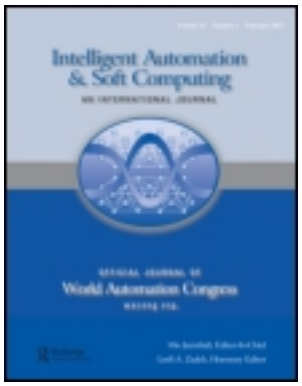

Intelligent Automation & Soft Computing
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tasj20

Implementation And Optimization For Tate Pairing

Guangming Dai [a] , Maocai Wang [a] , Lei Peng [a] & Ruijie Qin [a]
[a] School of Computer , China University of Geosciences , Wuhan , 430074 , P.R. China
Published online: 01 Mar 2013.

To cite this article: Guangming Dai , Maocai Wang , Lei Peng & Ruijie Qin (2011) Implementation And Optimization For Tate Pairing, Intelligent Automation & Soft Computing, 17:5, 607-617, DOI: 10.1080/10798587.2011.10643174

To link to this article: http://dx.doi.org/10.1080/10798587.2011.10643174

Intelligent Automation and Soft Computing, Vol. 17, No. 5, pp. 607-617, 2011

IMPLEMENTATION AND OPTIMIZATION FOR TATE PAIRING

GUANGMING DAI, MAOCAI WANG, LEI PENG, RUIJIE QIN
School of Computer
China University of Geosciences
Wuhan, 430074, P.R. China

ABSTRACT—Tate pairings has found several new applications in cryptography. However, how to compute Tate pairing is a research focus in all kinds of applications of pairing-based cryptosystems (PBC). In the paper, the structure of Miller's algorithm is firstly analyzed, which is used to implement Tate pairing. Based on the characteristics that Miller's algorithm will be improved tremendous if the order of the subgroup of elliptic curve group is low hamming prime, a method of generating primes with low hamming is presented. Then, a new method for generating parameters for PBC is put forward, which enable it feasible that there is certain some subgroup of low hamming prime order in the elliptic curve group generated. Moreover, an optimization implementation of Miller's algorithm for computing Tate pairing is given. Finally, the computation efficiency of Tate pairing using the new parameters for PBC is analyzed, which saves 25.4% of the time to compute the Tate pairing.

Key Words: Pairing-based cryptosystems, Optimization algorithm, Miller's algorithm, Tate pairing, low hamming

1. INTRODUCTION

Pairing-based cryptosystems (PBC) can provide us several novel cryptographic applications, e.g., ID-based cryptosystems, short digital signatures, broadcast encryption, etc. Some of them have not been achieved using the conventional public key cryptosystems[1]. Therefore, PBC have been attracted by researchers in cryptography. In most of these applications, the Weil pairing or Tate pairing of supersingular elliptic curves (or curves of small embedding degrees) are essential tools. Therefore efficient computation of the Weil or Tate pairings is a crucial factor for practical applications of the above mentioned cryptographic protocols[2-3]. The Weil pairing for a given elliptic curve is a symmetric bilinear pairing which can be thought of two applications of the Tate pairing. Thus the Weil pairing is slower to compute than the Tate pairing, and consequently, it is desirable to replace the Weil pairing as the Tate pairing whenever it is possible in PBC. The standard algorithm for computing the Tate pairing is Millers algorithm. Millers algorithm is about 5 times slower than 1024-bit RSA and 160-bit elliptic curve cryptosystem(ECC)[4]. It is an important research topic to find more efficient algorithms for computing Tate pairing. Recently many progresses have been made on the computation of the Tate pairing. A few refined techniques and ideas to speed up the computation of the Tate pairing are suggested in [5-9]. Barreto showed that Miller's algorithm can be modified to a new algorithm where division in a finite field can be omitted since the denominator becomes one after final powering[4]. Also Duursma and Lee[10] presented a closed formula for the computation of the Tate pairing for a finite field with characteristic three, which significantly reduces the cost of computation.

