aP-bQ\rangle \quad (16)$$

Finally we observe the state of the quantum computer. The probability of observing a state $|c,d,R\rangle$ with $R = aP - bQ$ is:

$$\left|\frac{1}{n^2}\sum_{\substack{a,b\\ a\equiv kb+r}} exp\left(\frac{2\pi i}{n}(ac+bd)\right)\right|^2 \quad (17)$$

where the sum is over all (a,b) such that

$$a \equiv kb + r \ (mod\ n) \quad (18)$$

since for these (a,b) we have that $aP - bQ$ is the same point over the elliptic curve, rQ, for some r. We use the above relation to reduce the probability to

$$\left|\frac{1}{n^2}\sum_{b=0}^{n-1} exp\left(\frac{2\pi i}{n}(rc+b(d+kc))\right)\right|^2 \quad (19)$$

In the case that $d+kc \neq 0\ (mod\ n)$ the above sum is over a set of nth roots of unity over the unit circle, and thus the probability reduces to 0. Then, the probability is not zero only if $d+kc \equiv 0\ (mod\ n)$. Therefore, for any result $|c,d,R\rangle$ we can obtain k as $-d/c$ in $\mathbb{Z}_n$.

In the general case, we first find a smooth number m such that $n \leq m \leq 2n$ (for the proof of this, see [Shor,1994]). Then, once we have our quantum computer in the state

$$\frac{1}{n}\sum_{a=0}^{n-1}\sum_{b=0}^{n-1}|a,b,aP-bQ\rangle \quad (20)$$

we apply the Fourier transform FT_m to get

$$\frac{1}{m\cdot n}\sum_{a,b=0}^{n-1}\sum_{c,d=0}^{m-1} exp\left(\frac{2\pi i}{m}(ac+bd)\right)|c,d,aP-bQ\rangle \quad (21)$$

The probability of observing a state $|c,d,R\rangle$ with $R = aP - bQ$ is:

$$\left|\frac{1}{m\cdot n}\sum_{\substack{a,b\\ a\equiv kb+r}} exp\left(\frac{2\pi i}{m}(ac+bd)\right)\right|^2 \quad (22)$$

where the sum is over all (a,b) such that

$$a \equiv kb + r \ (mod\ n) \quad (23)$$

Then we use the relation

$$a = kb + r - n\left\lfloor\frac{kb+r}{n}\right\rfloor \quad (24)$$

to obtain the following expression for the amplitude

$$\frac{1}{m\cdot n}\sum_{b=0}^{n-1} exp\left(\frac{2\pi i}{m}\left(bd+bkc+cr-cn\left\lfloor\frac{kb+r}{n}\right\rfloor\right)\right) \quad (25)$$

Without loss of generality, we can ignore the constant term $exp(2\pi icr/m)$. Splitting the exponent in two parts and factorizing b, we obtain

$$\frac{1}{m\cdot n}\sum_{b=0}^{n-1} exp\left(\frac{2\pi i}{m}bT\right) exp\left(\frac{2\pi i}{m}V\right) \quad (26)$$

where

$$T = kc + d - \frac{k}{n}\{cn\}_m \quad (27)$$

and

$$V = \left(\frac{kb}{n} - \left\lfloor\frac{kb+r}{n}\right\rfloor\right)\{cn\}_m \quad (28)$$

where $\{z\}_m$ is the residue of $z\ (mod\ m)$.

At this point, we get a expression for the amplitude of the state very similar to the obtained in the Shor algorithm for the DLP [Shor,1994]. Then we can follow the Shor method in order to obtain the integer k. For the details of the subsequent discussion see Shor's paper [Shor,1994].

As in the Shor algorithm, we have some "good" outputs, from which we can deduce the value of k. These outputs satisfy the following two conditions. First,

$$\{cn\}_m \leq \frac{m}{12} \quad (29)$$

and

$$\left|kc + d - \frac{k}{n}\{cn\}_m\right| \leq \frac{1}{2} \quad (30)$$

The first condition implies that $|V| \leq \frac{m}{12}$, and then we have that $exp(\frac{2\pi iV}{m})$ differs from 1 no more than $exp(\pi i/6)$. From the second condition we can deduce the value of k. To do this, first we can reformulate this condition as

$$\left|d + \frac{k[cn - \{cn\}_m]}{n}\right| \leq \frac{1}{2} \quad (31)$$

Notice that k is the only unknown. Also observe that m divides $[cn - \{cn\}_m]$. Then dividing both sides by m, we get

$$\left|\frac{d}{m} + \frac{kl}{n}\right| \leq \frac{1}{2m} \quad (32)$$

where $l = [cn - \{cn\}_m]/m$. Rounding d/m to the closest multiple of n, called e/n, we can make the approximation $e/n = -kl/n$ and then $k = -e/l$. Thus, we can approximate k as $-nd/[cn - \{cn\}_m]$ in $\mathbb{Z}_n$.

