中国剩余算法在多元多项式最大公因式提取中的应用*)

杨宁学　诸昌钤　龚　晖

(西南交通大学计算机与通信工程学院　成都610031)

摘　要　本文研究了利用中国剩余算法提取多元整系数多项式的最大公因式的算法，首先将多项式通过同态映射进行逆元化简，直至化为一元多项式，并对一元多项式提取最大公因式，然后利用中国剩余算法对一元最大公因式进行逆元反复提升，最后可得到多元多项式的最大公因式。大量的算例表明该算法是求解多元整系数多项式最大公因式的一种有效算法。

关键词　中国剩余定理，多元多项式，最大公因式，同态映射

The Application of Chinese Remainder Algorithm in the Acquisition of the Greatest Common Divisor from Multivariate Polynomial

YANG Ning-Xue　ZHU Chang-Qian　GONG Hui

(School of Computer & Comm., Southwest Jiaotong University, Chengdu 610031)

Abstract　The application of Chinese Remainder Algorithm to determine the greatest common divisor of multivariate polynomials is studied Using homostasis mapping, a multivariate polynomial is simplified step by step until only one main variable is left, and the greatest common divisor of the single variable polynomial is obtained by the extended Euclid Algorithm. Then, using the Chinese Remainder Algorithm raises the greatest common divisor from one variable to the original multivariate until the real greatest common divisor is gotten. It is an efficient algorithm to solve the greatest common divisor of multivariate polynomials.

Keywords　Chinese remainder theorem, Multivariate polynomial, Greatest common divisor, Homostasis mapping

1　前言

多项式最大公因式的计算是计算机代数中最基本的问题之一。在符号计算中，很多问题都要涉及到最大公因式的求解。例如，为使运算效率更高，在有理分式的表示中我们常常要求分子和分母互素，这样对于给定的多项式，就要求我们去掉最大公因子。在近年来发展迅速的计算机自动批改作业技术中，为了使识别运算快捷可靠，需要对学生和老师答案中的数学表达式提取最大公因式，然后去掉最大公因式，如果在多项式中含有分式，还要对分子分母提取最大公因式并去掉此最大公因式，使其化为最简分式，最后对剩余部分运用符号运算方法进行识别。因此，设计多项式最大公因式的有效算法，对符号计算效率的提高意义十分重大。

文[1～5]研究了单变量多项式的最大公因式提取方法，本文将对多元多项式最大公因式提取进行研究。其主要思想是：先通过同态映射将未知元逐元化简，直至最后只剩下主变元，此时多元多项式被化为一元多项式，并通过文[1～5]中介绍的方法求得最大公因式，然后利用中国剩余算法对一元最大公因式作提升，并得到二元最大公因式，再对二元最大公因式提升得到三元最大公因式，以此逐步提升，最后得到所求的多元最大公因式。

早在二千多年前，在孙子算经中提出了这样一类问题，即一个自然数，除以2余1，除以3余2，除以5余3，求该数。后人把这类问题叫做孙子问题，或者叫做中国剩余问题。对这类问题用严格的数学语言可定义如下：

中国剩余问题[7]　设 $P_1,P_2,\cdots,P_n$ 是两两互素的自然数。若一个非负整数 M 依次用 $P_1,P_2,\cdots,P_n$ 去除，所得余数依次为 $R_1,R_2,\cdots,R_n$，即

$$R_i=M \bmod P_i \quad i=1,2,\cdots,n$$

求 M。

宋代数学家秦九韶给出了中国剩余问题的解法，形成了著名的中国剩余定理(Chinese Remainder Theorem，CRT)。后来在证明中国剩余定理的过程中，得出了求 M 的中国剩余算法(Chinese Remainder Algorithm，CRA)[5～7]，本文将利用该算法来求整系数多元多项式的最大公因式。

2　中国剩余定理

定理(中国剩余定理)　设 $P_1,P_2,\cdots,P_n,R_1,R_2,\cdots,R_n\in Z,P_1,P_2,\cdots,P_n$ 两两互素，令 $P=P_1P_2\cdots P_n$，则存在唯一的整数 $M\in[0,P]$，使得

$$M\equiv R_i \bmod P_i,\ i=1,2,\cdots n$$

证明：(唯一性)设还存在另一整数 M' 满足条件，那么 $M-M'\equiv 0 \bmod P_i,\ i=1,2,\cdots,n$。又 $|M-M'|\in[0,P]$，所以 $M=M'$。

