# Finding Small Roots of Univariate Modular Equations Revisited

Nicholas Howgrave-Graham

University of Bath, nahg@maths.bath.ac.uk

**Abstract.** An alternative technique for finding small roots of univariate modular equations is described. This approach is then compared with that taken in (Coppersmith, 1996), which links the concept of the dual lattice (see (Cassels, 1971)) to the LLL algorithm (see (Lenstra *et al.*, 1982)). Timing results comparing both algorithms are given, and practical considerations are discussed. This work has direct applications to several low exponent attacks on the RSA cryptographic scheme (see (Coppersmith, 1996)).

## 1 Introduction

Let $p(x)$ be a univariate modular polynomial of degree $k$;

$$p(x) = x^k + a_{k-1}x^{k-1} + \ldots + a_1x + a_0 \pmod{N}. \qquad (1)$$

It is assumed that $p(x)$ is monic and irreducible, and that $N$ is not prime, but hard to factorise.

In this paper we describe a new method for finding all the small integer roots, $|x_0| < N^{1/k}$, of equation 1, and show the relationship between the approach taken here, and that taken in (Coppersmith, 1996). It will be proved, via a general result on dual lattices that these two algorithms are in fact equivalent, though the present approach may be preferred for computational efficiency.

It has been shown in (Coppersmith, 1996) how finding small solutions to equation 1 can lead to various attacks on the RSA cryptographic scheme when using a small encrypting exponent.

Since both approaches employ lattice basis reduction, the remainder of this section deals with the notation and technical results that will be required.

Sections 2 and 3 give expositions of the algorithms in question, together with proofs of their validity; examples of both algorithms are shown in section 4.

Section 5 proves a technical result about dual lattices with respect to the LLL algorithm, whilst section 6 shows that it is indeed this theory that links the two methods. Section 7 then discusses practical issues relating to the algorithms and gives relevant timing results.

### 1.1 Notation

For the sake of consistency, all the results stated in this paper will be with respect to the *rows* of the relevant matrices. We shall denote the $i$'th row of a matrix $M$ by $m_i$, and the $i$'th element of a vector $v$ by $v_i$.

The sum and Euclidean norm are denoted by $||v||_1 = \sum |v_i|$ and $||v||_2 = \sum v_i^2$ respectively. If no subscript is present then the norm should be taken to be Euclidean. The dot product of two vectors will be denoted $v \cdot w = \sum v_i w_i$.

The set of all $(n) \times (n)$ matrices of determinant $\pm 1$, and with integer coefficients will be denoted by $GL_n(Z)$. The symbol $M^r$ (resp. $M^c$) is used to denote a matrix $M$ that has had its rows (resp. columns) reversed.

## 1.2 Lattice reduction

For a thorough grounding on lattices see (Cassels, 1971), however for our purposes the following will suffice.

For a given basis $B = \{b_1, \ldots, b_n\}$ of $R^n$ a lattice $L$ is defined to be the set of points

$$L = \left\{ y \in R^n \,\middle|\, y = \sum_{i=1}^{n} a_i b_i,\ a_i \in Z \right\}.$$

Clearly many matrices will generate the same set of lattice points, in fact if we represent a basis $B$ by a matrix with rows $\{b_1, \ldots, b_n\}$, then it is exactly the bases $B' = HB$ where $H \in GL_n(Z)$ that generate $L$. All these matrices have the same (absolute) determinant $d(L) = |\det B|$, which is referred to as the determinant of the lattice.

The landmark paper of (Lenstra *et al.*, 1982) gives a definition of an *LLL-reduced* basis of $L$, and more importantly an effective way of computing one. The method is closely connected to the Gram-Schmidt orthogonalisation procedure which, given a basis $B$, forms an orthogonal basis $B^* = \{b_1^*, \ldots, b_n^*\}$, where

$$b_i^* = b_i - \sum_{j=1}^{i-1} \mu_{i,j} b_j^*,$$

and $\mu_{i,j} = (b_i \cdot b_j^*)/||b_j^*||^2$. Note that span $\{b_1^*, \ldots, b_i^*\}$ = span $\{b_1, \ldots, b_i\}$ for all $1 \le i \le n$, but $B^*$ is not typically a basis for $L$.

