Finding a Small Root of a Univariate Modular Equation

Don Coppersmith

IBM Research
T.J. Watson Research Center
Yorktown Heights, NY 10598, USA

Abstract. We show how to solve a polynomial equation (mod N) of degree k in a single variable x, as long as there is a solution smaller than $N^{1/k}$. We give two applications to RSA encryption with exponent 3. First, knowledge of all the ciphertext and 2/3 of the plaintext bits for a single message reveals that message. Second, if messages are padded with truly random padding and then encrypted with an exponent 3, then two encryptions of the same message (with different padding) will reveal the message, as long as the padding is less than 1/9 of the length of N. With several encryptions, another technique can (heuristically) tolerate padding up to about 1/6 of the length of N.

1 Introduction

Let N be a large composite integer of unknown factorization. Let

$$p(x) = x^k + a_{k-1}x^{k-1} + \ldots + a_2x^2 + a_1x + a_0$$

be an integer polynomial of degree k in a single variable x, which we may assume to be monic. Suppose there is an integer solution x_0 to

$$p(x_0) = 0 \pmod N$$

satisfying

$$|x_0| < N^{1/k} \ .$$

We show how to find such a solution x_0, using lattice basis reduction techniques, in time polynomial in $\log N$ and k.

An immediate application is to RSA encryption of stereotyped messages with small exponents. If we know the high order $\frac{2}{3}\log_2(N)$ bits B of a plaintext, and the ciphertext c resulting from RSA encryption with exponent 3, then we can recover the unknown bits x_0 of the plaintext by solving the equation $p(x) = (B+x)^3 - c = 0 \pmod N$.

Another important application is to RSA encryption with small exponents and random padding. Suppose a message m is padded with a random value t before encryption with a small exponent such as $e = 3$, so that the ciphertext is

$$c = (m+t)^3 \pmod N \ .$$

Ueli Maurer (Ed.): EUROCRYPT '96 Proceedings, LNCS 1070
© Springer-Verlag Berlin Heidelberg 1996

Suppose it happens that a single message is encrypted twice, using different values of the random padding:

$$c_1 = (m + t_1)^3 \pmod N \ ,$$

$$c_2 = (m + t_2)^3 \pmod N \ .$$

From these two ciphertexts we can recover an equation of degree 9 in the quantity $t_2 - t_1$ (using the resultant), and if t_1 and t_2 are small – less than 1/9 of the length of N – then we can solve that equation for $t_2 - t_1$. Then, using techniques developed by Franklin and Reiter [2], we recover the original message $m + t_1$.

This can be viewed as a warning that, when using RSA with small exponents, the use of random padding might not be helpful and might even be dangerous.

2 Solving a univariate polynomial

We show first how to find solutions x_0 to $p(x) = 0 \pmod N$ satisfying the tighter restriction $|x_0| < \frac{1}{2}N^{(1/k)-\epsilon}$ in time polynomial in $\log N$, k and $1/\epsilon$. Then by setting $\epsilon = 1/\log N$ and exhaustively searching the few unknown high bits of x_0 we can extend the range to $|x_0| < N^{1/k}$.

Begin by selecting an integer $h \geq \max\{7/k, \ (k + \epsilon k - 1)/(\epsilon k^2)\} \approx 1/(k\epsilon)$ so that

$$h - 1 \geq (hk - 1)\left(\frac{1}{k} - \epsilon\right) \text{ and } hk \geq 7 \ .$$

For each pair of integers i, j satisfying $0 \leq i < k$, $1 \leq j < h$, we set

$$q_{ij}(x) = x^i p(x)^j$$

and remark that, for the desired solution x_0, we know

$$q_{ij}(x_0) = 0 \pmod{N^j} \ .$$

Indeed, setting

$$y_0 = \frac{p(x_0)}{N}$$

and noting that y_0 is an integer, we see that

$$q_{ij}(x_0) = x_0^i y_0^j N^j \ .$$

We build a rational matrix M of size $(2hk - k) \times (2hk - k)$, using the coefficients of the polynomials $q_{ij}(x)$, in such a way that an integer linear combination of the rows of M corresponding to powers of x_0 will give a vector with relatively small Euclidean norm. Further, all such short vectors will satisfy a certain linear relation which we will discover by lattice basis reduction techniques [3]; this relation will translate to a polynomial relation on x_0 over $\mathbb{Z}$ (not mod N), which we can solve over $\mathbb{Z}$ to discover x_0.