(存在性)取 $C_i=P/P_i$，因为 $P_1,P_2,\cdots,P_n$ 互素，所以 C_i 与 $P_j(j\neq i)$ 不互素，C_i 与 P_i 互素，即有

$$C_iD_i=1 \bmod P_i,$$

其中 D_i 为 C_i 模 P_i 的逆。

$$\therefore M=C_1D_1R_1+C_2D_2R_2+\cdots+C_nD_nR_n=\sum_{i=0}^{n}C_iD_iR_i=C_jD_jR_j \bmod P_j=R_j \bmod P_j。$$

上述证明给我们提供了一个求整数 M 的算法，该算法叫

*)教育部世行贷款"高校考试技术手段的改革与实践"(1282C03041)。杨宁学　博士研究生，研究方向：虚拟现实技术，智能识别。诸昌钤　教授，博士生导师，研究方向：虚拟现实技术，智能交通。龚　晖　教授，博士，研究方向：人工智能、专家系统、现代教育技术的研究与应用。

做中国剩余算法，利用该算法不仅可以对整数作提升，而且可以对整系数多元多项式作提升。

3 多元多项式公因式提取

定义 设 R 和 R' 是两个环，从 R 到 R' 的一个映射 σ，如能保持加法和乘法运算，即$\forall\ a,\ b\in R$，有

$$\sigma(a+b)=\sigma(a)+\sigma(b)$$

$$\sigma(ab)=\sigma(a)\,\sigma(b)$$

则称 σ 是环 R 到 R' 的一个同态映射(简称态射)，或者称 σ 是环 R 到 R' 的一个态射[8]。

设多项式 $f(x,x_1,x_2,\cdots,x_{n-1})=\sum_i a_i x^k x_1^{k1}x_2^{k2}\cdots x_{n-1}^{kn-1}$

和 $g(x,x_1,x_2,\cdots x_n)=\sum_i \Phi_p(a_i)x^k x_1^{k1}x_2^{k2}\cdots x_{n-1}^{kn-1}$

其中 $\Phi_p(a_i)\equiv a_i \bmod p$，$|\Phi_p(a_i)|<p/2$，后面的叙述以 $\Phi_p(A)=A_p$。

由定义很容易得出 Φ_p 为环同态，因此由 $f\to g$ 的映射构成同态映射。

命题：设给定 $A,G\in Z[x,x_1,x_2,\cdots,x_n]$，若 $P|A$，则有 D 使得 $A=GD$，且其同态项满足

$$A_p=G_pD_p$$

多元多项式由于变量较多且变量个数不确定，因此较单变量多项式复杂，处理这类问题的方法是先消元再提升。设多项式 $A\in Z[x,x_1,x_2,\cdots,x_{n-1}]$可表示成如下的形式，

$$A=Q(x_i-a_j)+C$$

其中多项式 C 中不含 $x_i,a_j\in Z$，$j\in Z$，$j\leqslant M$，M 为 A 中变量 x_i 的最高项次数，

$$A\equiv C \bmod x_i-a_j$$

从上面式子可以看出，当 $M>1$时，Q 中还含有 x_i，如果将上式表示成 Q 不含变量 x_i 的形式，则可以得出如下形式：

$$A=Q\times\prod_{j=1}^{L}(x_i^2+b_jx_i+c_j)\times\prod_{k=1}^{N}(x_i-a_k)+C,$$

其中 $2L+N=M$，

定义 设 $A,\ B\in Z[x,x_1,x_2,\cdots,x_n]$，$a\in Z$，当满足

$$\gcd(A,B)_{x=a}=\gcd(A_{x=a},\ B_{x=a})$$

则称 a 为良约化的，否则称 a 为劣约化的。

由于在计算中我们很难事先找出 a_k，因此只能通过随机取值的办法来试算 a_k 的值，由式可知，最多可以取 M 个良约化的数($M=1+\min(\deg_x(A),\ \deg_x(B))$)。

