通信情報保護學會論文誌
第 7 卷 第 4 號, 1997. 12

유한체 위에서 다항식의 근에 관한 알고리즘

김 창 한*, 서 광 석**, 이 옥 연***

A root finding algorithm of a polynomial over finite fields

Changhan Kim, Kwangsuk Suh, Okyeon Yi

요 약

유한체 위에서 다항식의 근을 구하는 문제는 수학의 오래된 문제중 하나이고 최근들어 암호학과 관련하여 유한체 위서의 다항식 연산과 성질등이 쓰이고 있다. 유한체 위에서 다항식의 최대공약수(greatest common divisor)를 구하는데 많은 시간이 소요 된다. Rabin의 알고리즘에서 주어진 다항식의 근들의 곱 $(F(x), x^q-x)$를 구하는 과정을 $c \in F(p)$, $f_c(x)=(F(x), T_r(x)-c)$, $degf_c(x)>0$인 $f_c(x)$로 대체한 효율적인 알고리즘 제안과 Mathematica를 이용한 프로그램의 실행 결과를 제시한다.

Abstract

A root finding of polynomials over finite fields is an interesting old problem in number theory. We give a root finding algorithm based on Berlekamp's and Rabin's algorithms, and it's efficient implementation using Mathematica

1. 서 론

다항식의 근을 구하는 문제는 오래된 수학의 문제중 하나로서 최근들어 암호학과 관련하여 유한체 위에서 뿐만아니라 Z_n 위에서 다항식의 근을 찾는 문제에 관심이 고조되고 있다. 즉 RSA 암호시스템[3]은 Z_n에서 $x^e-C=0$의 근을 찾는 문제로, Z_p에서 quadratic residue

* 세명대학교 전산정보학부
** 서남대학교 수학과
*** 고려대학교 수학과
† 본 연구는 과학재단의 97년도 특정기초 연구비를 지원받아 수행 되었음.

문제[13]는 $x^2-a=0$의 문제로 볼 수 있다. 한편 J. Schwenk[7]는 n이 두 소수 p, q의 곱일때, Z_n 위에서 다항식의 곱을 이용한 암호시스템을 제안하였다. 이와같이 $Z_n(n=pq,\ p,q$는 소수) 과 유한체 위에서 다항식의 근을 찾는 문제는 암호학과 관련하여 중요한 문제이다.

유한체 위에서 다항식의 근을 찾는 문제는 Berlekamp에 의하여 제시된 이래, 속도를 개선한 Rabin[6]의 확률론적 알고리즘을 제안하였고 그 이후 계속 연구되고 있다. 유한체 위에서 다항식의 GCD(greatest common divisor) 계산은 효율적인 알고리즘이나 실행 시간이 많이 소요되는 알고즘이다. 이 논문에서는 Rabin의 알고리즘에서 주어진 다항식 $F(x)$의 근들의 곱 $(F(x),\ x^q-x)$를 구하는 과정을 $c \in GF(p)$, $f_c(x)=(F(x),\ Tr(x)-c)$, $deg f_c(x)>0$인 $f_c(x)$로 대체함으로 $(F(x),\ x^q-x)$를 $F(x),\ x^{p^{n-1}}-x)$, $q=p^n$를 구하는 과정으로 바꿀 수 있다. 이러한 알고리즘 제안과 Mathematica를 이용한 구현 결과를 제시한다.

2. 유한체의 성질

p를 소수, $q=p^n$, n을 양의 정수라 하자. q개의 원소를 갖는 유한체를 $GF(q)$라 하자. $f(x)$가 유한체 $GF(q)$ 위에서의 다항식이고 $f(\alpha)=0$, $\alpha \in GF(q)$일 때 α를 $f(x)$의 근이라 한다. 그러면 $GF(q)$는 다음과 같이 구성할 수 있다. $f(x)$를 $Z_p=GF(p)$ 위에서 n차 monic인 기약다항식(irreducible polynomial)이라 하면

$$GF(q) \cong Z_p[x]/(f(x)).$$

즉,

$$GF(q)=\{a_0+a_1x+\ldots+a_{n-1}x^{n-1} \mid ai \in GF(p)$$

이다. 그리고 α를 $f(x)$의 근이라 하면

$$GF(q)=\{a_0+a_1\alpha+\ldots+a_{n-1}\alpha^{n-1} \mid ai \in Z_p\}$$

와 같이 표현할 수 있다.