With this notation an LLL-reduced basis of $L$ is defined to be one where, if the Gram-Schmidt orthogonalisation procedure were applied to it, then the following conditions would hold.

$$|\mu_{i,j}| \le 1/2 \qquad \forall\ 1 \le j < i \le n, \tag{2}$$

$$||b_i^* + \mu_{i,i-1} b_{i-1}^*||^2 \ge (3/4)\, ||b_{i-1}^*||^2 . \tag{3}$$

Together these conditions imply that $||b_i^*||^2 \ge (1/2)||b_{i-1}^*||^2$ which enables one to prove the following essential results.

$$||b_1|| \le 2^{(n-1)/4} d(L)^{1/n} \tag{4}$$

$$||b_1|| \le 2^{(n-1)/2} ||x|| \quad \forall\, x \in L \tag{5}$$

$$||b_n^*|| \ge 2^{-(n-1)/4} d(L)^{1/n} \tag{6}$$

In fact these results do not rely on condition 2 being quite so strict, only that

$$|\mu_{i,i-1}| \leq 1/2 \quad \forall\, 1 \leq i \leq n. \tag{7}$$

For this reason we will refer to a basis that satisfies conditions 3 and 7 as being *effectively* LLL-reduced. To turn an effectively LLL reduced basis into an LLL reduced basis is very simple, and akin to one application of the Gram-Schmidt orthogonalisation procedure.

When the so called *LLL algorithm* for lattice reduction is applied to an integer basis $b_i \in Z^n$, $1 \leq i \leq n$, it has complexity $O(n^6 \log^3 R)$ where $n$ is the dimension of the basis, and $R = \max_{1\leq i\leq n}\{||b_i||^2\}$. This complexity however, is typically quite pessimistic, and faster times are often achieved in practice. If the entries of the basis are rational, then one can clear denominators before applying the LLL algorithm.

## 2 The new method

In this section we give an exposition of a new method for finding the small roots of a (monic) univariate modular equation $p(x) = 0 \pmod{N}$.

Observe that for any polynomial $r(x)$, and natural number $X$, we have the following upper bound on the absolute size of $r(x)$ in the region $|x| \leq X$.

$$\begin{aligned} |r(x)| &\leq |x^k| + |a_{k-1}x^{k-1}| + \ldots + |a_1 x| + |a_0| \\ &\leq |X^k| + |a_{k-1}X^{k-1}| + \ldots + |a_1 X| + |a_0| \quad \text{for all } |x| \leq X. \end{aligned}$$

For some integer $h \geq 2$, and natural number $X$ we define a lower triangular $(hk) \times (hk)$ matrix $M = (m_{i,j})$. The entry $m_{i,j}$ is given by $e_{i,j}X^{j-1}$, where $e_{i,j}$ is the coefficient of $x^{j-1}$ in the expression

$$q_{u,v}(x) = N^{(h-1-v)}x^u(p(x))^v, \tag{8}$$

with $v = \lfloor (i-1)/k \rfloor$, and $u = (i-1) - kv$. Notice that $q_{u,v}(x_0) = 0 \pmod{N^{h-1}}$ for all $u, v \geq 0$. All other entries of the matrix are zero, so it has determinant $X^{hk(hk-1)/2}N^{hk(h-1)/2}$.

Let $B$ be an LLL-reduced basis of the rows of $M$, and denote the first (small) row vector of $B$ by $b_1$. Equation 4 implies that

$$||b_1||_2 \leq 2^{(hk-1)/4}X^{(hk-1)/2}N^{(h-1)/2}. \tag{9}$$

Letting $b_1 = cM$ for some $c \in Z^n$ also gives

$$\begin{aligned} ||b_1||_2 &\geq \frac{1}{\sqrt{hk}}||b_1||_1 \\ &= \frac{1}{\sqrt{hk}}\left(\left|\sum_{i=1}^{hk} c_i m_{i,1}\right| + \left|\sum_{i=1}^{hk} c_i m_{i,2}\right| + \ldots + \left|\sum_{i=1}^{hk} c_{hk} m_{i,hk}\right|\right) \\ &= \frac{1}{\sqrt{hk}}\left(\left|\sum_{i=1}^{hk} c_i e_{i,1}\right| + \left|\left(\sum_{i=1}^{hk} c_i e_{i,2}\right)X\right| + \ldots + \left|\left(\sum_{i=1}^{hk} c_{hk} e_{i,hk}\right)X^{hk-1}\right|\right) \\ &\geq \frac{1}{\sqrt{hk}}\;|r(x)| \quad \text{for all } |x| \leq X, \end{aligned} \tag{10}$$

where

$$r(x) = \sum_{i=1}^{hk} c_i e_{i,1} + \left(\sum_{i=1}^{hk} c_i e_{i,2}\right) x + \ldots + \left(\sum_{i=1}^{hk} c_{hk} e_{i,hk}\right) x^{hk-1}$$
$$= c_1 \sum_{j=1}^{hk} e_{1,j} x^{j-1} + c_2 \sum_{j=1}^{hk} e_{2,j} x^{j-1} + \ldots + c_{hk} \sum_{j=1}^{hk} e_{hk,j} x^{j-1}. \qquad (11)$$

So $||b_1||$ is "almost" an upper bound for the polynomial $r(x)$ in the entire range $|x| \leq X$. Notice also that $r(x_0) = 0 \pmod{N^{h-1}}$ since each sum in equation 11 is zero modulo $N^{h-1}$.

