Euclid算法及扩展在密码学中的研究和应用

陈良臣[1,2],芦东昕[2],李春葆[3]
(1.华北电力大学 计算机科学与技术学院,北京 102206;
2.中兴软件技术(南昌)有限公司,江西 南昌 330058;
3.武汉大学 计算机学院,湖北 武汉 430079)

摘 要:信息安全是网络时代的焦点,密码技术是信息安全的核心,而算法是密码学的精髓。文中研究了基于因数分解的Euclid算法和扩展Euclid算法,包括算法的基本原理、算法流程及编程实现。分析了Euclid算法的算法复杂性,介绍了Euclid算法在RSA和Affine Cipher密码系统中的应用,最后指出了该算法存在的缺陷和算法需要改进的方向。

关键词:Euclid算法;加密算法;RSA;Affine Cipher
中图分类号:TP309.7 **文献标识码**:A **文章编号**:1673-629X(2006)11-0156-04

Research and Application of Euclid Algorithm and Extended Euclid Algorithm

CHEN Liang-chen[1,2], LU Dong-xin[2], LI Chun-bao[3]
(1. School of Computer Science & Technology, North China Electric Power University, Beijing 102206, China;
2. ZTE Software Engineering Co., Ltd, Nanchang 330058, China;
3. School of Computer Science & Technology, Wuhan University, Wuhan 430079, China)

Abstract: The information security is the focal point of the network times. Cryptology is the core of the information security, and algorithm is the soul of the cryptology. Investigated the Euclid algorithm based on factorization and extended E uclid algorithm, including their rationale, process and programme. Then analyzed the complexity of the Euclid algorithm, and introduced its application in RSA and Affine Cipher. At last, point out the limitation of the Euclid algorithm and where the algorithm should be improved.

Key words: Euclid algorithm; encrpytion techniques; RSA; Affine Cipher

0 引 言

随着计算机与网络技术的不断发展和广泛应用,Internet已经成为一个非常复杂且极不安全的信息载体,信息安全成为网络信息时代的焦点,密码技术是信息安全的核心,而算法又是密码技术的精髓。所以研究和设计好的算法是提高信息安全的关键。

Euclid算法是公元前希腊著名数学家欧几里得提出的,又称为辗转相除法,是基于因式分解求两个整数 a,b 最大公因数的最普遍的算法。Euclid算法对于早期算法和程序性过程的研究起到了非常重要的作用,而且一直沿用至今,在密码学领域和数论研究中具有极其重要的意义[1]。

收稿日期:2006-03-08
基金项目:中国下一代互联网示范工程(CNGI)移动奥运资助项目(CNGI-04-17-2A)
作者简介:陈良臣(1982-),男,湖北武汉人,硕士研究生,研究方向为网络与信息安全;芦东昕,博士后,教授,研究方向为网络控制、网络与信息安全;李春葆,教授,研究方向为软件工程、人工智能与信息安全。

Euclid算法在RSA和Affine Cipher密码系统中普遍应用,其中RSA是当前最著名、应用最广泛的公钥系统,基于因式分解的Euclid算法及扩展Euclid算法是RSA和Affine Cipher密码系统中实现数据加密的基础。所以研究和改进Euclid算法对于密码学的研究和发展有着非常重要的理论意义。

1 相关背景

定义1 设 a,b 是任意整数,如果存在整数 c,使有 $a=bc$,则称 a 是 b 的倍数,b 是 a 的因数;亦说 a 被 b 整除,或 b 整除 a,记为 $b \mid a$。

定理1 设 a,b 是任意整数且 $b \neq 0$,则惟一存在整数 q 和 r,使得 $0 \leqslant r < |b|$,$a = qb + r$。若 $r > 0$,则称 q 为带余除法的不完全商,称 r 为 b 除 a 的余数。

证明:

1) 证存在整数 q 和 r,使得 $0 \leqslant r < |b|$,$a = qb + r$。考虑 b:若 $b > 0$,则 b 的倍数数可递增排列为:$\cdots, -4b, -3b, -2b, -b, 0, b, 2b, 3b, 4b, \cdots$;若 $b < 0$,则 b 的倍数数可递增排列为:$\cdots, 4b, 3b, 2b, b, 0, -b, -2b, -3b,$

$-4b,\cdots$。有两种情况:

