Jacobi 기호에 의한 암호화 프로토콜

오 정환 , 김 칠 김 용대
연세대학교 광운대학교 연세대학교

Cryptographic Protocol Using Jacobi Symbol.

Oh Jung-Whan Kim Chul Kim Yong-Dae
Yonsei Univ. Kwangwun Univ. Yonsei Univ.

요약:본 논문은 정수론에 있어서 Legendre 기호의 응용인 Jacobi 기호를 이용한 암호화 프로토콜의 정의,성질 및 확장에 대해 논한다. 본 논문은 먼저 Legendre 기호와 Jacobi 기호의 정의와 성질에 대하여 살펴보고 이를 이용한 프로토콜에 대한 설명을 하며 그 알고리즘을 구축하여 다른 프로토콜과의 비교를 통하여 이 프로토콜의 안전성과 신속성을 고찰한다.

1.서론

정수론은 암호학의 시작,발전과 많은 관련을 가지고있을 뿐만 아니라 현대 암호학에도 많은 영향을 끼치고있다. 본 논제에서는 Legendre 기호를 응용한 Jacobi 기호를 사용하여 만든 암호화 프로토콜에 대해서 언급하겠다. 2절은 Legendre 기호와 Jacobi 기호의 정의와 몇 가지 성질에 대해 언급하였고 3절은 Jacobi 기호를 이용한 암호화 프로토콜의 정의들을 설명하고 다른 프로토콜과 비교했을 때 생기는 이 프로토콜의 안전성 및 신속성을 논한다. 2절의 수학적인증명은 참고문헌의 제시로 대신한다.

2. Legendre 기호와 Jacobi 기호.

Legedre 기호는 $x^2 \equiv a \pmod p$ ------ (1)의 정수근을 찾는 법 p의 이차 잉여류(qudratic residue modulo p)의 문제에서 시작된다. 이때 p는 임의의 소수이고 a는 임의의 정수이다. 만약 $a \equiv 0 \pmod p$ 이면 (1)의 근은 $x \equiv 0 \pmod p$ 밖에 없다. 따라서 $p \nmid a$라고 가정하더라도 일반성을 잃지않는다.

정의 2.1. p가 소수, a가 정수이고 $p \nmid a$라고 가정하자. a가 법 p의 이차 잉여류라는 것은 (1)이 정수근을 갖는다는 것이다. (1)이 근을 갖지않는 경우에는 a가 법 p의 이차 비잉여류라고 한다.

보조정리 2.2. $p > 2$ 인 소수, a가 법 p의 이차잉여류, 그리고 $p \nmid a$라고 하자. 그러면 식 (1)은 정확히 두 개의 근을 갖는다.

간단한 계산에 의해서 (2) 중의 어느 두 수도 법 p에 대해 합동이 아님을 알 수 있다.

정리 2.3. $p > 2$인 소수이고 $p \nmid a$라고 하자. 그러면 a가 법 p의 이차 잉여류이기 위한 필요충분조건은 a가 (2) 중의 하나와 합동인 것이다. 그리고 (2) 중의 어느 두 개도 법 p에 대해 합동이 아니다. 따라서 $0, 1, 2, \cdots, p-1$ 중 정확히 $\frac{p-1}{2}$ 개가 법 p의 이차 잉여류이고 $\frac{p-1}{2}$ 개가 법 p의 이차 비잉여류임을 알 수 있다.

이제 Legendre 기호를 정의해보자. Legendre 기호에 관한 정리의 증명은 [Ad]에 나와있다.

정의 2.4. p를 홀수인 소수라 하고 a는 p를 나누지 못하는 임의의 정수라 하자. 이때 Legendre 기호 $(\frac{a}{p})$는 다음과 같이 정의된다.

$$(\frac{a}{p}) = \begin{cases} +1 & : a\text{가 법 } p\text{의 이차 잉여류일 때} \\ -1 & : a\text{가 법 } p\text{의 이차 비잉여류일 때} \end{cases}$$

그러면 다음과 같은 정리들이 나온다.

정리 2.5. p는 홀수인 소수라 하고 a,b는 p를 나누지 못하는 임의의 정수라 하자. 그러면 다음이 성립한다.