Combining equations 9 and 10 implies that, from making the matrix $M$ with a natural number $X$, one can form a polynomial $r(x)$ that satisfies $r(x_0) = 0 \pmod{N^{h-1}}$ and

$$|r(x)| \leq \left(2^{(hk-1)/4}\sqrt{hk}\right) X^{(hk-1)/2} N^{(h-1)/2} \quad \text{for all } |x| \leq X.$$

Thus choosing

$$X = \left\lceil \left(2^{-1/2}(hk)^{-1/(hk-1)}\right) N^{(h-1)/(hk-1)} \right\rceil - 1 \qquad (12)$$

shows that one can form a polynomial $r(x)$ such that $r(x_0) = 0 \pmod{N^{h-1}}$ and $|r(x)| < N^{h-1}$ for all $|x| \leq X$. This implies that $r(x_0) = 0$ over the integers as well, for any $x_0$ such that $|x_0| \leq X$, and $p(x_0) = 0 \pmod{N}$. Solving this univariate equation over the integers can be done in polynomial time (for instance by Hensel lifting the linear factors, or by finding small factors of the trailing coefficient), and then one can test each solution to see if it satisfies $p(x_0) = 0 \pmod{N}$. Notice that the bound $X \to 2^{-1/2}N^{1/k}$ as $h \to \infty$.

The polynomial $r(x)$ can be formed from equation 11 or the coefficients may be obtained by dividing the entries of the vector $b_1$ by appropriate powers of $X$.

## 3 A review of Coppersmith's method

Below we outline the approach given in (Coppersmith, 1996) for finding small roots of univariate modular equations. One firstly chooses a natural number $X$, and forms the upper triangular $(2hk - k) \times (2hk - k)$ matrix

$$M = \left(\begin{array}{c|c} D & A \\ \hline O_{hk} & D' \end{array}\right),$$

where $D = (d_{i,j})$ is an $(hk \times hk)$ diagonal matrix with entries $d_{i,i} = X^{1-i}$, $A = (a_{i,j})$ is an $(hk \times (h-1)k)$ matrix, where the entry $a_{i,j}$ is the coefficient of $x^i$ in the expression $x^u((p(x))^v$, with $v = \lfloor (k+j-1)/k \rfloor$, and $u = (j-1)-k(v-1)$, and $D' = (d'_{i,j})$ is an $((h-1)k \times (h-1)k)$ diagonal matrix with entries $d'_{i,i} = N^v$ where $v = \lfloor (k+i-1)/k \rfloor$. This matrix has determinant $N^{hk(h-1)/2}X^{-hk(hk-1)/2}$.

Since there is a triangular sub-matrix of $A$ with 1's on the diagonal it is possible to transform the matrix $M$ (using integral elementary row operations implied by a matrix $H_1 \in GL_n(Z)$), to

$$\tilde{M} = H_1 M = \left(\begin{array}{c|c} \widehat{M} & 0_{(hk\times(h-1)k)} \\ \hline A' & I_{(h-1)k} \end{array}\right).$$

This means that the absolute value of the determinant of both $\tilde{M}$ and $\hat{M}$ are the same as $M$. We then reduce $\hat{M}$ using lattice basis reduction to give a matrix $B = H_2\hat{M}$. If $B^*$ (with row vectors $b_i^*$) denotes this basis after Gram-Schmidt orthogonalisation, then equation 6 implies

$$||b_{hk}^*|| \geq 2^{-(hk-1)/4}N^{(h-1)/2}X^{-(hk-1)/2}. \tag{13}$$

Assume that there exists an $x_0$ such that $p(x_0) = 0 \pmod{N}$ and let $y_0 = p(x_0)/N \in Z$. Define the following vector of length $(2hk-k)$,

$$c_0 = \left(1, x_0, \ldots, x_0^{hk-1}, -y_0, -y_0x_0, \ldots, -y_0x_0^{k-1}, -y_0^2, -y_0^2x_0, \ldots, -y_0^{h-1}x_0^{k-1}\right).$$

Further, when given a vector $v$ of length $(2hk-k)$ that has 0's for the last $(h-1)k$ entries, then denote by $[v]_{\mathrm{sh}}$ this vector "shortened" to just the first $(hk)$ elements.

