广义数域筛法的理论基础及其对公钥加密算法的威胁

侯方天，杨成，张雅琨

（中国传媒大学信息工程学院，北京 100024）

摘要：广义的数域筛法（GNFS）是目前最快的整数分解方法，由于公钥加密算法（RSA）是建立在大整数很难分解的基础之上的，所以 GNFS 也是 RSA 公钥加密算法最有效的破解方法之一，09 年末就有科学家通过 NFS 成功分解了 768bit 的大整数 RSA-768，本文将结合 RSA-768 的破解过程，分析 GNFS 的分解原理和步骤。

关键词：广义的数域筛法 GNFS；公钥加密算法 RSA；二次筛法 QS

中图分类号：TP309

What is The Generalized Number Field Sieve and How can it theart to The Public Key Encryption Algorithm

Hou Fangtian, Yang Cheng, Zhang Yakun

(Information Engineering school,The Communication University of China, Beijing 100024)

Abstract: The General Number Field Sieve algorithm(GNFS) is the fastest known method for factoring large Integers. It is the reason that The RSA relies upon the fact that it is computationally difficult to factor a “large" integer into its component prime integers, the GNFS is one of the most effective ways to crack the RSA. By the end of 2009, scientists have successfully factored the 768bit large integer (RSA-768) by the number field sieve (NFS). This paper will analysis the principles and procedures of the GNFS’s decomposition, combining the cracking process of the RSA-768.

Keywords: The General Number Field Sieve(GNFS); The Public Key Encryption Algorithm (RSA); The Quadratic Sieve(QS)

0 引言

RSA 是一种非常流行的公钥加密算法，如今被应用在许多的电子商业系统中，如网站的浏览器和服务器系统，电子邮件系统，远程会议安全和电子信用卡支付系统等，它是建立在大整数很难因式分解的基础之上的，所以对大整数做因数分解的难度决定了 RSA 算法的可靠性，随着整数分解算法的不断改进和计算机运算速度的加快，人们对 RSA 系统的安全性又产生了怀疑。本文将介绍大整数分解的一种重要的算法——广义数域筛法(GN FS)

广义数域筛法（GNFS）是数域筛法（NFS）的一般形式，比较适于分解 130 位以上的大整数。NFS 其还有一种特殊的形式，被称为特殊数域筛法（SNFS）,SNFS 分解的大整数 w 要满足形式 $w=r^e\pm s$，其中 $r,e,s\in\mathbb{Z}$，并且 $e>0$，在 07 年春天就有科学家通过 SNFS 分解了 1039bit 的大整数，但由于 SNFS 对所要分解的大整数有着特殊的形式要求，它比用 GNFS 分解 768bit 的大整数在难度上要小很多。

广义数域筛法（GNFS）是一种很好的因式分解的方法，到目前为止，有效的因式分解方法还有很多，譬如试除法，Pollard’s rho 分解法，Pollard’s P-1 分解法，椭圆曲线分解法，基于完全平方的分解算法[1]，各算法的特点如表 1 所示。

作者简介：侯方天，（1987-），男，硕士研究生，主要研究方向：媒体内容安全。

通信联系人：杨成，男，副教授 系副主任，主要研究方向：媒体内容安全. E-mail: 13810539335@126.com

表 1　因式分解各算法特点

Tab. 1　The Factorization algorithm's Characteristics

算法名称	*时间复杂度*	*分解整数的大小*
试除法	$O(p(\log n)^2)$	<110bit
Pollard's rho 分解法	$O(\sqrt{p}(\log n)^2)$	<110bit
Pollard's P-1 分解法	$O(B\log B(\log n)^2)$	<110bit
椭圆曲线分解法	$O(\exp((\sqrt{2}+o(1))(\ln p)^{1/2}(\ln\ln p)^{1/2}))$	<110bit
基于完全平方的分解算法	$O(\exp((64/9)^{1/3}(\ln p)^{1/3}(\ln\ln p)^{2/3}))$	>110bit

GNFS 是一种很好的基于完全平方的分解算法，它对分解大整数的效率相对较高，本文将结合 RSA-768 的破解过程[2]，对 GNFS 的相关理论和操作步骤做出相应说明。

