F_p 上不可约与本原多项式的高效确定算法*

王泽辉[1], 方小洵[2]
(1. 中山大学科学计算与计算机应用系, 广东 广州 510275;
2. 广东省科技情报研究所, 广东 广州 510033)

摘 要: 对于一大类整数 n(n 为素数乘于素数或 1 的积), 分别给出有限域 F_p 上 n 次多项式是不可约多项式与本原多项式的一个充要条件, 该条件可通过 $O(n^3)$ 次 F_p 上乘法加以验证, 易于硬件实现。提出可约多项式一个充分条件, 借此减少验证时间, 并得到用 $O(n^4)$ 次 F_p 上乘法确定一个 n 次不可约多项式及一个 n 次本原多项式的高效算法。对于 ECC 中构造 F_p^n 上椭圆曲线、序列密码中构造 LFSR, 有重要的应用价值。

关键词: 不可约多项式; 本原多项式; ECC; 序列密码; 多项式时间复杂性; 高效算法
中图分类号: O157.4 文献标识码: A 文章编号: 0529-6579 (2004) 06-0089-04

设 p、q 为素数, $F_p=Z/(p)=\{0, 1, \cdots, p-1\}$ 为有限域, F_p^m 为 F_p 的 m 次扩张。基于 F_p^m 的 ECC (椭圆曲线公匙密码系统) 与基于 F_q 的 ECC 相比, 优点是处理速度快、易于硬件实现, 但在处理多项式乘法之前必须先确定一个 F_p 上 m 次不可约多项式, 这时一般整系数多项式不可约性的判定定理用不上; 文[1]指出, 确定型 (构造性) 算法技术上比较复杂, 一般常采用概率型算法, 目前较好的算法成功率为 $1/m$ (m 较大时), 需计算 $p^{[m/2]}$ 次多项式与 m 次多项式的最小公因式, 算法的时间复杂度为 $O(p^{m/2})$, 属于指数时间算法。另外, 序列密码研究需要确定一个 m 次本原多项式 (取 $p=2$), 以便产生最大周期 2^m-1 的输出序列, 其确定难度更大[2,3]。文[4, 5]均指出确定 F_p 上不可约多项式的困难性[4], 的解决方法是由低阶构造高阶[5], 的解决方案要用到整数的标准分解。本文的方法不同于上述方案。

实用中 (如 ECC 或 LFSR) 不可约或本原多项式的次数 m 并非不可更改, 可用 $m\pm k$ 范围 (k 不大) 中某个数 n 代替 m。如原想构造 F_p^{36}, 按每 36 个 2 进制码一组加密, 即使改成 35 个或 37 个一组也可以, 即改成构造 F_p^{37} 或 F_p^{35}。基于此, 本文对于一大类正整数 n (素数乘以素数或 1 的积), 提出确定 n 次不可约多项式、n 次本原多项式的快速算法, 仅需作 $O(n^3)$ 次 F_p 上乘法, 属于多项式时间算法。

1 不可约多项式的高效确定算法

以下研究的多项式均为最高次项为 1 (简称首一的), 非首一可化为首一多项式乘以非零常数, 不影响不可约性。$F_p[x]$ 为系数在 F_p 上的多项式集合。$F_p[x]$ 上的整除、同余是 Z_p 上整除、同余的类推。$\deg(f)$ 表示 $f(x)$ 的次数。$\hat{}$ 表示幂, $x\hat{}n=x^n$。$\gcd(a, b)$ 表示 a, b 的最大公约数。

定义 1 称为 $f(x)\in F_p[X]$ 为 n 次不可约多项式, 如不存在

$$f_1(x), f_2(x)\in F_p[x],$$
$$\deg(f_1)\geqslant 1, \deg(f_2)\geqslant 1,$$
$$\text{使 } f(x)=f_1(x)f_2(x)$$