보조정리 1. $GF(q)[x]$에 있는 다항식 $f(x)$가 squarefree이기 위한 필요충분조건은

$$(f,\ f')=1$$

이다.

보조정리 2. $GF(q)[x]$에서

$$x^q-x=\prod_{r \in GF(q)}(x-r)$$

이다.

정리 3. $GF(q)[x]$에서

$$x^q-x=\prod_{c \in GF(p)}(Tr(x)-c)$$

이다.

증명. $\deg(Tr(x))=p^{n-1}$이므로

$$\deg(\prod_{c \in GF(p)}(Tr(x)-c))=p^n=q$$

이고 $a \in GF(q)$에 대하여 $Tr(a) \in GF(p)$이다. 그리고 $b \neq c \in GF(p)$에 대하여

$$(Tr(x)-b,\ Tr(x)-c)=1$$

이므로

$$x^q-x=\prod_{c \in GF(p)}(Tr(x)-c)$$

이다.

보조정리 4. $r \in GF(q)$에 대하여

$$x^q-x=r^{-1}\prod_{c \in GF(p)}(Tr(rx)-c)$$

이다.

증명. $r \in GF(q)$에 대하여

$$\begin{aligned}(rx)^q - rx &= r^q x^q - rx \\ &= rx^q - rx \\ &= \prod_{c \in GF(p)} (Tr(rx) - c)\end{aligned}$$

이므로

$$x^q - x = r^{-1} \prod_{c \in GF(p)} (Tr(rx) - c)$$

이다.

3. 유한체 위에서 다항식의 근 알고리즘

$F(x) \in GF(q)[x]$일 때, 정리 3에 의하여

$$\begin{aligned}f(x) &= (x^q - x,\ F(x)) \\ &= \prod_{c \in GF(p)} (Tr(x) - c,\ F(x))\end{aligned}$$

라 하면 $GF(q)$에 있어서 $F(x)$의 근은 모두 $f(x)$의 근이다. 또한 $x^q - x$가 square-free이므로 $f(x)$도 square-free이다. $GF(q)$를 $GF(p)$ 위에서의 벡터공간으로 보고 $B = \{1, \beta, \beta^2, \ldots, \beta^{n-1}\}$를 $GF(q)$의 기저(basis)라 하자.

정리 5.

1) $f(x) = \prod_{c \in GF(q)} (f(x),\ Tr(\beta^j x) - c)$, $0 \le j < n$.

2) $\deg f(x) > 1$ 이면 $0 \le j' < n$ 이고

$$Tr(\beta^{j'} x) \not\equiv c \bmod f(x),\ c \in GF(p)$$

인 적당한 j'가 존재한다.

증명. 1) $b \ne c \in GF(p)$에 대하여

$$(Tr(x) - b,\ Tr(x) - c) = 1$$

이고

$$x^q - x = r^{-1} \prod_{c \in GF(p)} (Tr(rx) - c)$$

이므로 $GF(q)[x]$에서

$$f(X) = \prod_{c \in GF(q)} (f(x),\ Tr(\beta^j x) - c)$$

이다.

2) 모든 $(0 \le j < n)$에 대해서

$$Tr(\beta^j x) \equiv t_j \bmod f(x),\ 0 \le j < n$$

를 만족하는 $t_j \in GF(p)$ 존재한다고 가정하자. 그러면 $GF(q)$에 있어서 $f(x)$의 근 $s_1, \ldots, s_l$에 대해서

$$\begin{aligned}Tr(\beta^j s_1) &= \ldots \\ &= Tr(\beta^j s_l) \\ &= t_j,\ 0 \ne j < n\end{aligned}$$

이 성립한다. 그러므로 $\mathrm{Char}(GF(q)) = p$를 이용하여

$$Tr(\beta^j (s_i - s_k)) = 0,\ 0 \le j < n,\ 1 \le i < k \le i$$

이 성립함을 알 수 있다. 따라서 $p_0, \ldots, p_{n-1} \in GF(p)$에 대하여

$$\begin{aligned}0 &= \sum_{j=0}^{n-1} p_i Tr(\beta^j (s_i - s_k)) \\ &= Tr\left(\left(\sum_{j=0}^{n-1} p_j \beta^j\right)(s_i - s_k)\right)\end{aligned}$$

이다. $i \ne k$에 대해서 $s_i \ne s_k$이므로 모든 $a \in GF(q)$에 대하여 $Tr(a) = 0$이다. 그러나 $\deg(Tr(x)) = p^{n-1}$이므로 모순이다.