All these improvements focus on the arithmetical operations above the finite field. In fact, the parameters of elliptic curve also impact the computation of the Tate pairing directly. Currently, there are many methods of generating elliptic curve, such as random method[11] and complex multiplication method. However, the purpose of generating these curves are only used them in elliptic curve cryptosystems(ECC), not in pairing-based cryptosystems (PBC). In this paper, according to the special demand of effective computation of Tate pairing, we present a new generation method of elliptic curve. In the elliptic curve group generated, it is sure that there is some subgroup of low hamming prime order, which enable to improve the computation of Tate pairing tremendous in the subgroup of low hamming prime order. Moreover, an optimization implementation of Miller's algorithm for computing Tate pairing is given.

2. MILLER'S ALGORITHM

Computing the Tate pairing is a costly process. To efficiently implement cryptosystems based Tate pairing, it is necessary to optimize the computation time for the Tate pairing. When the pairings were first used, the best known algorithm was exponential in the size of the input. Miller gives an algorithm for the computation which is linear in the size of the input in 1986[12]. For computing the Tate pairing, we need to find the function f_P and then evaluate its value at A_Q. By doing these steps separately, we need to evaluate and compute functions of large degrees. Instead, Miller's algorithm uses a "double and add" algorithm for elliptic curve point multiplication with an evaluation of points on the curve and on lines which appear during the addition of points. In the algorithm, we denote by TDBL an algorithm for point doubling and updating the value f, and by TADD an algorithm for point addition and updating f.

The input for the algorithm are two points $P \in G_1$ and $Q \in G_2$, where G_1 and G_2 are the subgroups $E(F_p)[r]$, $rP_1 = o$, $rP_2 = o$, $G_1 \cap G_2 = \varnothing$, r be a positive integer which is coprime to p. Usually we choose r as a prime divisor of $\#E(\mathrm{F}_q)$.

We also know the binary representation of r: $r = \sum_{i=0}^{m} b_i 2^i$.

The output of the algorithm is the value $f = f_p(A_Q) \in F_p$. To get a unique value of the Tate pairing of the points P and Q, we need to raise f to the power of $(q-1)|r$, thus eliminating all r-th powers.

Algorithm 1: Miller's algorithm

Input: The points $P \in G_1$, $Q \in G_2$ and an integer r with its binary representation $r = \sum_{i=0}^{m} b_i 2^i$.

Output: $f_p(A_Q) \in F_p$

for i=:m-1,m-2,...,1,0 do
  compute TDBL
 if b_i=1 then
compute TADD
 end if
end for
return $f_p(A_Q)$

According to Miller's algorithm, it is clear that the computation of Tate pairing will be improved tremendous if the prime order r of the subgroup $E(F_p)[r]$ of elliptic curve group $E(F_p)$ is low hamming prime, which means that there are few bit 1 in the binary representation of r while lots of bit 0. When the prime order r of the subgroup has the form as follows $r = 2^n + 1$, there are only two bits with 1 while others with 0 in the binary representation of r. Therefore, the process to compute TADD can be reduced to once time in the computation of Tate pairing.

3. GENERATING PRIMES OF LOW HAMMING WEIGHT

Primality tests come in two varieties: deterministic and probabilistic. Deterministic tests determine with absolute certainty whether a number is prime. Examples of deterministic tests include the Lucas-Lehmer test and elliptic curve primality proving. Probabilistic tests can potentially (although with very small probability) falsely identify a composite number as prime (although not vice versa). However, they are in general much faster than deterministic tests. Numbers that have passed a probabilistic prime test are therefore properly referred to as probable primes until their primality can be demonstrated deterministically.

Miller-Rabin strong pseudoprime test is a particularly efficient test. If n is a large positive integer, Miller-Rabin test will determine whether n is prime or composite, with arbitrarily small probability of error. To generating primes of low hamming, the alogrithm of generating low hamming with weight 3 is presented and then its primality is tested by Miller-Rabin test. The algorithm of generating primes of low hamming with weight 3 is as follows.