引理 1 $F_p[x]$ 上 1 次首一不可约多项式为 $f(x)=x+k$, $k=0, 1, \cdots, p-1$; 任意次不可约多项式 $f(x)$ 中, 满足 $f(0)=0$ 的惟有 $f(x)=x$。

由引理 1, 不可约多项式 $f(x)$ 次数 k 如 $k>1$, 则 $f(0)\neq 0$, 或 $x\nmid f(x)$。

引理 2 $F_p[x]$ 上 n 次首一不可约多项式 $f(x)$ 必为 $x^{(p^n)}-x$ 的因式, n 次 ($n\geqslant 2$) 首一不可约多项式 $f(x)$ 必为 x^N-1 的因式, $N=p^n-1$。

引理 3 设 $g(x)\in F_p[x]$ 为 k 次 ($k\geqslant 2$) 不可约多项式。如 $x^N\equiv 1 \pmod{g(x)}$, $N=p^n-1$, 则 k 整除 n; 反之, 如 k 整除 n, 则 $x^N\equiv 1 \pmod{g(x)}$, $N=p^n-1$。

引理 4 对任意 n, $F_p[x]$ 上多项式 x^N-1 ($N=$

* 收稿日期: 2004-06-23
基金项目: 广东省科技攻关资助项目 (B12206)
作者简介: 王泽辉 (1963 年生), 男, 副教授; 通讯联系人: 方小洵; E-mail: fangxx@stig.gd.cn

p^n-1) 无重因式。

引理 2、3、4 的证明主要思路见文 [2]。

定理 1 设 $g(x)$ 为 F_p 上 n 次($n\geqslant 2$) 首一多项式, n 为素数, $n\geqslant p$, 则 $g(x)$ 为不可约多项式的充要条件是: $x^N\equiv 1(\bmod g(x))$ 或 $g(x)\mid(x^N-1)$, $N=p^n-1$。

证明 先证充分性: 如 $g(x)$ 可约, 则存在 s 个首一不可约多项式 $g_i(x)$, $i=1,2,\cdots,s$; 使 $g(x)=\prod_{i=1}^{s}g_i(x)$, $\deg(g_i)=n_i\geqslant 1$, 故 $n=\sum_{i=1}^{s}n_i$, $n_i<n$, 由引理 4 $g_i(x)$ 互不相同。

因为 $g(x)\mid(x^N-1)$, 所以

$$\exists d(x)\in F_p[x],\ x^N-1=d(x)\times g(x)$$

$$x^N-1=(d(x)\times g_1(x)\times\cdots\times g_{i-1}(x)\times g_{i+1}(x)\times\cdots\times g_s(x))$$

$$x^N\equiv 1(\bmod g_i(x)),\ i=1,2,\cdots,s$$

由引理 3, n_i 整除 n, $n_i<n$, n 为素数, 故 $n_i=1$, $g(x)$ 所有首一不可约因式均为 1 次, 其中不含 x, 至多有 $p-1$ 个 1 次首一不可约因式(由引理 1), 当 $n\geqslant p$ 时, 必有重因式, 与引理 4 矛盾。

必要性由引理 2 易得。证毕。

推论 1 对任意 $g(x)\in F_p[x]$, 如 $g(k)\neq 0$, $\forall k\in F_p$; 则 $g(x)$ 不含 1 次不可约多项式作因式。

证明 由引理 1 易知, $(x+i)\mid g(x)$ 则必 $g(k)=0$, $k=p-i$。

定理 2 设 $g(x)\in F_p[x]$ 为 n 次($n\geqslant 2$) 首一多项式, $n=q_1q_2$, q_1、q_2 为素数, 允许 $q_1=q_2$。则 $g(x)$ 为不可约多项式的充要条件是(1)(2) 成立。其中

(1) $g(k)\neq 0$, $\forall k\in F_p$;