1　RSA 算数描述

GNFS 最主要的一个作用就是用来破解 RSA 加密信息。作为一种高效的因数分解的算法，如果其能够成功对 RSA 进行因式分解，那么用 RSA 加密的信息就会被破解，公钥加密体制将会失去作用。密钥长度决定了因数分解的难度，目前 RSA 密钥多采用 1024-bit，但随着 09 年末 RSA-768 地成功破解，2048-bit 长度密钥的使用迫在眉睫。

RSA 的公钥和私钥的产生过程如图 1 所示，其中 N 的大小一般为 1024bit，p 和 q 的大小接近于 N 大小的一半，我们把 N 称为 RSA 的模数，e 称为加密指数，d 称为解密指数，于是组合(N,e)就成为了公钥，（N,d）成为了私钥。

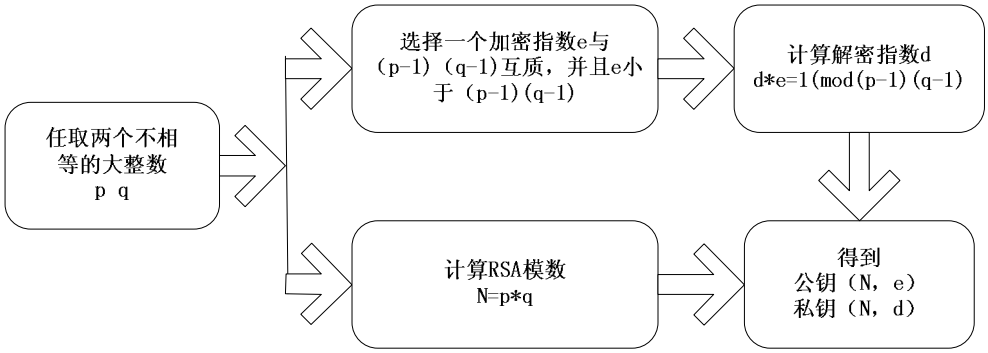

图 1　RSA 的公钥和私钥的产生过程

Fig. 1　generation process of RSA public key and private key

一个需要加密的信息 M，通过公钥(N,e)加密后，可以生成密文 C，$C = M^e \pmod N$，C 经由传输媒介传送的接受方，接收方通过私钥(N,d)，解出明文 M，$M \equiv C^d \pmod N$。举个简单的例子，当发送方选择 $N = 7\times13$和e=5，则可通过计算发现 d=29时，5×29≡1（mod72），于是公钥为（91,5），私钥为(91,29),发送的明文 M=3，加密后 $C = 3^5 \pmod{91} = 61$，解密后，$M = 61^{29} \pmod{91} = 3$。

通过对 RSA 算法理论分析，可以看出，对于一个攻击者来说，当得到公钥（N，e）后，如果能通过 N 分解出 p 和 q 的值，再由公式 $d = e^{-1} (\mathrm{mod}(p-1)(q-1))$ 就可解出私钥（N,d),GNFS 就是采用这个思路对 RSA 进行攻击的。

2　二次筛法（QS）

二次筛法与数域筛法有着相似的理论基础，为了更好的理解数域筛法，我们首先来研究

一下二次筛法。

从费马、欧拉、高斯开始一直到现在，一般整数分解方法基本上都是在“同余式的平方组合”上做文章，同时也会再加上一些现代技巧，如因数基、平滑数、筛法、线性代数等[3]。现在假定想分解的数 n，如果能够找到两个正整数 x 和 y，满足 $x^2 \equiv y^2 \pmod{n}$，其中 $0 < x < y < n, x \neq y, x + y \neq n$，则 $\gcd(x+y,n)$ 和 $\gcd(x-y,n)$ 有可能就是 n 分解的两个因数。

假设 n=77，由于 $9^2 \pmod{77} \equiv 2^2 \pmod{77}$，因此 $\gcd(9 \pm 2, 77) = (7,11)$，而 77=7·11，整数 77 分解的两个因数就为 7 和 11。不管是 QS 还是 NGFS，最后的目标就是找到足够多配对组合 (x, y)，满足等式 $x^2 \equiv y^2 (\mathrm{mod}\, n)$，从而找到分解的因数，完成分解。