(1) 存在整数 q,使得 $a = qb$,此时 $r = 0$,问题得证。

(2) 当 $b > 0$ 时,存在整数 q,使得 $qb \leqslant a < (q+1)b$;当 $b < 0$ 时,存在整数 q,使得 $qb \leqslant a < (q-1)b$。因而有 $a = qb + r\ (*), 0 \leqslant r < |b|$。

2) 证存在的整数对 q 和 r 惟一。如果另有 q' 和 r' 满足 $a = q'b + r'\ (**), 0 \leqslant r' < |b|$,则由式$(**)-(*)$得 $r' - r = (q - q')b$,并有 $|r' - r| = |q' - q|\,b$。鉴于 $|r' - r| < |b|$,$|q' - q| \geqslant 0$ 且皆为整数,故必有 $|q' - q| = 0$,从而 $|r' - r| = 0$,即 $q' = q$,$r' = r$。惟一性成立。

推论 1　设 a, b 是任意整数且 $b \neq 0$,则 $\gcd(a,b) = \gcd(b, a \bmod b)$。其中 gcd 表示两个整数的最大公因数。

证明:设正整数 $a = kb + r$,则 $r = a \bmod b$。

假设 d 是 a, b 的一个公约数,则有 $d \mid a$,$d \mid b$,而 $r = a - kb$,因此 $d \mid r$,因此 d 是 $(b, a \bmod b)$ 的公约数。

假设 d 是 $(b, a \bmod b)$ 的公约数,则 $d \mid b$,$d \mid r$,而 $a = kb + r$。因此 d 也是 (a,b) 的公约数,因此 (a,b) 和 $(b, a \bmod b)$ 的公约数是一样的,其最大公约数也必然相等,得证。

定理 2　整数 a, b 的最大公约数 $d = (a,b)$ 可以表示为 a, b 的倍数和,即存在整数 s, t,使 $d = sa + tb$。

证明:设在求取 $d(= (a,b) = r_n)$ 的辗转相除过程中得:

$$a = q_1 b + r_1,$$
$$b = q_2 r_1 + r_2,$$
$$r_1 = q_3 r_2 + r_3,$$
$$\cdots\cdots$$
$$r_{i-2} = q_i r_{i-1} + r_i,$$
$$\cdots\cdots$$
$$r_{n-2} = q_n r_{n-1} + r_n,$$
$$r_{n-1} = q_{n+1} r_n。$$

只需证明对每个正整数 $i(i = 1,2,3,\cdots,n)$,都存在整数 s' 和 t',r_i 总可以表示为 $r_i = s'a + t'b$ 的形式[1]。

数学归纳法证明:

当 $i = 1$ 时,$r_1 = a - q_1 b = a + (-q_1)b$。

当 $i = 2$ 时,$r_2 = b - q_2 r_1 = b - q_2(a - q_1 b) = (-q_2)a + (1 + q_1 q_2)b$。

设对 $r_{i-1}, r_{i-2}, 3 \leqslant i \leqslant n$,分别有整数 s', t' 和 s'', t'' 使得 $r_{i-1} = s'a + t'b$,$r_{i-2} = s''a + t''b$,则 r_i 也可表示为同样的形式:

$$r_i = -q_i r_{i-1} + r_{i-2} = -q_i(s'a + t'b) + s''a + t''b = (s'' - s'q_i)a + (t' - t'q_i)b。$$

即知所证成立。

定理 2 引申:使用矩阵知识,构造结论式 $d = sa + tb$ 中的 s 和 t。

改写并扩展定理 2 证明中的辗转相除式为:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} q_1 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} b \\ r_1 \end{pmatrix}$$

$$\begin{pmatrix} b \\ r_1 \end{pmatrix} = \begin{pmatrix} q_2 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$$

……

类而推之得:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} q_1 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} q_2 & 1 \\ 1 & 0 \end{pmatrix}\cdots\begin{pmatrix} q_i & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} r_{i-1} \\ r_i \end{pmatrix} = A_i \begin{pmatrix} r_{i-1} \\ r_i \end{pmatrix}$$

式中 $A_i = \begin{pmatrix} q_1 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} q_2 & 1 \\ 1 & 0 \end{pmatrix}\cdots\begin{pmatrix} q_i & 1 \\ 1 & 0 \end{pmatrix}$

且 $\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} T_i & V_i \\ S_i & U_i \end{pmatrix}\begin{pmatrix} r_{i-1} \\ r_i \end{pmatrix}$