The matrix M is broken into blocks. The upper right block, of size $(hk) \times (hk - k)$, has rows indexed by the integer g with $0 \leq g < hk$, and columns indexed by $\gamma(i,j) = hk + i + (j-1)k$ with $0 \le i < k$ and $1 \le j < h$, so that $hk \le \gamma(i,j) < 2hk - k$. The entry at $[g, \gamma(i,j)]$ is the coefficient of x^g in the polynomial $q_{ij}(x)$.

The lower right $(hk-k) \times (hk-k)$ block is a diagonal matrix, with the value N^j in each column $\gamma(i,j)$.

The upper left $(hk) \times (hk)$ block is a diagonal matrix, whose value in row g is a rational approximation to $X^{-g}/\sqrt{hk}$, where $X = \frac{1}{2}N^{(1/k)-\epsilon}$ is an upper bound to the solutions $|x_0|$ of interest.

The lower left $(hk-k) \times (hk)$ block is zero.

We illustrate the matrix M in the case $h = 3$, $k = 2$. (For this illustration we ignore the condition $hk \ge 7$.) Assume that $p(x) = x^2 + ax + b$ and $p(x)^2 = x^4 + cx^3 + dx^2 + ex + f$. For simplicity we write δ instead of $1/\sqrt{hk}$.

$$M = \begin{bmatrix} \delta & 0 & 0 & 0 & 0 & 0 & b & 0 & f & 0 \\ 0 & \delta X^{-1} & 0 & 0 & 0 & 0 & a & b & e & f \\ 0 & 0 & \delta X^{-2} & 0 & 0 & 0 & 1 & a & d & e \\ 0 & 0 & 0 & \delta X^{-3} & 0 & 0 & 0 & 1 & c & d \\ 0 & 0 & 0 & 0 & \delta X^{-4} & 0 & 0 & 0 & 1 & c \\ 0 & 0 & 0 & 0 & 0 & \delta X^{-5} & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & N & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & N & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N^2 \end{bmatrix}$$

We will need to estimate $\det(M)$, which is easy because M is upper triangular. Its determinant is

$$\det(M) = N^{kh(h-1)/2} X^{-(hk)(hk-1)/2} / \sqrt{hk}^{hk} = \left(N^{h-1} X^{-(hk-1)} (hk)^{-1} \right)^{hk/2} .$$

Because $hk \ge 7$, a calculation shows that $hk < 2^{(hk-1)/2}$. This implies

$$\det(M) > \left(N^{h-1} X^{-(hk-1)} 2^{-(hk-1)/2} \right)^{hk/2} .$$

Then from our choice of X we calculate

$$\det(M) > \left(N^{(h-1)-(hk-1)(\frac{1}{k}-\epsilon)} 2^{+(hk-1)/2} \right)^{hk/2} ,$$

and the condition $h - 1 \ge (hk-1)\left(\frac{1}{k} - \epsilon\right)$ gives

$$\det(M) > 2^{(hk)(hk-1)/4} .$$

We will do lattice basis reduction on the rows of M to find an expression for those integer linear combinations of rows of M with small Euclidean norm ("short" vectors).

One such short vector is related to the unknown solution x_0. Consider a row vector $\mathbf{r}$ whose left-hand elements are powers of the unknown x_0:

$$r_g = x_0^g$$

and whose right-hand elements are the negatives of powers of x_0 and y_0:

$$r_{\gamma(i,j)} = -x_0^i y_0^j$$

$$\mathbf{r} = (1, x_0, x_0^2, \ldots, x_0^{hk-1}, -y_0, -x_0y_0, \ldots, -x_0^{k-1}y_0, -y_0^2, -x_0y_0^2, \ldots, -x_0^{k-1}y_0^{h-1}) \ .$$