Assuming that $|x_0| \leq X$, the above implies

$$\begin{aligned}
\sqrt{hk} &\geq \left|\left|\left(1, x_0/X, \ldots (x_0/X)^{hk-1}, 0, \ldots, 0\right)\right|\right| \\
&= ||c_0M|| \\
&= ||c_0H_1^{-1}\tilde{M}|| \\
&= ||[c_0H_1^{-1}]_{\mathrm{sh}}\hat{M}|| \quad \text{since } (c_0H_1^{-1})_i = 0 \text{ for } i > hk \\
&= ||[c_0H_1^{-1}]_{\mathrm{sh}}H_2^{-1}B|| \\
&= ||c'B|| \quad \text{where } c' = [c_0H_1^{-1}]_{\mathrm{sh}}H_2^{-1} \\
&= ||c''B^*|| \quad \text{for some } c'' \in R^{hk} \\
&\geq ||c''_{hk}b^*_{hk}|| \\
&= ||c'_{hk}b^*_{hk}|| \quad \text{since } c''_{hk} = c'_{hk} \in Z \\
&= |c'_{hk}|\ ||b^*_{hk}|| \\
&\geq |c'_{hk}|2^{-(hk-1)/4}N^{(h-1)/2}X^{-(hk-1)/2},
\end{aligned} \tag{14}$$

which means, since $c'_{hk} \in Z$, that $c'_{hk} = 0$ for any

$$X < \left(2^{-1/2}(hk)^{-1/(hk-1)}\right)N^{(h-1)/(hk-1)}. \tag{15}$$

If instead of $c_0$ we consider the variable vector

$$\begin{aligned}
c(x) = \big(1, x, \ldots, x^{hk-1}, -p(x)/N, -xp(x)/N, \ldots, -x^{k-1}p(x)/N, \\
-(p(x)/N)^2, -x(p(x)/N)^2, \ldots, -x^{k-1}(p(x)/N)^{h-1}\big),
\end{aligned} \tag{16}$$

which satisfies $c(x_0) = c_0$, then $c'_{hk}(x)$ is a univariate polynomial given by

$$c'_{hk}(x) = [c(x)H_1^{-1}]_{\text{sh}} \cdot ((H_2^{-1})^t)_{hk}. \tag{17}$$

This has integer coefficients after multiplying through by $N^{h-1}$, and with $X$ chosen as large as possible (from equation 15), this polynomial must satisfy $c'_{hk}(x_0) = 0$ for any $|x_0| < X$.

The polynomial $c'_{hk}(x)$ is not identically zero since it is the sum of integer multiples of polynomials of differing degrees, and not all these multiples can be zero otherwise $H_2$ would be singular.

# 4 Examples

We examine the approach used by both methods to solve the equation $p(x) = x^2 + 14x + 19 = 0 \pmod{35}$ with $h = 3$ (thus we are guaranteed of finding any solutions of absolute size at most $X = 2$). Actually this polynomial has a solution $x_0 = 3$, but as we will see the methods still find it even though $x_0 > X$. It is often the case that the theoretical $X$ given in the previous two sections is a little pessimistic.

## 4.1 Coppersmith's method

Coppersmith's method would firstly form the (10) × (10) matrix below.

$$M = \left(\begin{array}{cccccccccc}
1 & & & 0 & & & 19 & 0 & 361 & 0 \\
 & 2^{-1} & & & & & 14 & 19 & 532 & 361 \\
 & & 2^{-2} & & & & 1 & 14 & 234 & 532 \\
 & & & 2^{-3} & & & & 1 & 28 & 234 \\
 & & & & 2^{-4} & & & & 1 & 28 \\
 & & & & & 2^{-5} & & & & 1 \\
 & & & & & & 35 & & & \\
 & & 0 & & & & & 35 & & \\
 & & & & & & & & 1225 & \\
 & & & & & & & & & 1225
\end{array}\right)$$

This is transformed (using elementary row operations) to $\tilde{M} = H_1 M$ given below.

$$\left(\begin{array}{cccccccccc}
1 & 0 & -19 \times 2^{-2} & 266 \times 2^{-3} & -3363 \times 2^{-4} & 42028 \times 2^{-5} & & & & \\
 & 2^{-1} & -14 \times 2^{-2} & 177 \times 2^{-3} & -2212 \times 2^{-4} & 27605 \times 2^{-5} & & & & \\
 & & -35 \times 2^{-2} & 490 \times 2^{-3} & -5530 \times 2^{-4} & 58800 \times 2^{-5} & & 0 & & \\
 & & & -35 \times 2^{-3} & 980 \times 2^{-4} & -19250 \times 2^{-5} & & & & \\
 & 0 & & & -1225 \times 2^{-4} & 34300 \times 2^{-5} & & & & \\
 & & & & & -1225 \times 2^{-5} & & & & \\
 & & 2^{-2} & -14 \times 2^{-3} & 158 \times 2^{-4} & -1680 \times 2^{-5} & 1 & & & \\
 & & & 2^{-3} & -28 \times 2^{-4} & 550 \times 2^{-5} & & 1 & & \\
 & 0 & & & 2^{-4} & -28 \times 2^{-5} & & & 1 & \\
 & & & & & 2^{-5} & & & & 1
\end{array}\right)$$