QS 是一种很好的因式分解的方法，一般适合分解 130 位以下的整数，它的核心是 Dixon 的随机二次理论，该理论的核心是要通过运用因数基、平滑和二次剩余类中向量的相关性等概念计算出相应的平方值。在 QS 中，因数基为非空的素数集合，假设为 F，如果一个整数 k 的所有分解的素数都在集合 F 中，那就称该整数 k 在因数基 F 上平滑。

现在固定一个因数基 $F = \{p_1, p_2, \cdots, p_m\}$，选取一个等式 $f(x) = x^2 (\mathrm{mod}\, n)$，计算出合适的值 r_i，使得 $f(r_i)$ 在因数基 F 上平滑，当有 m 个 r_i 被找到后

$$\prod_{r_i \in U} f(r_i) = p_1^{2e_1} p_2^{2e_2} \cdots p_m^{2e_m} = (p_1^{e_1} p_2^{e_2} \cdots p_m^{e_m})^2$$

令 $x = \prod_{r_i \in U} r_i$，$y = p_1^{e_1} p_2^{e_2} \cdots p_m^{e_m}$，则可以等到

$$x^2 = \prod_{r_i \in U} (r_i^2 - n) \equiv \prod_{r_i \in U} f(r_i) \equiv y^2 (\mathrm{mod}\, n)$$

这就得到了我们所需要的配对的（x,y）。

举个简单的例子，先假设因数基 F 为（-1,2,3,5,7），$f(r_i)$ 为一些随机整数的平方，但要保证 $f(r_i)$ 在因数基 F 上平滑，取 n=33，随机取 $f(r_i)$ 的值结果如表 2 所示:

表 2 $f(r_i)$ 的取值结果

Tab. 2 $f(r_i)$’s result

$f(r_i)$	$f(r_i)-n$	$f(r_i)-n$ 的质因数	质因数的模二值
3^2	-24	-3×2^3	(1,1,1,0,0)
5^2	-8	-2^3	(1,1,0,0,0)
6^2	3	3	(0,0,1,0,0)
7^2	16	2^4	(0,0,0,0,0)

从线性代数的角度上来看，只要能在质因数模二值构成的向量中找到一个线性相关组，就能找到所需要的配对组合（x,y）。从表中可以看到，当 $f(r_i) = 7^2$ 时，向量跟自身线性相关，于是可以得到 $7^2 \equiv 2^{2^2} \equiv 4^2 (\mathrm{mod}\, 33)$，从而可以计算出 $\gcd(7 \pm 4, 33) = (3,11)$，当向量跟自身不相关时，可以通过和别的向量组合，一起组成一个线性相关组，当把 $f(r_i)$ 等于 5^2 和 6^2 的质因数的模二值相加后，向量为（1,1,1,0,0），正好与 $f(r_i)$ 等于 3^2 的向量线性相关，

于是可以得到$3^2 \equiv (5\times 6)^2 (\mathrm{mod}\, 33)$， $\gcd(30\pm 3, 33)=(3,11)$。

从以上的分析可以看出，二次筛法最主要的步骤就是通过筛选获取在因数基上平滑的数，然后通过计算找到线性相关的向量组，最后以一定概率分解大整数，广义数域筛法的主要步骤也大体如此。当然广义数域筛法也有着其自身的特点，对于那些大于 130 位的整数，广义数域筛法有着更高的分解效率，筛法中所选取的不可约多项式 $f(r_i)$ 的最高次数不再仅仅是二次，高次的多项式可以产生更多在因数基上平滑的数。在二次筛法中， $f(r_i)$ 起到了一个桥梁的作用，他把整数 $\mathbb{Z}$ 映射到了 n 次剩余类 $\mathbb{Z}/n\mathbb{Z}$，把一个在 $\mathbb{Z}$ 中的完全平方数映射到了一个在 $\mathbb{Z}/n\mathbb{Z}$ 中的完全平方数，这种映射的方法在广义数域筛中也有着重要的作用，两者的不同点在于 QS 操作的数都是整数，映射是从 $\mathbb{Z}$ 到 $\mathbb{Z}/n\mathbb{Z}$，而广义数域筛操作的数除了整数，还有环 $\mathbb{Z}[\alpha]$，映射包括了两部分，一部分是从 $\mathbb{Z}$ 到 $\mathbb{Z}/n\mathbb{Z}$，还有一部分是从 $\mathbb{Z}[\alpha]$ 到 $\mathbb{Z}/n\mathbb{Z}$，这部分映射用 $\phi(\beta)$ 来表示，这就使广义数域筛法相比二次筛法显得更加得复杂。