The product $\mathrm{s} = \mathbf{r}M$ is a row vector with left-hand elements given by

$$s_g = (x_0/X)^g/\sqrt{hk}$$

and right-hand elements by

$$s_{\gamma(i,j)} = q_{ij}(x_0) - x_0^i y_0^j N^j = 0 \ .$$

The Euclidean norm of s is estimated by:

$$|\mathrm{s}| = \left[\sum_g {s_g}^2\right]^{1/2} < \left[\sum_g (1/\sqrt{hk})^2\right]^{1/2} = 1 \ .$$

Because $p(x)$ and hence $q_{ij}(x)$ are monic polynomials, the submatrix of M formed by rows k through $hk-1$ and the right-hand $hk-k$ columns is an upper triangular matrix with 1 on the diagonal. This implies that we can do elementary row operations on M to produce a block matrix $\tilde{M}$ whose lower right $(hk-k) \times (hk-k)$ block is the identity matrix and whose upper right $(hk) \times (hk-k)$ block is zero. The upper left $(hk) \times (hk)$ block $\hat{M}$ satisfies $\det(\hat{M}) = \det(M) > 2^{(hk)(hk-1)/4}$.

We now restrict our attention to the upper hk rows of $\tilde{M}$, or equivalently on $\hat{M}$: the lattice elements represented by vectors whose right-hand side is 0.

Let $n = hk = \dim(\hat{M})$. Perform lattice basis reduction on $\hat{M}$, using the procedure in [3]. Let $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ be the resulting row basis of $\hat{M}$, and let $\mathbf{b}_n^*$ denote the component of $\mathbf{b}_n$ orthogonal to the span of $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{n-1}$. From the discussion in [3] we know that the last basis element $\mathbf{b}_n$ satisfies

$$|\mathbf{b}_n^*| \geq \{\det(\hat{M})\}^{1/n} 2^{-(n-1)/4} > 1 \ ,$$

the latter estimate coming from our lower bound on $\det(\hat{M})$.

The Euclidean norm of any element $\sum c_i\mathbf{b}_i$ of the lattice of $\hat{M}$ is at least $|c_n| \times |\mathbf{b}_n^*| > |c_n|$. So any lattice element with norm less than 1 must have $c_n = 0$; it lies in the subspace spanned by $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{n-1}$. In particular, s is such a lattice element: it has norm less than 1, and it lies in the lattice of $\hat{M}$ because its right-hand entries are 0.

In terms of the original matrix M, let an arbitrary short lattice element (with 0 in the right-hand side and length less than 1) be given by

$$(d_0, d_1, \ldots, d_{hk-1}, e_{\gamma(0,1)}, \ldots, e_{\gamma(k-1,h-1)})M \ .$$

Because these short rows span a vector space of dimension at most $n-1 = hk-1$, simple linear algebra can produce a collection of integers $f_0, f_1, \ldots, f_{hk-1}$ (not all zero) satisfying

$$\sum_g f_g d_g = 0$$

for each short row; this equation holds over $\mathbb{Z}$. (Notice that the $e_{\gamma(i,j)}$ are not involved.)

Our unknown solution row $\mathbf{s} = \mathbf{r}M$ is such a short row, with $d_g = r_g = x_0^g$. So it must be true that

$$\sum_g f_g d_g = \sum_g f_g x_0^g = 0 \ .$$

This is a polynomial equation in x_0 which holds in $\mathbb{Z}$, not just mod N. We can solve this for x_0 in polynomial time, using known techniques for solving univariate polynomial equations over $\mathbb{Z}$. Thus we have produced the desired solution x_0.

Theorem 1. *Let $p(x)$ be a monic integer polynomial of degree k, N a positive integer of unknown factorization, and $\epsilon > 0$. In time polynomial in $\log N$, k and $1/\epsilon$, we can find all integer solutions x_0 to $p(x_0) = 0 \pmod N$ with $|x_0| < \frac{1}{2}N^{(1/k)-\epsilon}$.*

Proof. The lattice basis reduction algorithm from [3] operates in time polynomial in the dimension and the logarithms of the numerators and denominators of the matrix entries; the dimension is polynomial in k and $1/\epsilon$, and the numbers of bits in the matrix entries are polynomial in $\log N$, k and $1/\epsilon$. □