정리5의 2)는

$$f(x),\ Tr(\beta^j x) - c),\ c \in GF(p)$$

를 계산함으로 $f(x)$의 인수를 얻게되고 이 과정을 반복하여 $f(x)$의 일차 인수를 다 구할 수 있다.

알고리즘 6. (Deterministic root finding algorithm)

Input: $F(x) \in GF(q)[x]$, $q=p^n$.
Output: $\alpha \in GF(q) \mid F(\alpha)=0$.

1). $f(x)=F(x)$, x^q-x)를 계산하고 $S(x)=\sum_{j=0}^{n-1} x^{p^j}$ 라 한다.

1)'. $d \in GF(p)$, $f_d(x)=(F(x),\ Tr(x)-d)$, $\deg(f_d(x))>0$인 $f_d(x)$를 $f(x)$라 하고 $S(x)=\sum_{j=0}^{n-1} x^{p^j}$라 한다.

2). $GF(p)$ 위에서 $GF(q)$에 대한 기저 B={1, β, $\beta^2,\ldots,\ \beta^{n-1}$}를 구한다.

3). $K_0=\{(f(x),\ S(x)-c) \mid \deg((f(x),\ S(x)-c))>1,\ c \in GF(p)\}$와 $H_0=\{(f(x),\ S(x)-c \mid \deg((f(x),\ S(x)-c))=1,\ c \in GF(p),\ f(x) \in K_0\}$를 계산한다.

4). $|H_j|=\deg(f(x))$가 될때까지 $j=1$에서 $j=n-1$까지 다음 과정을 반복한다.

$K_j=\{(g(x),\ S(\beta^j x)-c) \mid \deg((g(x),\ S(\beta^j x)-c))>1,\ g(x) \in K_{j-1},\ c \in GF(p)\}$,
$H_j=H_{j-1} \cup \{(g(x),\ S(\beta^j x)-c) \mid deg((g(x),\ S(\beta^j x)-c))=1,\ g(x) \in K_{j-1},\ c \in GF(p)\}$.

5). $|H_j|=\deg(g(x))$일 때 H_j의 다항식의 근을 구한다.

주의. 알고리즘 6에서 1)을 이용하면 $F(x)$의 모든근을 구할수 있고 1)'를 이용하면 Trace값이 d인 모든 근을 구할 수 있다.

q를 홀수라 하자. 다항식 $F(x) \in GF(q)[x]$에 대하여 $GF(q)$에서 $F(x)$의 근중 0을 제외한 모든 근은 $f(x)=(F(x),\ x^{q-1}-1)$의 근이다. $f(x)=(x-\alpha_1)\ \ldots\ (x-\alpha_m)$, $(m>1)$라 하고 q가 홀수이므로 $d=\frac{q-1}{2}$ 라 하면

$$x^{q-1}-1=(x^d-1)(x^d+1)$$

이다. 그러면

$$f(x)=(f(x),\ x^d-1)(f(x),\ x^d+1)$$

이다.

정의 7. $\alpha \neq 0$, $\beta \neq 0 \in GF(q)$이고 d= $\frac{q-1}{2}$ 일 때 $\alpha^d=\beta^d$이면 α, β는 different type이라 한다.

정의 8. $\beta_1 \neq \beta_2 \in GF(q)$이면

$$|\{\delta \in GF(q) \mid \beta_1+\delta,\ \beta_2+\delta : \text{different type}\}| = \frac{q-1}{2}$$

이다.

증명. $\beta_1+\delta$, $\beta_2+\delta$가 different type이기위한 필요충분조건은 $\beta_1+\delta \neq 0$, $\beta_2+\delta \neq 0$ $\left(\frac{\beta_1+\delta}{\beta_2+\delta}\right)^d=1$ 이다. 그러므로 $\beta_1+\delta$, $\beta_2+\delta$가 different type 이면 $\left(\frac{\beta_1+\delta}{\beta_2+\delta}\right)^d \neq -1$ 이다.

그리고 $x^d-1=0$는 $GF(q)$에서 d개의 해를 갖고

$$\phi : GF(q)-\{-\beta_2\} \rightarrow \{GF(q)-\{1\}$$
$$\delta \rightarrow \frac{\beta_1+\delta}{\beta_2+\delta}$$

는 전단사 함수이므로 $\phi(\delta)^d-1=0$의 해는 d개 존재한다.