(i) $(\frac{a^2}{p}) = 1$

(ii) $(\frac{1}{p}) = 1$

(iii) If $a \equiv b \pmod p$, then $(\frac{a}{p}) = (\frac{b}{p})$.

정리 2.6. (Euler's criterion): $p>2$인 소수라고 하고 a는 p를 나누지 못하는 임의의 정수라고 하자. 그러면 $(\frac{a}{p}) \equiv a^{\frac{p-1}{2}} \pmod p$ 이 성립한다.

계 2.7. p는 홀수인 소수라하고 a,b는 p를 나누지 못하는 임의의 정수라 할 때, $(\frac{ab}{p}) = (\frac{a}{p})(\frac{b}{p})$ 가 성립한다.

계 2.8. p는 홀수인 소수라고 하자. 그러면 $(\frac{-1}{p}) = (-1)^{\frac{p-1}{2}}$ 이 성립한다.

예 2.9. $(\frac{31}{47}) = (\frac{-16}{47}) = (\frac{-1}{47})(\frac{4}{47})^2 = -1$이므로 $x^2 \equiv 31 \pmod{47}$의 해는 존재하지 않는다.

위의 예에서 볼수 있듯이 $a = \pm p_1^{a_1} p_2^{a_2} \cdots p_t^{a_t}$이고 $p_1, p_2, \cdots, p_t$가 서로 다른 소수이고 $p \nmid a$라 하자. 그러면 a가 법 p에 대해 이차 잉여류인가 $(\frac{a}{p}) = (\frac{-1}{p})(\frac{p_1}{p})^{a_1}(\frac{p_2}{p})^{a_2} \cdots (\frac{p_t}{p})^{a_t}$를 계산함으로써 알 수있다.

다음 정리는 [Ad]의 Gauss' theorem에 의해 나온다.

정리 2.11. p를 2가 아닌 소수라 하자. 그러면 $(\frac{2}{p}) = (-1)^{\frac{p^2-1}{8}}$ 이 성립한다.

실제적인 Legendre 기호의 계산은 주로 다음 정리에 의해 얻어진다.

정리 2.12.(이차 잉여류의 상호 법칙, Law of Quadratic Reciprocity) p와 q를 홀수인 서로 다른 소수라 하자. 그러면 $(\frac{q}{p})(\frac{p}{q}) = (-1)^{\frac{p-1}{2}\frac{q-1}{2}}$ 이 성립한다.

예 2.13. $(\frac{23}{59}) = (\frac{26}{59}) = (\frac{2}{59})(\frac{13}{59}) = -(\frac{59}{13}) = -(\frac{7}{13}) = -(\frac{13}{7}) = -(\frac{-1}{7}) = 1$ 라는 계산에 의해서 $x^2 \equiv 23 \pmod{59}$의 정수근이 존재함을 알 수 있다.

다음에는 Jacibi 기호를 정의해보자. Jacobi 기호에 대한 여러가지 정리의 증명은 [Ke]에 있다.

정의 2.14. n이 1보다 큰 홀수이고 $\gcd(a, n) = 1$ 일 때 n은 소수들의 곱인 $p_1 p_2 \cdots p_t$라고 하자. 이때 Jacobi 기호 $(\frac{a}{n})$는 다음과 같이 정의 된다.

$(\frac{a}{n}) = (\frac{a}{p_1})(\frac{a}{p_2}) \cdots (\frac{a}{p_t})$

정의에 의하여 다음과 같은 Legendre 기호와 유사한 정리들을 얻을 수 있다.

정리 2.15. a와 n을 위에서와 같이 정의하자.그러면 다음 식이 성립한다.

(i) $a \equiv a' \pmod{n} \Rightarrow (\frac{a}{n}) = (\frac{a'}{n})$

(ii) $(\frac{aa'}{n}) = (\frac{a}{n})(\frac{a'}{n})$

(iii) $(\frac{a}{n})(\frac{a}{m}) = (\frac{a}{nm})$

(iv) $(\frac{-1}{n}) = (-1)^{\frac{n-1}{2}}$

(v) $(\frac{2}{n}) = (-1)^{\frac{n^2-1}{8}}$

(vi) (Reciprocity Law) 만약 m과 n이 홀수이고 서로소이면 $(\frac{n}{m})(\frac{m}{n}) = (-1)^{\frac{n-1}{2}\frac{m-1}{2}}$ 이 성립한다.