命题：a 是良约化的且 x_i-a 不整除 A、B 关于 x 的首项系数，则

$$\deg_x(\gcd(A,B))=\deg_x(\gcd(A_{x=a},B_{x=a}))$$

有了上述的理论，我们便可以对多元多项式提取公因式，其原理即是先将多元逐步化为一元，并求出一元多项式的最大公因式，然后将求得的最大公因式通过中国剩余算法从一元逐步提升到多元。现将其算法用伪 delphi 语言描述如下：

```
//Input:fa, fb-多元多项式(含一元多项式)
//       x-主变元
//       i-变量下标的最大值
//Output: 多元多项式 fa 和 fb 的最大公因式
Function mygcd(fa, fb, x, i)
Begin
    If i=0 then
    begin
        Result:=gcd(fa,fb,x); //对单变量多项式提取公因式并返回
        Exit;
    End;
    M:=1+min(deg(fa,x),deg(fb,x));
    While(True) do
    begin
        a:=random();
        Subfa:=fa mod (x[i]-a);
        Subfb:=fb mod (x[i]-a);
        If deg(subfa,x[i])=deg(fa,x[i]) and deg(subfb,x[i])=deg(fb,x[i]) then
        Begin
            M_gcd:=mygcd(subfa,subfb,x,i-1);
            Break;
        End;
    End;
    P:=x[i]-a;
    n:=1;M_gcd1:=M_gcd;
    While n<=M do
    Begin
        a:=random();
        Subfa:=fa mod (x[i]-a);
        Subfb:=fb mod (x[i]-a);
        If (deg(subfa,x[i])=deg(fa,x[i])) and (deg(subfb,x[i])
          =deg(fb,x[i])) then
        Begin
            M_gcd:=mygcd(subfa,subfb,x,i-1);
            If deg(M_gcd,x[i])<deg(M_gcd1,x[i])then Continue;
            Pa[n]:= a;R[n]:=m_gcd; //随机数 a 和公因式用于中国剩余算法作提升
            m_gcd1:=m_gcd;
            Inc(n);
        End;
    End;
    Result:=CRT(Pa,R,i,n);
    If (Result=0 mod fa) and (Result=0 mod fb) then Exit;
End;
Function CRT(Pa,R,i,n)
Begin
    P:=(x[i]-Pa[1])(x[i]-Pa[2])…(x[i]-Pa[n]);
    C[j]:=P/P[j]:= (x[i]-Pa[1])(x[i]-Pa[2])…(x[i]-Pa[j-1])( x[i]-Pa[j+1])…(x[i]-Pa[n]);
    C[j]D[j]= 1 mod P[j]; //可通过扩展的 Euclid 算法求得
    Result:=sum(C[j]D[j]R[j],j=1…n);
End;
```

4 算例

已知两个多元整系数多项式 $f_a(x,x_1,x_2)$和 $f_b(x,x_1,x_2)$

$$f_a(x,x_1,x_2):=2x^2x_1^3+10x_2+80x^2x_2-17x_2x_1^5x^3-85x_2^2x_1^2x-680x_2^2x_1^2x^3+x^5x_1^3+5x^3x_2+40x^5x_2+13x_2^2x_1^4x^3+65x_2^3x_1x+520x_2^3x_1x^3$$

$$f_b(x,x_1,x_2):=2x+2x_1^3x_2+10x^2x_2-17x_2x_1^2x^2-17x_2^2x_1^5x-85x_2^2x_1^2x^3+x^4+x^3x_1^3x_2+5x^5x_2+13x_2^2x_1x^2+13x_2^3x_1^4x+65x_2^3x_1x^3$$

试求 $f_a(x,x_1,x_2)$和 $f_b(x,x_1,x_2)$的最大公因式。

根据前文所述的方法，首先将3元多项式化为2元，进而化为1元，然后反向利用中国剩余算法求出2元公因式，进而求出3元公因式。我们不妨以 x 为主变元(为与其它变元区别，此处不设下标)，先将 x_1和 x_2化去，由于 x_1的最高次数为5，x_2的最高次数为3，因此 x_1和 x_2应分别取6个和4个随机数，不妨取 $x_1=5,7,4,-2,3,-3$；$x_2=2,-7,3,-3$。以 $x_2=2$为例，此时可计算出 $f_a(x,x_1[i],2)$和 $f_b(x,x_1[i],2)$，此时 f_a 和 f_b 化为一元多项式，可通过扩展的 Euclid 算法[6]求得最大公因式。设 $R_2(x,x_1[i],2)=\gcd(f_a(x,x_1[i],2),\ f_b(x,x_1[i],2))$有：

