ON NONLINEAR POLYNOMIAL SELECTION FOR THE
NUMBER FIELD SIEVE

NICHOLAS COXON

Abstract. The number field sieve is asymptotically the fastest known integer
factorisation algorithm. The algorithm begins with the selection of a pair of
low-degree integer polynomials. The coefficient size of the chosen polynomi-
als then plays a key role in determining the running time of the algorithm.
Nonlinear polynomial selection algorithms approach the problem of construct-
ing polynomials with small coefficients by employing a reduction to the well-
studied problem of finding short vectors in lattices. The reduction rests upon
the construction of modular geometric progressions with small terms. In this
paper, tools are developed to aid in the analysis of nonlinear algorithms. Pre-
cise criteria for the selection of geometric progressions are given. Existing
nonlinear algorithms are extended and analysed.

1. Introduction

To factor an integer \( N \), the number field sieve (NFS) \([14]\) begins with the selection
of low-degree coprime irreducible polynomials \( f_1, f_2 \in \mathbb{Z}[x] \) with a common root
modulo \( N \). If \( F_i \in \mathbb{Z}[x, y] \) denotes the homogenisation of \( f_i \), for \( i = 1, 2 \), then the
time taken to factor \( N \) depends on the supply of coprime integer pairs \( (a, b) \) for
which \( F_1(a, b) \) and \( F_2(a, b) \) are smooth. Pairs with this property, called relations,
are identified by sieving. The polynomials selection problem is concerned with
determining a choice of polynomials that minimises the time taken by the sieve
stage of the algorithm.

The size of the values taken by the polynomials \( F_1 \) and \( F_2 \) is a key factor in
determining the supply of relations (see \([23, 24]\)). Polynomial selection algorithms
address this factor by seeking to generate polynomials with small coefficients. The
efforts of research into this problem have been divided between two different ap-
proaches: so-called linear and nonlinear algorithms. Linear algorithms were in-
troduced during the development of the number field sieve \([3]\) and subsequently
improved by Montgomery and Murphy \([24]\), and Kleinjung \([11, 10]\). They have
been used in a string of record setting factorisations, culminating in the factorisa-
tion of a 768-bit RSA modulus \([12]\). Polynomials are found by selecting nonzero
integers \( a_d, m \) and \( p \) such that \( a_d m^d \equiv N \pmod{p} \). Then \( N \) is represented in the
form \( N = \sum_{i=0}^{d} a_i m^{i} p^{d-i} \), for integers \( a_0, \ldots, a_{d-1} \). Each representation gives rise
to a degree \( d \) polynomial \( f_1 = \sum_{i=0}^{d} a_i x^{i} \) and a linear polynomial \( f_2 = px - m \) with
common root \( mp^{-1} \) modulo \( N \).

Polynomials produced by linear algorithms experience an imbalance in the size
of the values \( F_1(a, b) \) and \( F_2(a, b) \); for most pairs \( (a, b) \in \mathbb{Z}^2 \), the nonlinear poly-
nomial produces values that are larger and thus less likely to be smooth. Nonlinear
algorithms address this problem by producing pairs of nonlinear polynomials with equal or almost equal degrees. Current methods for nonlinear polynomial selection rely on the construction of geometric progressions with small terms modulo \( N \) and techniques from the algorithmic geometry of numbers. The first example of a nonlinear algorithm, Montgomery’s two quadratics algorithm (reported in [8, Section 5]), produces pairs of quadratic polynomials with provably optimal coefficient size. However, quadratic polynomial pairs are only competitive for the factorisation of integers containing at most 110-120 digits (see [24, Section 2.3.1]). Montgomery [21, 20] outlined a generalisation of the quadratic algorithm to arbitrary degrees. Constructing geometric progressions that meet the requirements of Montgomery’s generalisation remains a largely open problem.

Recent developments in geometric progression construction and relaxations of the requirements of Montgomery’s approach have lead to a string of new nonlinear algorithms. This line of research begins with Williams’ [34] algorithms for producing quadratic and cubic polynomial pairs. Refinements to Williams’ algorithms and extensions to arbitrary degree were given by Prest and Zimmermann [30]. Finally, Koo, Jo and Kwon [13] extended methods for constructing geometric progressions.

In this paper, tools from the geometry of numbers are developed to aid in the analysis of nonlinear algorithms. The tools allow precise criteria for selecting geometric progressions to be given. A family of geometric progressions modulo \( N \) containing those used in existing algorithms is characterised. The characterisation enables minor extensions to existing nonlinear algorithms to be made. Parameter selection for the new algorithms is considered. Due to the work of Brent, Montgomery and Murphy [22, 23, 24], it is understood that an abundance of roots modulo small primes can significantly increase the yield of a number field sieve polynomial. This factor, called the root properties of a polynomial, is considered where possible.

The remainder of the paper is organised as follows. In the next section, notation is establish and relevant background material provided. Particular attention is given to the methods used to measure coefficient size throughout this paper. These methods differ from the existing literature on nonlinear polynomial selection. Their motivation is therefore discussed in detail. In Section 3 nonlinear polynomial selection is reviewed. Section 4 and 5 contain new nonlinear generation algorithms.

2. Preliminaries

This section introduces notation, definitions, and preliminary results required for subsequent sections.

2.1. Skewed coefficient norms and the resultant bound. In this section, the coefficient norms used throughout this paper to measure coefficient size are introduced. Then a lower bound on the coefficient size of polynomial pairs with a common root modulo \( N \) is derived. Throughout, it is assume that a sieve is used to identify all relations contained in a region \( \mathcal{A} \) of the form \( \mathcal{A} = [-A, A] \times [0, B] \). The actual form of the region depends on the method of sieving. Furthermore, it is known that a rectangular sieve region is not optimal in general [32]. The area \( 2AB \) of \( \mathcal{A} \) is approximately determined by the size of the input \( N \). Therefore, it will be assumed that the quantity \( D = \sqrt{AB} \) is fixed. Then the region \( \mathcal{A} \) is determined by the parameter \( s = A/B \) called the skew of the region: \( A = D \sqrt{s} \) and \( B = D / \sqrt{s} \).
2.1.1. Skewed coefficient norms. Given two polynomials $f_1$ and $f_2$, the size of the values taken by the respective homogenisations $F_1$ and $F_2$ over the sieve region $A$ can be roughly quantified by the integral
\[
\int_A |F_1(x, y) F_2(x, y)| \, dx \, dy.
\]
Using Hölder’s inequality to bound this integral suggests that the size properties of a degree $d$ polynomial $f$ can be quantified by the integral
\[
\int_A F(x, y)^2 \, dx \, dy = \int_D \int_D \left( f \left( \frac{sx}{y} \right) \cdot \left( \frac{y}{\sqrt{s}} \right) \right)^2 \, dx \, dy.
\]
Montgomery and Murphy consider a similar quantity in their linear algorithm [24, Procedure 5.1.6]. The integrand on the right motivates the following choice of norm:

**Definition 2.1.** Let $f = \sum_{i=0}^d a_i x^i \in \mathbb{R}[x]$ be a degree $d$ polynomial. For a given skew $s > 0$, the skewed $2$-norm of $f$ is defined to be
\[
\|f\|_{2,s}^2 = \left( \sum_{i=0}^d |a_i s^{i-\frac{d}{2}}|^2 \right)^{\frac{1}{2}}.
\]
The case $s = 1$ corresponds to the $2$-norm of $f$, simply denoted $\|f\|_2$.

An $\infty$-norm analogue of the skewed $2$-norm, called the sup-norm, appears in [11]. A skew of a polynomial $f$ is any value $s > 0$ for which $\|f\|_{2,s}$ is minimal.

2.1.2. The resultant bound. For nonzero coprime polynomials $f_1, f_2 \in \mathbb{Z}[x]$ with a common root modulo $N$, the resultant bound provides a lower bound on the $2$-norms of $f_1$ and $f_2$:
\[
N \leq \|f_1\|_{2}^{\deg f_2} \cdot \|f_2\|_{2}^{\deg f_1}.
\]
The $2$-norm may greatly overestimate the coefficient size of polynomials with large skew. To provide tighter bounds, a generalisation of inequality (2.1) is now derived for the skewed $2$-norm. To begin, the definition and some properties of the resultant of two polynomials must first be introduced.

Let $f = \sum_{i=0}^m a_i x^i$ and $g = \sum_{i=0}^n b_i x^i$ be non-constant polynomials with complex coefficients and $a_m, b_n \neq 0$. The Sylvester matrix of $f$ and $g$, denoted $\text{Syl}(f, g)$, is the $(m + n) \times (m + n)$ matrix
\[
\text{Syl}(f, g) = \begin{pmatrix}
a_m & a_{m-1} & \cdots & \cdots & a_0 \\
a_m & a_{m-1} & \cdots & \cdots & a_0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
b_n & b_{n-1} & \cdots & \cdots & b_0 \\
b_n & b_{n-1} & \cdots & \cdots & b_0 \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
b_n & b_{n-1} & \cdots & \cdots & b_0 \\
\end{pmatrix}
\]
where there are $n$ rows containing the $a_i$, $m$ rows containing the $b_i$, and all empty entries are 0. The resultant of $f$ and $g$, denoted $\text{Res}(f, g)$, is equal to the determinant of the Sylvester matrix $\text{Syl}(f, g)$. For the purpose of generalising the resultant bound, the following well-known properties of resultants are required:
• If $\alpha_1, \ldots, \alpha_m \in \mathbb{C}$ are the roots of $f$ and $\beta_1, \ldots, \beta_n \in \mathbb{C}$ the roots of $g$, then
  \[ \text{Res}(f, g) = a_m^n b_n^m \prod_{i,j} (\alpha_i - \beta_j). \]

• If $f, g \in \mathbb{Z}[x]$, then Res$(f, g)$ belongs to the ideal $\langle f, g \rangle \cap \mathbb{Z}$.