考虑到 $\begin{vmatrix} q_1 & 1 \\ 1 & 0 \end{vmatrix} = \begin{vmatrix} q_2 & 1 \\ 1 & 0 \end{vmatrix} = \cdots = \begin{vmatrix} q_i & 1 \\ 1 & 0 \end{vmatrix} = -1$

及 $|A_i| = (-1)^i$,知存在 $\begin{pmatrix} T_i & V_i \\ S_i & U_i \end{pmatrix}^{-1}$,且

$$\begin{pmatrix} T_i & V_i \\ S_i & U_i \end{pmatrix}^{-1} = \frac{A_i^*}{|A_i|} = \begin{pmatrix} (-1)^i U_i & (-1)^{i+1} V_i \\ (-1)^{i+1} S_i & (-1)^i U_i \end{pmatrix}$$

于是有

$$\begin{pmatrix} r_{i-1} \\ r_i \end{pmatrix} = \begin{pmatrix} T_i & V_i \\ S_i & U_i \end{pmatrix}^{-1}\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} (-1)^i U_i & (-1)^{i+1} V_i \\ (-1)^{i+1} S_i & (-1)^i U_i \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix}$$

且 $r_i = (-1)^{i+1} s_i a + (-1)^i T_i b$。

特别还有 $r_n = (-1)^{n+1} s_n a + (-1)^n T_n b$,即定理 2 结论式中 $d = sa + tb$ 中的

$$s = (-1)^{n+1} s_n, t = (-1)^n T_n,$$

$$\begin{pmatrix} T_i & V_i \\ S_i & U_i \end{pmatrix} = \begin{pmatrix} q_1 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} q_2 & 1 \\ 1 & 0 \end{pmatrix}\cdots\begin{pmatrix} q_i & 1 \\ 1 & 0 \end{pmatrix} =$$

$$\begin{pmatrix} T_{i-1} & V_{i-1} \\ S_{i-1} & U_{i-1} \end{pmatrix}\begin{pmatrix} q_i & 1 \\ 1 & 0 \end{pmatrix}$$

因为 $U_i = S_{i-1}, V_i = T_{i-1}$

所以 $U_{i-1} = S_{i-2}, V_{i-1} = T_{i-2}(i > 2)$　　(1)

$S_i = q_i S_{i-1} + U_{i-1}, T_i = q_i T_{i-1} + V_{i-1}(i > 2)$　　(2)

将式(1)代入式(2)得:

$S_i = q_i S_{i-1} + S_{i-2}, T_i = q_i T_{i-1} + T_{i-2}(i > 2)$　　(3)

补充定义 $U_1 = S_0, V_1 = T_0$,则式(1)和式(3)对 $i \geqslant 2$ 也成立;而且从 $\begin{pmatrix} T_1 & V_1 \\ S_1 & U_1 \end{pmatrix} = \begin{pmatrix} q_i & 1 \\ 1 & 0 \end{pmatrix}$ 得到递推公式 $S_i = q_i S_{i-1} + S_{i-2}, T_i = q_i T_{i-1} + T_{i-2}(i \geqslant 2)$ 的初始值条件 $S_0 = 0, S_1 = 1; T_0 = 1, T_1 = q$[2]。

2　Euclid 算法及其扩展

2.1　Euclid 算法

Euclid 算法是求最大公因数的最普遍的算法,有时也称为辗转相除法。Euclid 算法就是这个式子:$\gcd(a,b) = \gcd(b, a\ \%b)$。

表述如下:设给定 $m, n(m > n)$,令 $r_0 = m, r_1 = n$,有:

$$\begin{cases} r_0 = r_1 q_1 + r_2, & 0 \leqslant r_2 < r_1 \\ r_1 = r_2 q_2 + r_3, & 0 \leqslant r_3 < r_2 \\ \vdots & \\ r_{k-2} = r_{k-1} q_{k-1} + r_k, & 0 \leqslant r_k < r_{k-1} \\ r_{k-1} = r_k q_k \end{cases} \tag{4}$$

则得 $r_k = \gcd(r_{k-1}, r_k) = \gcd(r_{k-2}, r_{k-1}) = \cdots = \gcd(r_2, r_3) = \gcd(r_1, r_2) = \gcd(r_0, r_1) = \gcd(m, n)$。算法中做带余除法的次数 k 可由 m 和 n 确定[3]。