3 广义数域筛法（GNFS）

广义的数域筛法的分解理论和技术包括多项式选择、筛选、矩阵形成和线性相关以及平方根的求解，同二次筛法一样，广义的数域筛法也是基于“同余式的平方组合”，目的是找到更多的配对组合(x,y)，从而分解大整数，正如上文提到的，广义的数域筛法的操作的数还包括环 $\mathbb{Z}[\alpha]$， 其中 α 是多项式 $f(x)$ 的根，根的集合形成了集合 Ψ，在这个环 $\mathbb{Z}[\alpha]$ 上能产生更多的平滑，于是广义的数域筛法产生配对组合(x,y)的基础方程就有两个

$$\prod_{(a,b)\in U}(a+b\theta)=\alpha^2 \qquad \prod_{(a,b)\in U}(a+bm)=\mathrm{z}^2 .$$

其中$\alpha\in\Psi$， $z\in\mathbb{Z}$，令$\beta=f'(\theta)\cdot\alpha\in\mathbb{Z}[\theta], y=f'(m)\cdot z, x=\phi(\beta)\in\mathbb{Z}/n\mathbb{Z}$

$$x^2\equiv\phi(\beta)^2\equiv\phi(\beta^2)\equiv\phi(f'(\theta)\cdot\prod_{(a,b)\in U}(a+b\theta))$$

$$\equiv\phi(f'(\theta))^2\cdot\prod_{(a,b)\in U}(a+b\theta)\equiv f'(m)^2\cdot\prod_{(a,b)\in U}(a+bm)\equiv y^2(\mathrm{mod}\, n)$$

通过这个恒等式可以看到，广义的数域筛法要想找到合适的配对组合(x,y)，就得找到足够的配对组合(a,b)，使得$a+b\theta$ 在代数因 $\mathbb{Z}[\theta]$ 上平滑，$a+bm$ 在有理数因数基 $\mathbb{Z}$ 上平滑，然后计算出(x,y),以 50%左右的可能性得到整数 n 的分解因式。

广义的数域筛法的分解步骤如下所示：

3.1 多项式选择

多项式的选择是广义数域筛法的第一步，它对整个筛法的耗时量和筛选的复杂程度有着重要的作用。

广义数域筛法一般会选取一个不可约的 d 阶多项式 $f(\mathrm{x})$， α 为这多项式的一个复根，$\mathbb{Z}[\alpha]\cong\mathbb{Z}[x]/f(x)$，然后选取一个参数 $\mathrm{m}\in\mathbb{Z}/n\mathbb{Z}$，通过 Murphy 的多项式选取法，使得 $f(m)\equiv 0(\mathrm{mod}\, N)$， N 为所要分解的大整数，令 m 为 N 的一个分解基，则 $N=\sum_{i=0}^{\mathrm{d}} c_i m^i$,其中$c_i\in\mathbb{Z}$并且$0\le c_i\le$ m-1 ， 当 $f(x)=\sum_{i=0}^{\mathrm{d}} c_i x^i$ 时，很容易得到

$f(m)=N\equiv 0(\bmod N)$，满足等式的要求。通过映射$\phi:\mathbb{Z}[\alpha]\to\mathbb{Z}/n\mathbb{Z}$,得到$\phi(\alpha)$=m，这样就使$\mathbb{Z}[\alpha]$的值同整数有了对应关系。

在实践中，需要通过大量的实验，参数的细致调整，凭借经验，有时再加上点运气，才能寻找到一个最合适的多项式。一般来说，首先会确定多项式的次数 d，d 的大小由所要分解的大整数 N 的位数有关[4]，当分解 50~80 位的十进制整数时，会取 d 值为 3;80~110 位的十进制整数时，取 d 值为 4;120~220 为十进制整数时，取 d 值为 5；当分解 220~300 位十进制整数时，取 d 值为 6。然后确定参数 m 的值，通常取 m 值为$\sqrt[d]{N}$或其附近的某一整数，多项式的选择有很多不确定因素，多项式的次数 d 的选取，参数 m 的取值都有很强的随机性，即使确定了一组 d 和 m，多项式的系数还可以有不同的选择，只有通过实验，根据得到的合适的整数对(a,b)的数量才能确定多项式的好坏。