Algorithm 2: The algorithm of generating primes of low hamming with weight 3

Input: the length $m \geq 160$ for an integer tested; a positive integer t for the number of trials

Output: prime n of low hamming with weight 3

Step1. Choose random k in the interval $0 < k < m$.

Step2. $n \leftarrow 2^m + 2^k + 1$

Step3. Compute v and odd w such that $n - 1 = 2^v w$.

Step4. For j from 1 to t do

4.1 Choose random a in the interval $0 < a < n$.

4.2 Set $b \leftarrow a^w \bmod n$.

4.3 If $b = 1$ or $n - 1$, goto step 4.6

4.4 For i from 1 $v - 1$ to do

4.4.1 Set $b \leftarrow b^2 \bmod n$.

4.4.2 If $b = n - 1$ goto step 4.6

4.4.3 if $b = 1$, goto step 1

4.4.4 Next i.

4.5 goto step 1

4.6 Next j.

Step 5. Output n

The algorithm outputs n, which $n = 2^m + 2^k + 1$ is almost certainly prime of low hamming with weight 3. For example, given $m = 189$, n is one prime of low hamming with weight 3, where

$n=2^{188}+2^{101}+1=392318858461667547739736841485780351462856018272408567809$.

4. GENERATING ELLIPTIC CURVES INCLUDING THE SUBGROUP OF LOW HAMMING ORDER

It is a well known fact that the endomorphism ring of an elliptic curve over a number field is isomorphic to either Z or an order in an imaginary quadratic field. If the latter holds then the curve is said to have complex multiplication (CM.) Elliptic curves with complex multiplication have found applications in cryptography and coding theory, since there are closed form expressions for the number of points on such curves modulo prime ideals. This property was also utilized in the Atkin-Morain primality proving method[13]. Constructing elliptic curves with complex multiplication is computationally very expensive. In this paper, we produce an effective method constructing elliptic curves with CM, which can generate elliptic curves including the subgroup of low hamming order.

If E is a non-supersingular elliptic curve over F_p of order u, then

$$z=4p-(p+1-u)^2$$

is positive by the Hasse bound. Thus, there is a unique factorization

$$z=DV^2$$

Where D is squarefree. Thus, for each non-supersingular elliptic curve over F_p of order u, there exists a unique squarefree positive integer D such that

$$4p=W^2+DV^2 \tag{1}$$

$$\#E=u=p+1\pm W \tag{2}$$

for some W and V. It is said that E has complex multiplication by D. D is called a discriminant for p.

Theorem 1[14]: If E is a non-supersingular elliptic curve over F_p with order u, then

$$4u=A^2+DB^2$$

$$4p=(A\pm 2)^2+DB^2$$

For some A and B, where D is a discriminant for p.

According to "IEEE Standard Specifications for Public-Key Cryptography" in [15], while constructing a curve with prescribed CM, if $D=3$, the coefficients a_0 and b_0 of E is 0 and 1 respectively. Therefore, theorem 1 can be written as:

$$4u=A^2+3B^2 \tag{3}$$

$$4p=(A\pm 2)^2+3B^2 \tag{4}$$

Theorem 2[16]. If group H is the subgroup of group G, $\#G = \#H \bullet |G:H|$. Where $\#G$ is the order of group G, $\#H$ is the order of group H.

If the order $\#G$ of group G satisfies $\#G = r^* r$, then there are certain subgroups with order r in group G according theorem 2.

The algorithm of generating elliptic curves including the subgroup of low hamming order is as follows.

Algorithm 3: The algorithm of generating elliptic curves including the subgroup of low hamming order

Input: the length $m \geq 160$ for the subgroup order

Output: a, b and prime p as the parameter of the elliptic curve $y^2 \equiv x^3 + ax + b \mod p$, low hamming order r as the order of subgroup, point P_1 as the based point for generating subgroup G_1 while calculating Tate pairing, where $rP_1 = o$ and point P_2 as the based point for generating subgroup G_2, while $rP_2 = o$ and $G_1 \cap G_2 = \varnothing$.