(2) $g(x)\mid(x^N-1)$, $g(x)\nmid(x^{(N_1)}-1)$, $g(x)\nmid(x^{(N_2)}-1)$,

或 $x^N\equiv 1(\bmod g(x))$ 成立, $x^t\equiv 1(\bmod g(x))$, $t=N_1$, N_2 不成立, $N=p^n-1$, $N_1=p^{(q_1)}-1$, $N_2=p^{(q_2)}-1$。

证明 先设 $q_1\neq q_2$, 不妨设 $q_2>q_1$。充分性: 由(1) $g(x)$ 无 1 次不可约多项式作因式, 当 $g(x)\mid(x^N-1)$ 时, 由引理 3, 如 $g(x)$ 可约则其不可约因式次数只能为 q_1、q_2, 设

$$g(x)=\prod_{i=1}^{k_1}g_{i_1}(x)\prod_{i=1}^{k_2}g_{i_2}(x),$$

$$\deg(g_{i_1})=q_1,\deg(g_{i_2})=q_2$$

故

$$\deg(g)=n=q_1q_2=$$

$$k_1\times q_1\otimes k_2\times q_2\Rightarrow q_2\mid k_1,\ q_1\mid k_2$$

$$k_1\neq 0\Rightarrow k_1\geqslant q_2\Rightarrow k_1\times q_1\geqslant n$$

故

$$k_2=O,\ k_1=q_2,$$

$$k_2\neq O\Rightarrow k_2\geqslant q_1\Rightarrow k_2\times q_2\geqslant n,$$

$$k_1=O,\ k_2=q_1$$

如 $g(x)=\prod_{i=1}^{q_2}g_{i_1}(x)$, $g_{i_1}(x)$ 互不相同而次数均为 q_1, 由引理 3, $g_{i_1}(x)$ 为 $x^{(N_1)}-1$ 的不同因式, 故 $g(x)$ 也为 $x^{(N_1)}-1$ 之因式, 即 $g(x)\mid(x^{(N_1)}-1)$, 与(2) 矛盾。其余情况类似。

必要性: 设 $g(x)$ 为 n 次不可约多项式, 则 $g(x)\mid(x^N-1)$, $n\geqslant 2$ 故(1) 成立; 如 $g(x)\mid(x^{(N_1)}-1)$, 由引理 3, $n\mid q_1$, 矛盾; 同理 $g(x)\mid(x^{(N_1)}-1)$, 也矛盾; 故(2) 得证。

当 $q_1=q_2$, 即 n 为素数的平方是上述情况特例, 类似可证。证毕。

因此, 当 $n=q_1q_2$, q_2 为素数, q_1 为素数或 1 时, n 次首一多项式 $g(x)$ 是否可约至多只需求 3 次形如 $x^t(\bmod g(x))$ 之值来确定, $t=p^n-1$, $p^{(q_1)}-1$, $p^{(q_2)}-1$; 如设 $t=\sum_{i=0}^{s-1}c_i2^i$, 求 $x^t(\bmod g(x))$ 可采用下列快速算法。

算法 1 求 $x^t(\bmod g(x))$ 快速算法

$z(x)\leftarrow 1$

for $i\leftarrow s-1$ to O step -1

$z(x)\leftarrow z(x)\times z(x)(\bmod g(x))$

if $c_i=1$ then

$z(x)\leftarrow x\times z(x)(\bmod g(x))$

return$(z(x))$

完成算法只需作 $O(sn^2)$ 次 F_p 上乘法, $s=[\log_2 t]$, 即 t 的 2 进制位, $t=p^n-1$ 时 $s=n$, 判断 $g(x)$ 是否可约总共只需作 $O(n^3)$ 次 F_p 上乘法。

引理 5 设 $F_p[x]$ 上全部 n 次不可约多项式的个数为 $\Phi_p(n)$, 则 $\Phi_p(n)=\sum^{k\mid n}\mu(k)p^{n/k}$, 其中 $\mu(k)$ 为麦比乌斯函数。