We then examine (after clearing denominators and swapping the rows and columns for efficiency), the top left (6) × (6) sub-matrix below.

$$\hat{M} = \begin{pmatrix} -1225 & & & & & 0 \\ 34300 & -1225 \times 2 & & & & \\ -19250 & 980 \times 2 & -35 \times 2^2 & & & \\ 58800 & -5530 \times 2 & 490 \times 2^2 & -35 \times 2^3 & & \\ 27605 & -2212 \times 2 & 177 \times 2^2 & -14 \times 2^3 & 2^4 & \\ 42028 & -3363 \times 2 & 266 \times 2^2 & -19 \times 2^3 & 0 & 2^5 \end{pmatrix}$$

This is LLL reduced to $B_2 = H_2 \hat{M}$ where (the inverses of) the relevant matrices are given below.

$$\begin{array}{cc} H_1^{-1} & H_2^{-1} \\ \begin{pmatrix} 1 & & & & & & 19 & 0 & 361 & 0 \\ & 1 & & & & & 14 & 19 & 532 & 361 \\ & & & & & & 1 & 14 & 234 & 532 \\ & & & & & & & 1 & 28 & 234 \\ & & & & & & & & 1 & 28 \\ & & & & & & & & & 1 \\ & & 1 & & & & 35 & & & \\ & & & 1 & & & & 35 & & \\ & & & & 1 & & & & 1225 & \\ & & & & & 1 & & & & 1225 \end{pmatrix} & \begin{pmatrix} -5 & 4 & -2 & 1 & -1 & -2 \\ 138 & -109 & 56 & -18 & 31 & 57 \\ -77 & 60 & -32 & 8 & -18 & -32 \\ 231 & -171 & 104 & -7 & 59 & 98 \\ 109 & -82 & 48 & -6 & 27 & 46 \\ 166 & -125 & 73 & -9 & 41 & 70 \end{pmatrix} \end{array}$$

The vector $[c(x)H_1^{-1}]_{\text{sh}}$ has (as is typical) the following form

$$\left(1, x, \frac{-p(x)}{35}, \frac{-xp(x)}{35}, \frac{-p^2(x)}{1225}, \frac{-xp^2(x)}{1225}\right),$$

and so taking the dot product of $[c(x)H_1^{-1}]_{\text{sh}}$ with the last column of $H_2^{-1}$ (and then multiplying by 1225) gives the polynomial

$$r(x) = 2x^5 - x^4 - 8x^3 - 24x^2 + 8x + 3,$$

which evaluates to zero over the integers at the root of $(p(x) \pmod{35})$, $x_0 = 3$.

## 4.2 The new method

The approach given in section 2 would immediately form the (6) × (6) matrix below.

$$M = \begin{pmatrix} 1225 & & & & & 0 \\ 0 & 1225 \times 2 & & & & \\ 665 & 490 \times 2 & 35 \times 2^2 & & & \\ 0 & 665 \times 2 & 490 \times 2^2 & 35 \times 2^3 & & \\ 361 & 532 \times 2 & 234 \times 2^2 & 28 \times 2^3 & 2^4 & \\ 0 & 361 \times 2 & 532 \times 2^2 & 234 \times 2^3 & 28 \times 2^4 & 2^5 \end{pmatrix}$$

This is then LLL reduced to

$$B = \begin{pmatrix} 3 & 8\times 2 & -24\times 2^2 & -8\times 2^3 & -1\times 2^4 & 2\times 2^5 \\ 49 & 50\times 2 & 0 & 20\times 2^3 & 0 & 2\times 2^5 \\ 115 & -83\times 2 & 4\times 2^2 & 13\times 2^3 & 6\times 2^4 & 2\times 2^5 \\ 61 & 16\times 2 & 37\times 2^2 & -16\times 2^3 & 3\times 2^4 & 4\times 2^5 \\ 21 & -37\times 2 & -14\times 2^2 & 2\times 2^3 & 14\times 2^4 & -4\times 2^5 \\ -201 & 4\times 2 & 33\times 2^2 & -4\times 2^3 & -3\times 2^4 & 1\times 2^5 \end{pmatrix},$$

where $B = HM$, and

$$H = \begin{pmatrix} 70 & 46 & -98 & 32 & -57 & 2 \\ 73 & 48 & -104 & 32 & -56 & 2 \\ 55 & 36 & -74 & 27 & -50 & 2 \\ 125 & 82 & -171 & 60 & -109 & 4 \\ -175 & -115 & 254 & -74 & 126 & -4 \\ 41 & 27 & -59 & 18 & -31 & 1 \end{pmatrix}.$$

The polynomial relationship required can be obtained by dividing the entries of $b_1$ by $1, 2, \ldots 2^5$; this gives the polynomial $r(x) = 2x^5 - x^4 - 8x^3 - 24x^2 + 8x + 3$. Alternatively one may form the sum

$$r(x) = \alpha_1 N^2 + \alpha_2 N^2 x + \alpha_3 N p(x) + \alpha_4 N x p(x) + \alpha_5 p^2(x) + \alpha_6 x p^2(x),$$

where the $\alpha_i$ are the elements of $h_1$.