Corollary 2. *Let $p(x)$ be a monic integer polynomial of degree k and N a positive integer of unknown factorization. In time polynomial in $\log N$ and k, we can find all integer solutions x_0 to $p(x_0) = 0 \pmod N$ with $|x_0| < N^{1/k}$.*

Proof. Set $\epsilon = 1/\log_2 N$ and do exhaustive search on $O(1)$ unknown high order bits of x. □

Remark: We have not attempted to find the shortest vector(s) of the lattice, but rather to confine all sufficiently short vectors to a subspace. This appears to be a novel use of lattice basis reduction techniques. It is fortunate, because our desired vector need not be the shortest one. The technique allows us to claim that we will always find the solution, not just with high probability.

Remark: If there are several short solutions x_0, this procedure will find all of them simultaneously.

2.1 Comparison to Previous Work

Vallée et al. [4] apply an LLL-based solution to solving $p(y) = 0 \pmod N$, but require $y < N^{2/[k(k+1)]}$ where $k = \deg(p)$. Our methods are similar to theirs, except that they use only one polynomial $q_{01}(y) = p(y)$ where we use several.

Considering our requirement that $\det(M) > 2^{(hk)(hk-1)/4}$: Each equation $q_{ij}(x) = 0 \pmod{N^j}$ gives a factor of N^j to $\det(M)$, while each unknown x^g costs a factor of about X^{-g}. We must balance the two contributions in order to achieve $\det(M) > 2^{(hk)(hkj-1)/4}$. In the present paper we are able to amortize the cost of the variables over several equations, and this yields the improvement in the bound from $N^{2/[k(k+1)]}$ to $N^{1/k}$.

Another difference is that, because of our technique of confining all small lattice elements to a subspace, we always find the solution if it exists, while Vallée et al., searching for the smallest lattice elements themselves, will succeed with high probability but not always.

3 Extension to Multivariate Polynomials

We encountered some technical difficulties trying to extend this procedure to multivariate polynomials. The guarantee breaks down at a crucial step, so this extension is heuristic. Our sketch is brief because this is irrelevant to the present application.

If we are given a polynomial $p(x_1, \ldots, x_m) \pmod N$ of total degree k, and we know there is a solution $x_i = y_i$ with $|y_i| < N^{\alpha_i}$, we hope to find this solution as long as $\sum \alpha_i < (1/k) - \epsilon$.

Define

$$z = p(y_1, \ldots, y_m)/N$$
$$q_{i_1 \ldots i_m j}(x_1, \ldots, x_m) = x_1^{i_1} \cdots x_m^{i_m} p(x_1, \ldots, x_m)^j \quad ,$$

and notice that $q_{i_1 \ldots i_m j}(y_1, \ldots, y_m)$ is divisible by N^j. Set a limit T and develop the modular equations $q_{i_1 \ldots i_m j}(y_1, \ldots, y_m) = 0 \pmod{N^j}$ for all nonnegative integer indices $(i_1, \ldots, i_m, j)$ with $i_m < k$ and $i_1 + i_2 + \ldots + i_m + kj \leq T$.

Build the matrix M analogous to that of Section 2. The vector $\mathbf{r}$ contains all monomials of total degree at most T, so the sum of the total degrees of these monomials is $m \binom{T+m}{m+1}$, and the sum of the degrees in each $x_i, i = 1, 2, \ldots, m-1$, is $\binom{T+m}{m+1}$. These appear as negative exponents of α_i in the diagonal entries of the upper left block of M.

The powers of N appearing in the lower right of M (the moduli) add to

$$\frac{1}{k}\binom{T+m}{m+1} - O(\binom{T+m}{m})$$

(asymptotically for large T). With the requirement $\sum \alpha_i < (1/k) - \epsilon$, we will have $\det(M) > 1$.

The vector s will be shorter than 1, so it will be among the shorter vectors in the lattice. By the methods of Section 2 we can get at least one polynomial equation satisfied by the y_i over $\mathbb{Z}$. But to solve for y_i over $\mathbb{Z}$ we would need m independent equations. We might or might not get the required equations; if we get m equations, they might not be independent. So the procedure might work or might fail in a particular application. Much work needs to be done in this area.