$$R_2(x,5,2)=x^3-590x+2;$$

$$R_2(x,7,2)=x^3-1302x+2;$$

$$R_2(x,4,2)=x^3-336x+2;$$

$$R_2(x,-2,2)=x^3-240x+2;$$

$$R_2(x,3,2)=x^3-150x+2;$$

$$R_2(x,-3,2)=x^3-462x+2;$$

上面的最大公因式，实际上也可以叙述为将 $f_a(x,x_1,2)$和 $f_b(x,x_1,2)$分别模 x_1-5，x_1-7，x_1-4，$x_1-(-2)$，x_1-3，$x_1-(-3)$所得到的余数再提取最大公因式得到的，因此由中国剩余算法可知：

$$P=(x_1-5)(x_1-7)(x_1-4)(x_1+2)(x_1-3)(x_1+3)$$

(下转第232页)

频率分辨率 Δf 和最小输出频率 fom 相等，即：

$$\Delta f = \text{fmin} = \text{fclk}/m \quad (2)$$

往往在要求较低频率输出时，数据区可能会较长，采用了类似硬件相位截断的方法，也就是将同一数据重复输出多次 r（由波形的复杂程度、D/A 转换器的位数决定），这样可以成倍减小数据区。

(3)相位控制：通过启用相移字寄存器，就可以实现多通道间的相位关系控制了。如果需要 θ 角的相移，那么相移字为：

$$w = \theta * m/360 \quad (3)$$

(4)幅值控制：为了保证幅值改变时不改变波形质量，采用了双 D/A，分别用于波形输出和幅值调节。幅值 D/A 的输出电压作为波形 D/A 的参考电压 V_{ref} 输入，这样可以快速控制波形幅值。

结束语 将 DDS 软件化，波形、频率和相位等控制更加容易，硬件电路也大大简化。高速的 PCI 总线，再加上 DMA 方式，为产生长周期或非周期波形提供了保障。该多波形信号发生器可以产生标准的正弦、方波、三角波、锯齿波、白噪音、扫频正弦和其它非规则波形，其主要性能指标为：(1)外部时钟：8MHz；(2)输出幅值：±5V 可调；(3)输出频率0.1Hz～500KHz 可调，频率分辨率0.1Hz。

参 考 文 献

1 Spoedler H T W. Virtual Instruments and Virtual Environments [J]. IEEE Instrument. Meas. Mag, 1999, 2(9): 14～19

2 Cristaldi L, Ferrero A, Piuri V. Programmable instruments, virtual instruments, and distributed measurement systems: what is really useful, innovative, and technically soud [J]. IEEE Instum. Meas, 1999, 2: 20～27

3 The Measurement and Automation Catalog 2001. National Instruments

4 Benetazzo L, Bertocco M, Piuri V. A Web-based distributed virtual educational laboratory [J]. IEEE Transaction on Instrum. And Meas, 2000, 49(2): 355～394

5 Bertocco M, Ferraris F, Offelli C, Parvis M. A Client-servers architecture for distributed measurement systems [J]. In: Proc. IMTC98, St. paul, MN, May 1998. 67～72

6 Autodesk Inc. Object ARX Developer's Guide and Reference Manual. 1997

(上接第229页)

由公式 $C[i]=P/P[i]$ 可求出 $C[i]$ ($i=1,2,3,4,5,6$)。

又 $C[i]\times D[i]\equiv 1 \bmod P[i]$ 且 $C[i]$ 与 $P[i]$ 互素，于是可通过扩展 Euclid 计算出 $D[i]$：

$D[1]=-1/224$；$D[2]=1/2160$；$D[3]=1/126$；$D[4]=1/1890$；$D[5]=-1/240$；$D[6]=-1/3360$。再利用中国剩余算法求得：

$$R_1(x,x_1,2)=\gcd(f_a(x,x_1,2),f_b(x,x_1,2))=\sum_{i=1}^{6}C[i]\times D[i]\times R_2(x,x_1[i],2)=x^3-34x_1^2x+52x_1x+2$$