This new method may be thought of as "flattening" the polynomial $p(x)$ around the origin, making it continuous in this region even modulo $N^{h-1}$.

## 5 The dual lattice and LLL

The dual (or polar) lattice, as given in (Cassels, 1971), is defined as the following.

**Definition 5.1** *If $\{b_1, \ldots, b_n\}$ is a basis for a lattice $L$, then there do exist orthogonal vectors $\{d_1, \ldots, d_n\}$ such that*

$$d_j \cdot b_i = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ otherwise.} \end{cases} \tag{18}$$

*The lattice which is spanned by $\{d_1, \ldots, d_n\}$ is called the* dual *lattice of $L$.*

In terms of matrices, if the rows of $B$ form a basis for a lattice $L$, then the rows of $(B^{-1})^t$ form a basis (the *dual* basis) for the dual lattice of $L$. In (Cassels, 1971) the notation $L^*$ and $B^*$ are used for the dual lattice and basis respectively, however to avoid confusion with the Gram-Schmidt procedure we shall adopt the notation $L^{-t}$ and $B^{-t}$ for these concepts. Notice $B^{-t} = (B^{-1})^t = (B^t)^{-1}$. We now give a theorem linking the dual lattice and the LLL algorithm.

**Theorem 5.2** *Let the rows of an $(n) \times (n)$ matrix $A$ form a basis for a lattice $L$, and let $B$ be an effectively LLL reduced basis for this lattice. Further let $A^{-t}$ denote the dual basis, and $A^r$ denote the matrix $A$ with it's rows reversed. Then the rows of the matrix $D = (B^{-t})^r$ form an effectively LLL reduced basis for the dual lattice $L^{-t}$ generated by the rows of $A^{-t}$.*

*Moreover, if $\{b_1, \ldots, b_n\}$ and $\{d_1, \ldots, d_n\}$ denote the rows of $B$ and $D$ respectively, then the following relationships hold for all $1 \leq i \leq n$;*

$$b_i^* = \frac{d_{n+1-i}^*}{||d_{n+1-i}^*||^2}, \tag{19}$$

*and*

$$\frac{b_i \cdot b_{i-1}^*}{||b_{i-1}^*||^2} = \frac{d_{n+2-i} \cdot d_{n+1-i}^*}{||d_{n+1-i}^*||^2}. \tag{20}$$

*Proof.* It is relatively easy to see that $D$ is a basis for $L^{-t}$, since $(H^{-t})^r \in GL_n(Z)$. To show that it is effectively LLL reduced consider the definition of the dual lattice;

$$b_i \cdot d_j = \begin{cases} 1 \text{ if } i + j = n + 1, \\ 0 \text{ otherwise.} \end{cases}$$

By induction on $j$ we have $b_i \cdot d_j^* = 0$ for all $j \leq n - i$, and $b_i \cdot d_{n+1-i} = 1$. Further, since $b_1 = b_1^* = \sum \alpha_{1,i} d_i^*$ with $\alpha_{1,i} = (b_1 \cdot d_i^*)/||d_i^*||^2$ this gives $b_1^* = d_n^*/||d_n^*||^2$.

Now assume $b_i^* = d_{n+1-i}^*/||d_{n+1-i}^*||^2$ and induct on $i$. Thus we write $b_{i+1}^* = \sum \alpha_{i+1,j} d_j^*$ where

$$\begin{aligned} ||d_j^*||^2 \alpha_{i+1,j} &= b_{i+1}^* \cdot d_j^* \\ &= \left( b_{i+1} - \sum_{k=1}^{i} \mu_{i+1,k} b_k^* \right) \cdot d_j^* \\ &= b_{i+1} \cdot d_j^* - \sum_{k=1}^{i} \mu_{i+1,k} b_k^* \cdot d_j^*. \end{aligned}$$

If $j < n - i$ then both terms on the right hand side are 0, so $\alpha_{i+1,j} = 0$. If $j = n - i$ then $b_{i+1} \cdot d_j^* = 1$ and the terms in the sum are 0, so $\alpha_{i+1,n-i} = 1/||d_{n-i}||^2$. Finally if $j > n - i$ then $d_j^* = ||d_j^*|| b_{n+1-j}^*$ by the inductive hypothesis (since $(n + 1 - j) \leq i$) which implies $\alpha_{i+1,j} = 0$. Thus only $\alpha_{i+1,n-i}$ is non-zero, and so equation 19 is true.