4 RSA with Stereotyped Messages

An easy application is to RSA encryption with low exponent where most of the message is fixed or "stereotyped".

Suppose we use an RSA exponent of 3 to encrypt a plaintext consisting of two pieces:

(1) A known piece $B = 2^k b$, such as the ASCII representation of "May 14, 1996. The secret key for the day is "

(2) An unknown piece m, such as "Squeamish Ossifrage".

If we know B and the ciphertext $c = (B+m)^3 \pmod{N}$, then we can recover m as long as $|m| < N^{1/3}$. Here $p(m) = (B+m)^3 - c = 0 \pmod{N}$.

This is obvious when $B = 0$, but the present paper makes it possible for nonzero B as well.

5 Application to RSA with Random Padding

The present work was motivated by the following result of Franklin and Reiter [2]; see also [1]. Suppose two messages m and m' satisfy a *known* affine relation, say

$$m' = m + t$$

with t known. Suppose we know the RSA-encryptions of the two messages with an exponent of 3:

$$\begin{aligned} c &= m^3 && \pmod{N} \\ c' &= (m')^3 = m^3 + 3m^2t + 3mt^2 + t^3 && \pmod{N} \end{aligned}$$

Then we can recover m from c, c', t and N:

$$m = \frac{t(c' + 2c - t^3)}{c' - c + 2t^3} = \frac{t(3m^3 + 3m^2t + 3mt^2)}{3m^2t + 3mt^2 + 3t^3} \pmod{N}$$

What if we do not know the exact relation between m and m', but we do know that t is small, say

$$\begin{aligned} m' &= m + t \\ |t| &< N^{1/9} \end{aligned}$$

Can we still find m?

One can imagine a protocol in which messages M are subjected to random padding before being RSA-encrypted with an exponent of 3. Perhaps M is left-shifted by k bits, and a random k-bit quantity T is added, to form a plaintext m; the ciphertext c is then the cube of $m \pmod{N}$:

$$c = m^3 = (2^k M + T)^3 \pmod{N}$$

Now suppose the same message is encrypted twice, but with a different random pad each time. Let the second random pad be $T' = T + t$ so that the second plaintext is $m' = m + t$. Then we see the two ciphertexts

$$\begin{aligned} c &= m^3 &&= (2^k M + T)^3 && \pmod{N} \\ c' &= (m')^3 &&= (2^k M + T')^3 = (m+t)^3 && \pmod{N} \end{aligned}$$

Can we recover t and m?

We can eliminate m from the two equations above by taking their resultant:

$$\begin{aligned}&\mathrm{Resultant}_m(m^3 - c, (m+t)^3 - c') = \\ &= t^9 + (3c - 3c')t^6 + (3c^2 + 21cc' + 3(c')^2)t^3 + (c - c')^3 = 0 \pmod N\end{aligned}$$

This is a univariate polynomial in t of degree 9 (mod N). If $|t| < N^{1/9}$, we can apply the present work to recover t. We can then apply Franklin and Reiter's result to recover m, and strip off the padding to get M.

This works just as well if the padding goes in the high order bits, or in the middle; just divide each ciphertext by the appropriate power of 2, in order to divide each plaintext by another power of 2, to move the random bits to the low order bits.

The warning is clear: If the message is subject to random padding of length less than 1/9 the length of N, and then encrypted with an exponent of 3, multiple encryptions of the same message will reveal the message.

Notice that for a 1024-bit RSA key, this attack tolerates 100 bits of padding fairly easily.

Some possible steps to avoid this attack:

(1) Spread the random padding into several blocks (not one contiguous block). Then the present attack needs to be modified, and apparently will tolerate only a smaller total amount of padding. The padding could be two small blocks t and u, positioned so that the encryption is $c = (2^{\ell}t + 2^k m + u)^3 \pmod N$. Two encryptions of the same message would yield a resultant which is a single equation in two small integer variables t and u. The generalized attack of Section 3 might work, provided that $|t|$ and $|u|$ are subject to bounds T and U with $TU < N^{1/9}$. The computation is more complicated and results are not guaranteed.

(2) Spread the padding throughout the message: two bits out of each eight-bit byte, for example. This seems to be a much more effective defense against the present attack.