정리 2.16. $x^2 \equiv a \pmod{n}$의 근이 존재하면 $(\frac{a}{n}) = 1$ 이다.

정리 2.15와 2.16의 증명은 Legendre 기호의 성질, Jacobi 기호의 정의와 중국인의 나머지 정리를 이용하면 쉽게 구할수 있다.

*그러나 일반적으로 정리2.16의 역방향은 성립하지 않는다. 다음 예는 역방향이 성립하지 않음을 보여준다.

예 2.17. $(\frac{2}{15}) = 1$임을 우리는 위의 공식들을 쓰면 쉽게 알 수 있다. 그렇지만 $x^2 \equiv 2 \pmod 5$와 $x^2 \equiv 2 \pmod 3$의 근이 존재하지 않으므로 $x^2 \equiv 2 \pmod{15}$의 근도 존재하지 않는다.

3. Jacobi 기호를 이용한 암호화 프로토콜

많은 프로토콜이 전화를 이용한 동전 던지기(coin flipping by telephone)를 기본 구조로 하여 만들어졌다. 동전 던지기는 1982년 Manuel Blum이 처음 제안했으며 현재에도 많이 쓰여지고 있다. 일반적인 유형은 다음과 같다. 먼저 A와 B가 일방함수(one - way funcion) f 를 알지만 역함수 f^{-1}는 알지 못한다고 하자. 다음에 B가 임의의 수 x를 택하고 A에게 $f(x)$를 알려준다. 그러면 A는 50 대 50의 확률로 x를 예상한다. B는 A에게 그 예측이 맞았는가 그렇지 않은가 알려준다. 다음에 A는 B에게 x를 알려준다. 이제부터 언급해 나가려는 Jacobi 기호를 이용한 프로토콜도 동전 던지기의 한 유형이라 말할 수 있다. 특히 $(\frac{a}{n})=1$ 일 때 a가 이차 잉여류냐 아니냐를 결정하는 문제는 50%의 확률을 가지고 있어 A의 예상하는 과정이 유사하다.

1)단순한 암호화 프로토콜

제 2절에서 $(\frac{23}{57})$을 계산하는 과정이 보여주듯이 $(\frac{a}{p})$를 계산하는 데 걸리는 시간은 $O(\log^2 p)$ 이다. Jacobi 기호를 계산하는 알고리즘은 [Sh]에 있다. 제 2절의 마지막 부분에 있는 (*)에서 보듯이 $0 < a < n$ 이고 $\gcd(n, a) = 1$일 때 $(\frac{a}{n}) = 1$이더라도 a가 법 n의 이차 잉여류일 수도 있고 그렇지 않을 수도 있다. 따라서 a가 법 n의 이차 잉여류인가 를 결정하는 유일한 방법은 n을 인수분해하는 것이다. 그래서 $n = p_1 p_2 \cdots p_t$가 되었을 때 $(\frac{a}{p_i})$가 모든 i에 대해 1이 되는가를 확인하면 된다. 이것이 이제부터 설명하려는 프로토콜의 가장 중요한 모티브이다. 프로토콜은 다음과 같이 진행된다.

(1)B가 큰 소수 p와 q를 택하여 그 곱 n을 구하고 $(\frac{a}{n}) = 1$인 임의의 정수 a를 택하여 A에게 n과 a를 알려준다.

이 때 A가 n과 a를 알더라도 a가 법 p의 이차 잉여류인지 아닌지는 알 수가 없다.

(2)다음에 A는 a가 법 p의 이차 잉여류인가 아닌가를 추측하여 B에게 알려준다.

(3)B는 그 추측이 옳은지 그른지를 A에게 알려준다.

(4)후에 B는 A에게 p와 q를 알려준다.

(5)A가 p와 q를 받으면 A는 그것이 소수인지를 반드시 확인해야 한다. 왜냐하면 B는 A를 다음과 같이 속일 수 있기 때문이다.