Step 1. Generating low hamming prime r under given bits m using algorithm 2.

Step 2. Calculating the order u of elliptic curve over F_p: $\#E = u = r^* r$.

Step 3. Specifying $D = 3$, calculating p according to equation (3) and (4).

Step 4. If p is not a prime, goto Step 1.

Step 5. Selecting an integer ξ with $0 < \xi < p$.

Step 6. Setting $0 \to a$ and $b_0 \xi \mod p \to b$.

Step 7. Looking for a point P_1 with order r on the curve $y^2 \equiv x^3 + ax + b \mod p$.

Step 8. If the output of Step 7 is wrong order, goto Step 1.

Step 9. Looking for a point P_2 with order r on the curve $y^2 \equiv x^3 + ax + b \mod p$, where $P_2 \notin \{kP_1 \mid k \in \{1,2,...,r\}\}$.

Step 10. Ouput p, a, b as the parameter of the elliptic curve $y^2 \equiv x^3 + ax + b \mod p$, low hamming order r as the order of subgroup, point P_1 as the based point for generating subgroup G_1 while calculating Tate pairing, where $rP_1 = o$ and point P_2 as the based point for generating subgroup G_2, while $rP_2 = o$ and $G_1 \cap G_2 = \varnothing$.

It is feasible that the elliptic curve generated by algorithm 3 includes two different subgroups with low hamming order r as subgroup G_1 and G_2 respectively for computing Tate pairing. Because the order r of subgroup is a public parameter, these parameters generated by the algorithm presented in the paper don't affect the security of PBC itself.

The test result for algorithm 3 is as followings.

Test result with 189 bits.

r= $2^{188} + 2^{101} + 1$ = 392318858461667547739736841485780351462856018272408567809

u=r*r= 153914086704665934422965002380478937065239300146784650062493196265043891122629691760314855922519876939750551060481

p=153914086704665934422965002380478937065239300146784650062100877406582223574889954918829075571057020921478142492673

a=0,b=19939371

The based point P_1 for generating subgroup G_1 while calculating Tate pairing is:

P_1=(101632576, 645655967795359524324907144091481243867041983049751623166184516901626696125274037381986432971258553239372450708 8)

Another based point P_2 for generating subgroup G_2 is:

P_2=(787, 622073653175303030798684744246819686582909998874224851239448448036666458824030646893254612406319452289407143765 79)

5. OPTIMIZATION FOR MILLER ALGORITHM

Suppose $P=(x_1,y_1)$ and $Q=(x_2,y_2)$ are points on *E*. if $x_1 \neq x_2$, the TADD computation $\mathrm{P}+\mathrm{Q}$ will have coordinates (x_3,y_3), where

$\lambda=(y_2-y_1)/(x_2-x_1)$

$x_3=\lambda^2-x_1-x_2$

$y_3=(x_1-x_3)\lambda-y_1$

if $x_1=x_2$, the TDBL computation $2P$ will have coordinates (x_4,y_4), where

$\lambda=(3x_1^2+a)/(2y_1)$

$x_4=\lambda^2-2x_1$

$y_4=(x_1-x_4)\lambda-y_1$

In Miller's algorithm, we first need to compute TDBL and then to compute TADD if $b_i=1$. Therefore, the costs of operations on E have 3 squares, 2 multiplications and 2 divisions when $b_i=1$.