With this result we have

$$\frac{d_{n+2-i} \cdot d_{n+1-i}^*}{||d_{n+1-i}^*||^2} = d_{n+2-i} \cdot b_i^* = d_{n+2-i}^* \cdot b_i^* = d_{n+2-i}^* \cdot b_i = \frac{b_i \cdot b_{i-1}^*}{||b_{i-1}^*||^2},$$

which shows equation 20 is valid, and hence equation 7 holds for the basis $D$ of $L'$, assuming $B$ is itself effectively LLL reduced.

Finally to show equation 3 also holds for the basis $D$ of $L'$ when $B$ is effectively LLL reduced, observe that this condition is equivalent to

$$||b_i^*||^2 \geq \left(\frac{3}{4} - \left(\frac{b_i \cdot b_{i-1}^*}{||b_{i-1}^*||^2}\right)^2\right) ||b_{i-1}^*||^2,$$

which implies

$$\frac{1}{||d_{n+1-i}^*||^2} \geq \left(\frac{3}{4} - \left(\frac{d_{n+2-i} \cdot d_{n+1-i}^*}{||d_{n+1-i}^*||^2}\right)^2\right) \frac{1}{||d_{n+2-i}^*||^2},$$

$$||d_{n+2-i}^*||^2 \geq \left(\frac{3}{4} - \left(\frac{d_{n+2-i} \cdot d_{n+1-i}^*}{||d_{n+1-i}^*||^2}\right)^2\right) ||d_{n+1-i}^*||^2$$

as required.

This theorem implies that a vector satisfying condition 6 can alternatively be found by LLL-reducing the dual basis.

**Corollary 5.3** *Let the rows of a matrix $C$ form a basis for a lattice $L'$. A vector $d_n^*$ such that $||d_n^*|| \geq 2^{-(n-1)/4}|\det C|^{1/n}$ for some basis $D$ of $L'$ can be found by LLL reducing the matrix $C^{-t}$.*

*Proof.* Let $A = C^{-t}$, and LLL-reduce this to form the matrix $B$. From theorem 5.2 we know that $D = (B^{-t})^r$ is an effectively LLL reduced basis for $C$ (where $d_n^* = b_1/||b_1||^2$), so condition 4 implies $||d_n^*||^2 \geq 2^{-(n-1)/4}|\det C|^{1/n}$.

If, as in the method in section 3, it is not explicitly the vector $d_n^*$ that is required but a coefficient $\gamma$ such that $||vC|| \geq |\gamma|\ 2^{-(n-1)/4}d(L')^{1/n}$, then the following corollary is more useful.

**Corollary 5.4** *Given a basis $C$ of a lattice $L'$ and a vector $v \in Z^n$, one can find a constant $\gamma \in Z$ such that*

$$||vC|| \geq |\gamma|\ 2^{-(n-1)/4}d(L')^{1/n}, \tag{21}$$

*by LLL reducing the matrix $C^{-t}$.*

*Proof.* As the theory in section 3 shows, the normal way to find such a $\gamma$ is to form an LLL reduced basis D from the initial basis $C$, and then $\gamma = (v(H')^{-1})_n$ will satisfy equation 21, where $D = H'C$.

Instead if we LLL reduce $A = C^{-t}$ to form a basis $B$, where $B = HA$, and $H$ has rows $\{h_1, \ldots h_n\}$, then

$$||vC|| = ||vA^{-t}|| = ||vH^tB^{-t}|| = \left|\left|v\left(H^t\right)^c\left(B^{-t}\right)^r\right|\right|,$$

where $(H^t)^c$ is $H^t$ with its *columns* reversed, and we know $D = (B^{-t})^r$ is an effectively LLL-reduced basis for $L'$. Thus

$$\begin{aligned} ||vC|| &\geq \left|\left|\left(v\left(H^t\right)^c\right)_n d_n^*\right|\right| \\ &\geq |\gamma|\ 2^{-(n-1)/4}d(L')^{1/n}, \end{aligned}$$

where $\gamma = (v(H^t)^c)_n = v \cdot h_1$.

This theory suggests that if the LLL algorithm is being used for a purpose concerning a large vector $d_n^*$, it may be better to reduce the dual lattice searching for a small vector $b_1$. The advantage of this is that the LLL algorithm (since it works its way up through the vectors) can have an "early exit" when it has found a small enough $b_1$, rather than reducing the whole basis to find a large $d_n^*$.

## 6 The connection between the methods

We must actually show that it is the theory in section 5 that links the lattices produced by the two methods given in sections 2 and 3.

