MATH 467, THE QUADRATIC SIEVE (QS)

Algorithm QS. We are given an odd number n which we know to be composite and not a perfect power. The objective is to find a non–trivial factor of n. A number $m \in \mathbb{N}$ is called *B–smooth* when it has no prime factor exceeding B.

1. *Initialization.*

1.1. Pick a number B for the size of the factor base. Theory says take $B = \lceil L(n)^{1/2} \rceil$ where $L(n) = \exp(\sqrt{\log n \log\log n})$, but in practice a B somewhat smaller works well. Also, adding extra primes suggested by the sieving process can be useful and if one uses the wrinkle in 5.3 the prime p is adjoined to the factor base.

1.2. Set $p_0 = -1$, $p_1 = 2$ and find the odd primes $p_2 < p_3 < \ldots < p_K \le B$ such that $\left(\frac{n}{p_k}\right)_L = 1$. Algorithm LJ is useful here.

1.3. For $k = 2, \ldots, K$ find the solutions $\pm t_k$ to $x^2 \equiv n \pmod{p_k}$ by using algorithms QC357/8 and QC1/8 (described elsewhere).

2. *Sieving.*

2.1. Let $N = \lceil \sqrt{n} \rceil$. Sieve the sequence $x^2 - n$ for $x = N + j$, $j = 0, \pm \ldots$ until one has obtained a list of at least $K+2$ B-smooth $x_j^2 - n$ and their factorizations. This could be done by using a matrix, with B^2 rows (B^2 is somewhat arbitrary and can be increased if necessary) so that the j–th row is a $K+3$ dimensional vector in which the first entry is x_j, the second is $x_j^2 - n$, and the $k+3$–rd entry is the exponent of p_k in $x_j^2 - n$.

2.2. For each prime p_k in the factor base divide out all the prime factors p_k in each entry $x_j^2 - n$ with $x_j \equiv \pm t_k \pmod{p_k}$, recording the exponent in the $k+3$-rd entry in the associated j-th vector.

2.3. If the second entry in the j–th vector has reduced to 1, then $x_j^2 - n$ is B–smooth.

3. *Linear Algebra.*

3.1. Form a $(K+2) \times (K+1)$ matrix $\mathcal{M}$ with the rows being formed by the 3–rd through $K+3$–rd entries of the row vectors arising in 2.2, but with the entries reduced modulo 2.

3.2. Use linear algebra (Gaussian elimination, for example) to solve $\boldsymbol{\lambda}\mathcal{M} = \mathbf{0} \pmod 2$ where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_{K+2})$ is a $K+2$ dimensional vector of 0s and 1s (not all 0!).

4. *Factorization.*

4.1. Compute $x = x_1^{\lambda_1} x_2^{\lambda_2} \ldots x_{K+2}^{\lambda_{K+2}}$ modulo n and

$$y = \sqrt{(x_1^2 - n)^{\lambda_1} (x_2^2 - n)^{\lambda_2} \ldots (x_{K+2}^2 - n)^{\lambda_{K+2}}}$$

modulo n. The value of x can be computed by using the first entries in the j–vectors and the square root can be computed quickly using the factorizations in 2.2.

4.2. Compute $m = \gcd(x - y, n)$.

4.3. Return m.

5. *Aftermath.*

5.1. If m is not a proper factor of n try one or more of the following.

5.2. Extend the sieving in 2.1 to obtain more pairs. As a matter of policy the original sieving probably should be conducted so as to obtain $K + L$ pairs where L is somewhat larger than 2.

5.3. Use another polynomial in place of $x^2 - n$, or rather, be a bit more cunning about the choice of the x in 2.1. Choose a large prime p for which $b^2 - n \equiv 0 \pmod p$ is soluble, and compute b. Then $(px+b)^2 - n \equiv 0 \pmod p$ and x can be chosen so that $f(x) = ((px+b)^2 - n)/p$ is comparatively small since p is large, so the sieving proceeds relatively speedily, there is a better chance of a complete factorization of $f(x)$, and we only have to augment the factor base with the prime p.