Proofs of these two properties can be found in [7, Section 1.3.2]. The two properties imply that coprime non-constant polynomials $f_1, f_2 \in \mathbb{Z}[x]$ with a common root modulo $N$ must satisfy $N \leq |\text{Res}(f_1, f_2)|$. The resultant bound (2.1) is then obtained by applying Hadamard’s inequality (see [31, Section 1.3]) to bound the absolute value of $\text{det Syl}(f_1, f_2)$. The following lemma generalises the upper bound obtained from Hadamard’s inequality.

**Lemma 2.2.** Let $f = \sum_{i=0}^{m} a_i x^i$ and $g = \sum_{i=0}^{n} b_i x^i$ be non-constant polynomials with complex coefficients and $a_m, b_n \neq 0$. Then

\[ |\text{Res}(f, g)| \leq \|f\|_{2,s}^n \cdot \|g\|_{2,s}^m, \]

for all $s > 0$.

**Proof.** Let $\alpha_1, \ldots, \alpha_m$ be the roots of $f$ and $\beta_1, \ldots, \beta_n$ the roots of $g$. For $s > 0$,

\[ \text{Res}(f, g) = a_m^n b_n^m \prod_{i,j} (\alpha_i - \beta_j) = (a_m s^{-\frac{m}{2}})^n (b_n s^{-\frac{n}{2}})^m \prod_{i,j} \left( \frac{\alpha_i}{s} - \frac{\beta_j}{s} \right). \]

Hence,

\[ |\text{Res}(f, g)| \leq \left\| s^{-\frac{m}{2}} f(sx) \right\|^n \cdot \left\| s^{-\frac{n}{2}} g(sx) \right\|^m = \|f\|_{2,s}^n \cdot \|g\|_{2,s}^m, \]

where the inequality is obtained by applying Hadamard’s inequality. \qed

**Corollary 2.3.** Let $f_1, f_2 \in \mathbb{Z}[x]$ be non-constant coprime polynomials with a common root modulo $N$. Then

\[ N \leq \|f_1\|_{2,s}^{\deg f_2} \cdot \|f_2\|_{2,s}^{\deg f_1}, \]

for all $s > 0$.

The complexity of the number field sieve is largely determined by the size of $N$ and the degree sum $\deg f_1 + \deg f_2$ of the polynomials used [8, Section 11]. For values of $N$ within the current range of interest, the optimal choice of degree sum remains small (see [24, Section 3.1] for a relevant discussion). For example, the factorisation of a 768-bit RSA modulus by Kleinjung et al. [12] and the special number field sieve [15] factorisation of $2^{1039} - 1$ by Aoki et al. [2] both used polynomial pairs with a degree sum of 7. Corollary 2.3 shows that the restriction to small degree sum implies that a pair of number field sieve polynomials will necessarily have large coefficients. For large $N$ without special form, the problem of determining polynomials that meet the lower bound in Corollary 2.3 remains open.

### 2.1.3. A note on measuring coefficient size.

The resultant of two coprime polynomials $f_1, f_2 \in \mathbb{Z}[x]$ is a homogeneous polynomial of degree $\deg f_1 + \deg f_2$ in their coefficients. As a result, some authors consider a pair of number field sieve polynomials $f_1, f_2 \in \mathbb{Z}[x]$ to have optimal coefficient size whenever $\text{Res}(f_1, f_2) = \pm N$. However, the resultant only provides a lower bound on the coefficient size. Therefore, on its own, the resultant of two polynomials does not serve as an accurate
measure of coefficient size. This is demonstrated by the following lemma which proves the existence of integer polynomial pairs with arbitrary large coefficients and resultant equal to $\pm N$.

**Lemma 2.4.** Let $X, s > 0$. For integers $N$ and $d \geq 2$ with $N > 1.5(d/\log 2)^d$, there exist degree $d$ polynomials $f_1, f_2 \in \mathbb{Z}[x]$ such that $\text{Res}(f_1, f_2) = \pm N$ and $\|f_i\|_{2,s} \geq X$, for $i = 1, 2$.

**Proof.** Let $m = \lceil N^{1/d} \rceil$ and write $N$ in base-$m$:

$$N = a_dm^d + a_{d-1}m^{d-1} + \ldots + a_1m + a_0,$$

where the coefficients $a_0, \ldots, a_d \in [0, m) \cap \mathbb{Z}$. Then $a_d = 1$ since $N > 1.5(d/\log 2)^d$ [6, Exercise 6.8]. Define polynomials

$$f_1(x) = c_1 \cdot (x - m) + \sum_{i=0}^{d} a_ix^i \quad \text{and} \quad f_2(x) = c_2 \cdot f_1(x) + (x - m),$$

where $c_1, c_2 \in \mathbb{Z}$, $c_2 \neq 0$, are chosen sufficiently large to ensure $\|f_i\|_{2,s} \geq X$, for $i = 1, 2$. Then $f_1$ and $f_2$ are degree $d$ integer polynomials with $f_1(m) = N$. Finally, by subtracting $c_2$ times row $i$ of Syl$(f_1, f_2)$ from row $d + i$, for $1 \leq i \leq d$, it follows that

$$\text{Res}(f_1, f_2) = \text{Res}(f_1(x), c_2 \cdot f_1(x) + (x - m)) = a_d^{d-1}, \text{Res}(f_1(x), x - m) = \pm f_1(m).$$

Throughout this paper, the coefficient size of a pair of number field sieve polynomials $f_1$ and $f_2$ will be measured by their product of coefficient norms $\|f_1\|_{2,s} \cdot \|f_2\|_{2,s}$. In the case that both polynomials are of degree $d$, Corollary 2.3 implies that the product is bounded below by $N^{1/d}$. The choice is further motivated by the observation that the polynomial values $F_1(a, b)F_2(a, b)$, with $(a, b) \in A \cap \mathbb{Z}^2$, satisfy

$$|F_1(a, b)F_2(a, b)| = |\text{Res}(f_1(x), bx - a)| \cdot |\text{Res}(f_2(x), bx - a)| \leq \|f_1\|_{2,s} \cdot \|f_2\|_{2,s} \cdot \|bx - a\|_{2,s}^{\deg f_1 + \deg f_2}.$$ 

Thus a choice of number field sieve polynomials with $\|f_1\|_{2,s} \cdot \|f_2\|_{2,s}$ small should yield more relations compared to another pair with equal degree sum and a larger product of coefficient norms.

### 2.2. Lattices in $\mathbb{R}^n$.

Throughout this paper, results and algorithms from the geometry of numbers are extensively used. Here necessary background on lattices and lattice algorithms is reviewed. The reader is referred to [5, 13, 17, 28] for further background on the concepts discuss in this section.

A **lattice** in $\mathbb{R}^n$ is a subgroup $\Lambda$ of $\mathbb{R}^n$ with the following property: there exists $\mathbb{R}$-linear independent vectors $b_1, \ldots, b_k \in \mathbb{R}^n$ such that $\Lambda = \sum_{i=1}^k \mathbb{Z}b_i$. The vectors $b_1, \ldots, b_k$ are said to form a **basis** for $\Lambda$, denoted throughout by a $k$-tuple $B = (b_1, \ldots, b_k)$; and $k$ is called the **dimension** or **rank** of $\Lambda$. When written with respect to the canonical orthonormal basis of $\mathbb{R}^n$, if $b_i = (b_{i,1}, \ldots, b_{i,n})$, for $1 \leq i \leq k$, then the $k \times n$ matrix $B = (b_{i,j})_{1 \leq i \leq k, 1 \leq j \leq n}$ is called a **basis matrix** for $\Lambda$. The **Gram matrix** of $B$ is the $k \times k$ symmetric matrix $BB^t$. Let $B_1$ and $B_2$ be bases for $\Lambda$ with respective basis matrices $B_1$ and $B_2$. Then there exists a matrix $U \in \text{GL}_k(\mathbb{Z})$ such that $UB_1 = B_2$. Thus the Gram matrix of $B_2$ is $Q_2 = UQ_1U^t$, where $Q_1$ is the
Gram matrix of $B_1$. Therefore, the determinant of the Gram matrix is independent of the choice of basis. The determinant of $\Lambda$ is defined to be $\det \Lambda = \sqrt{\det Q}$, where $Q$ is the Gram matrix of one of its bases.

The sublattices of a lattice are its subgroups. A sublattice $\Lambda'$ of a lattice $\Lambda$ is called full-rank whenever $\dim \Lambda' = \dim \Lambda$. This occurs if and only if $[\Lambda : \Lambda']$ is finite. In this case, the determinant of $\Lambda'$ is related to the determinant of $\Lambda$ by $\det \Lambda' = [\Lambda : \Lambda'] \cdot \det \Lambda$. Let $\langle x, y \rangle \mapsto x \cdot y$ denote the usual inner product on $\mathbb{R}^n$.

The dual lattice of $\Lambda$ is $\Lambda^* = \{ x \in \text{span}(\Lambda) \mid \forall y \in \Lambda, \langle x, y \rangle \in \mathbb{Z} \}$.