证明见[2]。

推论 2 当 $n=q_1q_2$, q_2 为素数, q_1 为素数或 1 时, 随机选择一个 n 次首一多项式 $f(x)$, $f(x)$ 不可约的古率不低于 $1/(3n)$。

证明

(1) 当 $n=q$ 为素数时时, $\Phi_p(n)=(p^q-p)/n=(p^n-p)/n>(p^n/n)/2$;

(2) 当 $n=q^2$, q 为素数时, $\Phi_p(n)=(p^{(q^2)}-$

$p^q)/n=(p^n-p^q)/n>(p^n/n)/2$;

(3) 当 $n=q_1q_2$, q_1、q_2 为素数, $q_1<q_2$ 时, $\Phi_p(n)=(p\uparrow(q_1q_2)-p\uparrow(q_1)-p\uparrow(q_2)+p)/n=[p^n-p\uparrow(q_1)-p\uparrow(q_2)+p]/n>(p^n/n)/3$。而 $F_p[x]$ 上 n 次多项式总个数为 p^n, 故 $f(x)$ 不可约的概率 $>[(p^n/n)/3]/p^n=1/(3n)$。

因此, 随机选择 $3n$ 个 n 次首一多项式 $f(x)$, 以总共作 $O(n^4)$ 次 F_p 上乘法的代价, 可确定 1 个 n 次不可约多项式, 利用下面第二节的技巧, 还可大大提高成功率。

2　本原多项式的高效确定算法

定义 2　设 $g(x)\in F_p[x]$ 为 n 次不可约多项式, 如 $g(x)\mid(x^N-1)$, $g(x)\nmid(x^t-1)$, $\forall t, 0<t<N$, $N=p^n-1$, 则称 $g(x)$ 为 n 次本原多项式。

引理 6　设 $\gcd(n,k)=d$, 则 $\gcd(x\uparrow(p^n-1)-1, x\uparrow(p^k-1)-1)=x\uparrow(p^d-1)-1$。

证明见文[2]。

定理 3　设 $g(x)\in F_p[x]$ 为 n 次($n\geqslant 2$) 首一多项式, n 为素数, $n\geqslant p$。则 $g(x)$ 为 n 次本原多项式的充要条件为: $g(x)\mid(x^N-1)$, $N=p^n-1$, $g(x)\nmid(x^{N/r}-1)$, 对任意 N 的素数因子 r 成立。

证明

充分性: 由定理 1, $g(x)$ 为不可约多项式, 又 $g(x)\mid(x^N-1)$, 如有 $t, 0<t<N$, 使 $g(x)\mid(x^t-1)$, 则 $g(x)\mid\gcd(x^N-1, x^t-1)$, 由引理 6, $g(x)\mid(x^d-1)$, $d=\gcd(N,t)<N$, 故

$$x^d\equiv 1(\bmod g(x)), N/d>1,$$

设 N/d 素数因子为 r, r 也为 N 的素数因子,

$$N/d=rh, N/r=dh,$$

所以 $x^{N/r}=x^{dh}=(x^d)\Lambda h\equiv 1(\bmod g(x))$ 与 $g(x)\nmid(x^{N/r}-1)$ 矛盾。故 $g(x)\nmid(x^t-1)$, $\forall t, 0<t<N$, $N=p^n-1$, 即 $g(x)$ 为 n 次本原多项式。

必要性: $g(x)$ 为 n 次本原多项式必不可约, 故 $g(x)\mid(x^N-1)$, 本原性定义中令 $t=N/r$ 得证。证毕。

定理 4　设 $g(x)$ 为 n 次($n\geqslant 2$) 首一多项式, $n=q_1q_2$, q_1、q_2 为素数, 允许 $q_1=q_2$。则 $g(x)$ 为 n 次本原多项式的充要条件是(1)(2)(3) 均成立。其中:

(1) $g(k)\neq 0$, $\forall k\in F_p$;

(2) $g(x)\mid(x^N-1)$, $g(x)\nmid(x\uparrow(N_1)-1)$, $g(x)\nmid(x\uparrow(N_2)-1)$, $N=p^n-1$, $N_1=p\uparrow(q_1)-1$, $N_2=p\uparrow(q_2)-1$;

(3) 对任意 N 的素数因子 r, $g(x)\nmid(x^{N/r}-1)$。

证明类似于定理 2 及定理 3, 略。

类似可证, 选择 n 使 $N=p^n-1$ 素数因子个数不多时, 以总共作 $O(n^4)$ 次 F_p 上乘法的代价, 可确定 1 个 n 次本原多项式。

定理 5　设 $g(x)$ 为 F_2 上 n 次首一多项式, n 为梅森素数, 即 $N=2^n-1$ 为素数, 则 $g(x)$ 为 n 次本原多项式的充要条件为 $g(0)\neq 0$, $g(1)\neq 0$, $g(x)\mid(x^N-1)$。

构造 ECC 只需找到一个 n 次不可约多项式即可, 故可选择非零系数尽可能少的, 基于 F_p^m 的 ECC 一般取 $p=2$。序列密码中构造 LFSR(线性反馈移位寄存器) 取 $p=2$, 只需找到一个 n 次本原多项式即可时, 其中仅含 3 个非零项的多项式(称三项式) 特别重要。利用下列定理可大大提高生成不可约或本原多项式的效率。

定理 6　设 $g(x)=\sum_{i=0}^{n-1}a_ix^i$ 为 F_2 上 n 次多项式, $a_i\in\{0,1\}$, $a_n=a_0=1$,

(1) $\sum_{i=1}^{n-1}a_i\bmod 2=0$

(2) $\forall a_i$, $i \bmod 2=1$ 时 $a_i=0$;

(3) 求 $i>0$ 时定义

$$I_0=\{i;\ a_i=1, i \bmod 3=0\}, s_0=\sum_{i\in I_0}a_i,$$

$$I_1=\{i;\ a_i=1, i \bmod 3=1\}, s_1=\sum_{i\in I_1}a_i,$$

$$I_2=\{i;\ a_i=1, i \bmod 3=2\}, s_2=\sum_{i\in I_2}a_i,$$

$s_0\bmod 2=0, S_1=S_2, S_1\bmod 2=1$。

当条件(1) 或(2) 或(3) 成立时, $g(x)$ 为可约多项式。

证明　F_2 上 $-1=1$, $-x=x$, $2x=0$。当(1) 成立时, $g(1)=0$, $(x+1)\mid g(x)$。

当 (2) 成立时, 设

$$g(x)=\sum_{2i\leqslant m}a_{2i}x^{2i}=\sum_{x^{2i}\leqslant m}x^{2i}=\left(\sum_{2i\leqslant m}x^i\right)^2$$

$g(x)$ 为一个 n 次多项式的平方, 故可约。当(3) 成立时, 如 $s_0>0$, $g(x)$ 有几对形如 $x^{3s}+x^{3t}$ 的项, 设 $s>t$, $x^{3s}+x^{3t}=x^{3t}(x^{3(s-t)}+1)=x^{3t}(x^{3(s-t)}-1)$ 有因子 x^3-1, 从而有因子 $1+x+x^2$; 另外 $g(x)$ 有 s_1(奇数) 对形如 $x^{3s+2}+x^{3t+1}$ 的项。

