Performance analysis of cryptosystems using algebraic geometric code over various fields Fp

Manju C

Department of Computer Science Bharathidasan Government College for Women Puducherry, India

Summary
In this paper, the performance analysis of a cryptosystem using algebraic geometric code over various fields is studied. Implementation is done by an algorithm developed using Mat lab. A comparison on time taken for finding key generation, encryption and decryption over various fields is done. Result indicates that key generation, encryption and decryption time increases as the size of field size increase. Security level also increases with field size.

Key words: Algebraic geometric code, elliptic curves, Encryption, Decryption

1. Introduction

Cryptosystem using algebraic geometric code was developed by Mc-Eliece [1] during early 1970's.. Later Niederaiter [2] and others developed cryptosystem using algebraic geometric code. But due to excessive size of key they were not efficiently used. In this paper we will deal with a cryptosystem using algebraic geometric code which makes use of elliptic curve and we will have a study about the time taken to execute it over various fields.

The organization of this paper is as follows. In section 2, an overview of algebraic geometric code is presented. Next section deals with a brief overview of cryptosystem using elliptic curves. In section 4: key generation, encryption and decryption are given. In section 5 implementation details and comparison over various field is given. Finally conclusion is given.

2. Algebraic geometric code

In 1948[3] Claude Shannon's paper "theory of communication" led to twin disciplines- information theory and coding theory. Main aim of these two disciplines is to provide efficient and reliable communications. To be efficient the transfer of information must not require a prohibitive amount of time and effort and to be reliable the received data stream must resemble the transmitted stream to with the narrow tolerances.

Manuscript received November 5, 2009
Manuscript revised November 20, 2009

Coding theory deals with error correction and detection of the information transmitted. It involves generation of codes, encoding of information transmitted, decoding of information received. Algebraic geometric codes are code generated using curves. As in any other code these codes also generator and parity check matrices.

Algebraic geometric code is code defined over a curve. The code is defined by Goppa [4]. The curve used in generation of algebraic geometric code is defined over a finite field Fq. The curve should be absolutely irreducible nonsingular and equations of curve should be polynomials with coefficients Fq.

2.1 Divisor, Rational functions and Function field

A divisor[5] D on a curve X is a formal sum of form $D=\sum n_p P$ where $n_p \varepsilon Z$ and $n_{p=0}$ for all but a finite number of points P on X. Divisors are often thought to be the key stone to understand how algebraic geometry is formed and its relationship to curve. Another important thing in the construction of algebraic geometric code is order function. The order is a generalization of the degree of a function as well as its zeroes. There are two candidates, the x-order and the y-order. Usually they are the same; however care must be taken to ensure their accuracy.

Let X: f (x, y) =0 be a curve and P (x=α, y=β) be a point on curve X, with α and β ε F, Let g (x, y) ε F [X], then the largest power n for which there exists polynomials $g^0 \varepsilon F[X]$ and h^0 (x, y) ε F [x, y] with $h^0 (0, 0) \neq 0$ such that

$$g = ((x- \alpha)\, g^0(x- \alpha) / h^0(x- \alpha, y- \beta)) \bmod f$$

is called the x-order of g at P and denoted by $ord_{p,x}(g)$.The x-order can be defined using the notation $V_{p,x}(g/h)$ is defined as $V_{p,x}(g)- V_{p,x}(h)$ and y order is defined analogously.

Rational function [6] can be defined as follows: Let X is a curve defined by a field F. On the points of X, any two polynomials that differ by multiplies of F have same value. So when we compare it with the curve they will be the same. So we can say or define rational function R as the ratio f =A(x, y, z)/B(x, y, z) of two homogeneous polynomials of the same degree up to factorization modulo F(x, y, z) A

rational function [11] f is defined at a point P, if there exists a representation f = A/B such that B (P) ≠ 0.
Another important thing we have to discuss before the construction and definition of algebraic geometric code is the space associated with the divisor. The space associated with the divisor can be called linear space.
Let D =$\sum n_p P$, be a divisor and space associated to D [5] denoted by L (D) is the linear vector space which contains set of all functions satisfying Vp (f) ≥ -n_p at every point P, together with the zero function. For an effective divisor D, L (D) consists of rational functions and all its poles lies in the Supp (D) and the multiplicity of each of them is not greater than n_p