For any basis $\mathcal{B}$ of $\Lambda$, the dual basis $\mathcal{B}^*$ for $\text{span}(\Lambda)$ is a basis for $\Lambda^*$. A lattice with $\Lambda^* = \Lambda$ is called unimodular. The lattice $\mathbb{Z}^n$ is unimodular.

Let $\| \cdot \|_2$ be the norm on $\mathbb{R}^n$ induced by $\langle \cdot, \cdot \rangle$. For a $k$-dimensional lattice $\Lambda$ and all $1 \leq i \leq k$, the $i$th minimum $\lambda_i(\Lambda)$ of $\Lambda$ is defined to be the minimum of $\max_{1 \leq j \leq i} \| v_j \|_2$ over all linearly independent lattice vectors $v_1, \ldots, v_i \in \Lambda$. Minkowski’s second theorem (see [28, Theorem 5 p. 35]) provides an upper bound on the geometric mean of consecutive minima: if $\Lambda$ is a $k$-dimensional lattice and $t$ an integer satisfying $1 \leq t \leq k$, then

$$\left( \prod_{i=1}^{t} \lambda_i(\Lambda) \right)^{\frac{1}{t}} \leq \sqrt[k]{\gamma_k \det(\Lambda)}^\frac{1}{t},$$

where $\gamma_k \leq 1 + k/4$ denotes Hermite’s constant (see [28 p. 33]).

Algorithms for lattice reduction aim to produce bases consisting of short vectors. The most widely used reduction algorithm, due to Lenstra, Lenstra and Lovás [16], is the LLL algorithm. Given a basis for a lattice $\Lambda \subseteq \mathbb{Z}^n$, the LLL algorithm produces an LLL-reduced basis for $\Lambda$ in polynomial time.

**Definition 2.5.** Let $\Lambda \subseteq \mathbb{R}^n$ be a $k$-dimensional lattice and $\mathcal{B} = (b_1, \ldots, b_k)$ one of its bases. Let $(b_1^*, \ldots, b_k^*)$ be the Gram–Schmidt orthogonalisation of $\mathcal{B}$ and define $\mu_{i,j} = \langle b_i, b_j^* \rangle / \langle b_i^*, b_j^* \rangle$, $1 \leq j < i \leq k$. Then $\mathcal{B}$ is LLL-reduced with factor $\delta \in (1/4, 1]$ if and only if the following conditions hold:

1. $|\mu_{i,j}| \leq 1/2$, for $1 \leq j < i \leq k$; and
2. $\left\| b_{i+1}^* + \mu_{i+1,i} b_i^* \right\|_2^2 \geq \delta \left\| b_i^* \right\|_2^2$, for $1 \leq i < k$.

For simplicity, it is assumed throughout this paper that LLL-reduced means LLL-reduced with factor $\delta = 3/4$. Accordingly, the following properties of LLL-reduced bases hold:

**Theorem 2.6.** Let $(b_1, \ldots, b_k)$ be an LLL-reduced basis of a $k$-dimensional lattice $\Lambda \subseteq \mathbb{R}^n$. Then

1. $\|b_1\|_2 \leq 2^{(k-1)/4} |\det \Lambda|^{1/k}$.
2. $\|b_i\|_2 \leq 2^{(k-1)/2} \lambda_i(\Lambda)$, for $1 \leq i \leq k$.
3. If $\Lambda \subseteq \mathbb{Z}^n$, then $\|b_i\|_2 \leq 2^{k(k-1)/4} \sqrt[k]{\gamma_k} \det \Lambda^{1/k}$, for $1 \leq i \leq k$.

Proofs of the first two properties occur in [16]. The third property is due to May [19, Theorem 4].

Given a basis $(b_1, \ldots, b_k)$ of a $k$-dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$, with $\max_i \|b_i\|_2 \leq M$, the LLL algorithm returns an LLL-reduced basis in time $O(k^5 n \log^3 M)$ with arithmetic operations performed on integers of bit-length $O(k \log M)$. For instances
where \( \log M \) is large, it is preferable to use a floating point variant of the LLL algorithm such as the \( L^2 \) algorithm [27, 26]. The \( L^2 \) algorithm returns an LLL-reduced basis in time \( O(k^4 n(k + \log M) \log M) \) and requires a precision of \( (\log_2 3) \cdot k \) bits thus giving an improved overall complexity and requiring precision independent of \( \log M \).

3. Nonlinear polynomial selection

Nonlinear polynomial generation algorithms are based on the observation that polynomials with bounded degree and a prescribed root modulo \( N \) can be characterised by orthogonality conditions on their coefficient vectors modulo \( N \). As a trivial example, an integer polynomial \( f = \sum_{i=0}^{d} a_i x^i \) of degree at most \( d \) has \( m \) as a root modulo \( N \) if and only if the coefficient vector \( (a_0, \ldots, a_d) \) is orthogonal to \( (1, m, \ldots, m^d) \) modulo \( N \). The set of all such coefficient vectors, denoted throughout by \( L_{m,d} \), forms a lattice in \( \mathbb{Z}^{d+1} \) [3, Section 12.2]. Nonlinear algorithms employ lattice reduction to search for short vectors in sublattices of \( L_{m,d} \). Bounds such as those in Theorem 2.6 suggest this approach will be most successful for sublattices with small determinants.

Using an approach introduced by Montgomery (see [8, Section 5] and [24, Section 2.3.1]), and since applied by several authors [21, 20, 34, 30, 13], nonlinear algorithms construct sublattices of \( L_{m,d} \) with small determinants from “small” geometric progressions modulo \( N \). Formally, a geometric progression (GP) of length \( l \) and ratio \( r \) modulo \( N \), denoted throughout by a vector \([c_0, \ldots, c_{l-1}]\), is an integer sequence with the property that \( c_i \equiv c_0 r^i \pmod N \), for \( 0 \leq i < l \). Central to the construction of lattices for nonlinear algorithms is the observation that

\[
L_{m,d} = \left\{ (a_0, \ldots, a_d) \in \mathbb{Z}^{d+1} \mid \sum_{i=0}^{d} a_i c_i \equiv 0 \pmod N \right\},
\]

for any length \( d + 1 \) GP \([c_0, \ldots, c_d]\) with ratio \( m \) modulo \( N \), nonzero terms and \( \gcd(c_0, N) = 1 \). Given such a GP, nonlinear algorithms consider sublattices of \( L_{m,d} \) contained in the \( \mathbb{Q} \)-vector space orthogonal to \([c_0, \ldots, c_d]\). The role of \( N \) in the definition of the sublattices is therefore made implicit, resulting in determinants that depend on the terms of the GP and not on \( N \) itself. Roughly speaking, a GP with terms that are small when compared to \( N \) is then expected to lead to a sublattice of \( L_{m,d} \) with small determinant. More generally, lattices contained in the \( \mathbb{Q} \)-vector space orthogonal to multiple linearly independent geometric progressions are considered.

There are two main problems that immediately arise from this approach: firstly, establishing a relationship between the size of terms in the geometric progressions and the determinant of the resulting lattices; and secondly, constructing geometric progressions with small terms. In the next section, tools are developed to address the first problem. There the object of study is the orthogonal lattice. A detailed description of nonlinear algorithms is given in Section 3.2. Based on the results of Section 3.1, criteria for the selection of geometric progressions are also given. In Section 3.3, existing solutions to the second problem are reviewed.

Throughout this paper, big-\( O \) estimates may have implied constants depending on the degree parameter \( d \).
3.1. **The orthogonal lattice.** Let $\Lambda$ be a lattice in $\mathbb{Z}^n$ and denote by $E_\Lambda$ the unique $\mathbb{Q}$-vector subspace of $\mathbb{Q}^n$ that is generated by any of its bases. The dimension of $E_\Lambda$ over $\mathbb{Q}$ is equal to the dimension of $\Lambda$. Let $E_\Lambda^\perp$ be the orthogonal complement of $E_\Lambda$ with respect to $\langle \ , \rangle$. The orthogonal lattice of $\Lambda$ is defined to be $\Lambda^\perp = \mathbb{Z}^n \cap E_\Lambda^\perp$. The intersection $\mathbb{Z}^n \cap E_\Lambda$ forms a lattice in $E_\Lambda$ (see [18 Proposition 1.1.3]), which shall be denoted throughout by $\overline{\Lambda}$. Clearly, $\overline{\Lambda}$ contains $\Lambda$ as a sublattice, and $\dim \overline{\Lambda} = \dim \Lambda$.

Proposition 1.3.4 in [18] implies that $\dim \Lambda^\perp = \dim E_\Lambda^\perp$ if and only if the dimension of $(\mathbb{Z}^n)^\perp \cap E_\Lambda^\perp$ is equal to $\dim E_\Lambda$. This clearly holds since $\overline{\Lambda} = \mathbb{Z}^n \cap E_\Lambda^\perp$. Hence, dim $\Lambda + \dim \Lambda^\perp = n$. Nguyen and Stern [20] Theorem 1] showed that the determinants of $\Lambda$ and $\Lambda^\perp$ are related as follows:

$$
\det \Lambda = [\overline{\Lambda} : \Lambda] \cdot \det \Lambda^\perp.
$$

Therefore, $\det \Lambda^\perp \leq \det \Lambda$ with equality if and only if $\Lambda = \overline{\Lambda}$. A lattice $\Lambda \subseteq \mathbb{Z}^n$ for which equality holds is called primitive. Let $B$ be a basis matrix for a $k$-dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$. Then $\Lambda$ is primitive if and only if the greatest common divisor of all $k \times k$ minors of $B$ is 1 (see [31 Corollary 4.1c]). The following lemma determines the index $[\overline{\Lambda} : \Lambda]$ in general.