Euclid 算法求 gcd 的递推算法流程：

取两个正整数 m, n，第一步是比较 m, n 的值，确定 m 的值是较大的，如果不是，互换 m, n 的值。第二步是用 n 除 m 并令 r 为余数，若 $r = 0$，n 就是 m，n 的最大公约数，若 $r \neq 0$，使 $m = n, n = r$，并返回步骤二。Euclid 求 gcd 递推流程如图 1 所示[4]。

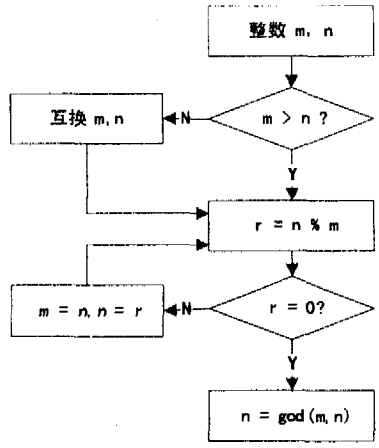

图 1 Euclid 递推算法流程

下面是著名求最大公约数的辗转相除算法的代码实现：

```
int Euclid_Algorithm (int m,int n)
{
    int temp=m;
    if (! m || ! n) return 0;
    if (m < n) {m=n; n=temp;}
    while (1) {
        if (! (m=m % n)) return n;
        if (! (n = n % m)) return m;
    }
}
```

2.2 扩展 Euclid 算法

定义 2 对于整数 a, p，如果存在整数 b，满足 $ab \bmod p = 1$，则说 b 是 a 的模 p 乘法逆元。

定理 3 对于整数 a 和 p，a 存在模 p 的乘法逆元的充要条件是 $\gcd(a, p) = 1$。

证明：首先证明充分性。

如果 $\gcd(a, p) = 1$，根据欧拉定理，$a^{\varphi(p)} \equiv 1 \bmod p$，因此显然 $a^{\varphi(p)-1} \bmod p$ 是 a 的模 p 乘法逆元。

再证明必要性。

假设存在 a 模 p 的乘法逆元为 b，$ab \equiv 1 \bmod p$ 则 $ab = kp + 1$，所以 $1 = ab - kp$。

因为 $\gcd(a, p) = d$，所以 $d \mid 1$，故 d 只能为 1。

推论 2 如果 $\gcd(a, b) = d$，则存在 m, n，使得 $d = ma + nb$，称这种关系为 a、b 组合整数 d，m、n 称为组合系数。当 $d = 1$ 时，有 $ma + nb = 1$，此时可以看出 m 是 a 模 b 的乘法逆元，n 是 b 模 a 的乘法逆元。

扩展的 Euclid 算法是 Euclid 算法的推广，该算法和 Euclid 算法在计算最大公约数上是一致的，不仅能得出任意两个正整数 a 和 b 的最大公约数 d，还能计算出满足 $d = \gcd(a, b) = ax + by$ 的整系数 x 和 y，即 a 模 b 和 b 模 a 的乘法逆元[5]。

1) 如果 $b = 0$，则 $\gcd(a, b) = a, x = 1, y = 0$；

2) 如果 $b \neq 0$，则首先计算 $d' = \gcd(b, a \bmod b)$ 和满足 $d' = bx' + (a \bmod b)y'$ 的 x', y'。这种情况下，有 $d = \gcd(a, b) = d' = \gcd(b, a \bmod b)$。

3) 为了得到满足 $d = ax + by$ 的 x, y，利用等式 $d = d' = bx' + (a \bmod b)y'$，得出 $d = bx' + (a - [a/b]b)y' = ay' + b(x' - [a/b]y')$。

因此，当选择 $x = y', y = x' - [a/b] * y'$，就可以满足等式 $d = ax + by$。

现在考虑一般的 $ax + by = 1$ 如何求解。因为满足条件的 x, y 存在的条件是 $\gcd(a, b) = 1$。然后有 $ax + by = \gcd(a, b)$，而同时有 $bx' + (a \% b)y' = \gcd(b, a \% b)$。由 Euclid 定理 $\gcd(a, b) = \gcd(b, a \% b)$，所以有 $ax + by = bx' + (a \% b)y' = bx' + (a - [a/b] * b)y' = bx' + ay' - [a/b] * by' = ay' + b(x' - [a/b]y')$，对应 $x = y', y = x' - [a/b]y'$。$[a/b]$ 是 a/b 再取整。特别地，在 $b = 0$ 时，$\gcd(a, b) = a = a * 1 + b * 0$，即 $x = 1, y = 0$。