We have seen above that, to compute $2\mathrm{P}+\mathrm{Q}$, we need a TDBL and a TADD. In fact, it can be achieved as follows[17]. To form $2\mathrm{P}+\mathrm{Q}$, where $P=(x_1,y_1)$ and $Q=(x_2,y_2)$, we first find $\mathrm{P}+\mathrm{Q}$ while we omit its y-coordinate, because we will not need it for the next stage. This saves a field multiplication. Next we form $(\mathrm{P}+\mathrm{Q})+\mathrm{P}$. One additional squaring is saved when $\mathrm{P}\neq\mathrm{Q}$ because the order of our operations avoids a point doubling. Therefore, according the algorithm, performing a doubling and an addition, $2\mathrm{P}+\mathrm{Q}$, on an elliptic curve *E* use only 2 squares, 1 multiplication, and 2 divisions, which saves a square and a multiplication. The detailed process for computing $2\mathrm{P}+\mathrm{Q}$ in order $(\mathrm{P}+\mathrm{Q})+\mathrm{P}$ as follows.

Suppose $P=(x_1,y_1)$ and $Q=(x_2,y_2)$ are distinct points on *E* and $x_1 \neq x_2$, the TADD computation $\mathrm{P}+\mathrm{Q}$ will have coordinates (x_3,y_3), where

$\lambda=(y_2-y_1)/(x_2-x_1)$

$x_3=\lambda^2-x_1-x_2$

$y_3=(x_1-x_3)\lambda-y_1$

Now suppose we want to add $(\mathrm{P}+\mathrm{Q})$ to P. We must add (x_1, y_1) to (x_3, y_3) using the above rule. Assume $x_1 \neq x_3$. The result has coordinates $(\mathrm{x}_4, \mathrm{y}_4)$, where

$$\lambda' = (\mathrm{y}_3 - \mathrm{y}_1) / (\mathrm{x}_3 - \mathrm{x}_1)$$

$$\mathrm{x}_4 = \lambda'^2 - \mathrm{x}_1 - \mathrm{x}_3$$

$$\mathrm{y}_4 = (\mathrm{x}_1 - \mathrm{x}_4)\lambda' - \mathrm{y}_1$$

We can omit the y_3 computation, because it is used only in the computation of λ', which can be computed without knowing y_3 as follows:

$$\lambda' = -\lambda - 2\mathrm{y}_1 / (\mathrm{x}_3 - \mathrm{x}_1)$$

In this improvement algorithm, we denote by TBAD an algorithm for point addition twice in the form $(\mathrm{P}+\mathrm{Q})+\mathrm{P}$ and updating the value f. Other parameter is same with Miller's algorithm.

Algorithm 4: optimization of Miller's algorithm

Input: The points $P \in G_1$, $Q \in G_2$ and an integer r with its binary representation $r = \sum_{i=0}^{m} b_i 2^i$.

Output: $f_p(A_Q) \in F_p$

for i=:m-1,m-2,...,1,0 do
  if $\mathrm{b_i}=1$ then
    compute TBAD
  else
    compute TDBL
  end if
end for
return $f_p(A_Q)$

Estimating a division as 5.18 multiplications, we see that the improvement algorithm saves an estimated 13% of the time to compute the Tate pairing as follows:

2/(3+2+5.18*2)=13%

6. EFFICIENCY ANALYZE

In Miller algorithm, for every bit of the order r of subgroup, it needs 16 multiplies, 7 inverses. In addition, if the bit is 1, it still needs 11 multiplies, 5 inverses. For the order r of subgroup with 190 bits, in ordinary Pairing-based cryptosystems (PBC) there are 95 bits with 1 in average. It needs 4085 multiplies and 1805 inverses. While using the parameters presented in the paper, it needs 4085 multiplies and 1805 inverses. The detailed operators are shown in Table I.

Estimating an inverse as 5.18 multiplications[17], PBC over parameters presented in the paper saves 25.4% of the time to compute the Tate pairing as follows:

(3062+1340*5.18)/(4074+1800*5.18)=74.6%

Therefore, the computation efficiency of Tate pairing on PBC with parameters generated in the paper is faster much than these on ordinary PBC.

In the environment of Pentium 4 PC (CPU 3.06GHz, RAM 512MB), the test for computing Tate pairing with parameters with 189 bits presented in the paper is shown in Figure 1.