For an effective divisor D, L (D) consists of rational functions and all its poles lie in the Supp (D) and the multiplicity of each of them is not greater than n_p.
By making use of above mentioned concepts an algebraic geometric code is defined as follows. Let X be a curve, P be the points on the curve (P1, P2…..Pn) and divisor D = P1 + P2+Pn. Let L (D) denote the vector space and length of the vector is defined by Rienmann-Roch theorem [6] is given by l (D) = n+g-1 and let f1, f2….fk form basis of vector space L (D). The algebraic geometric code ((X, P, D) is the image of evaluation map

$$E: L(D) \rightarrow F_q^n$$
$$F= (f(p1) \ldots . f(pn)).$$

The code can be converted into (n, k, d) code where n is the number of points on curve, k is the dimension and d is the distance. Dimension k = degD+1-g and min distance d>n-degD. Generator matrix[5] is defined as

$$\begin{pmatrix} F_1(P_1) & \ldots & F_1(P_n) \\ & & \\ F_l(P_1) & \ldots & F_l(Pn) \end{pmatrix}$$

3. Cryptosystems using elliptic curves

In 1985 Kobliz and Miller [7] independently proposed a public key cryptosystems based on elliptic curve as an analogue of the Elgammal scheme [8] in which group Zp* is replaced by points on the elliptic curve defined over a finite field. The main attractions of elliptic curve cryptography over competing technologies such as RSA, DSA is that various algorithms are known for solving the underlying hard mathematical problems in elliptic curve cryptography .Elliptic curve discreet logarithm problem takes fully exponential time. On other hand, the best algorithm known for solving the underlying hard mathematical problem in RSA and DSA (Integer Factorization problem and DLP problem) take sub-exponential time. This means that significant parameters used in ECC is small compared to RSA and DSA but with significant equivalent levels of security.
The lack of sub exponential attack on ECC offers potential reductions in processing power, storage space, band width and electrical power. These advantages are especially important in applications on devices such as smart card, pagers, cellular phones etc. [88]
The performance of ECC depends mainly on the efficiency of finite field computations and fast algorithm for elliptic scalar multiplication. In addition to the numerous known algorithms [8] the performance of ECC can be speeded up by selecting particular underlying finite field and/ or elliptic curve

4. Algorithm for a cryptosystem using algebraic geometric code

Any cryptographic algorithm includes the following steps.

4.1 Key generation

Key generation is a process of generation of keys for the process of encryption and decryption. Security of the cryptosystem is highly dependent on keys. So we must be very careful in generating keys. Every public key cryptosystem has two keys . Public key and Private Key .

The following section describes the key generation

Input : Elliptic curve X, Fq
Output: Public key (Fq, k, X) and Private Key (α, β)

1. Compute base point B also the set the basis of linear vector space.
2. User A selects a random integer β between 0 and ord_B
3. User B selects a random integer α between 0 and ord_B
4. Public key information include (Fq, k, X) and private key (α, β)

4.2 Encryption and decryption

Encryption is the process of converting the received message into cipher text. It is done as follows. The message m is divided into m_1, m_2… m_k and is converted into a code C by making use of generator matrix [5]. A private random key is generated and is multiplied by C and sent to the receiver. That is the cipher text of message m and is sent to the receiver.

At the receiving end decryption process is done. Decryption process is the process of retrieving the original message m from the cipher text by using the private key. .By using inverse generator matrix and private key β , the fragments of message m_1, m_2…m_k can be obtained. This is in turn converted into message M. This is possible because of the linear dependent property of generator matrix.

5. Implementation

The algorithm was implemented by a program developed using Mat lab for various fields and executed in an Intel Pentium processor. The system was tested for time required for key generation, encryption and decryption. Five fields were chosen which include 13, 31, 83,127 and 167. An elliptic curve E is of form $y^2=x^3+ax+b$ and is defined over a finite field Fp and is represented as E_p (a, b).

1. q=13,a=1,b=1curveE_{13}(1,1) Number of points n=15, base point(12,8) Random key limit: 11 Key generation=0.0630 μs Encryption time=0.008667 μs Decryption time=0.0630 μs
2. q=31,a=1,b=1 CurveE_{31}(1,1) Number of points n =32:,base point(17,31): Random key limit: 31 Key generation =0. 6090 μs Encryption time=0.002 μs Decryption time=0.082 μs
3.q=83,a=1,b=1 CurveE_{83}(1,1) Number of points n =90:,base point(12,9) Random key limit: 89 Key generation =10. 14100 μs Encryption time=0.04264μs Decryption time=0.055 μs
4.q=127,a=1,b=1 CurveE_{127}(1,1) Number of points n =131, base point(18,3): Random key limit: 131 Key generation =34.1 μs Encryption time=0.06233 μs Decryption time=0.63 μs
5.q=167,a=1,b=1 CurveE_{167}(1,1) Number of points n =147, Base point (35,21) Random key limit: 144 Key generation: 62.8280 μs Encryption time=025 μs Decryption time=0.85 μs