따름정리 9. $\delta \in GF(q)$에 대하여 $f_\delta(x)=(f(x),\ (x+\delta)^d-1)$라 하면

$$\frac{1}{2} \leq \Pr(\delta \mid 0<\deg(f_\delta(x))<\deg(f(x)))$$

이다.

증명. $f(\alpha_i)=0$, $(1 \le i \le m)$인 α_i에 대해서 $f_\delta(\alpha_i)=0$의 가능성을 생각해 보자.
정리 10에 의하여 $f_\delta(\alpha_2) \neq 0$ 즉, $(\alpha_2+\delta)^d \neq 0$, 이면 $f_\delta(\alpha_1)=0((\alpha_1++\delta)^d=1)$일 가능성은 $\frac{1}{2}$ 이다. 따라서 따름정리가 성립한다.

알고리즘 10. (Probabilistic root finding algorithm)

Input: $F(x) \in GF(q)[x]$, $q=p^n$, p:odd prime
Output: $GF(q)^*$에 있는 $F(x)$의 근 α

1). $c \in GF(p)$에 대하여 $f(x)=(F(x),\ Tr(x)-c)$, $\deg(f(x))>1$인 $f(x)$를 구한다.
2. $d=\frac{q-1}{2}$ 라 하고, $GF(q)$이 있는 $f_\delta(x)=(f(x),\ (x+\delta)^d-1)$, $0<\deg(f_\delta(x))<\deg(f(x))$인 δ를 구한다.
3). $f_\delta(x)$와 $\frac{f(x)}{f_\delta(x)}$ 중 차수가 작은 것을 $f(x)$로 놓는다.
4). 다항식 $f(x)$의 차수가 1일때까지 2, 3을 반복한다.
5). $\deg(f(x))=1$일 때 $f(x)$의 근을 구한다.

주의. 알고리즘 10에서 1)과정은 Rabin의 알고리즘에서 $f(x)=(F(x),\ x^q-x)$를 대체하는 과정으로 $F(x)$와 q차 다항식의 GCD과정을 $F(x)$와 $p^{n-1}(q=p^n)$차의 다항식의 GCD과정으로 변환하고자 한다. 그러나 Trace 함수의 선택에 따라 더많은 시간이 소요 될 수 있다.

4. 실행결과

유한체 $GF(q)$위에 주어진 다항식 $a_0+a_1x+a_2x^2+\ldots+a_mx^m$, $a_i \in GF(q)$를 $a_0, a_1, \ldots, a_m\}$로 나타낸다.

4.1 알고리즘 6에 의한 실행결과

1) $GF(2)[x]/(1+x+x^4+x^9+x^{10})$에서 105차의 다항식 $F(x)$가

$\{x^5+x^6+x^7+x^8,\ 1+x^3+x^7,\ x^3+x^4+x^5+x^9,$
$1+x+x^2+x^3+x^4+x^5+x^8+x^9,\ 1+x+x^2+x^4+x^5,$
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, $x^2+x^3+x^6+x^7+x^8,\ 1+x+x+x^4+x^5+x^6+x^9,$
$x+x^3+x^5+x^6+x^7+x^8,\ 1+x^4+x^5,\ x^4+x^5+x^6,$
$x+x^2+x^4+x^5+x^7,\ 1+x+x^2+x^4+x^5,\ 1+x^5+x^7,$
$x+x^2+x^4+x^5,\ 1\}$

일 때

$$f(x)=\prod_{c \in GF(p)}(F(x),\ Tr(x)-c)$$

는 $\{1+x^5+x^6+x^7+x^8,\ x+x^2+x^4+x^9,\ x^4+x^7+x^8+x^9,\ 1\}$이므로 $F(x)$의 모든 근은 $x+x^2$, x^4+x^5, $x+x^2+x^5+x^7+x^8+x^9$이다.

2) $GF(13)/(2+x^2+x^3)$에서 145차의 다항식 $F(x)$

$x+x^3,\ x+x^3+x^4,\ 1+x+x^2+x^4,\ x^2+x^3+x^4,\ 1,$
$1+x,\ x^2,$ 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 2, 3, 4, x, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1, 1,
0, 0, 2, 0, x, x, 1, 1, 0, 0, 0, 1, 2, 0, 9,

1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 11, 0, 0, 1}

의 근은 $11+12x+4x^2$이다.

4.2 알고리즘 10을 이용한 실행결과

유한체 $GF(p^n)$ 위에서 차수가 p^n보다 작은 다항식을 선택하여 근을 구하였다.