As Shor have demonstrated [Shor,1994], we have a constant, high probability to found pairs c,d that satisfy both conditions, so we can determinate the value of k after a few run trials.

Example 2 Consider $E(\mathbb{Z}_{23})$ as was introduced in the example 1. Let $P = (0,1)$ and $Q = (18,20)$. We want to found k such that $Q = kP$. The order of P, denoted by n, is 28. As we have seen $|E(\mathbb{Z}_{23})| = 28$, and then P is a generator of $E(\mathbb{Z}_{23})$. In the first step of our algorithm, we put the first two registers of the quantum computer in the state

$$\frac{1}{28}\sum_{a=0}^{27}\sum_{b=0}^{27}|a,b\rangle \tag{33}$$

which is a superposition of $28^2 = 784$ states. Next, we apply the unitary operator to compute $aP - bQ$, and the state of our quantum computer will be

$$\begin{aligned}\frac{1}{28}\sum_{a,b=0}^{27}|a,b,aP-bQ\rangle \\ \frac{1}{28}(|0,0,(0,0)\rangle &+ |0,1,(18,3)\rangle + \cdot \\ +|27,26,(1,7)\rangle &+ |27,27,(7,12)\rangle)\end{aligned} \tag{34}$$

Now, we apply the Fourier transform to get the state

$$\frac{1}{784}\sum_{a,b=0}^{27}\sum_{c,d=0}^{27}e^{\left(\frac{\pi i}{14}(ac+bd)\right)}|c,d,aP-bQ\rangle \tag{35}$$

The probability to observe the state $|c,d,R\rangle$ is given by

$$\left|\frac{1}{784}\sum_{a,b}exp\left(\frac{\pi i}{14}(ac+bd)\right)\right|^2 \tag{36}$$

If, for example, we have $R = (12,4)$ then the above sum is over the following pairs of a and b

a	0	1	2	3	4	5	6
b	2	13	24	7	18	1	12
a	7	8	9	10	11	12	13
b	23	6	17	0	11	22	5
	14	15	16	17	18	19	20
b	16	27	10	21	4	15	26
a	21	22	23	24	25	26	27
b	9	20	3	14	25	8	19

since, as can be easily verified, for all those pairs $aP-bQ = (12,4)$.

The above sum would be zero except for the following values of c and d

c	0	1	2	3	4	5	6
d	0	5	10	15	20	25	2
c	7	8	9	10	11	12	13
d	7	12	17	22	27	4	9
c	14	15	16	17	18	19	20
d	14	19	24	1	6	11	16
c	21	22	23	24	25	26	27
d	21	26	3	8	13	18	23

For such values of c and d (except when $c = d = 0$) we have $d/c = 5$ in $\mathbb{Z}_{28}$ and since $-5 = 23$ in $\mathbf{Z}_{28}$ then $k = 23$. We can verify that $Q = 23P$.

Moreover, any observed state $|c,d,R\rangle$, except in the case when $c = d = 0$,will give us the value of k. Then, after one single run of our quantum algorithm, we will obtain the answer with a probability of $756/784 \approx 0.964$.

5 Conclusions

As we have seen in this paper, the basic idea behind Shor's quantum algorithm to solve both the IFP and DLP, also can be used to solve, in polynomial time, the ECDLP.

Acknowledgments

J.M. Garcia wish to thank to Carol A. Martinez for her lovely support.

References

ANSI X9.62, *Public key cryptography for the financial services industry - the Elliptic Curve Digital Signature Algorithm (ECDSA)*, draft, 1997.

ANSI X9.63, *Public key cryptography for the financial services industry - Elliptic Curve Key Agreement and Transport Protocol*, draft, 1997.

Beckman D., A. N. Chari, S. Devabhaktuni, and J. Preskill, "Efficient networks for quantum factoring", *Physical Review A*, volume 54, pages 1034-1063, 1996.

Boneh D. and R. Lipton. "Quantum cryptoanalysis of hidden linear forms", *Proceedings of Crypto'95*, LNCS 963, pp. 424-437, 1995.

Certicom Corp., "Remarks on the Security of Elliptic Curve Criptosystem", *Certicom Whitepaper*, September 1997.
Available at http://www.certicom.com/.

Diffie W. and M. Hellman, "New directions in cryptography", *IEEE Transaction on Information Theory*, volume 22, pages 644-654, 1976.

Johnson D. and A. J. Menezes, "Elliptic curve DSA (ECDSA): An enhanced DSA", *Certicom whitepaper*, March 1997.
Available at http://www.certicom.com/.