RSA-768 的团队通过三个月的时间，从2^{60}个多项式中选出了 3 个符合要求的质量很高的多项式，而他们最终决定采用的多项式为

$$\begin{aligned}f_1(X)=&265482057982680X^6\\&+1276509360768321888X^5\\&-5006815697800138351796828X^4\\&-46477854471727854271772677450X^3\\&+652543726193598939710966737189478 5X^2\\&-1818577935208859435672601886243480 3054X\\&-2775652667915438819952161997138011033431 20\end{aligned}$$

RSA-768 的团队同时也选取了一个一阶的多项式用来在整数环上进行计算，多项式为

$$\begin{aligned}f_2(X)=&34661003550492501851445829X\\&-1291187456580021223163547791574810881\end{aligned}$$

3.2 平滑和因数基

从二次筛法中可以看出，平滑和因数基是筛法的基础，通过找到足够多的在因数基上平滑的数，才能顺利分解大整数，广义的数域筛法也是这样，不过因为其不光在整数域上操作，还要在环$\mathbb{Z}[\alpha]$上操作，所以在环$\mathbb{Z}[\alpha]$对于平滑和因数基就有了新的定义，因数基不再是$\mathbb{Z}[\alpha]$中的素数，而是其理想。

α 是多项式的根，定义$N(\alpha)=\sigma_1(\alpha)\sigma_2(\alpha)\sigma_3(\alpha)\cdots\sigma_d(\alpha)$，其中$\sigma_i$是从$\mathbb{Q}(\theta)$到$\mathbb{C}$的映射，$D$为一绰金环，$P$为$D$的一个素理想，$p$代表一些素数，当$N(P)=$p 时，定义$P$为$D$的第一阶素理想，当$r$属于$\mathbb{Z}/p\mathbb{Z}$，并且$f(r)\equiv 0(\bmod p)$时，集合$(r,p)$同第一阶素理想的集合一一映射，理想$p$决定了组合$(r,p)$，广义数域筛法要找的因数基就是集合$(r,p)$。

广义数域筛法中定义了三个因数基，分别为有理数因数基，代数因数基和二次特征基。有理数因数基跟二次筛法中的因数基一样，都是在整数域上操作，通过对（a，b）的取值，找到在 a+bm 上平滑的值，为了和代数因数基和二次特征基的形式保持一致，在实践中

一般选取因数基的形式为$(m(\mathrm{mod}\ p), p)$。

上文提到过不可约多项式$f(\mathrm{x})$根形成的集合为○，由数论的相关知识可知，环○是一个绰金环，它的一些理想可以被分解为素理想，从○的理想中选取一个集合I，这个集合I就被称为代数因数基，然后找一对组合(a,b)，使得$a+b\theta$有一个主理想，能完全分解成I上的素理想，这样的元素$a+b\theta$就被称为在I上平滑。

二次特征基是用来确定$a+b\theta$在$\mathbb{Z}[\theta]$上是否为一个完全平方数，广义数域筛法的一个基础方程为

$$\prod_{(a,b)\in U}(a+b\theta)=\alpha^2$$

给定一个第一阶素理想Q，它决定了组合(s,q)，当Q不能分解主理想$\langle a+b\theta\rangle$,并且$f'(s)\neq 0(\mathrm{mod}\ q)$时，$\prod_{(a,b)\in U}(\frac{a+bs}{q})=1$，也就是说，如果$a+b\theta$在$\mathbb{Z}[\theta]$上是一个完全平方数，那么$\prod_{(a,b)\in U}(\frac{a+bs}{q})=1$，当然这是一个必要条件，不是说当等式$\prod_{(a,b)\in U}(\frac{a+bs}{q})=1$成立时，$a+b\theta$就一定是完全平方数，通常在实际计算的时候，q 会取比 p 大一点的值。

3.3 筛选

筛选是广义数域筛法中最重要也是最耗时的步骤之一，可以通过并行处理的方式提高效率，筛选的目的是要得到足够多的整数对(a,b)，使得$a+bm$在有理数因数基 $\mathbb{Z}$ 上平滑，$a+b\theta$在代数因数基$\mathbb{Z}[\theta]$ 上平滑，而对两者筛选的方法大体一致。