** : Rabin의 알고리즘, ## : 제안된 알고리즘, 단위 : 초

유한체 / 순서	$GF(2)[x]/(2+2x+x^6)$		$GF(5)[x]/(4+x+x^4)$		$GF(13)[x]/(2+x^2)$		$GF(23)[x]/(1+x^2)$	
	**	##	**	##	**	##	**	##
1	934.84	79.95	753.37	52.84	24.6	4.99	219.7	19.22
2	568.00	58.5	574.08	134.96	615.3	2.47	513.01	47.68
3	989.98	299.57	1570.1	64.48	571.48	2.53	416.72	7.2
4	1642.2	1944.5	1564.32	123.87	76.07	48.11	4454.23	4379.27
5	1014.64	486.48	1588.91	457.99	144.12	61.68	450.55	152.2
6	1666.4	530.64	659.17	46.47	390.03	58.6	553.76	97.99
7	119.21	219.43	727.27	260.13	86.94	70.96	598.63	368.77
8	706.06	3365.4	1078.27	761.92	675.21	28.83	2896.88	142.04
9	497.95	1516.6	8247.55	402.05	414.8	112.49	679.87	325.93
10	499.82	317.03	14849.72	7231.27	424.47	88.87	1961.44	439.08

5. 결 론

유한체 위에서의 다항식의 근을 찾는 Berlekamp의 알고리즘은 다항식 $F(x)$의 유한체 $GF(q)$의 근들을 $(x^q-x,\ F(x))$를 이용하여 찾아낸 다음 각 근들을 구하는 알고리즘이다. 그러나 유한체의 원소의 개수가 많아지면 $(x^q-x,\ F(x))$을 계산하는데 많은 시간을 소요 되는 바 정리 3을 이용하여 $c \in GF(p)$, $f_c(x) =(F(x),\ Tr(x)-c)$, $\deg(f_c(x))>0$를 구하게 함으로, 4.2에서와 같이 대부분의 경우에 계산 시간을 단축할 수 있었다. 그리고 Mathematica에 내장된 다항식 연산을 활용하기 위하여 polynomial basis를 이용한 유한체를 사용하여 실행하였다.

참 고 문 헌

[1] D. Copersmith, "Finding a small root of a univariate modular equation", Erocrypt'96, pp. 155-165(1996).

[2] D.E. Knuth, "The art of computer programing", 2nd, Addison-Wesley, New York, 1981.

[3] N. Koblitz, "A course in number theory and cryptography", 2nd, Springer-Verlag, Berlin, 1994.

[4] R. Lidl and H. Niederreiter, "Introduction to finite fields and their applications", Revised edition, Cambridge Unversity press, Cambridge, 1994.

[5] A. Menezes, "Applications of finite

fields", Kluwer academic publishers, Boston, 1993.

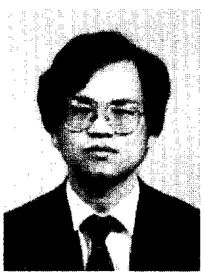
[6] M.O. Rabin, "Probabilistic algorithms in finite fields", Siam J. Comput. 9(2), pp. 273-280, 1980.

[7] J. Schwenk and J. Eisfeld, "Public key encryption and signature schemes based on polynomials over Z_n", Eurocrypt'96, pp. 60-71, 1996.

□ 箸者紹介

김 창 한

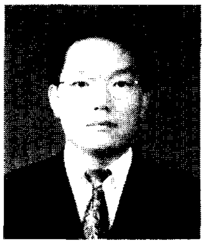
1985년 2월 고려대학교 수학과 학사
1987년 8월 고려대학교 수학과 석사
1992년 2월 고려대학교 수학과 박사
1992년 3월 ~ 현재 세명대학교 전산정보학부 조교수



서 광 석

고려대 수학과 학사(1978)
고려대 수학과 석사(1982)
고려대 수학과 박사(1989)
1991년 서남대 수학과 부교수

※ 주관심 분야 : 전산수론 및 암호학



이 옥 연

1984 ~ 1988 고려대학교 이과대학 수학과 졸업
1988 ~ 1990 고려대학교 이과대학 대학원 수학과 졸업
1990 ~ 1996 University of Kentuket 이학박사

※ 주관심 분야 : Lattice를 이용한 trapdoor one-way function 개발
Lattice와 LLL 알고리듬을 이용한 public key cryptosystem 개발
Ringtheory를 이용한 trapdoor one-way function 개발