**먼저 B는 다음을 만족하는 세 개의 소수 p_1, p_2, q_1과 정수 a를 택한다.

$(\frac{a}{p_1}) = (\frac{a}{p_2}) = -1$, $(\frac{a}{q_1}) = 1$

그리고 A의 옳은 추측을 H(head), 그른 추측을 T(tail)이라고 하자.

B가 H라고 보내기를 원하면 다음과 같이 진행한다. A가 잉여류라고 대답하면 B는 A에게

$p = p_1p_2$, $q = q_1$을보내준다. A가 비잉여류라고 대답하는 경우는 $p = p_1$, $q = p_2\, q_1$을 보내준다.
B가 T라고 보내기를 원하면 다음과 같이 진행한다. A가 잉여류라고 대답하면 B는 $p = p_1$, $q = p_2\, q_1$을 보내준다. A가 비잉여류라고 대답하는 경우는 $p = p_1\, p_2$, $q = q_1$을 보내준다.

위의 프로토콜은 [통신]을 참조하였다. 예 3.1은 이 프로토콜의 예이고 예3.2는 (5)에서 소수인가 아닌가를 확인하지 않았을 때 B가 속는 예이다.

예 3.1. B가 두 소수 p, q를 택하여 p를 479891, q를 479939, a를 414977로 택하였다고 하자. 그러면 n은 230318406649가 되고 p, q는 법 a의 이차 잉여류가 된다. B가 n, a를 A에게 보내면 A는 a가 n의 이차 잉여류인가 아닌가 모르므로 단지 추측만 하여 B에게 보내준다. A가 이차 잉여류라고 보내주면 B는 맞다는 회신을 보내고 나중에 p와 q를 A에게 보내준다.그러면 A는 p와 q가 소수인가 확인해야 한다. 왜냐하면 B는 A를 예3.2와 같이 속일 수 있기때문이다.

예3.2. B가 A를 속이기 위해 p_1을 479909, p_2를 541, q_1을 479939, a를 414977로 택하면 그들은 다음과 같은 관계를 만족한다: $(\frac{a}{p_1}) = (\frac{a}{p_2}) = -1$, $(\frac{a}{q_1}) = 1$
그리고 세 수의 곱인 230318406649를 n으로 하여 A에게 n과 a를 보낸다. 이 때 A는 a가 법 n의 이차 잉여류인가 아닌가 추측하여 B에게 그 결과를 보낸다. B는 A의 결과를 가지고 다음과 같이 조작할 수 있다.
먼저 옳은 추측이라고 보내고 싶다고 가정하자. A가 이차 잉여류라고 대답하면 B는 A에게 p_1과 p_2의 곱인 259630769를 p로 가장하여 보내고 q_1인 479939를 q로 보낸다. 그러면 A는 $(\frac{a}{p}) = (\frac{a}{q}) = 1$ 이라는 것을 알 수 있다. 이때 A가 B가 보낸 p, q를 그대로 믿으면 A는 B의 속임수에 넘어가게 된다. 그렇지만 A는 p, q가 소수인지 아닌 지 확인하고 그것으로 B의 정보가 거짓임을 확인할 수 있다.
또 A가 이차 비잉여류라고 보내오면 B는 p를 p_1인 479909로, q를 p_2와 q_1의 곱인 259646999로 하여 보낸다. 그러면 위에서와 같이 A는 $(\frac{a}{p}) = (\frac{a}{q}) = -1$라는 것을 알 수 있고 B에게 속게 된다. A는 p, q 가 소수인지 아닌지를 확인해서 B의 속임수를 방지한다.
틀린 추측이라고 보내고 싶으면 B는 위에서 잡은 p, q를 반대로 잡아 A를 속일 수 있으며 A는 속지 않기 위해 p, q 가 소수인지를 확인해야 한다.