$$x^{3s+2}+x^{3t+1}+1=$$

$$x^{3s+2}-x^2+x^2+x^{3t+1}-x+x+1=$$

$$x^2(x^{3s}-1)+x(x^{3t}-1)+x^2+x+1$$

有因子 x^2+x+1, 故 $(x^2+x+1)\mid g(x)$, $g(x)$ 可约。证毕。

故可选 $n=2q$, q 为素数, 素数的确定见文[6],

8 ‖m, n 次不可约多项式很可能有三项式(参见文[2]), 设 $g(x)=x^n+x^i+1$ 为待选 n 次不可约多项式, 则 i 只能选奇数, 如 $n \bmod 3=2$, 则 i 只能在满足 $i \bmod 3=0$ 或 $i \bmod 3=2$ 中选; 如 $n \bmod 3=1$ 则 i 只能在满足 $i \bmod 3=1$ 或 $i \bmod 3=0$ 中选。i 选择范围由{1, 2, …, n - 1} 中 n - 1 个数降为约 1 /2 *2 /3(n - 1) = (n - 1) /3 个。再调用算法 1, 利用上述定理确定一个 F_2 上 n 次不可约或本原多项式仅需作 $O(n^4)$ 次 F_2 上乘法, 实即逻辑"异或" 运算, 易于硬件实现。

例: 构造基于 F_2^{36} 的 ECC, 将 $m=36$ 用 n 代替, $|n-m| \leqslant 5$, 可取 $n=31$、$33=3\times 11$、$34=2\times 17$、$35=5\times 7$、37、$38=2\times 19$、$39=3\times 13$、41。用上述算法求出不可约多项式 $1+x^3+x^{31}$, $1+x^{13}+x^{33}$, $1+x^2+x^{35}$, $1+x^4+x^{39}$, $1+x^3+x^{41}$, 都满足要求; 它们也是本原多项式, 可用于 31、33、35、39、41 级的 LFSR。

对 $m=150$ 取 $n=151$ 代替, 生成本原多项式(也是不可约多项式) $1+x^3+x^{151}$, $1+x^9+x^{151}$, 也可生成 5 项式 $1+x+x^2+x^3+x^{151}$, 或满足其他要求的多项式。

参考文献:

[1] 裴定一, 祝跃飞. 算法数论[M]. 北京: 科学出版社, 2002.

[2] 柯召, 孙琦. 数论讲义[M]. 北京: 高等教育出版社, 2003.

[3] 丁存生, 肖国镇. 流密码学及其应用[M]. 北京: 国防工业出版社, 1994.

[4] 虞培全. 确定有限域上给定周期的不可约多项式的个数以及利用低次不可约多项式构造高次不可约多项式[J]. 数学研究, 2002, 35(12): 439-444.

[5] 郑一. 完全找到整系数不可约多项式的一种新方法[J]. 数学理论与应用, 2003, 23(6): 100-104.

[6] 王泽辉. 超强伪素数及素数检验加速技术[J]. 中山大学学报(自然科学版), 2004, 43(2): 25-28.

Highly Efficient Deriving Calculation Method of Irreducible Polynomial and Primitive Polynomial Over F_p

WANG Ze-hui[1], *FANG Xiao-xun*[2]

(1. Department of Scientific Computation and Computer Science, Sun Yat-sen University, Guangzhou 510275, China;
2. Guangdong Institute of Sci Tech Information, Guangzhou 510033, China)

Abstract: For a wide range of integers n (n is the product of prime number and prime number or 1), a necessary and sufficient condition is given for a polynomial of degree n over the finite field F_p being an irreducible polynomial or primitive polynomial. Such kind of condition can be verified by multiplication of $O(n^3)$ over F_p and easy to be realized by hardware. A sufficient condition of reducible polynomial is proposed to cut down the validating time. A highly efficient calculation method is derived via confirming the irreducible polynomial of degree n or primitive polynomial of degree n using multiplication of $O(n^3)$ over F_p. The result has important application in constructing the elliptic curve of the Elliptic Curve Cryptosystem over F_p^n, and in constructing the Linear Feedback Shift Register of the stream cipher.

Key words: irreducible polynomial; primitive polynomial; Elliptic Curve Cryptosystem; stream cipher; polynomial timing complexity; highly efficient calculation method