Table I. Efficiency Compare

	The ordinary PBC				PBC with parameter in the paper			
	Every bit (190 bits)		Every bit with 1 (94 bits)		Every bit (190 bits)		Every bit with 1 (2 bits)	
	Multiple	Inverse	Multiple	Inverse	Multiple	Inverse	Multiple	Inverse
	16	7	11	5	16	7	11	5
	3040	1330	1034	470	3040	1330	22	10
Total	Multiple:4074 Inverse:1800				Multiple:3062 Inverse:1340			

In Figure 1, it takes 203.2ms to compute 5 Tate pairings including $t(P,Q)$, $t(P,2Q)$, $t(2P,Q)$, $t(P,3Q)$ and $t(3P,Q)$ and other computations, which means that it takes less 40ms for computing a pairing in average. The computing result is also verified the bilinear characteristic is as followings.

$$t(P,2Q)=t(2P,Q)=t(P,Q)^2$$

$$t(P,3Q)=t(3P,Q)=t(P,Q)^3$$

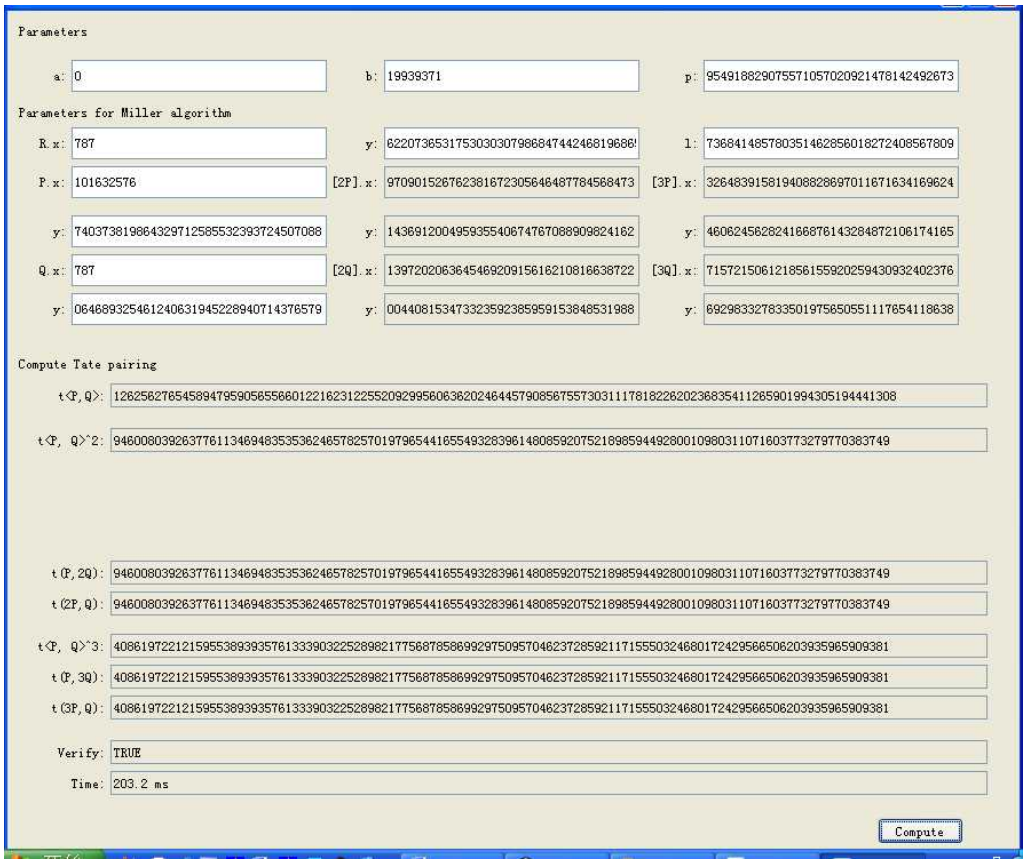

Figure 1. Test result with 189 bits.