扩展的 Euclid 算法程序如下[2]：

```
int EuclidExp_Algorithm (int a,int b,int &ar,int &br)
{
  int x1,x2,x3;
  int y1,y2,y3;
  int t1,t2,t3;
  int k;
    if (0 == a ) {ar=0,br=0, return b };
    if (0 == b ) {ar=0,br=0, return a };
    else{
    x1 = 1;x2 = 0;x3 = a;
    y1 = 0;y2 = 1;y3 = b;
    for(t3 = x3%y3;t3! =0;t3 = x3%y3)
    {
      k = x3%y3;
      t1 = x1 - k * y1;
      t2 = x2 - k * y2;
      x1 = y1;x2 = y2;x3 = y3;
      y1 = t1;y2 = t2;y3 = t3;
    }
```

```
If (y3 = =1) //有乘法逆元
{ar = y2;br = x1;return 1;
}else{ //公约数不为一,无乘法逆元
  ar = 0;br = 0;return y3;
}
}
```

2.3 Euclid 算法的复杂性分析

定理4　若 $n_1 > n_2 > 0$,则欧几里得算法求 $\gcd\{n_1, n_2\}$ 需要 $d < 2 * \log_2 n_1$ 次除法。不仅如此,实际上对于 $n_1 > n_2 \in Z \setminus \{0\}$,存在 $a, b \in Z$,使得 $\gcd\{n_1, n_2\} = a * n_1 + b * n_2$。

定理5　式(4)中的 $k < 5\log_{10} n + 1$,即得辗转相除的次数不大于 n 的十进位表示的位数的5倍。

证明:引入斐波那契序列 $F_0 = 0, F_1 = 1, F_n = F_{n-1} + F_{n-2}, n = 2,3,4,\cdots$,易知有:

$$F_n = \frac{1}{\sqrt{5}} \cdot \left[\left(\frac{1+\sqrt{5}}{2} \right)^n - \left(\frac{1-\sqrt{5}}{2} \right)^n \right]$$

由式(4)得,

$F_2 + F_3 = F_4, \cdots, r_1 \geqslant r_2 + r_3 \geqslant F_k + F_{k-1} = F_{k+1}$。

而 $F_{k+1} > \left[\left(\frac{1+\sqrt{5}}{2} \right)^{k-1} \right], r_k \geqslant 1 = F_2, r_{k-1} > r_k$,故 $r_{k-1} \geqslant r_{k+1} \geqslant 2 = F_3, r_{k-2} \geqslant r_{k-1} + r_k$,当 $k \geqslant 2$ 时成立,故 $n = r_1 \geqslant F_{k+1} \left[\left(\frac{1+\sqrt{5}}{2} \right)^{k-1} \right]$,

即 $\log_{10} n > (k-1)\log_{10} \left(\frac{1+\sqrt{5}}{2} \right)$,而 $\log_{10} \left(\frac{1+\sqrt{5}}{2} \right) > \frac{1}{5}$,故 $\log_{10} n > \frac{1}{5}(k-1)$,因而 $k < 5\log_{10} n + 1$。

设 n 的十进位数表示的位数是1,则有 $n < 10$,故 $k < 51 + 1$,而 k 和 $51 + 1$ 都是整数,故 $k \leqslant 51$。

推论3　用欧几里得算法求 $m, n (m \geqslant n)$ 的最大公因数的计算量是 $O(\log_2 m^3)$。

证明:因为式(4)中的 $k < 5\log_{10} n + 1$,而且,其中每次带余除法的被除数和除数都不大于 m,每次带余除法的计算量是 $O(\log_2 m^2)$,$\gcd(m, n)$ 的计算量是 $O(\log_2 m^3)$。

熟知,由式(4)的最后一个等式往回推演,可以得到 u 和 v,使 $\gcd(m, n) = um + vn$,而且计算出 u, v 的计算量也可以证明是 $O(\log_2 m^3)$,故对一次不定方程 $ax + by = c$(其中 $\gcd(a, b) \mid c$)和一次同余式 $ax \equiv c(\mathrm{mod} b)(\gcd(a, b) \mid c)$ 求解的计算量是 $O(\log_2 m^3)$,这里 $m = \max(a, b, c)$。