**Lemma 3.1.** Let $\Lambda \subseteq \mathbb{Z}^n$ be a $k$-dimensional lattice and $B$ one of its basis matrices. Let $\Omega$ denote the greatest common divisor of all $k \times k$ minors of $B$. Then $[\overline{\Lambda} : \Lambda] = \Omega$.

**Proof.** Let $\overline{B}$ denote a basis matrix for $\overline{\Lambda}$. The lattice $\Lambda$ is a full-rank sublattice of $\overline{\Lambda}$, thus there exists a $k \times k$ integer matrix $U$ with $|\det U| = [\overline{\Lambda} : \Lambda]$ such that $B = U \cdot \overline{B}$. Hence, the lemma will follow by showing that $\Omega = |\det U|$.

For indices $1 \leq i_1 < \ldots < i_k \leq n$, let $B_{i_1 \ldots i_k}$ (resp. $\overline{B}_{i_1 \ldots i_k}$) denote the $k \times k$ submatrix of $B$ (resp. $\overline{B}$) formed by columns $i_1, \ldots, i_k$. Then $B_{i_1 \ldots i_k} = U \cdot \overline{B}_{i_1 \ldots i_k}$, for all $1 \leq i_1 < \ldots < i_k \leq n$. Therefore, $\Omega = |\det U| \cdot \overline{\Omega}$, where $\overline{\Omega}$ is the greatest common divisor of all $k \times k$ minors of $\overline{B}$. However, $\overline{\Omega} = 1$ as the lattice $\overline{\Lambda}$ is primitive. \hfill \Box

3.1.1. **The determinant under transformation.** For a $k$-dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$ and $S \in \text{GL}_n(\mathbb{R})$, define $\Lambda_S = \{x \cdot S \mid x \in \Lambda\}$. Given a basis $(b_1, \ldots, b_k)$ of $\Lambda$, define $(b_1, \ldots, b_k)_S = (b_1 S, \ldots, b_k S)$. Then $\Lambda_S$ is a $k$-dimensional lattice in $\mathbb{R}^n$ with basis $(b_1, \ldots, b_k)_S$.

**Lemma 3.2.** Let $\Lambda$ be a lattice in $\mathbb{Z}^n$ and $S \in \text{GL}_n(\mathbb{R})$. Then

$$
\det \Lambda_S^\perp = |\det S| \cdot \det \overline{\Lambda}_{S^{-1}},
$$

where $S^{-t} = (S^{-1})^t$ denotes the inverse transpose of $S$.

**Proof.** Fix a basis $(b_1, \ldots, b_k)$ for $\overline{\Lambda}$. The lattice $\overline{\Lambda}$ is primitive, thus $(b_1, \ldots, b_k)$ can be extended to a basis $(b_1, \ldots, b_n)$ for $\mathbb{Z}^n$ [4 Lemma 2, Chapter 1]. Since $\mathbb{Z}^n$ is unimodular, the dual basis $(b_1^\perp, \ldots, b_n^\perp)$ for $\mathbb{R}^n$ forms a basis for $\mathbb{Z}^n$. The dual basis is characterised by the equalities $\langle b_i^\perp, b_j \rangle = \delta_{i,j}$, where $\delta_{i,j}$ is the Kronecker delta. Therefore, $(b_1^\perp, \ldots, b_k^\perp)$ forms a basis for the orthogonal lattice $\Lambda^\perp$. Hence, $(b_1, \ldots, b_n)_{S^{-t}}$ forms a basis for $\mathbb{Z}^n_{S^{-t}}$, $(b_1, \ldots, b_k)_{S^{-t}}$ forms a basis for $\overline{\Lambda}_{S^{-t}}$ and $(b_1^\perp, \ldots, b_n^\perp)_{S^{-t}}$ forms a basis for $\Lambda_{S^{-t}}$.

For all $1 \leq i, j \leq n$,

$$
\langle b_i^\perp S, b_j S^{-t} \rangle = b_i^\perp SS^{-1}b_j^t = \langle b_i^\perp, b_j \rangle = \delta_{i,j}.
$$
Thus \((b_1^*, \ldots, b_n^*)_S\) is a dual basis of \((b_1, \ldots, b_n)_{S-\tau}\). Therefore, by applying \cite{18} Corollary 1.3.5] with \(E = \mathbb{R}^n\) and \(F\) equal to the subspace of \(\mathbb{R}^n\) generated by \((b_1, \ldots, b_k)_{S-\tau}\), it follows that
\[
|\det S|^{-1} = \det \mathbb{Z}_{S-\tau} = \det (\Lambda_{S-\tau}) \cdot \det (\Lambda_{S-\tau}^\perp)^{-1}.
\]

Given a basis for a lattice \(\Lambda \subseteq \mathbb{Z}^n\) and a diagonal matrix \(S \in \text{GL}_n(\mathbb{R})\), the following theorem provides a method for computing the determinant of \(\Lambda_{S}^\perp\).

**Theorem 3.3.** Let \(\Lambda \subseteq \mathbb{Z}^n\) be a \(k\)-dimensional lattice and \(B\) one of its basis matrices. For all indices \(1 \leq i_1 < \ldots < i_k \leq n\), denote by \(B_{i_1, \ldots, i_k}\) the \(k \times k\) submatrix of \(B\) formed by columns \(i_1, \ldots, i_k\). For nonzero reals \(S_1, \ldots, S_n\), define \(S = \text{diag}(S_1, \ldots, S_n)\). Then
\[
\det \Lambda_{S}^\perp = |S_1 \cdots S_n| \cdot \Omega^{-1} \sum_{1 \leq i_1 < \ldots < i_k \leq n} \left( \frac{\det B_{i_1, \ldots, i_k}}{S_{i_1} \cdots S_{i_k}} \right)^2,
\]
where \(\Omega\) is the greatest common divisor of all \(k \times k\) minors of \(B\).

**Proof.** The index of \(\Lambda\) in \(\mathbb{K}\) is invariant under scaling by the matrix \(S^{-1}\), i.e., \([\Lambda_{S^{-1}} : \Lambda_{S^{-1}}] = [\Lambda : \Lambda]\). Therefore, it follows from Lemma 3.1 and Lemma 3.2 that
\[
\det \Lambda_{S}^\perp = |S| \cdot \det \Lambda_{S^{-1}} = |S_1 \cdots S_n| \cdot \Omega^{-1} \cdot \det \Lambda_{S-1}.
\]
The matrix \(P = BS^{-1}\) forms a basis matrix for \(\Lambda_{S-1}\). For all indices \(1 \leq i_1 < \ldots < i_k \leq n\), let \(P_{i_1, \ldots, i_k}\) denote the \(k \times k\) submatrix of \(P\) formed by columns \(i_1, \ldots, i_k\). Using the Cauchy-Binet formula (see \cite{1} p. 86) to compute \(\det PP^t\), shows that
\[
\det \Lambda_{S-1} = \sum_{1 \leq i_1 < \ldots < i_k \leq n} \det (P_{i_1, \ldots, i_k})^2.
\]
The theorem then follows from the fact that \(P_{i_1, \ldots, i_k} = B_{i_1, \ldots, i_k} \cdot \text{diag}(S_{i_1}, \ldots, S_{i_k})^{-1}\), for all \(1 \leq i_1 < \ldots < i_k \leq n\). \(\square\)

### 3.1.2. Constructing a basis for the orthogonal lattice.
Let \(\Lambda\) be a \(k\)-dimensional lattice in \(\mathbb{Z}^n\) and \(B = (b_{i,j})\) one of its basis matrices. A basis for the orthogonal lattice \(\Lambda^\perp\) can be found by using either Algorithm 2.4.10 or Algorithm 2.7.2 in \cite{5} to compute a basis for the integer kernel of the matrix \(B\). The former algorithm is based on Hermite normal form computation (see \cite{5} Section 2.4.2) and the latter algorithm on the MLLL algorithm of Pohst \cite{29}. In practice, the MLLL based algorithm is preferable since it is more likely to avoid large integer arithmetic (see \cite{5} Section 2.4.3). Similarly, one can use LLL HNF algorithm of Havas, Majewski and Matthews \cite{9} Section 6]. If \(M = \max_j \|(b_{1,j}, \ldots, b_{n,j})\|_2^2\), then the algorithm performs \(O((n+k)^4 \log(nM))\) operation on integers of size \(O(n \log(nM))\) \cite{33}. The algorithm of Nguyen and Stern \cite{25} Algorithm 5] directly computes an LLL-reduced basis for \(\Lambda^\perp\). Given an \(n \times n\) diagonal matrix \(S\) with integer entries and nonzero determinant, the following modification of their algorithm produces an LLL-reduced basis for \(\Lambda_{S}^\perp\).

**Algorithm 3.4.**

**INPUT:** A basis matrix \(B = (b_{i,j})\) for a lattice \(\Lambda \subseteq \mathbb{Z}^n\), where \(k < n\). An \(n \times n\) diagonal matrix \(S\) with integer entries and nonzero determinant.

