An Implementation of Multiple Polynomial Quadratic Sieve

Kai Sun, 5110309061
Shunning Jiang, 5110309084

1. 引言

二次筛法是目前渐进复杂度第二低的大整数分解算法（第一是数域筛法，第三是椭圆曲线法）。在110位十进制数的范围内，重多项式二次筛法（MPQS）的运行速度是最快的。本文对二次筛法与相关的理论基础进行描述，并且对给出的实现进行简要解释。

2. 理论基础

2.1. Fermat's Algorithm

Fermat's Algorithm基于如下事实：

如果$N = a^2 - b^2 = (a+b)(a-b)$，那么当$(a+b)$与$(a-b)$都不是1时，$N = (a+b)(a-b)$是一个可行的分解式。

简单变形后有$b^2 \equiv a^2 - N (mod\, N)$，找出一对$a, b$且$a \neq b (mod\, N)$满足该式，此时$(N, a \pm b)$便是$N$的非平凡因子。

自然地，看上去可以对a^2和b^2进行进一步的工作，于是就有：

2.2. Dixon's Method

Dixon's Method是建立在Fermat's Algorithm上的：

定义素数集合$prime_B = \{p \mid p \leq B, p\ is\ a\ prime\}$。称数$a$是B-光滑的，当且仅当$a$的分解式$a = \prod_i p_i^{k_i}$满足$\forall p_i \in prime_B$。容易发现，$p \in prime_B$中的数同时需要满足条件$\left(\frac{N}{p}\right) = 1$，即$N$是模$p$的二次剩余，称满足该条件的素数集合为分解基。

若$a^2 \equiv \prod_{p_i \in prime_B} p_i^{k_i} (mod\ N)$(i.e. a^2是B-光滑的)，则我们得到一个指数向量$E_i = (k_1, k_2, \ldots, k_n)$。当有$n+1$个指数向量时，这些向量

必定线性相关。在$\mathbf{F}_2$中，这个结论便是当找到$n+1$个指数向量后

$$\exists(s_1,s_2,\ldots,s_{n+1})\ s.t.\ \sum_{i=1}^{n+1} s_iE_i \equiv \mathbf{0}(mod\ 2)$$

更一般的有

$$a_1^2a_2^2\ldots a_m^2 \equiv \prod_{p_i\in prime_B} p_i^{k_{i,1}+k_{i,2}+\ldots+k_{i,m}}(mod\ N)$$

再用Fermat's Algorithm来判定即可。

注意到此时数a_i^2与指数向量E_i是一一对应的，这启发我们通过一些别的方法找到一些B-光滑的数。

3. 二次筛法

3.1. 自行初始化二次筛法（SIQS）

SIQS事实上便是单多项式的二次筛法，是一个Dixon's Method的优化，目的还是找到一组指数向量$\mathbf{E}_i$，接着通过高斯消元求出解向量。

设$P(x)\in\mathbf{Z}[x]$，若$p|P(a)$，则$\forall k\in Z, m|P(a+kp)$，这一系列数都有因子$p$。令$P(x)=([\sqrt{N}]+x)^2-N(mod\ N)$，则$P(x)\equiv([\sqrt{N}]+x)^2(mod\ N)$，可以用来进行求解。

由B-光滑的条件，我们取出那些$p_i^{e_{i,j}}<B$的素数p_i与指数$e_{i,j}$，由二次剩余解个数的定理，当$p_i>2$时，$x^2\equiv N(mod\ p_i^{e_{i,j}})$恰有两个解$x_{i,j},y_{i,j}$。

对$x_{i,j}$($y_{i,j}$的过程相同)，令$L=x_{i,j}-[\sqrt{N}]$，此时$p_i^{e_{i,j}}|P(L)$。那么在区间I中任意与L相差$p_i^{e_{i,j}}$的倍数的数都被$p_i^{e_{i,j}}$整除，这便构成了筛法的核心。

可以发现，$P(x)$增长的比x快，所以当$P(x)$的素因子不在分解基中时，我们还需要做一些另外的工作：

3.2. 重多项式二次筛法（MPQS）

考虑使用多个多项式进行筛法，从而减少$P(x)$的大小：

令$P(x)=Ax^2+Bx+C, A>0$，配方得到

$$AP(x)=(Ax+B)^2-(B^2-AC)$$

类似SIQS，选取A,B,C使$N|B^2-AC$，从而有$AP(x)\equiv(Ax+B)^2(mod\ N)$。

牢记我们的目的，$P(x)$的极小值点为$x_m=-\frac{B}{A}$，此时$P(-\frac{B}{A})=\frac{B^2-AC}{A}$。设搜索区间$|I|=2M$，$I$的中心就定在$x_m$。极差为

$$P(x_m\pm M)-P(x_m)=AM^2-2\frac{B^2-AC}{A}$$

再由限制，$N|B^2-AC$，不妨取$N=B^2-AC$，令极差为0，得到$A=\frac{\sqrt{2N}}{M}$。

这样我们就得到了一个选取A,B,C的流程：

- 确定筛区间大小M。
- 令$A\approx\frac{\sqrt{2N}}{M}$，且$A$为素数。
- $B^2-AC=N\implies B^2\equiv N(mod\ A)$，即求二次剩余解$B$。
- $C=\frac{B^2-N}{A}$

4. 具体实现

4.1. Knuth-Schroeppel's Multiplier

Knuth在实现MPQS的时候，做了一个优化，我们在实现时也应用了这个优化：

最基本的二次筛法中的分解基满足$\forall p\in prime_B$，N是模p的二次剩余。事实上，如果将N换成$k\times N$(k是一个很小的常自然数)，把分解基也换成满足$\left(\frac{k\times N}{p}\right)=1$的那些素数，运行速度可以获得增长。

令$K=\{1,2,3,5,6,7,10,11,13,14,15,17\}$，即去掉所有4的倍数后的一些小自然数。定义估价函数$f(k,S_k)$，目标找出$k=argmax_{i\in K}f(i,S_i)$：

$$
\begin{aligned}
&S_k=\{p\mid p\ is\ a\ prime\ \left(\frac{k\times N}{p}\right)=1\}\\
&|S_k|=Constant\\
&\forall q\notin S_k,\left(\frac{k\times N}{q}\right)=1,q>\max_{p\in S_k}\{p\}\\
&p\mid K\implies distribute_p=\frac{ln(p)}{p}\\
&p\nmid K\implies distribute_p=2\times\frac{ln(p)}{p-1}\\
&f(k,S_k)=\sum_{p\in S_k}distribute_p
\end{aligned}
$$

4.2. 进行筛法

筛法实现时只需要保存一张表$Sieve[]$，如果$p_i^{e_{i,j}}|P(l)$，那么表中$Sieve[l]$就增加lnp_i，最后若$Sieve[l]$接近$lnP(l)$，那么$P(l)$就可以被完全分解。

4.3. 使用的数学库

我们使用了一个开源的数学运算库：GMP(The GNU MP Bignum Library)，程序使用了里面的一系列函数。

5. 运行速度

$N = 56951612469251461658462625796415380008766223201484908903684 3$
runtime = 38s
(result: 576812034936156067406846219857$\times$987351321051321350761324103099)
$N = 315116204677568382361696961575648011579608451627825950469156873430637$
runtime = 8min 37s
(result: 476193485719832475183247698347 6029$\times$661739847619348571983247769834 75953)

6. 参考文献

[1] http://en.wikipedia.org/wiki/Quadratic_sieve
[2] 李超，计算机代数系统，Bachelor
[3] GMP Manual，Torbjorn Granlund
[4] qsieve from miracl