7. SECURITY ANALYZE

In this paper, an efficient method of generating parameters for PBC was presented. The specific cryptograph scheme is not mentioned. In fact, the security of any scheme based the algorithms presented in the paper could be analyzed using these methods based ordinary PBC[18-19].

Goldwasser gave the standard model for studying the security of signature schemes[20]. There an adversary A challenged with a fixed public key, is allowed to adaptively request signatures on messages of his choice and is tasked to produce an existential forgery for that key, i.e. a valid signature for any previously unrequested message. To capture security in the identity based setting, we extend this model by additionally allowing A to obtain private keys S_{ID} corresponding to identities ID of his choice and to request signatures on messages and for identities of his choice. The adversary's task is now to produce a signature on a message and identity of his choice, but not for an identity for which he has requested the private key, and not for a message/identity combination for which he has already requested a signature. The adversary's advantage is the probability that his final output is accepted as a valid signature for his choice of message and identity.

We consider the security of the scheme against such an extended adversary in the random oracle model. Suppose then that the hash function H is replaced by a random function in the scheme. Then we can show, using techniques similar to those gave by Boneh[21], that an adversary A with advantage ε against the scheme can be used to build an adversary B who can produce forgeries for a related, non-identity based signature scheme with advantage ε/cN. Here N is the number of H queries made by A and c is a small constant. In PBC, the fixed public key is Q_{ID}, the corresponding private key is S_{ID} and the verification condition holds. This ordinary signature scheme resembles the generalized ElGamal signature scheme[22]. Thus the security of the identity based scheme is linked to the security of an ordinary signature scheme which resembles a well-known scheme. If the ordinary scheme is secure, then the signature scheme is also secure, and we can say that the ability to make private key extractions is of essentially no use to an adversary A.

The only difference is the order of subgroup for computing Tate pairing. The order in the algorithms is a prime with low hamming weight while the order in ordinary PBC is any prime. In pairing-based cryptosystems, the order of subgroup is a public parameter. Moreover, the order with low hamming weight is only used in computing Tate pairing. So the special order could not affect the security of PBC based the algorithms presented in the paper.

8. CONCLUSION

In this paper, we analyze the structure of Miller's algorithm, which is used to implement Tate pairing. According to the characteristics that Miller's algorithm will be improved tremendous if the prime order r of the subgroup $E(F_p)[r]$ of elliptic curve group $E(F_p)$ is low hamming prime, we firstly present a method of generating primes with low hamming. Then, a new method generating elliptic curve is put forward, which enable it feasible that there is certain some subgroup of low hamming prime order in the elliptic curve group generated. Moreover, an optimization implementation of Miller's algorithm for computing Tate pairing is given. Finally, we analyze the computation efficiency of Tate pairing using the new parameters and optimization algorithm.

ACKNOWLEDGMENT

This research is supported by National Nature Science Foundation of China under Grant No. 60873107 to G.M. Dai, Nature Science Foundation in Hubei under Grant No. 2010CDB04104 to M.C. Wang and No. 2008CDB348 to G.M. Dai and Special Funds to Finance Operating Expenses for Basic Scientific Research of Central Colleges in China under Grant No. CUGL090241 to M.C. Wang and No. CUGL090238 to L. Peng.