3 欧几里得算法的应用

Euclid 算法是早在电子计算机时代之前最有名的一个算法。现在 Euclid 算法和扩展 Euclid 算法在数据加密领域广泛应用,主要用于 RSA 密码系统和仿射密码(Affine Cipher)系统中,在量子加密技术中也有很重要的作用。

3.1 RSA 算法

RSA 公钥密码算法是一种公认十分安全的公钥密码算法。该算法是目前网络上进行保密通信和数字签名的最有效、最成功的安全算法。RSA 算法原理:随机选择两个大素数 p, q,计算 $N = p \cdot q, \phi(N) = (p-1)(q-1)$。选择 e 使得 $1 < e < \phi(N)$,且 $\gcd(e, \phi(N)) = 1$,计算 d,满足 $e \cdot d = 1 \bmod \phi(N)$ 且 $0 \leqslant d \leqslant N$[6]。

公布公钥:$KU = \{e, N\}$;

保存私钥:$KR = \{d, N\}$;

若 M 为信息,

加密:$C = M^e \bmod N$,

解密:$M = C^d \bmod N$。

RSA 算法的安全性在于对于一个大数 n,没有有效的方法能够将其分解,从而在已知 n, d 的情况下无法获得 e;同样在已知 n, e 的情况下无法求得 d。在 RSA 算法中,使用 Euclid 算法验证所选择的 e 满足 $\gcd(e, \phi(N)) = 1$,已知 e,使用到扩展 Euclid 算法计算出 d 值[7]。

3.2 Affine Cipher 算法

仿射密码(Affine Cipher)算法是加法密码和乘法加密的组合。Affine Cipher 算法的加密函数取形式为:$E(x) = ax + b(\mathrm{mod}\ 26)$,a、b $\in Z/(26)$。要求唯一解的充要条件是 $\gcd(a, 26) = 1$[8]。

Affine Cipher 算法描述为:

设 $P = C = Z/(26)$,

$K = \{(a, b) \in Z/(26) \times Z/(26) \mid \gcd(a, 26) = 1\}$。

密钥对 $k, k = (a, b) \in K$。

加密:$eK(x) = ax + b(\mathrm{mod}\ 26)$;

解密:$dK(y) = a^{-1}(y - b)(\mathrm{mod}\ 26)$。

其中 $x, y \in Z/(26)$。

解密仿射密码过程中,需要使用 Euclid 算法来计算两个整数的最大公约数。给定两个不同的整数 r_0 和 r_1,Euclid 算法计算 $\gcd(r_0, r_1)$;使用扩展的 Euclid 算法,根据等式 $y = (x * m + b) \bmod 26$,可以得出 $x = [m^{-1}(y - b)] \bmod 26 = [m^{-1} \bmod 26][(y - b) \bmod 26]$,此时扩展的 Euclid 算法能够被用于计算 $m^{-1} \bmod 26$。

4 结　论

Euclid 算法是计算两个数最大公约数的传统算法,它无论从理论还是从效率上都是很好的,更具可读性且易交流。在加密算法中,面对一般情况计算两个数最大公约数,Euclid 算法是优先考虑的方法。但是这种算法有一个致命的缺陷,这个缺陷只有在大素数时才会显现出来。

考虑现在的硬件平台,一般整数最多也就是 64 位,对于这样的整数,计算两个数之间的模是很简单的。对于字长为 32 位的平台,计算两个不超过 32 位的整数的模,只需要一个指令周期,而计算 64 位以下的整数模,也不过几

(下转第 184 页)

又如何能做出人性化的设计。因此在采用虚拟技术后,使用者与产品之间多了一些亲切的感觉,而不是被动的、机械地接受产品。

第三是降低了开发的风险性:业内专家指出,利用虚拟技术通常可使产品的开发效率提高 3 至 5 倍。使用者的直接参与使得产品在开发阶段就得到了使用者各方面的鉴定,产品本来就是为使用者所使用,既然各方面都通过了使用者的鉴定,那么自然会得到使用者的青睐。同时由于可以直接参与感受,也可以使设计师之间进行更好的交流互动,这就避免了一个产品只刻上某一个设计师的烙印。

