A Heuristic Cryptosystem Based on Bernstein Polynomial on Galois Fields GF(P) and GF(2^m)

Smitha Sasi[1], Dr. L. Swarna Jyothi[2]

[1] *Assistant Professor, Department of Telecommunication Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka*

[2] *Principal, Jnana Vikas Institute of Technology, Bangalore, Karnataka*

***Abstract*— Public key cryptosystem or the asymmetric crypto system is more secure than secret key method because a pair of related key is being used by both sender and receiver. The problem occurring in most of the cryptosystem is plain text being considered as an integer number that leads to poor security. In this paper we propose an efficient polynomial based public key cryptography technique over Galois field GF(p) and extended to GF(2^m), which considers plaint text as a (x,y) coordinate elements derived from the polynomial.**

Keywords- Bernstien polynomial, Encryption, Decryption, Public Key, Private Key

I. INTRODUCTION

Today's world faces data security problems while transmitting of data. The solution to provide confidentiality in the data transaction is encrypting the data using cryptographic algorithms. Cryptography is an effective way of protecting sensitive information as it is stored on media or transmitted through network communication path. Cryptographic algorithms can be implemented as symmetric key or asymmetric key systems. In the symmetric key crypto system same key is used for both decryption and encryption. A disadvantage of symmetric key cryptography is that the 2 parties sending messages to each other must agree to use the same private key before they start transmitting secure information. In the public key cryptosystem public key used to encrypt the data in sender side and private key is used by receiver to decrypt the data. The primary advantage of public-key cryptography is increased security and convenience, private keys never need to be transmitted or revealed to anyone. But in most of the public key algorithm like RSA the main problem is that the plain text is represented as a integer number[4]. In this paper we propose the polynomial based cryptographic method where the plain text represents points based on the polynomial[5]. The mathematical computation is effective in Bernstein polynomial method. So this paper proposes Bernstein polynomial cryptographic technique.

II. BERNSTEIN POLYNOMIAL

In the mathematical field of numerical analysis, a Bernstein polynomial, named after Sergei Natanovich Bernstein, is a polynomial in the Bernstein form, that is a linear combinationof Bernstein basis polynomials[1,2]. A numerically stable way to evaluate polynomials in Bernstein form is de Casteljau's algorithm.

The (n + 1) Bernstein basis polynomials of degree n are defined as $n(f,t)=\sum_{r=0}^{n} f\left(\frac{i}{n}\right) nc_i\, t^i (1-t)^{n-i}$

Where nc_i is a binomial coefficient. The Bernstein basis polynomials of degree n form a basis for the vector space Пn of polynomials of degree at most n.

The coefficient nc_i can be obtained from pascal's triangle; the exponent on the (1-t) term decrease by one as i increases.

- The Bernstein polynomials of degree 1 are

  $B_{0,1}(t) = 1-t$ $B_{1,1}(t) = 1-t$ can be plotted for $0<t<1$ as shown in fig 1

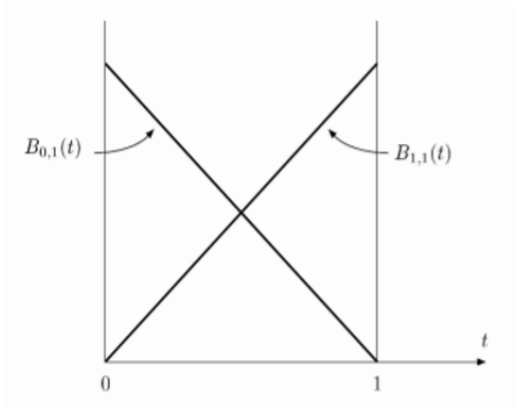

Fig 1 linear bernstien polynomial

- Bernstein polynomials degree 2 are

  $B_{0,2}(t) = (1-t)^2$

  $B_{1,2}(t) = 2t(1-t)$ $B_{2,2}(t) = t^2$ can be plotted for $0<t<1$ as shown in fig 2

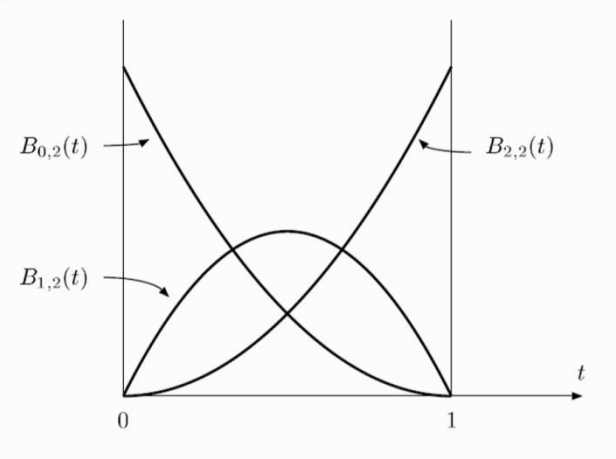

Fig 2 quadratic Bernstein polynomial on

- Bernstein polynomials degree 3 are
  $B_{0,3}(t) = (1-t)^3$
  $B_{1,3}(t) = 3t(1-t)^2$
  $B_{2,3}(t) = 3t^2(1-t)$ $B_{3,3}(t) = t_3$ can be