REFERENCES

1. M.C. Wang, H.P. Hu, G.M. Dai. An Identity-based Signature Scheme for Mobile Business. ICIC Express Letters, 2010, 4(2):565-569.
2. X.A. Boyen. Promenade through the new cryptography of bilinear pairings// Proceedings of IEEE Information Theory Workshop. Washington, USA: IEEE Press, 2006:19–23.
3. P. Barreto, S. Galbraith, C. Eigeartaigh, et al. Efficient pairing computation on supersingular abelian varieties. Designs, Codes and Cryptography, 2007, 42(3):239–271.
4. P.S.L.M. Barreto, B. Lynn, M. Scott. Efficient Implementation of Pairing-Based Cryptosystems. Journal of Cryptology, 2004, 17(4): 321-334.
5. S.D. Galbraith, K. Harrison, D. Soldera. Implementing the tate pairing// Proceedings of the Fifth Symposium on Algorithmic Number Theory, LNCS 2369, Springer-Verlag, 2002:324-337.
6. B. Lynn. On the implementation of pairing-based cryptosystems. USA: Stanford University, 2007.
7. M. Stogbauer. Efficient Algorithms for pairing-based cryptosystems. Germany: Darmstadt University of Technology, 2004.
8. M. Scott, N. Costigan, W. Abdulwahab. Implementing Cryptographic Pairings on Smartcards//Proceedings of Cryptographic Hardware and Embedded Systems, LNCS 4249, Springer-Verlag, 2006:134-147.
9. S. Kwon. Efficient Tate Pairing Computation for Elliptic Curves over Binary Fields// Proceedings of Australasian Conference on Information Security and Privacy, LNCS 3574, Springer-Verlag, 2005:134-145.
10. I. Duursma I, H. Lee. Tate pairing implementation for hyperelliptic curves $y^2 = x^p - x + d$ // Proceedings of AsiaCrypt 2003, LNCS 2894, Springer-Verlag, 2003:111-123.
11. M.C. Wang, G.M. Dai, H.P. Hu, et al. Selection of Security Elliptic Curve Based on Evolution Algorithm// Proceedings of international conference on Computational Intelligence and Natural Computing, IEEE Press, 2009:55–57.
12. V.S. Miller. Short Programs for Functions on Curves. New York: Exploratory Computer Science IBM,1986.
13. Atkin A.O.L. AND F. Morain F. Elliptic curves and primality proving. Research Report 1256, INRIA, Juin 1990. Submitted to Math. Comp.
14. M.C. Wang, G.M. Dai, L. Pen, et al. An efficient method of generating parameters for pairing-based cryptosystems. China Communications, 2010(10):162-165.
15. IEEE-SA Standards Board. IEEE standard specifications for public-key cryptography.2000-1-30.
16. K.C. Lu, H.M. Lu. Algorithms Introduce of elliptic curve cryptosystems]. Beijing: Tsinghua University Press, 2008:22-22. (in Chinese).

17. K. Eisentrager, K. Lauter, P.L. Montgomery. Fast elliptic curve arithmetic and improved Weil pairing evaluation// Proceedings of the 2003 RSA conference on The cryptographers' track, LNCS 2612, Springer-Verlag, 2003: 343-354.
18. Li J.G., Jiang P.J. An Efficient and Provably Secure Identity-Based Signature Scheme in the Standard Model. Chinese Journal of Computers, 2009, 32(11):2130-2136(in Chinese).
19. Koblitz N. and Menezes A. Pairing-based cryptography at high security levels. In Cryptography and Coding, Springer-Verlag LNCS 3796, 13–36, 2005.
20. S. Goldwasser, S. Micali and R. Rivest. A digital signature scheme secure against adaptive chosen-message attachs. SIAM J. Computing, Vol. 17(2), 1988, 281-308.
21. D. Boneh, M. Franklin. Identity-Based encryption from the Weil pairing. In Advances in Cryptology'01, Springer-Verlag, 2001, 213-229.
22. K. G. Paterson. ID-based Signatures from Pairings on Elliptic Curves. Electronics Letters 38(18), 2002, 1025-1026.

ABOUT THE AUTHORS

G. Dai, Ph.D., professor, Ph. D. supervisor, research fields: optimization theory and algorithm.

M. Wang, Ph.D., postdoctor, associate professor, research fields: cryptograph theory and optimization algorithm.

L. Peng, Ph.D., lecture, research fields: optimization theory and algorithm.

R. Qin, an undergraduate student, research fields: cryptograph theory.