3 虚拟设计在工业产品中的应用

虽然目前虚拟技术才刚刚起步,但它已经取得了可喜的进步。例如波音 777 飞机的设计制造过程就是一个较为成功的范例,它利用虚拟现实技术进行各种条件下的模拟试飞,工程师们在工作站上实时采集和处理数据并及时解决设计问题。使得最终制造出来的波音 777 飞机与设计方案误差小于 0.001 英寸,保证了机身和机翼一次对接成功和飞机一次上天成功,整个设计制造周期从 8 年缩短到 5 年[5];美国福特汽车公司采用网络并行设计技术制造的新型 SS1 型赛车从开始设计到上道测试仅用了 9 个月时间。

各个国家也已经开始认识到虚拟技术的巨大潜力并逐步开始大力发展虚拟技术。美国已经从虚拟制造的环境和虚拟现实技术、信息系统、仿真和控制、虚拟企业等方面进行了系统的研究和开发,多数单元技术已经进入实验和完善的阶段;欧洲以大学为中心也纷纷开展了虚拟制造技术研究,如虚拟车间、建模与仿真工程等的研究;中国在虚拟制造技术方面的研究只是刚刚起步,其研究也多数是在原先的 CAD/CAE/CAM 和仿真技术等基础上进行的,目前主要集中在虚拟技术的理论研究和实施技术准备阶段,系统的研究尚处于国外虚拟制造技术的消化和与国内环境的结合上。

4 结束语

目前,虚拟设计技术在产品方面还具有很大的发展空间,但是由于计算机等硬件设备问题等的原因使其还不能得到充分的应用。但是展望未来,它将是一种崭新的设计方式,更好地利用虚拟设计技术将使产品设计发展到一个全新的领域,而虚拟设计的进一步发展则有待于做更进一步的研究。

参考文献:

[1] 刘宏增,黄靖远. 虚拟设计[M]. 北京:机械工业出版社. 1999:20-22.

[2] 罗天龙,孙克豪. 虚拟设计与网络化制造研究综述[J]. 机械制造,2004(7):31-34.

[3] 韩伟力,陈 刚,董金祥. 面向个性化服务的虚拟设计系统[J]. 计算机集成制造系统,2001(12):13-18.

[4] 沈 璞. 虚拟现实技术在现代工业设计中的应用[J]. 制造业自动化,2004(6):76-78.

[5] 刘翠娟. 虚拟现实技术在工业产品设计中的应用[J]. 江苏煤炭,2004(2):88-89.

(上接第 159 页)

个周期而已。但是对于更大的素数,这样的计算过程就不得不由用户来设计,为了计算两个超过 64 位的整数的模,用户也许不得不采用类似于多位数除法手算过程中的试商法,这个过程不但复杂,而且消耗了很多 CPU 时间[9]。

对于现代密码算法,要求计算 128 位以上的素数的情况比比皆是,设计这样的程序迫切希望能够抛弃除法和取模。所以在密码学领域,需要找到一些更好的、效率更高、速度更快、更加安全的算法用于加密系统中。

参考文献:

[1] 陈景润. 初等数论Ⅲ[M]. 北京:科学出版社,1998:140-150.

[2] Dr Liam M. Electronic Payment &Security Systems[J/OL]. 2004-11-05. http://eee.ucc.ie/staff /marnanel/ Files/handoutee5251/lec7.pdf.

[3] vckbase. 欧几里德算法和扩展欧几里德算法[J/OL]. 2005-08-04. http://www.sssdf.com//show.jsp? categoryid=4&id=7.

[4] Ekert A. Cracking the Code-The Mathematics of Cryptanalysis[M]. Washington: The Brookings Institution Press, 2005: 100-150.

[5] 刘文江,董 威,戎蒙恬. 有限域逆元算法的实现[J]. 计算机工程,2004,30(17):184-185.

[6] 鞠宏伟,李凤银,禹继国,等. 基于 RSA 的证实数字签名方案[J]. 计算机应用研究,2006,23(1):93-95.

[7] 张亚玲,禹 勇,王晓峰,等. 基于 RSA 签名的安全数字时间戳方案[J]. 计算机应用,2005,25(2):381-382.

[8] 易大进,杨千里. 基于仿射密码原理的差分跳频频率转移函数研究[J]. 空军工程大学学报:自然科学版,2005(3):50-52.

[9] Loy J. Euclid's Algorithm[J/OL]. 2000. http://www.jimloy.com/number/euclids.htm.