2) 확장된 암호화 프로토콜
다음에 또 하나의 프로토콜을 생각하기 위해 다음과 같은 계산을 해보자. 두 개의 큰 소수 p, q를 잡고 n을 두 수의 곱이라 하고 (1, n / 2) 사이의 정수 a를 생각하자. 이때 다음과 같은 방정식을 생각해 보자. $x^2 \equiv k \pmod n$, $k \equiv a^2 \pmod n$ 이고 k는 0과 n사이의 정수라 하면 이 방정식의 근을 찾는 문제는 n을 인수분해하는 것과 같은 복잡도를 가진다. 만약 n을 인수분해하여 p , q를 알게 되면 $x^2 \equiv k \pmod p$와 $x^2 \equiv k \pmod q$의 근을 알면 중국인의 나머지 정리를 써서 $x^2 \equiv k \pmod n$, $k \equiv a^2 \pmod n$의 네 개의 근인 $\pm c \pmod n$, $\pm d \pmod n$ 을 구할 수 있다. 이런 사실을 이용하여 다음과 같은 것을 생각할 수 있다. 만약 $s_1, s_2, \cdots, s_t$가 A의

비밀이고 모든 s_j는 이진수열이라 하자. 이제 A는 자신이 가지고 있는 비밀 중 일부만을 원하는 사람에게 팔려고한다. 예를 들어 s_i를 B라는 사람에게 팔려고 한다고 하자. 그러면 다음과 같은 RSA와 Jacobi 기호를 이용한 다음과 같은 프로토콜을 생각할 수 있다.

먼저 각 비밀이 RSA를 사용해서 암호화 되고 각 비밀들의 RSA 암호체계가 서로 다르다고 가정하자. 즉 각 s_j 에 대해 p_j, q_j와 그 곱 n_j를 생각하자. 그러면 각 비밀의 암호문은 n_j를 인수분해함으로서 쉽게 풀 수 있다. 그리고 A는 먼저 각 비밀들의 목록을 공개한다.

(1) 먼저 A는 k개의 p_j, q_j(여기서 p_j, q_j는 법 4에 대해 3과 합동이다.)를 가지고 k개의 RSA 암호체계를 만든다. 여기서 $x^2 \equiv k \pmod{n_j}$의 두 개의 서로 다른 근은 Jacobi 기호가 다르다. 그리고 A는 B에게 암호화 키 (e_j, n_j) 와 $s_j^{e_j} \pmod{n_j}$를 j는 1부터 k까지 보내자. 물론 numerical encoding과 block division은 사전에 약속되어 있었다고 가정하자.

(2) B는 $x_1, x_2, \cdots, x_k$를 택하고 Jacobi 기호 $(\frac{x_j}{n_j})$와 $x_j^2 \pmod{n_j}$ $j = 1, 2, \cdots, k$ 를 계산한다. 그리고 A에게 $j \neq i$ 인 경우에 $x_j^2 \pmod{n_j}$와 $(\frac{x_j}{n_j})$를 보낸다. $j = i$ 인 경우는 $x_i^2 \pmod{n_i}$, $-(\frac{x_i}{n_i})$를 보낸다.

(3) A는 모든 j에 대해 제곱근을 구한 후 각 Jacobi기호와 대응되는 근을 B에게 보낸다.

(4) B는 $x_i^2 \pmod{n_i}$의 두개의 서로 다른 근을 가졌기 때문에 n_i를 인수분해할 수 있어서 복호화 키 d_i를 구할 수 있고 알고자 하는 비밀 s_i를 알게 된다. $j \neq i$인 경우에 B는 아무것도 알 수 없다는 것은 쉽게 알 수 있고 이 프로토콜은 끝난다.

확장된 암호화 프로토콜은 [Sa]를 참조하였다.

위의 두 개의 프로토콜은 oblivious transfer의 예로 보내는 사람은 보내는 사람은 누가 비밀을 가졌는가 그렇지 않은가 확인 할 수 없다. 그러나 B는 누구로부터 받는 지 알 수 있다. 그리고 이 oblivious transfer 가 성공할 확률은 50%이다. 다음은 Mathematica를 이용해서 계산한 확장된 프로토콜의 한 예이다.

참고문헌

[통신] 오정환,김철, Legendre 기호와 암호학,통신정보보호학회지 제2권 제2부,pp.25-30,1992.

[Ad] W.Adams, and J.Goldstein, *Introduction to Number Theory*, Prentice-Hall, 1976.