**OUTPUT:** An LLL-reduced basis for \(\Lambda_{S}^\perp\).
Let $(x)$ algorithm. Then

$$\text{Therefore, for } 1$$

$$\text{Thus } X$$

$$\text{Let } \Delta \text{ be the lattice with basis matrix } D.$$  

$$\text{Proof. Let } \Delta \text{ be the lattice with basis matrix } D. \text{ Given a vector } y = (y_1, \ldots, y_n) \in \mathbb{Z}^n,$$

$$\text{Therefore, } y \in \Lambda^\perp \text{ if and only if } (y_1 S_1, \ldots, y_n S_n, X(y, b_1), \ldots, X(y, b_n)).$$

$$\text{The existence of an LLL-reduced basis for } \Lambda^\perp \text{ gives rise to linearly independent vectors } y_1, \ldots, y_{n-k} \in \Delta \text{ such that }$$

$$\text{Let } (x_1, \ldots, x_n) \text{ be the LLL-reduced basis for } \Delta \text{ computed in Step 3 of the algorithm. Then }$$

$$\text{Thus } \mathcal{X}' = (\pi_{i_1}(x_1), \ldots, \pi_{i_1}(x_{n-k})) \text{ forms a basis for a sublattice of } \Lambda^\perp. \text{ If } \mathcal{X}' \text{ is not a basis of } \Lambda^\perp \subseteq \pi_{i_1}(\Delta), \text{ then there exist integers } z_{n-k+1}, \ldots, z_n, \text{ not all zero, such that }$$

$$\text{which is absurd. Hence, } \mathcal{X}' \text{ forms a basis for } \Lambda^\perp. \text{ It remains to show that } \mathcal{X}' \text{ is LLL-reduced. From the definition of an LLL-reduced basis, it follows that } (x_1, \ldots, x_{n-k}) \text{ inherits the property of being LLL-reduced from } (x_1, \ldots, x_n). \text{ If } (x_1^*, \ldots, x_{n-k}^*) \text{ is the Gram–Schmidt orthogonalisation of } (x_1, \ldots, x_{n-k}), \text{ then the last } k \text{ consecutive entries of } x_i^* \text{ must be 0, for } 1 \leq i \leq n-k. \text{ Therefore, }$$

$$\langle x_i, x_j^* \rangle = \langle \pi_i(x_i), \pi_i(x_j^*) \rangle \quad \text{and} \quad \langle x_i^*, x_j^* \rangle = \langle \pi_i(x_i^*), \pi_i(x_j^*) \rangle,$$

for $1 \leq i, j \leq n-k$. Hence, the Gram–Schmidt orthogonalisation of $\mathcal{X}'$ is equal to $(\pi_i(x_1^*), \ldots, \pi_i(x_{n-k}^*))$. Thus $\mathcal{X}'$ inherits the property of being LLL-reduced from $(x_1, \ldots, x_{n-k})$. □
3. Nonlinear polynomial generation in detail. To address the problem of constructing lattices with small determinants, the use of small geometric progressions modulo $N$ was briefly introduced in Section 3. To make matters more concrete, the ideas introduced there are now discussed in detail.

Nonlinear algorithms search for polynomials with coefficient vectors contained in the lattice orthogonal to linearly independent geometric progressions with ratio $m$ modulo $N$:

$$c_1 = [c_{1,0}, \ldots, c_{1,d}], c_2 = [c_{2,0}, \ldots, c_{2,d}], \ldots, c_k = [c_{k,0}, \ldots, c_{k,d}], \quad 1 \leq k < d.$$ 

Let $L$ denote the $k$-dimensional lattice with basis $(c_1, \ldots, c_k)$. Geometric progressions that are also a rational GP must be avoided. Otherwise, any nonlinear polynomial with coefficient vector in $L$ will be reducible. In general, $L$ may not be a sublattice of $L_{m,d}$. However, $L^⊥ \subset L_{m,d}$ whenever at least one GP $c_i$ has nonzero terms and $\gcd(c_i, N) = 1$. To obtained skewed polynomials, a skew parameter $s > 0$ is introduced and weights $S_i = s^{i-d/2}$ computed for $0 \leq i \leq d$. With $S = \text{diag}(S_0, \ldots, S_d)$, lattice reduction is then used to find an $\text{LLL}$-reduced basis $(b_1, \ldots, b_{d-k+1})$, with $b_i \in L^⊥$, for the lattice $L^⊥_S$. Finally, those polynomials with corresponding coefficient vectors $b_1$ and $b_2$ are returned.

In practice, the weights $S_i$ can be replaced by arbitrary positive real values. However, defining $S_i = s^{i-d/2}$ ensures that the length of a vector $(a_0 S_0, \ldots, a_d S_d) \in L^⊥_S$ and the skew 2-norm of the corresponding polynomial $f = \sum_{i=0}^d a_i x^i$ are related:

$$\|f\|_{2,s} = s^{\frac{d-k}{2}} \cdot \|(a_0 S_0, \ldots, a_d S_d)\|_2.$$ 

Therefore, if the vectors $b_1$ and $b_2$ correspond to degree $d$ polynomials $f_1, f_2 \in \mathbb{Z}[x]$ with nonzero resultant, then Corollary 2.6 and Theorem 2.6 imply that

$$N^{\frac{d}{2}} \leq \|f_1\|_{2,s} \cdot \|f_2\|_{2,s} \leq 2^{d-k} \cdot \gamma_{d-k+1} \cdot \det(L^⊥_S) \cdot \left(\frac{s}{a_{d-k+1}}\right)^{\frac{d-k}{2}}.$$ 

Consequently, when aiming to produce two polynomials of equal degree $d \geq 2$, the determinant of $L^⊥_S$ is of optimal size whenever $\det(L^⊥_S) = O(N^{(d-k)/2d})$.

The determinant of $L^⊥_S$ can be computed exactly using Theorem 3.3. However, this approach does not provide a clear intuition as to the relationship between the size of $\det(L^⊥_S)$ and the size of the geometric progressions $c_1, \ldots, c_k$. For $s > 0$ and $(x_0, \ldots, x_n) \in \mathbb{R}^{n+1}$, define

$$\|(x_0, \ldots, x_n)\|_{2,s} = \sqrt{\sum_{i=0}^{n} |x_i s^{i - \frac{d}{2}}|^2}.$$ 

Then a more illustrative relationship between the size of $\det(L^⊥_S)$ and the size of the geometric progressions is given by the following theorem.

**Theorem 3.6.** For linearly independent geometric progressions $c_1 = [c_{1,0}, \ldots, c_{1,d}], c_2 = [c_{2,0}, \ldots, c_{2,d}], \ldots, c_k = [c_{k,0}, \ldots, c_{k,d}], \quad 1 \leq k < d,$
with ratio in modulo $N$ and $\gcd(c_{i,0}, N) = 1$, let $L$ denote the lattice with basis $(c_1, \ldots, c_k)$. Then $L_S^\perp$ is $(d - k + 1)$-dimensional and

$$\det L_S^\perp \leq \frac{1}{N^{k-1}} \|c_1\|_{2,s-1} \cdots \|c_k\|_{2,s-1}.$$  

**Proof.** Observe that $c_i - (c_{i,0}c_{i,0}^{-1})c_1 \equiv 0 \pmod{N}$, for $2 \leq i \leq k$. Thus $N^{k-1}$ divides each $k \times k$ minor of the basis matrix $(c_{i,j})$ for $L$. Hence, Lemma 3.1 and Lemma 3.2 imply that $L_S^\perp$ is a $(d - k + 1)$-dimensional lattice and

$$\det L_S^\perp \leq (S_0 \cdots S_d) \cdot \frac{1}{N^{k-1}} \cdot \det L_{S-1} = \frac{1}{N^{k-1}} \cdot \det L_{S-1}.$$  

The proof is completed by using Hadamard’s inequality (see [31, Section 1.3]) to bound $\det L_{S-1}$.  

Theorem 3.6 provides a simple criterion for selecting geometric progressions: for a given skew $s > 0$, the best geometric progressions $c_1, \ldots, c_k$ are precisely those for which $\|c_i\|_{2,s-1}$ are small.

The construction of small geometric progressions is, by a large margin, the most difficult part of nonlinear polynomial generation. One approach to this problem, introduced by Montgomery [21, 20] and later extended by Koo, Jo and Kwon [13, Section 3], suggests constructing an initial GP $c = [c_0, \ldots, c_{l-1}]$ of length $l$, where $d < l < 2d$. Then $l - d$ geometric progressions of length $d + 1$ are obtained by taking successive terms:

$$c_1 = [c_0, \ldots, c_d], c_2 = [c_1, \ldots, c_{d+1}], \ldots, c_{l-d} = [c_{l-d-1}, \ldots, c_{l-1}].$$

If the vectors $c_1, \ldots, c_{l-d}$ do not form a basis for an $(l - d)$-dimensional sublattice of $L_{m,d}$, then $c$ is rejected. For $s > 0$, the product of the norms $\|c_i\|_{2,s-1}$ is bounded in terms of the initial GP:

$$\prod_{i=1}^{l-d} \|c_i\|_{2,s-1} = \prod_{i=1}^{l-d} s^{-\frac{(l-d)(l-d-1)}{2(l-d)}} \|c_i\|_{2,s-1} \leq \|c\|_{2,s-1}^{l-d}.$$  

To generate two degree $d$ polynomials with optimal size, Theorem 3.6 and (3.1) suggests that the initial geometric progression $c$ should satisfy

$$\|c\|_{2,s-1} = O(N^{\frac{(2d-1)(l-d)-(d-1)}{2d(l-d)}}).$$

For fixed $d$, the exponent of $N$ in (3.2) is a strictly increasing function of $l$. Therefore, the weakest size requirements on $c$ occur for $l = 2d - 1$ (corresponding to Montgomery’s algorithm). For this case, the orthogonal lattice is 2-dimensional, thus two linearly independent vectors of shortest length can be computed in polynomial time by using Lagrange’s algorithm (often called Gauss’ algorithm, see 28 and references therein). For large $N$, the problem of efficiently constructing geometric progressions satisfying (3.2) remains open for all parameters $(d, l) \notin \{(2,3), (3,5)\}$.