同理可得：

$R_1(x,x_1,-7)=x^3+119x_1^2x-474x_1x+2$

$R_1(x,x_1,3)=x^3-51x_1^2x+117x_1x+2$

$R_1(x,x_1,-3)=x^3+51x_1^2x+117x_1x+2$

继续运用中国剩余算法：

$P=(x_2-2)(x_2+7)(x_2-3)(x_2+3)$

$C[i]=P/P[i]=>$

$C[1]=(x_2+7)(x_2-3)(x_2+3)$

$C[2]=(x_2-2)(x_2-3)(x_2+3)$

$C[3]=(x_2-2)(x_2+7)(x_2-3)$

又由公式 $C[i]\times D[i]=1 \bmod P[i] =>$

$D[1]=-1/45$；$D[2]=-1/360$；$D[3]=1/60$；$D[4]=1/120$；

此时求得的公因式

$$R(x,x_1,x_2)=\sum_{i=1}^{4}C[i]\times D[i]\times R_1(x,x_1,x_2[i])=x^3+13xx_1x_2^2-17xx_1^2x_2+2$$

经验证，该多项式确为原多项式 $f_a(x,x_1,x_2)$ 和 $f_b(x,x_1,x_2)$ 的最大公因式。

总结 本文研究了利用中国剩余算法对多元多项式进行提升从而求得最大公因式的算法，理论研究和实验结果表明用该算法来提取多元多项式的最大公因式是可靠和高效的。下表是运用本算法运行的时间开销，实验环境为：CPU：PIII 450；内存：SDR 512M；Windows 2000 professional 操作系统，实验数据为10次运算的算术平均值。由实验结果可知，计算所需的时间与多项式的次数有密切关系，因此如果能在化元期间对变量的随机取值方面作一些优化，将会使得该算法对高次多项式的公因式提取有更高的效率。

mygcd(f_1, f_2)	0.055(s)
mygcd(f_3, f_4)	0.079(s)
mygcd(f_5, f_6)	0.112(s)

其中：

$f_1=(123x^2+23x_1^3x_2+11x_2^2)(58x^2-17x_1x_2+3x^3+23x_1)$

$f_2=(11x^2+5x_1^2x_2+x_2^2)x^3x_1^2(123x^2+23x_1^3x_2+11x_2^2)(58x^2-17x_1x_2+3x^3+23x_1)$

$f_3=(123x^2+23x_1^3x_2+11x_2^2)^2(58x^2-17x_1x_2+3x^3+23x_1)^2$

$f_4=(11x^2+5x_1^2x_2+x_2^2)x^3x_1^2(123x^2+23x_1^3x_2+11x_2^2)^2(58x^2-17x_1x_2+3x^3+23x_1)^2$

$f_5=(123x^2+23x_1^3x_2+11x_2^2)^7(58x^2-17x_1x_2+3x^3+23x_1)^3$

$f_6=(11x^2+5x_1^2x_2+x_2^2)^2x^3x_1^2(123x^2+23x_1^3x_2+11x_2^2)^3(58x^2-17x_1x_2+3x^3+23x_1)^3$

参 考 文 献

1 Parikh S N, Matula D W. A redundant binary Euclidean GCD algorithm [J]. Computer Arithmetic. In: Proc. 10th IEEE Symposium on, 1991. 220～225

2 Mansour Y, Schieber B, Tiwari P. Lower bounds for integer greatest common divisor computations [J]. Foundations of Computer Science. In: 29th Annual Symposium on, Oct. 1988. 54～63

3 Wu P-Y, Chen C-L, Parallel J. Extended GCD algorithm [J]. Parallel Processing Symposium. In: Proc. Eighth Int., April 1994. 357～361

4 Brown W S. On Euclid's Algorithm and the Computation of Polynomial Greatest Divisors [J]. J. ACM, 1971(18): 476～504

5 Calvez L C, Azou S, Vilbe P. Variation on Euclid's algorithm for polynomials [J]. Electronics Letters, 1997, 33(11): 939～940

6 Singh B, Siddiqi M U. Multivariate polynomial products over modular rings using residue arithmetic [J]. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], 1995, 43(5): 1310～1312

7 袁仁保，马建. 求解孙子问题的算法及符号计算软件[J]. 小型微型计算机系统，1996, 17(12): 20～25

8 丘维声. 高等代数[M]. 高等教育出版社，2000. 14～15