(3) Increase the amount of padding. This decreases efficiency; also if the padding is less than 1/6 the length of N, the alternate solution shown in Section 6 might still recover the message if multiple encryptions have been done.

(4) Make the "random" padding depend on the message deterministically. For example we could subject the message to a hashing function, and append that hash value as the random padding. Then two encryptions would be identical, because the random padding would be identical. A possible weakness still exists: suppose a time-stamp is included in each message, and this time-stamp occupies the low order bits, next to the padding. Then two plaintexts for the same message (with different time stamps) will differ in the time-stamp and (possibly) the pad; just let t combine these two fields and proceed as before.

(5) Use larger exponents for RSA encryption. If the exponent is e, the attack apparently tolerates random padding of length up to $1/e^2$ times the length of N. So already for $e = 7$ the attack is useless: on a 1024-bit RSA key with $e = 7$, the attack would tolerate only 21 bits of padding, and this would be better treated by exhaustion.

6 Another Solution for Multiple Encryptions

If instead of two encryptions of the same message we have several, say $k+1$, then we can mount other attacks which might tolerate larger fields of random padding. We sketch here an attack which (heuristically) seems to tolerate random padding up to α times the length of N where

$$\alpha < \frac{k-2}{6k-3} < \frac{1}{6} \ .$$

We begin with

$$\begin{aligned} A_0 &= m^3 && \pmod{N} \\ A_i &= (m+t_i)^3 && \pmod{N} \\ c_i = A_i - A_0 &= 3m^2t_i + 3mt_i^2 + t_i^3 && \pmod{N} \end{aligned}$$

where we know A_0, A_i, c_i and N, but not m or t_i. We assume the padding is small:

$$|t_i| \leq \frac{1}{2}N^{\alpha}.$$

For indices $i < j < \ell$ define $d_{ij} = t_it_j(t_i - t_j)$ and $e_{ij\ell} = -t_it_jt_\ell(t_i - t_j)(t_j - t_\ell)(t_\ell - t_i)$. The $C(k,2) = \binom{k}{2}$ linearly independent quantities d_{ij} each satisfy $|d_{ij}| < N^{3\alpha}$, and the $C(k,3)$ linearly independent quantities $e_{ij\ell}$ each satisfy $|e_{ij\ell}| < N^{6\alpha}$. One can check the following identity:

$$d_{ij}c_\ell + d_{j\ell}c_i - d_{i\ell}c_j = e_{ij\ell} \pmod{N} \ .$$

This suggests lattice basis reduction on the row basis of the following matrix. M is a square upper triangular integer matrix of dimension $(C(k,2)+C(k,3))$. Its upper left $C(k,2) \times C(k,2)$ block is the identity times an integer approximation to $N^{3\alpha}$. Its lower left $C(k,3) \times C(k,2)$ block is 0. Its lower right $C(k,3) \times C(k,3)$ block is N times the identity. Its upper right $C(k,2) \times C(k,3)$ block has rows indexed by pairs of indices $(i,j), i < j$, and columns indexed by triples of indices $(i,j,\ell), i < j < \ell$. Column (i,j,ℓ) has three nonzero entries: c_ℓ at row (i,j), c_i at row (j,ℓ), and $-c_j$ at row (i,ℓ).

We illustrate the matrix M for the case $k = 4$. The first $C(k,2) = 6$ rows are indexed by (1,2), (1,3), (1,4), (2,3), (2,4) and (3,4). The last $C(k,3) = 4$ columns are indexed by (1,2,3), (1,2,4), (1,3,4) and (2,3,4).

$$M = \begin{bmatrix} N^{3\alpha} & 0 & 0 & 0 & 0 & 0 & c_3 & c_4 & 0 & 0 \\ 0 & N^{3\alpha} & 0 & 0 & 0 & 0 & -c_2 & 0 & c_4 & 0 \\ 0 & 0 & N^{3\alpha} & 0 & 0 & 0 & 0 & -c_2 & -c_3 & 0 \\ 0 & 0 & 0 & N^{3\alpha} & 0 & 0 & c_1 & 0 & 0 & c_4 \\ 0 & 0 & 0 & 0 & N^{3\alpha} & 0 & 0 & c_1 & 0 & -c_3 \\ 0 & 0 & 0 & 0 & 0 & N^{3\alpha} & 0 & 0 & c_1 & c_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & N & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & N & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N \end{bmatrix}$$