plotted for 0<t<1 as shown in fig 3

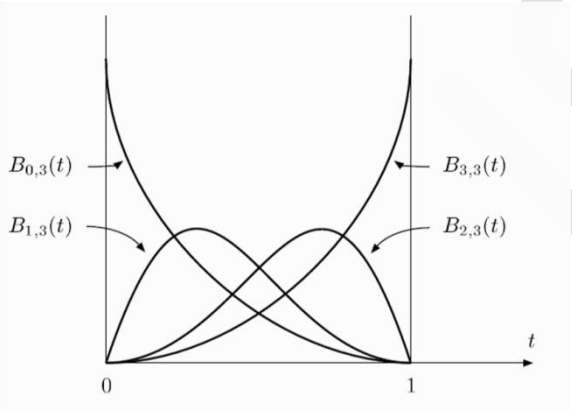

Fig 3 Ternary Bernstein polynomial

A. *Degree Raising*

Any of the lower-degree Bernstein polynomials (degree < n) can be expressed as a linear combination of Bernstein polynomials of degree n. In particular, any Bernstein polynomial of degree n−1 can be written as a linear combination of Bernstein polynomials of degree n.

$$tB_{i,n}(t) = nc_it_{i+1}(1-t)_{n-i}$$
$$= nc_it_{i+1}(1-t)_{(n+1)-(i+1)}$$
$$=(nc_i)/(n+1\ c_{i+1})\ B_{i+1,,n+1}(t)$$
$$=i+1/n+1\ B_{i+1,,n+1}(t)$$
$$(1-t)B_{i,n}(t) = nc_it_i(1-t)_{n-+1-i}$$
$$=(nc_i)/(n+1\ c_i)\ B_{i,,n+1}(t)$$
$$=n-i+1/n+1\ B_{i,,n+1}(t)$$
$$1/(nc_i)\ B_{i,n}(t)+1/(n+1\ c_i)\ B_{i+1,n}(t)$$
$$=t_i(1-t)_{n—i}+\ t_{i+1}(1-t)_{n—(i+1)}$$
$$= t^i(1-t)^{n—i-1}((1-t)+t)$$
$$= t^i(1-t)^{n—i-1}$$
$$=1/(n-1c_i)(B_{i,,n-1}(t))$$

III. GALOISFIELD

In abstract algebra a finite field or Galois field (so named in honor of Évariste Galois) is a field that contains a finite number of elements. Finite fields are important in number theory, algebraic geometry, Galois theory, cryptography, coding theory and quantum error correction[6,7]. The order of a finite field is always a prime or power of a prime. Cryptography focuses on finite fields. It turns out that for any prime integer p and any integer n greater than or equal to 1, there is a unique field with pn elements in it, denoted GF(pm).In case n is equal to 1, the field is just the integers mod p, In cryptography, one almost always takes p to be 2 in this case called binary extension and represented as GF(2m). Let α €$GF(p^m)$ be the root of a primitive polynomial of degree m over GF(p). The polynomial basis of $GF(p^m)$ is then
$\{1,\alpha,\ldots..,\alpha^{m-1}\}$

A. Addition in GF(2^m)

In $GF(2^m)$, addition is especially easy, since addition and subtraction modulo 2 are the same thing, and furthermore this operation can be done in hardware using the basic XOR logic gate. For example, using the Galois Field $GF(2^3) = GF(8)$ based on the primitive $P(x) = x^3 + x + 1 = (1011) = 11$ (decimal) .values in $GF(2^3)$ are 3-bits each, spanning the decimal range [0..7]. Addition takes place on these 3bit binary values using bitwise XOR. 5 + 5 = (101) + (101) = (000) = 0 .The addition table for $GF(2^3)$ as shown in Table1

Table 1 Galois field addition table $GF(2^3)$

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	0	3	2	5	4	7	6
2	2	3	0	1	6	7	4	5
3	3	2	1	0	7	6	5	4
4	4	5	6	7	0	1	2	3
5	5	4	7	6	1	0	3	2
6	6	7	4	5	2	3	0	1
7	7	6	5	4	3	2	1	0

B. Multiplication

In most algorithms the modular product is computed in two steps: polynomial multiplication followed by modular reduction. Let A(x), B(x) €GF(2^m) and P(x) be the irreducible field generator polynomial. Eg: GF(2^3),P(x)=x^3+x+1, A(x)=x + 1,B(x) =x + 1 Polynomial multiplication:

A(x)* B(x) =(x+1)*(x+1)=x^2+1 Modular reduction X^2 + 1 mod(x^3+x+1)=x^2+1 Using the Galois Field GF(2^3) = GF(8) based on P(x)=x^3+x+1=(1011)=11 Multiplication table for GF(2^3) as shown in Table 2.