Koo, Jo and Kwon observed that at least one degree $d$ polynomial can be obtained for all $l/2 \leq d < l$. Therefore, distinct degree polynomial pairs can be obtained by varying the parameter $d$. This approach allows for nonlinear algorithms to be applied to $N$ of any size.
3.3. Existing Algorithms. In this section, existing nonlinear generation algorithms are briefly reviewed. A uniform analysis of the algorithms that appear in this section is provided in Section 4 and Section 5. Therefore, attention is limited to each algorithm’s methods of GP and basis construction. Examples are given for comparison between the algorithms.

3.3.1. Montgomery’s two quadratics algorithm. In Montgomery’s two quadratics algorithm (see [8, Section 5] and [24, Section 2.3.1]), geometric progressions of length \(d + 1 = 3\) are constructed by first selecting an integer \(p \geq 2\) (usually chosen to be prime) such that \(\gcd(p, N) = 1\) and \(N\) is quadratic residue modulo \(p\). Then one of the two possible values of \(m \in \mathbb{Z}\) satisfying \(m^2 \equiv N \pmod{p}\) and \(|m - N^{1/2}| \leq p/2\) is chosen. Finally, the GP is taken to be \([c_0, c_1, c_2] = [p, m, (m^2 - N)/p]\), with ratio \(mp^{-1}\) modulo \(N\). For any integer \(t \equiv c_2c_1^{-1} \pmod{c_0}\), the matrix

\[
\begin{pmatrix}
c_1 & -c_0 & 0 \\
-c_1 & 1 & 0 \\
c_0 & -t & 1
\end{pmatrix},
\]

forms a basis matrix for the orthogonal lattice of \([c_0, c_1, c_2]\).

For a given skew \(s > 0\), choosing \(p = O(s^{-1}\sqrt{N})\) guarantees that (3.2) holds. As a result, Montgomery’s algorithm is capable of producing polynomials with optimal coefficient size. However, the restriction to quadratic polynomials means that the algorithm is not suitable for \(N\) containing more than 110–120 digits [24, Section 2.3.1]. Examples of polynomials generated using Montgomery’s two quadratics algorithm can be found in [8, Section 10].

3.3.2. The Williams and Prest–Zimmermann algorithms. Williams [34, Chapter 4] introduced another length 3 GP construction for producing pairs of quadratic polynomials. Roughly speaking, the new geometric progressions are obtained by setting \(p = 1\) in Montgomery’s construction. Williams also provided a length 4 GP construction for producing pairs of cubic polynomials. In both of Williams’ algorithms, the skew parameter is restricted to \(s = 1\). Prest and Zimmermann [30] extended Williams’ algorithms to include skews \(s \neq 1\) leading to a reduction in coefficient norms for the cubic algorithm. In addition, they generalised their algorithm to arbitrary degrees.

In the algorithms of Williams and Prest–Zimmermann, geometric progressions of length \(d + 1\) are constructed by first selecting an integer \(m\) with \(|md^2 - N| = O(N^{1-1/d})\). Then the GP is taken to be

\([c_0, \ldots, c_d] = [1, m, \ldots, m^{d-1}, md - N]\),

with ratio \(m\) modulo \(N\). The matrix

\[
\begin{pmatrix}
-c_1 & 1 & 0 & \ldots & 0 \\
-c_2 & 0 & 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
-c_d & 0 & 0 & \ldots & 1
\end{pmatrix},
\]

forms a basis matrix for the orthogonal lattice of \([c_0, \ldots, c_d]\).

Examples of polynomials found with the Williams and Prest–Zimmermann algorithms are found in [34, Chapter 5] and [30]. For comparison between the algorithms of this section, the following example is considered throughout.
Example 3.7. Let

\[ N = c91 = 456717603989410870435875216065628192034927306 \]

96982839773907434662898832715547522284379393.

With \( m = \left\lceil N^{1/3} \right\rceil = 16591382811472980794587079218 \), Williams [34, Chapter 5] obtained the cubic polynomials:

\[
\begin{align*}
    f_1 &= 8962732699933084116x^3 - 2027077443432188756x^2 \\
        &- 9743585171161776159x + 98228437934803482,
    \\
    f_2 &= 62526200906654277101x^3 - 141413847455697130658x^2 \\
        &- 161279695637696264892x - 88601408057407884491.
\end{align*}
\]

The product of coefficient norms \( \|f_1\|_2 \cdot \|f_2\|_2 \) is approximately \( N^{0.445} \). The product \( \|f_1\|_{2,s} \cdot \|f_2\|_{2,s} \) is minimised for \( s_{opt} \approx 1.763 \) with \( \|f_1\|_{2,s_{opt}} \cdot \|f_2\|_{2,s_{opt}} \approx N^{0.443} \).

Applying Prest and Zimmermann’s algorithm with \( m = \left\lceil N^{1/3} \right\rceil \) and \( s = 10^8 \), the following pair of cubic polynomials is obtained:

\[
\begin{align*}
    g_1 &= 10363104x^3 - 23437957x^2 - 2114716857651221243486x \\
         &- 109084939899748327411476171840,
    \\
    g_2 &= 4776851x^3 - 10803677x^2 + 150352771504116048021555x \\
         &- 10008782251408057407884491.
\end{align*}
\]

The product of coefficient norms \( \|g_1\|_{2,10^8} \cdot \|g_2\|_{2,10^8} \) is approximately \( N^{0.422} \). The product \( \|g_1\|_{2,s} \cdot \|g_2\|_{2,s} \) is minimised for \( s_{opt} \approx 45278023 \) with \( \|g_1\|_{2,s_{opt}} \cdot \|g_2\|_{2,s_{opt}} \approx N^{0.419} \). Consequently, the polynomials \( g_1 \) and \( g_2 \) have an optimised product of coefficient norms that is approximately 147 times smaller than that of \( f_1 \) and \( f_2 \).

3.3.3. The Koo–Jo–Kwon algorithms. Koo, Jo and Kwon [13, Section 4.1] generalised Montgomery’s GP construction to arbitrary degrees. They construct geometric progressions of length \( d + 1 \) by first selecting positive integers \( p = O((kN)^{1/d}) \) and \( k = O(1) \) such that \( x^d \equiv kN \pmod p \) has a nonzero solution. An integer \( m \) satisfying \( m^d \equiv kN \pmod p \) and \( |m - \sqrt[kN]{p}| \leq p/2 \) is chosen. Then the GP is taken to be

\[
[c_0, \ldots, c_d] = \left[p^{d-1}, p^{d-2} m, \ldots, m^{d-1}, \frac{m^d - kN}{p}\right],
\]

with ratio \( mp^{-1} \) modulo \( N \). This construction is seen to reduce to Montgomery’s construction for parameters \( d = 2, k = 1 \); and the constructions of Williams and Prest–Zimmerman for \( p = k = 1 \).

The Koo–Jo–Kwon and Prest–Zimmermann algorithms each produce polynomials which satisfy the same theoretical bounds on coefficient norms (see Section 4.1). However, for any given \( N \), the additional parameters \( p \) and \( k \) allow for a wealth of new geometric progressions to be constructed. As a result, polynomials with significantly smaller coefficients may be found in practice.

Example 3.8. Let \( N = c91 \). Applying the Koo–Jo–Kwon algorithm with \( s = 10^8, k = 1, p = 776112641898 \) and \( m = \left\lceil N^{1/3} \right\rceil + 5 \), the following pair of cubic
polynomials is obtained:

\[
\begin{align*}
    h_1 &= 124932x^3 - 276x^2 + 590020231905564605626x \\
    &+ 79893857071973416869543365671, \\
    h_2 &= 156165x^3 - 345x^2 + 737525290075983917507x \\
    &- 314917248946851224111717562717.
\end{align*}
\]

The product of coefficient norms \(\|h_1\|_{2,10^8} \cdot \|h_2\|_{2,10^8}\) is approximately \(N^{0.383}\). The product \(\|h_1\|_{2,s} \cdot \|h_2\|_{2,s}\) is minimised for \(s_{\text{opt}} \approx 106759349\) with \(\|h_1\|_{2,s_{\text{opt}}} \cdot \|h_2\|_{2,s_{\text{opt}}} \approx N^{0.383}\). Consequently, the polynomials \(h_1\) and \(h_2\) have an optimised product of coefficient norms that is approximately 1770 times smaller than that of \(g_1\) and \(g_2\) from Example 4.1.

By extending their length \(d+1\) GP construction, Koo, Jo and Kwon [13, Section 4.2] obtained a construction for length \(d+2\) geometric progressions. The construction begins by selecting positive integers \(p = \Theta((kN)^{1/d})\) and \(k = O(1)\) such that \(x^d \equiv kN \pmod{p^2}\) has a nonzero solution \(m = \Theta(p)\). Then the GP is taken to be

\[
[c_0, \ldots, c_{d+1}] = \left[ p^{d-1}, p^{d-2}m, \ldots, m^{d-1}, \frac{m^d - kN}{p}, \frac{m(m^d - kN)}{p^2} \right],
\]

with ratio \(mp^{-1}\) modulo \(N\). Koo, Jo and Kwon do not analyse their algorithm for skews \(s \neq 1\). This analysis is undertaken in Section 5 where it is shown that the algorithm improves upon previous algorithms for \(d \geq 3\) with polynomials of optimal size produced when \(d = 3\). However, this improvement is offset in part by the additional complexity of determining suitable parameters \(m, p\) and \(k\).