[Hu] H.L.Keng, *Introduction to Number Theory*, Springer-Verlag, 1982.

[Sa] A.Salomaa, *Public - Key Cryptography*, Springer-Verlag,1990.

[Sh] J.Shallit, *On the Worst Case of Three Algorithms of Computing the Jacobi Symbol*, J. of Symbolic Compution, to appear

```
(* Example3.3.Assume that A wants to transfer obviously*)
(* the factorization of n=59989=239*251                 *)
(* Among secrets si,we may assume B wants to buy s1     *)
(*Start of step 1*)
p1:=Prime[52]
(*Take prime integer pi which is 3 mod 4*)

p1

239

q1:=Prime[54]          (*Take other prime qi 3 mod 4*)

q1

251

n1:=p1*q1              (*Find pi*qi=ni *)

n1

59989

s:=Prime[200]*Prime[200]      (*Take secret s*)

s

1495729

s1:=BaseForm[s,2]         (*Find the value of s base 2*)

s1

10110110100101011000 1
                     2

phin:=(p1-1)*(q1-1) (*Calculate Euler phi function of n*)

phin

59500

FactorInteger[phin]

{{2, 2}, {5, 3}, {7, 1}, {17, 1}}

e1:=Prime[9]  (*Take encryption key e1 relatively *)
              (* prime to phi n                    *)
```

```
e1

23

d1:=PowerMod[e1,-1,phin]
 (*Since e1 is relatively prime to phi n1, we can*)
 (* find the inverse d1 modulo n1                  *)

d1        (* End of step 1 *)

2587

x1:=PowerMod[s,e1,n1]  (*Start of step 2*)
                       (*Calculate s to the e1 mod n1*)

x1

19044

JacobiSymbol[x1,n1]   (*Caculate Jacobi symbol of*)
                      (*  x1 over n1 *)

1

xp:=PowerMod[x1,2,n1] (*Calculate x1 to the 2 modulo n1*)

xp    (* A tells B xp,Jacobi symbol of x1 over n1.*)
      (* End of step 2*)

40431

pmo:=Mod[xp,p1] (*Start of step 3,to find the solution*)
                (*calculate xp modulo p1*)

pmo

40

qmo:=Mod[xp,q1] (*Calculate xp modulo q1*)

qmo

20

(*Solve the eqations modulo p1 and q1*)
Solve[x^2-pmo==0 && Modulus==p1,x]

{{Modulus -> 239, x -> -163}, {Modulus -> 239, x -> -76}}
```

```
Solve[x^2-qmo==0 && Modulus==q1,x]

{{Modulus -> 251, x -> -32}, {Modulus -> 251, x -> -219}}

px:=76

qx:=32

(*In order to use Chinese Remainder Theorem,find*)
(*the inverse of p1 (mod q1) and q1 (mod p1)    *)
mq:=PowerMod[q1,-1,p1]

mq

20

mp:=PowerMod[p1,-1,q1]

mp

230

(*By Chinese Remainder Theorem,calculate 4 square roots*)
r[1]:=q1*px*mq+p1*qx*mp

r[2]:=-q1*px*mq+p1*qx*mp

r[3]:=q1*px*mq-p1*qx*mp

r[4]:=-q1*px*mq-p1*qx*mp

l[1]:=Mod[r[1],n1]
l[2]:=Mod[r[2],n1]
l[3]:=Mod[r[3],n1]
l[4]:=Mod[r[4],n1]

Table[l[i],{i,4}]  (*The solutions*)

{40945, 57762, 2227, 19044}

Table[PowerMod[l[i],2,n1],{i,4}]
 (*Certify the l[i]'s are solution of the equatin*)

{40431, 40431, 40431, 40431}

(*If B originally had 40945,he gets in step2 decisive*)
(*new information if A returns 57762,2227,whereas B  *)
(*gets nothing new if A returns 40945,19044          *)
(*If B wants to buy s,he tells A the pair (xp,-1) and*)
(*gets back either 57762 or 2227 in step3.Since      *)
(*p1 divides 57762+40945 and 40945-2227, B is able to*)
(*factor n1*)
                (*END OF EXAMPLE*)
```