X	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7
2	2	4	6	3	1	7	5
3	3	6	5	7	4	1	2
4	4	3	7	6	2	5	1
5	5	1	4	2	7	3	6
6	6	7	1	5	3	2	4
7	7	5	2	1	6	4	3

Table 2 Galois field multiplication table GF(2^3)

values in GF(2^3) are 3-bits each, spanning the decimal range [0..7]. Multiplication takes place on 3-bit binary values (with modulo 2 addition) and then the result is computed modulo P(x) = (1011) = 11 (decimal).
For example: 4 × 4 = (100) × (100) = (10000) = (110) mod (1011) = 6 For exponential multiplication
5^7=(5 x 5 x 5 x 5 x 5 x5 x 5)mod p(x) =(7 x 7 x 7 x 5) =(3 x 7 x 5) =(2 x 5)=1

IV. POLYNOMIAL BASED CRYPTOGRAPHIC ALGORITHM

The general form of the Bernstein polynomial is

$$n(f,t)=\sum_{r=0}^{n} f\left(\frac{i}{n}\right) nc_i\, t^i (1-t)^{n-i}$$

When n=5; this polynomial is Quintic polynomial. In mathematics, a quintic function is a function of the form $g(x) = ax^5 + bx^4 + cx^3 + dx^2 + ex + f,$
where a, b, c, d, e and f are members of a field, typically the rational numbers, the real numbers or the complex numbers, and a is nonzero. In other words, a quintic function is defined by a polynomial of degree five.
Generate (x,y) points based ion the degree of the polynomial.map those points on to text.
Encryption: Step 1:

Choose (K_u,k_r) and (α_1, α_2) are the pairs of the keys Where K_u is the public key and k_r is the private key.

Step 2:

Choose the secret reference point on the curve based on Quintic polynomial

Step3:

Choose the plain text point (P_x,P_Y) on the curve based on the Quintic polynomial Step 4:

For GF(p): Perform (P_x,P_Y) /(α1, K_u) mod(n^2n+41)=(a,b) this is the another point on the curve.(perform point division operation).
For GF(2^m):Perform (P_x,P_Y) /(α1, K_u) mod GF(2^m)=(a,b) this is the another point on the curve.(perform point division operation) Step5:

For GF(p):Perform (a,b)/secret reference point mod(n^2-n+41)=(Cx, C_Y) this is the cipher text.

For GF(2^m): Perform (a,b)/secret reference point mod GF(2^m)=(Cx, C_Y) this is the cipher text.

Cipher text will be the point on the curve. So the final resultant curve which contains all the points of cipher text is transmitted to the receiver. Decryption:

On receipt of the cipher text points, the receiver starts to perform decryption.
Step 1:

For GF(p): Perform (Cx,Cy) (secret reference point) mod(n^2-n+41)=(a_1,b_1)
Perform (Cx,Cy) (secret reference point)mod GF(2^m)=(a_1,b_1)
(secret reference point will be exchanged between two parties by using any secure key exchange algorithm) Step 2:

For GF(p): Perform (a_1,b_1)/(α_2, K_r) mod (n^2-n+41)= (P_X,P_Y) this is the plain text value,if receiver use proper private key.
For GF(2^m): Perform (a_1,b_1)/(α_2, K_r) mod GF(2^m)=(P_X,P_Y) this is the plain text value,if receiver use proper private key.
Relation between K_u and K_r is

K_r=n[(b-nK_u)/(1-n) +b(1-n)/n] mod GF(2^m).

The relation between α_1, α_2 is α_2=n[a-nα_1)/(1-n)+a(1-n)/n)] mod GF(2^m)

V. RESULT

A. Cryptographic method by using GF(p) Encryption:

(P_x,P_y)=(4,43); is the plain text

(α_1, K_u)=(7,3) Public keys

(a,b)=(P_x,P_y)/(α_1,K_u)mod(n^2 -n+41)

=(4,43)/(7,3)mod(n^2-n+41)

(a,b)=(19,26)

(a,b)/secret reference point=(C_X,C_Y) Where

(C_X,C_Y) cipher text.

Secret reference point=(2,7) is exchanged between both the parties by using some security exchange algorithm.