4. LENGTH \(d + 1\) CONSTRUCTION REVISITED

Each of the length \(d+1\) GP constructions discussed in Section 4.3 led to geometric progressions \([c_0, \ldots, c_d]\) for which \([c_0, \ldots, c_{d-1}]\) forms a rational GP. The following theorem determines all such geometric progressions that, in addition, satisfy the properties necessary for polynomial generation.

**Theorem 4.1.** Let \([c_0, \ldots, c_d]\) be a GP modulo \(N\) with \(d \geq 2\) and nonzero terms. Suppose that the following properties are satisfied:

1. \(\gcd(c_0, N) = 1\);
2. \([c_0, \ldots, c_{d-1}]\) is a rational GP;
3. \([c_0, \ldots, c_{d-1}, c_d]\) is not a rational GP.

Then there exist nonzero integers \(a, m\) and \(k\) with \(\gcd(m, p) = 1\) such that

\[
[c_0, \ldots, c_d] = \left[ ap^{d-1}, ap^{d-2}m, \ldots, am^{d-1}, \frac{am^d - kN}{p} \right].
\]

**Proof.** Let \([c_0, \ldots, c_d]\) satisfy the conditions of the theorem. The second property implies the existence of nonzero integers \(a\) and \(m\) with \(\gcd(m, p) = 1\) such that \(c_i = ap^{d-i}m^i\), for \(0 \leq i \leq d-1\). Consequently, \(\gcd(ap, N) = 1\) as a result of the first property. If \(c_{d/2}c_{d/2} - c_0c_d = 0\), then \([c_0, \ldots, c_d]\) is a rational GP violating the third property. Therefore, there exists a nonzero integer \(l\) such that

\[
lN = c_{d/2}c_{d/2} - c_0c_d = ap^{d-2} \left( am^d - pc_d \right).
\]

Hence, \(am^d - pc_d = kN\) for some nonzero \(k\) in \(\mathbb{Z}\). \(\square\)
Given an arbitrary GP \([c_0, \ldots, c_{l-1}]\) with nonzero terms and length \(l \geq 3\), \([c_0, c_1]\) forms a rational GP with ratio \(c_1c_0^{-1}\). The following corollary is therefore a direct consequence of Theorem 4.1.

**Corollary 4.2.** Let \([c_0, \ldots, c_{l-1}]\) is a GP modulo \(N\) with \(l \geq 3\) and nonzero terms. Suppose that the following properties are satisfied:

1. \(\gcd(c_0, N) = 1\);
2. \([c_0, \ldots, c_{l-1}]\) is not a rational GP.

If \(2 \leq d < l\) is the largest index such that \([c_0, \ldots, c_{d-1}]\) forms a rational GP, then there exist nonzero integers \(a, m, p, k\) with \(\gcd(m, p) = 1\) such that \([c_0, \ldots, c_d]\) is given by (4.1).

As a consequence of Theorem 4.1, the following nonlinear generation algorithm is obtained:

**Algorithm 4.3.**

**Input:** An integer \(d \geq 2\). Positive integers \(a, p, m\) and \(k\) such that \(\gcd(ap, N) = 1\), \(\gcd(m, p) = 1\); and \((am^d - kN)/p\) is a nonzero integer. A positive integer \(s\).

**Output:** A pair of integer polynomials \(f_1\) and \(f_2\) with common root \(mp^{-1}\) modulo \(N\).

1. Compute \(c_i = ap^d - i - 1\) for \(0 \leq i \leq d - 1\); and \(c_d = (am^d - kN)/p\).
2. Compute weights \(S_i = s^{i-d/2}\), for \(0 \leq i \leq d\).
3. Let \(S = \text{diag}(S_0, \ldots, S_d)\). Use Algorithm 3.4 to compute an \(\text{LLL}-\text{reduced basis} (b_1, \ldots, b_d)\) for the lattice \(L_S\) (see Remark 2.1 below).
4. For \(i = 1, 2\), write \(b_i = (a_{i,0}, \ldots, a_{i,d})\) and return the polynomial \(f_i = \sum_{j=0}^{d} a_{i,j}x^{j}\).

The length \(d + 1\) GP construction in Step 1 of Algorithm 4.3 reduces to the construction of Montgomery’s two quadratics algorithm for parameters \(d = 2\), \(a = k = 1\); the constructions of the Williams and Prest–Zimmerman algorithms for \(a = p = k = 1\); and the construction of the Koo–Jo–Kwon algorithm for \(a = 1\). In the next section, parameter selection for Algorithm 4.3 is considered.

**Remark 4.4.** In Step 3 of Algorithm 4.3, a reduced basis for \(L_S\) can be found by first computing an \(\text{LLL}-\text{reduced basis} for L_{S'}\), where \(S' = \text{diag}(1, s, \ldots, s^d)\). Given a reduced basis \((b_1, \ldots, b_d)\) for \(L_{S'}\), the definition of \(\text{LLL}-\text{reduced bases} (\text{Definition 2.3})\) then implies that \((s^{-1}b_1, \ldots, s^{-1}b_d)_{S'}\) is also reduced. The later is equal to \((b_1, \ldots, b_d)_{S}\), a basis for \(L_S\). The restriction by the algorithm to \(s \in \mathbb{Z}\) ensures that a reduced basis for \(L_{S'}\) can be found by using Algorithm 3.4.

### 4.1. Parameter selection for Algorithm 4.3

Throughout this section, notation from Algorithm 4.3 is retained. In addition, let \(c = [c_0, \ldots, c_d]\). Then the polynomials \(f_1\) and \(f_2\) satisfy

\[
(4.2) \quad f_i \left( \frac{m}{p} \right) \cdot p^d = \frac{p}{a} \langle b_i, c \rangle + \frac{a_{i,d}k}{a}N = \frac{a_{i,d}k}{a}N, \quad \text{for } i = 1, 2.
\]

In Section 1, it was noted that root properties play a key role in determining the yield of number field sieve polynomials. Polynomial roots are divided into two classes: projective and non-projective (see [24] Section 3.2 for a definition). When \(a = 1\), Koo, Jo and Kwon [13] Remark 5] noted that choosing \(k\) to contain a
product of small primes improves the non-projective root properties of \( f_1 \) and \( f_2 \). More generally, (4.2) shows that selecting \( a \) and \( k \) to contain small prime factors can be used to aid both projective and non-projective root properties. However, the parameters \( a \) and \( k \) should be chosen small as \( a / \gcd(a,k) \) divides \( a_{i,d} \), for \( i = 1, 2 \); and \( kN / \gcd(a,k) \) divides \( \text{Res}(f_1, f_2) \).

For \( k = 1 \), the parameter spaces of Algorithm 4.3 and Kleinjung’s algorithm [11] coincide. The methods discussed by Kleinjung for efficiently generating parameters can be carried over to this setting and are readily extended to include \( k \neq 1 \). The reader is also referred to [13, Section 4.1] for a discussion of parameter selection when \( a = 1 \). Therefore, the problem of generating parameters will not be discussed here. Instead, it is shown that under an appropriate choice of parameters, Algorithm 4.3 can be used to obtain degree \( d \) polynomials \( f_1, f_2 \in \mathbb{Z}[x] \) with

\[
\| f_i \|_{2,s} = O \left( N^{(1/d)(d^2-2d+2)/(d^2-d+2)} \right), \quad \text{for } i = 1, 2.
\]

This yields polynomials of size \( O(N^{1/4}) \), for \( d = 2 \); \( O(N^{5/24}) \), for \( d = 3 \); and \( O(N^{5/28}) \), for \( d = 4 \). The exponent for \( d = 2 \) is optimal as a result of Corollary 2.9. The bound (4.3) is obtained without any assumptions on the size of vectors in LLL-reduced bases. This is in contrast to the previous analyses of [30, 13].

Applying Theorem 2.6 the determinant of \( L_S^1 \) satisfies

\[
\det L_S^1 \leq (S_0 \cdots S_d) \cdot \sqrt{\frac{c_0^2}{S_0^2} + \cdots + \frac{c_d^2}{S_d^2}} = \sqrt{\frac{c_0^2}{S_0^2} + \cdots + \frac{c_d^2}{S_d^2}}.
\]

For \( 0 \leq i \leq d - 1 \),

\[
\frac{c_i}{S_i} = a p^{d-i-1} m^i s^{\frac{d}{2}-i} = a p^{d-1} s^{\frac{d}{2}} \left( \frac{m}{ps} \right)^i.
\]

Let \( \tilde{m} = \sqrt{\frac{kn}{a}} \) and assume \( m \geq \tilde{m} \). Then

\[
\frac{c_d}{S_d} = \frac{am^d - kN}{p} s^{-\frac{d}{2}} = a \left( m^d - \tilde{m}^d \right) s^{-\frac{d}{2}} < \frac{d(m - \tilde{m})}{ps} a p^{d-1} s^{\frac{d}{2}} \left( \frac{m}{ps} \right)^{d-1}.
\]

Therefore, for parameters \( p \) and \( s \) satisfying \( \sqrt{d}(m - \tilde{m}) \leq ps \leq m \),

\[
(4.4) \quad \det L_S^1 \leq \sqrt{2^d a s^{1-\frac{d}{2}} m^{d-1}}.
\]

To minimise the determinant of \( L_S^1 \), it follows that the skew parameter \( s \) should be chosen as large as possible and \( m \approx \tilde{m} \). However, the size of \( s \) is limited by the requirement that two degree \( d \) polynomials are found.