Performance analysis can be viewed by the following graph

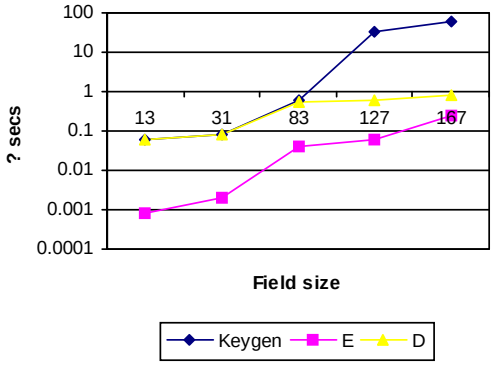

Fig 1: Graph showing time requirement for Key generation, Encryption and Decryption

From Fig 1 we can see that computation time increases as field size increase. Computational time is dependent on factors such as points of elliptic curve, scalar multiplications, point doubling and generator matrix generation. The computation time for other cryptosystem using ECC is as follows. An elliptic curve crypto system with a field size F_{127} has taken around 6.1 ms and F_{167} took 8.1 ms for key generation in an ULTRA Sparc III processor.

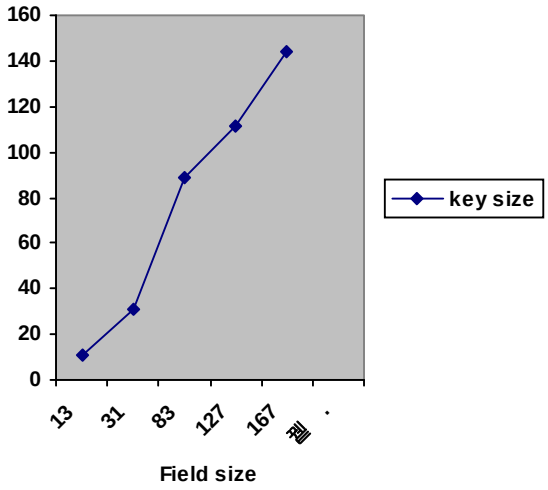

Fig 2: Graph showing Field size Vs Key size

Fig 2 shows that whenever field size increase security level goes high. System will be prone to less attacks when security increase. So it can be concluded that performance and security of a cryptosystem using algebraic geometric code can be improved by selecting a field of sufficiently large prime.

6. Conclusion

Here we have computed key generation, encryption, decryption time and key size for a curve. We can see that computational complexity increases with field size at the same time security level also increases. When ever we develop a system, field size should be a large prime. Overhead in computation can be solved by making use of processor of higher capacity.

References:

[1] R.J McEliece, A public key cryptosystem based on algebraic coding theory, DSN Progress report, Jet population laboratory, Pasadena, CA (Jan/ Feb, 1978)

[2] Niederaiter," Knapsack type cryptosystems and algebraic coding theory', Probl. Control and information theory Vol.15, pp 19-312, 1986

[3] Coding theory and cryptography the essentials: D.R Hanker son, D.Hoffman, D.A Leonard, C.C Linder, T.T Phelps, C.A Rodger, J.R Wall

[4] V.D Goppa Codes on algebraic curves, Soviet Math, Dokl, Translation Pages 207-214

[5] H.Stichenoth, "Algebraic function fields and codes", University, Springer-Verlag, Berlin, 1993

[6] J.H Van Lint," Introduction to coding theory', Grad. Texts in Math, Vol. 86, Springer – Verlag, New York - Heidelberg-Berlin, 1982

[7] V.Miller 'Use of elliptic curve in cryptography', Proceedings of crypto'85, LCNS 218, and pp 417-426, New York: Springer – Verlag 1986

[8] N. Kobliz "Elliptic curve cryptosystems", Mathematics of computation, 48, pp: 2203-209, 1987

Manju C is an Assistant Professor in the Department of Computer Science, Bharathidasan Government College for Women, Puducherry ,India. She is doing her PhD in Cochin University of Science and Technology. Her research area includes Coding theory, Cryptography and Network Security.