((19,26)/(2,7)) mod(n^2-n+41)=(56,53) Decryption:

(C_X,C_Y)(secret reference point)mod(n^2- n+41)=(a1,b 1)

(56,53) (2,7) mod(n2-n+41)=(19,26)

(a1,b1)/(α_2, K_r) mod(n^2-n+41)= (P_X,P_Y) From the relation ;

K_r=n[(b-nK_u)/(1-n) +b(1-n)/n] mod n^2-n+41 α2=n[a-nα_1)/(1-n)+a(1-n)/n)] mod n^2 -n+41.

(α2,K_r)=(5,50)

(19,26)/(5,50) mod(n^2-n+41)=(4,43)

B.Cryptographic method by using GF(2 ^{m}) Encryption:

(P_x,P_y)=(5,3); is the plain text

(α_1, K_u)=(7,3)

GF(2^m)= GF(2^3)

(a,b)=(P_x,P_y)/(α_1,K_u)mod GF(2^3)

=(5,3)/(7,3)mod GF(2^3)

(a,b)=(4,3)

(a,b)/secret reference point=(C_X,C_Y)

Secret reference point=(2,7) ((4,3)/(7,2))

mod GF(2^3)=(7,1)) Decryption:

(C_X,C_Y)(secret reference point)mod(GF (2 ^{m})=(a1 ,b 1)

(7,1) (2,7) mod GF(2^3)=(4,3)

(a1,b1)/(α_2, K_r) mod GF (2 ^{m})=((P_X,P_Y)

From the relation ;

K_r=n[(b-nK_u) /(1-n) +b(1-n)/n] mod GF (2 ^{m}) α_2 =n[a-nα_1)/(1-n)+a(1-n)/n)] mod GF(2^m)

(α_2 ,K_r)=(1,3)

((4,3)/(1,3)) mod GF(2^3)=(5,3)

Consider the Plaintext=(2,23)(3,2)(4,4)(5,83)(6,67) from the eighth order Bernstein polynomial (n=8(octic curve)e=3, alpha1=7).

The Plain text points on (x,y) coordinates for GF(p) is shown in Fig4.

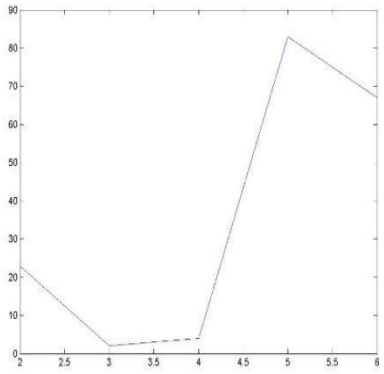

Fig. 4 plain text points on (x, y) coordinates

The corresponding cipher text is

(44,17)(93,55)(45,66)(94,47)(46,39) is plotted on (x,y) Coordinates for GF(p) is as shown in Fig 5.

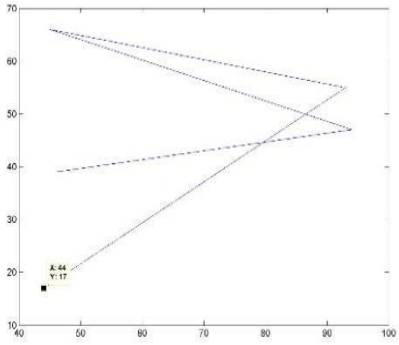

Fig. 5 Cipher text points on (x, y) coordinates

VI CONCLUSION

Bernstein polynomial cryptographic approach provides security and reduces computational complexity. The algorithm is implemented using MATLAB. Results are verified for the Bernstein polynomial up to a degree of n=10(Decic polynomial) and galois field GF(2^6). Encryption and decryption done successfully and since the algorithm involves encrypted data and key in the form of (x,y) coordinates improves the strength of the security.

REFERENCES

[1] H. Caglar and A. N. Akansu, *"A Generalized Parametric PRQMF Design Technique Based on Bernstein Polynomial Approximation*," IEEE Transactions on Signal Processing, vol. 41, no. 7, pp. 2314–2321, July 1993.

[2] Online geometric modeling notes: *Bernstein; Visualization and graphics research group; department of computer science*, University of California

[3] http://mathworld.wolfram.com

[4] William Stallings, "*Cryptographyand Network Security*", Principles and Practices, 3rd Edition, Prentice Hall 2003.

[5] *Interpolation and approximation of polynomials* by Philips, G.M.ISBN:978-0-387-00215-6 http://www.springer.com/978-0-387-00215-6;

[6] The Encyclopedia of design theory: Galois fields by Peter J. Cameron May 30, 2003

[7] Error Control Coding by Shu Lin, Daniel J Costello; 2nd edition.