Consider the integer row vector $\mathbf{r}$ whose first $C(k,2)$ entries are d_{ij}, and whose last $C(k,3)$ entries are the integers $(e_{ij\ell} - (d_{ij}c_\ell + d_{j\ell}c_i - d_{i\ell}c_j))/N$. The product $\mathbf{r}M = \mathbf{s}$ has left-hand elements $d_{ij}N^{3\alpha}$ and right-hand elements $e_{ij\ell}$; all its entries are bounded by $N^{6\alpha}$. We hope that lattice basis reduction will find this row.

The determinant of M is $N^{3\alpha C(k,2)+C(k,3)}$. Because of our choice of α, this is larger than $(N^{6\alpha})^{C(k,2)+C(k,3)}$. So $\mathbf{s}$ is among the shorter elements of the lattice generated by the rows of M.

The difficulty in finding $\mathbf{s}$ depends on its rank among the short elements. If $|t_i|$ are much smaller than N^α then we can hope that $\mathbf{s}$ is the shortest lattice element, and that lattice basis reduction methods can recover it efficiently. We do not here supply efficiency estimates or probabilities of success; we treat this as a heuristic attack.

Assuming that we can actually find $\mathbf{s}$, we will be able to recover the values t_i by taking *g.c.d.* of elements of $\mathbf{r} = \mathbf{s}M^{-1}$:

$$\begin{aligned} g.c.d.\{d_{1,2}, d_{1,3}, \ldots, d_{1,k}\} &= g.c.d.\{t_1t_2(t_1-t_2), t_1t_3(t_1-t_3), \ldots, t_1t_k(t_1-t_k)\} \\ &= t_1 \times g.c.d.\{t_2(t_1-t_2), t_3(t_1-t_3), \ldots, t_k(t_1-t_k)\}\ , \end{aligned}$$

and hopefully the latter *g.c.d.* will be small enough to discover by exhaustive search. Having found t_i, we can recover m by Franklin and Reiter's technique.

If we have 14 encryptions of the same message ($k = 13$), then we can tolerate a random padding of about 150 bits in a 1024-bit RSA message.

7 Conclusions and Open Problems

We have shown how to solve a univariate polynomial equation (mod N) of degree k if there is a solution smaller than $N^{1/k}$. We have applied this to stereotyped RSA messages with small encryption exponent. We have also applied it to the case of RSA encryption with exponent 3 with random padding of less than 1/9 (resp. 1/6) of the length of N, to recover a message which has been enciphered twice (resp. several times) with different random padding each time.

We warn against RSA encryption with exponent 3 and with random padding of such length, in the case where a protocol allows a message to be enciphered several times with different values of the padding.

An important open problem is to find conditions under which the multivariate case works. In particular, how effectively does it work for the case of two messages with random padding in two blocks?

8 Acknowledgments

The author gratefully acknowledges several enlightening discussions with Andrew Odlyzko and Jean-Jacques Quisquater. Matt Franklin and Mike Reiter's Crypto 95 rump session paper and subsequent discussions were most useful. Jacques Patarin was independently working on the idea of unknown offsets in the RSA setting. The Eurocrypt program committee supplied us with missing references to existing literature.

References

1. D. Coppersmith, M. Franklin, J. Patarin and M. Reiter, "Low Exponent RSA with Related Messages," Proceedings of Eurocrypt 96.
2. M. Franklin and M. Reiter, "A Linear Protocol Failure for RSA with Exponent Three," presented at the rump session, Crypto 95, but not in the proceedings.
3. A. K. Lenstra, H. W. Lenstra and L. Lovasz, "Factoring Polynomials with Integer Coefficients," Matematische Annalen **261** (1982), 513–534.
4. B. Vallée, M. Girault and P. Toffin, "How to Guess ℓ-th Roots Modulo n by Reducing Lattice Bases," Proceedings of AAECC–6, Springer LNCS **357** (1988) 427–442.