Ekert A. and R. Josza, "Shor's quantum algorithms for factorizing numbers", *Review of Modern Physics*, volume 68, pages 733-753, 1996.

ElGamal T., "A public-key cryptosystem and a signature scheme based on discrete logarithms", *IEEE Transactions of Information Theory*, volume 31, pages 469-472, 1985.

FIPS 186, *Digital signature standard*, N.I.S.T., 1993.
Available from http://csrc.ncsl.nist.gov/fips/

Gordon D., "Discrete logarithms in GF(p) using the number field sieve", *SIAM Journal on Discrete Mathematics*, volume 6, pages 124-138, 1993.

Griffiths R. B. and C. S. Niu, "Semiclassical Fourier transform for quantum computation", *Physics Review Letters*, volume 76, pages 3228-3231, 1996.

IEEE P1363, *Standard specifications for public key cryptography*, draft, 1997.

ISO/IEC 14888, *Digital signature with appendix*, draft, 1997.

Josza R., "Quantum algorithms and the Fourier transform", to appear in Proceedings of Santa Barbara Conference on Quantum Coherence and Decoherence, preprint available at LANL quant-ph/9707033.

Kitaev A., "Quantum measurements and the Abelian Stabilizer Problem", preprint available at LANL quant-ph preprint 9511026, 1995.

Koblitz N., "Elliptic Curve Cryptosystems", *Mathematics of computation*, volume 48, pages 203-209, 1987.

Lenstra H.W., "Factoring integers with elliptic curves", *Annals of Mathematics*, volume 126, pages 649-673, 1987.

Lenstra A. K., H. W. Lenstra, Jr., M. S. Manasse and J.M. Pollard, "The number field sieve", *The Development of the Number Field Sieve*, Lecture Notes in Mathematics, volume 1554, Springer-Verlag, pages 11-42, 1993.

Lercier R. and F. Morain, "Counting the number of points on elliptic curves over finite fields: strategies and performances", *Advances in Cryptology - EUROCRYPT 95*, Lecture Notes in Computer Science, volume 921, Springer Verlag, pages 79-94, 1995.

Menezes A., *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, 1993.

Menezes A., T. Okamoto and S. Vanstone, "Reducing elliptic curves logarithms to logarithms in a finite field", *IEEE Transactions on Information Theory*, volume 39, pages 1639-1646, 1993.

Miller V., "Uses of elliptic curves in cryptography", *Advances in Cryptology CRYPTO 85*, Lecture Notes in Computer Science, volume 218, Springer-Verlag, pages 417-426, 1986.

Nyberg K. and R. Rueppel, "Message recovery for signature schemes based on the discrete logarithm problem", *Designs, Codes and Cryptography*, volume 7, pages 61-81, 1996.

Pohlig S. and M. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance", *IEEE Transactions on Information Theory*, volume 24, pages 106-110, 1978.

Pollard J., "Monte Carlo methods for index computation mod p", *Mathematics of Computation*, volume 32, pages 918-924, 1978.

Pomerance C., "The quadratic sieve factoring algorithm", *Advances in Cryptology - EUROCRYPT 84*, Lecture Notes in Computer Science, volume 209, Springer-Verlag, pages 169-182, 1985.

Rivest R.L., A. Shamir and L.M. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM*, volume 21, pages 120-126, 1978.

Rabin M.O., "Digitalized signatures and public-key functions as intractable as factorization", *MIT/LCS/TR-212*, MIT Laboratory for Computer Science, 1979.

Schnorr C.P., "Efficient signature generation by smart cards", *Journal of Cryptology*, volume 4, pages 161-174, 1991.

Schoof R., "Elliptic curves over finite fields and the computation of square roots mod p", *Mathematics of Computation*, volume 44, pages 483-494, 1985.

Shor P.W., "Algorithms for Quantum Computation: discrete logarithms and factoring", *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society Press, pages 124-134, 1994.

Williams H.C., "A modification of the RSA public-key encryption procedure", *IEEE Transactions on Information Theory*, volume 26, pages 726-729, 1980.

Juan Manuel Garcia *received a MSc degree in Computer Science from the Universidad Nacional Autonoma de Mexico (UNAM) in 1994 and he is currently a PhD student at the Centro de Investigacion en Computacion (CIC) of the Instituto Politecnico Nacional (IPN). Since 1991 he is Associate Professor in the Computer Systems Department of the Instituto Tecnologico de Morelia (ITM). His research interests includes Quantum computing, Quantum and Classical cryptography, and Network security.*

Rolando Menchaca Garcia *received a PhD degree in Electric Engineering from the Centro de Investigaciones y Estudios Avanzados (CINVESTAV) of the Instituto Politecnico Nacional (IPN). Currently he is a full time researcher at the Centro de Investigacion en Computacion (CIC) of the Instituto Politecnico Nacional (IPN).*