对于$a+bm$来说，m 已经在上文确定，还有两个变量 a 和 b，当固定 b 的值（通常先取 b=1），再取-u<a<u（u 是 a 的取值的范围，可视情况而定），a 从-u 开始一直取到 u，通过计算得到能使$a+bm$在因数基上平滑的组合(a,b)，当取完 a 的值发现组合还不够时，增加 b 的值直到组合够为止，这种方法需要计算每个$a+bm$的值是否平滑，很消耗时间。在实际中一般会固定一个因数基里的素数 p 和一个正整数 b，当$a+bm\equiv 0(\mathrm{mod}\ p)$，也就是$a\equiv -bm(\mathrm{mod}\ p)$的时候,p 就能分解$a+bm$，那么在筛选的时候，在-u 到 u 的范围内取满足等式$a=-bm+kp$的 a 的值就可以了，这就大大提高了效率。

$a+b\theta$的筛选是在代数因数基上进行的，是要通过第一阶素理想 P 来分解主理想$\langle a+b\theta\rangle$，而 P 可以由组合(r,p)决定，当$a\equiv -br(\mathrm{mod}\ p)$的时候，p 能分解$N(a+b\theta)$，而当 p 能分解$N(a+b\theta)$的时候，第一阶素理想 P 也就能分解主理想$\langle a+b\theta\rangle$,于是筛选的方法就是固定 b 的值和组合$(r,p)$，取$a=-br+kp$，并且-u<a<u,然后计算$N(a+b\theta)$的值，当值能被 p 分解时，$(a,b)$就是筛选出来的可用的整数对。

通过筛选，RSA-768 的团队总共获取了64334489730对合适的组合，然后通过分布式筛选理论，其中有 24.7%的组合是重复记录的，然后通过一种判别算法，删去那些包含素数（小概率出现）的组合，于是就只剩下2458248361对组合，只占原来的 3.4%。

3.4 矩阵生成及过滤

通过筛选，得到了 I 对整数对 (a,b)，接下来就要构建一个矩阵，假设有理数因数基有 k 个素数，代数因数基有 l 个第一阶素理想，二次特征基有 m 个第一阶素理想，则矩阵应该有 k+l+m+1 行，列数为 I，矩阵的每一列都由一对整数对 (a,b) 决定，构建矩阵的最终目的是寻找到矩阵列与列之间的线性相关性，从而找到不同的 $a+bm$，$\langle a+b\theta \rangle$ 和 $a+b\theta$，使他们的乘积值为完全平方数。

每一列矩阵都是由四部分组成，第一部分就一位，它表示 $a+bm$ 的正负性，正的话为 0，负的话为 1；第二部分有 k 位，有理数因数基中的素数有 k 个，$a+bm$ 由这些素数分解后素数的指数模二的值就对应了该 k 位，假设因数基为{2,3,5}，选定的整数对为（3,1），m 为 27，则 $a+bm=30=2\times3\times5$，前 k 位就为（1,1,1）；第三部分有 l 位，代数因数基中有 l 对 (r,p)，通过公式 $N(a+b\theta)=(-b)^d f(-a/b)$ 计算出 $N(a+b\theta)$ 的值，把 $N(a+b\theta)$ 分解得到 p，当 $a\equiv -br(\mathrm{mod}\ p)$ 时，记该位为 1；第四部分有 m 位，由二次特征基决定，当满足公式 $\prod_{(a,b)\in U}(\frac{a+bs}{q})=1$ 时，值为 0，否则为 1.

以上生成的是一个原始矩阵，矩阵维数和重量会直接决定线性相关求解的时间，原始矩阵显然不是最合适的矩阵，必须通过相应的过滤方法，减小矩阵的维数，同时控制重量的增长，使两者达到一个最佳平衡点，这样能大大提高分解效率。

3.5 线性相关

当最佳矩阵找到后，就得通过计算得到矩阵列向量之间的线性相关性了，这也可以理解为求解一个大规模稀疏线性方程组，用传统的高斯消元法去处理一个维数很大的矩阵时会消耗很多时间，不能被采用，目前最常用的求解的方法为 Lanczos 和 Wiedemann 算法[5]，寻找线性相关性的这一步是数域筛法中耗时最多的步骤之一，实际中通常会采用并行处理。