Define the $(hk) \times (hk)$ matrix,

$$E = \left( \begin{matrix} I_k \\ 0_{(h-1)k} \end{matrix} \middle| A \right),$$

where $A$ is defined as in section 3. By the process also given in section 3, the matrix $\hat{M}$ is actually $PE^{-1}Q$, where $P = (p_{i,j})$ is diagonal and has entries $p_{i,\imath} = -N^v$ $(v = \lfloor (j-1)/k \rfloor)$, and $Q = \text{diag}\ \{1, X^{-1}, \ldots, X^{-(hk-1)}\}$.

This implies that $\hat{M}^{-t} = P^{-t}E^tQ^{-t}$, with $P^{-t} = (p'_{i,j})$ diagonal and such that $p'_{i,i} = -N^{-v}$, and $Q^{-t} = \text{diag}\ \{1, X, \ldots, X^{hk-1}\}$. After clearing denominators we verify that $\hat{M}^{-t} = M'$, where $M'$ is the matrix formed by the method given in section 2.

## 7 Practical implementations and results

There are (at least) two relatively small improvements that can be made to the algorithm given in section 2. Firstly it can be shown that the column corresponding to the linear terms (i.e. the left hand one) can be removed because of it's small contribution to $||b_1||$. Secondly the following lemma can be utilised to increase the permissible bound $X$ slightly, by including rows corresponding to different polynomials.

**Lemma 1.** *Given a polynomial $p(x)$ modulo $N$ of degree $k$, and provided that $N$ and $k!$ are coprime, then one can produce a polynomial $q(x)$ modulo $(k!)N$ also of degree $k$, which shares the same roots as $p(x)$.*

*Proof.* Simply change each coefficient of $p(x)$ by multiples of $N$ (using the Euclidean algorithm) until the resulting polynomial is congruent to $\prod_{i=1}^{k}(x-i)$ (mod $k!$), and then apply the Chinese remainder theorem.

Implementations of both algorithms have been written in `C` using `Gnu MP` as a multi-precision integer package (source code available on request). The timing results are from runs on a SGI Indy with one 100MHz IP22 processor (further results are also available on request). The main part of the program was an efficient implementation of the integral LLL algorithm, details of which may be found in (Cohen, 1991).

It should be stated that the algorithms only find solutions of univariate modular equations up to $O(N^{1/k})$, and that the time to find these solutions is of complexity $t = O(h^9 k^6 \log^3 N)$. Therefore as the degree of the polynomial $k$ increases, less possible solutions are checked in greater time, i.e. the method becomes increasingly bad compared to a brute force search.

When $h = 3$ we find solutions up to $O(N^{2/(3k-1)})$. In this case average times (in seconds) for polynomials of degree $k$ and various $N$ are shown below on the left. On the right we give average times for cubic polynomials, but varying $h$ and $N$.

| $k\backslash \log_{10}(N)$ | 50 | 100 | 120 | 200 |
|---|---|---|---|---|
| 2 | 0.68 | 3.3 | 5.5 | 19 |
| 3 | 8.4 | 52 | 83 | 320 |
| 4 | 47 | 290 | 470 | 1900 |
| 5 | 170 | 1100 | 1900 | – |

| $h\backslash \log_{10}(N)$ | 50 | 80 | 150 | 200 |
|---|---|---|---|---|
| 2 | 0.29 | 0.80 | 3.4 | 7.3 |
| 3 | 8.4 | 28 | 150 | 320 |
| 4 | 89 | 320 | 1700 | – |
| 5 | 560 | 1900 | – | – |

Comparing the above tables shows (as expected) that the algorithm is far more sensitive to an increase in $h$ than one in $k$. Furthermore since $X \to O(N^{1/k})$ as $h \to \infty$ (i.e. $t \to \infty$), there must be a compromise as to which $h$ to use to maximise the number of $X$ checked per unit time. For cubic polynomials modulo $N = 10^{50}$ this turns out to be at $h = 4$.

The effect of using the dual approach as opposed to that taken in (Coppersmith, 1996) gave a saving of about 5%, when the root $x_0$ was approximately as large as the bound $X$; it is thought that this saving can be attributed, in part, to the fact that the theoretical $X$ is actually a little pessimistic. In the cases that $x_0$ was significantly smaller than $X$, the dual approach became increasingly preferable.

## 8 Acknowledgements

I would like to thank Professor James Davenport for his supervision and his help in the production of this report. Thanks also go to Dr. Richard Pinch for pointing me to references on the dual lattice.

## References

Cassels, J. W. S. 1971. *An introduction to the geometry of numbers.* Springer.

Cohen, H. 1991. *A Course in Computational Algebraic Number Theory.* Springer-Verlag.

Coppersmith, D. 1996. Finding a small root of a univariate modular equation. *In: Proceedings of Eurocrypt 96.*

Lenstra, A. K., Lenstra, H. W., & Lovasz, L. 1982. Factoring polynomials with integer coefficients. *Mathematische Annalen*, **261**, 513–534.