RSA-768 的团队最后生成了一个 192796550×192795550 的矩阵，矩阵的重量为 27797115920，也就是说每行平均有 144 个非 0 数，通过 119 天的计算，团队最终在 2009 年的十月八号完成了矩阵的计算，得到了能使矩阵列向量线性相关的组合 (a,b)。

3.6 平方根的求解

用广义数域筛法分解大整数，利用的是”同余式的平方组合”,这就需要利用 LLL 算法或是 Montgomery 的算法[6]，来求解整数的平方根，同过大规模稀疏线性方程组的求解，就能找到对应的平方关系，这些平方关系都是由几对组合 (a,b) 确定的，通过组合 (a,b) 得到 $\alpha^2=\prod_{(a,b)\in U}(a+b\theta)$，而 $\beta^2=f'(\theta)^2\cdot\alpha^2$，从而计算出 β 的含 θ 的多项式 $f_2(\theta)$，于是 $x=f_2(m)\equiv\phi(\beta)(\mathrm{mod}\ N)$；通过组合 (a,b) 得到 $\alpha^2=\prod_{(a,b)\in U}(a+b\theta)$，而 $y^2=f'(m)^2\cdot z^2$，通过平方根的求解就能得到 y 的值。

RSA-768 的团队通过 Montgomery 的算法，在很短的时间内就完成了平方根的求解，最终的分解结果为

$$
\begin{aligned}
&RSA-768\\
&=123018668453011775513049495838496272077285356959533479219732245215172640050726365751874520219978646938995647494277406384592519255732630345373154826850791702612214291346167042921431160222124047927473779408066535141959745985690214341334780716989568987860441698482126908177147949837137685689124313889828837938780022876147116525317430877378144679994893674604366679959042824463379962795263227915816434308764267603228381573966651127923337341714339681027009279873630891760322838157396665112792333734171433968102700927987363089176032283815739666511279233373417143396810270092798736308917
$$

4 结论

本文讨论了数域筛法中广义数域筛法的基本理论，从中可以看出，广义数域筛法的分解步骤是相对固定的，首先要选择一个合适的多项式，多项式质量的好坏决定了筛选的效率，然后要确定三个因数基，以平滑为标准进行筛选，筛选可以采取并行处理[7]，当筛选出有效的组合后，就可以组成一个矩阵，并对其进行处理，找到线性相关的列向量，这一步骤计算量很大，需进行并行处理，通过平方根的求解，得到组合(x,y),最终以 50%左右的可能性完成对大整数的分解。

对于广义数域筛法来说，虽然还存有很多问题需要解决，但作为目前对大整数分解最快的方法，通过硬件设备的不断改进和算法的改良，其每一分解步骤都有改进的空间，相信今后广义数域筛法能有更多的改进，从而能够更加高效的完成大整数的分解，真正对公钥加密算法 RSA 产生威胁。

[参考文献] (References)

[1] 孙克全.RSA 密码分析中分解大整数的判定算法作者[J].计算机工程，2010,36(15):142-144

[2] Thorsten Kleinjung, Kazumaro Aoki, Jens Franke, Arjen Lenstra, Emmanuel Thom, Joppe Bos, Pierrick Gaudry, Alexander Kruppa, Peter Montgomery, Dag Arne Osvik, Herman te Riele, Andrey Timofeev, and Paul Zimmermann. Factorization of a 768-bit rsa modulus. Cryptology ePrint Archive, Report 2010/006,2010. http://eprint.iacr.org/.

[3] 颜松远.整数分解[M].北京：科学出版社，2009

[4] 王洪涛，刘春雷.数域筛法中多项式的选择[J].信息工程大学学报,2003,4(3)

[5] M. E. Briggs. An introduction to the general number field sieve[D]. Master's thesis, Virginia Polytechnic Institute and State University, USA, 1998.

[6] Joshua Hill.The Number Field Sieve: An Extended Abstract.March 12, 2010

[7] Laurence Tianruo Yang, Li Xu, Jong Hyuk Park.A Parallel GNFS Algorithm with the Improved Linbox Montgomery Block Lanczos Method for Integer Factorization[J].International Conference on Information Security and Assurance,2008