For a nonzero polynomial \( f \) with coefficient vector \( x \in L^1 \) and degree less than \( d \), (4.2) implies that \( f(mp^{-1}) = 0 \). Thus \( f \) must contain a monomial with nonzero coefficient divisible by \( m \). Accordingly, the coefficient vector \( x \) satisfies \( \|x\|_{2,s} > s^{-d/2}m \). Therefore, if the basis vectors \( b_1 \) and \( b_2 \) in the reduced basis for \( L_S^1 \) both satisfy \( \|b_i\|_{2,s} \leq s^{-d/2}m \), then \( f_1 \) and \( f_2 \) each have degree equal to \( d \).

Below it is shown that selecting \( s \) so that \( \|b_1\|_{2,s} \leq s^{-d/2}m \) holds is sufficient to guarantee that two degree \( d \) polynomial satisfying (4.3) can be found.

Theorem 2.6 and (4.3) imply that \( \|b_1\|_{2,s} \leq s^{-d/2}m \) whenever

\[
2^{d-1} \left( \sqrt{2^d a s^{1-\frac{d}{2}} m^{d-1}} \right) \leq s^{-\frac{d}{2}} m.
\]
Rearranging for \( s \) gives the bound

\[
s \leq \frac{1}{\sqrt{2}} \left( \frac{m}{\sqrt{d} \ a} \right)^{\frac{3}{2d^2 - d + 2}}.
\]

Recall that \( s \) should be chosen as large as possible and \( m \approx \tilde{m} \) in order to minimise the determinant of \( L_\perp \). Therefore, parameters should be chosen to satisfy

\[
m \geq \left( \frac{kN}{a} \right)^{\frac{1}{2}} \quad \text{and} \quad s = \left[ \frac{1}{\sqrt{2}} \left( \frac{m}{\sqrt{d} \ a} \right)^{\frac{3}{2d^2 - d + 2}} \right], \quad \sqrt{d} (m - \tilde{m}) \leq ps \leq m,
\]

with \( m = \Theta(\tilde{m}) \). For such parameters, \( f_1 \) is of degree \( d \) with \( f_1(mp^{-1}) \neq 0 \), and substituting into the bound \( \|b_1\|_{2,s} \leq s^{-d/2} m \) shows that

\[
\|f_1\|_{2,s} = O \left( \frac{k^{\frac{d}{2}(d-1)}}{a^d} \cdot \frac{k^2}{d^2 - d + 2} \cdot N \cdot \frac{k^{\frac{d}{2}-d/2}}{d^2 - d + 2} \right).
\]

Setting \( a = O(1) \) and \( k = O(1) \) leads to \( f_1 \) satisfying the bound in (1.3).

Repeating the analysis for \( m \leq \tilde{m} \) once again leads to parameters for which (1.3) is obtained. In both cases, the parameters satisfy (4.6)

\[
m = \Theta \left( \left( \frac{kN}{a} \right)^{\frac{1}{2}} \right), \quad s = \Theta \left( \left( \frac{kN}{a^{d+1}} \right)^{\frac{3}{2d^2 - d + 2}} \right), \quad \sqrt{d} |m - \tilde{m}| \leq ps \leq m.
\]

For parameters satisfying (4.6), the condition \( \|b_1\|_{2,s} \leq s^{-d/2} m \) is now used to show that \( b_2 \) satisfies \( \|b_2\|_{2,s} = O(s^{-d/2} m) \). Therefore, if the degree of \( f_2 \) is equal to \( d \), then (4.3) holds. Otherwise, (1.3) is satisfied by the degree \( d \) polynomials \( f_1 \) and \( f_1 + f_2 \).

Assume (4.6) holds and \( \|b_1\|_{2,s} \leq s^{-d/2} m \). Then the vector \( b = (-m, p, 0, \ldots, 0) \) in \( L_\perp \) satisfies

\[
\|b\|_{2,s} = \sqrt{(s^{-d/2} m)^2 + (s^{1-d/2} p)^2} \leq \sqrt{2} s^{-d/2} m.
\]

Moreover, the vectors \( b_1, b \in L_\perp \) are linearly independent since \( \deg f_1 = d \). Hence, \( \lambda_2(L_\perp^\perp) = O(s^{-d/2} m) \) and Theorem 2.6 implies that \( \|b_2\|_{2,s} = O(s^{-d/2} m) \).

**Remark 4.5.** The above arguments show that a degree \( d \) polynomial

\[
f_{j_1,j_2,j_3}(x) = j_1 \cdot f_1(x) + j_2 \cdot f_2(x) + j_3 \cdot (px - m), \quad j_1, j_2, j_3 \in \mathbb{Z}_n
\]

will satisfy \( \|f_{j_1,j_2,j_3}\|_{2,s} = O(s^{-d/2} m) \) whenever \( j_i = O(1) \), for \( i = 1, 2, 3 \). Therefore, it is possible to obtain multiple pairs of degree \( d \) polynomials that satisfy (1.3). Moreover, a sieve-like procedure such as that described in [24] Procedure 5.1.6 may be used to identify \( f_{j_1,j_2,j_3} \), with good root properties.

**5. The Koo–Jo–Kwon Length \( d + 2 \) Construction Revisited**

By utilising their length \( d + 2 \) GP construction, Koo, Jo and Kwon obtained an algorithm for producing nonlinear polynomials of degree at most \( d \) such that the coefficient of \( x^{d-1} \) in each polynomial is equal to zero \([13]\) Corollary 4]. Number field sieve polynomials with second highest coefficient equal to zero had previously been considered for linear algorithms by Kleinjung \([10]\). There the motivation was to produce polynomials with large skew in order to leverage practical advantages. In this section, it is shown that larger skews, when compared to those in Section 4.1, are able to be used in the Koo–Jo–Kwon algorithm. As a result, nonlinear
polynomial pairs with smaller coefficient norms are obtained. To begin this section, minor improvements to the Koo–Jo–Kwon algorithm are now given.

It follows immediately from Corollary 4.2 that the length $d + 2$ GP construction of Koo, Jo and Kwon [13, Section 4.2] can be extended: if $a$, $p$, $k$, and $m$ are positive integers that satisfy $\gcd(ap, N) = 1$, $\gcd(m, p) = 1$ and $am^d \equiv kN \pmod{p^2}$, then

\begin{equation}
[c_0, \ldots, c_{d+1}] = \left[ ap^{d-1}, ap^{d-2}m, \ldots, am^{d-1}, \frac{am^d - kN}{p}, \frac{m(am^d - kN)}{p^2} \right],
\end{equation}

is a GP with ratio $mp^{-1}$ modulo $N$. The Koo–Jo–Kwon construction then corresponds to the special case $a = 1$. Given a GP defined by (5.1), the proof of [13, Corollary 4] is readily modified to show that an integer polynomial $f = \sum_{i=0}^{d} a_i x^i$ with coefficient vector orthogonal to both $[c_0, \ldots, c_d]$ and $[c_1, \ldots, c_{d+1}]$ must have $a_{d-1} = 0$. A stronger statement is given by the following lemma.

**Lemma 5.1.** Let $a$, $p$, $m$, $k$ and $N$ be nonzero integers and $[c_0, \ldots, c_{d+1}]$ be defined by (5.1). For any vector $(a_0, \ldots, a_d) \in \mathbb{Z}^{d+1}$, the following conditions are equivalent:

1. $(a_0, \ldots, a_d)$ is orthogonal to $[c_0, \ldots, c_d]$ and $[c_1, \ldots, c_{d+1}]$.
2. $a_{d-1} = 0$ and $(a_0, \ldots, a_d)$ is orthogonal to $(c_1, \ldots, c_{d-1}, 0, c_{d+1})$.

**Proof.** By construction,

$$[c_0, \ldots, c_d] - pm^{-1}[c_1, \ldots, c_{d+1}] = [0, \ldots, 0, m^{-1}kN, 0].$$

Hence, $(a_0, \ldots, a_d) \in \mathbb{Z}^{d+1}$ is orthogonal to $[c_0, \ldots, c_d]$ and $[c_1, \ldots, c_{d+1}]$ if and only if $a_{d-1} = 0$ and $(a_0, \ldots, a_d)$ is orthogonal to the linearly dependent vectors $(c_0, \ldots, c_{d-2}, 0, c_d)$ and $(c_1, \ldots, c_{d-1}, 0, c_{d+1})$. \[\square\]

Lemma 5.1 permits a somewhat smaller lattice to be used in the Koo–Jo–Kwon algorithm, thus offering a minor practical advantage. The improved algorithm can be described as follows:

**Algorithm 5.2.**

**Input:** An integer $d \geq 3$. Nonzero integers $a$, $p$, $k$ and $m$ such that $\gcd(ap, N) = 1$, $\gcd(m, p) = 1$; and $(am^d - kN)/p^2$ is a nonzero integer. A positive integer $s$.

**Output:** A pair of integer polynomials $f_1$ and $f_2$ with common root $mp^{-1}$ modulo $N$.

1. Compute $c_i = ap^{d-i-2}m^i$, for $0 \leq i \leq d - 2$; and $c_{d-1} = (am^d - kN)/p^2$.
2. Compute weights $S_i = s^{d-i/2}$, for $0 \leq i \leq d - 2$; and $S_{d-1} = s^{d/2}$.
3. Let $L = (c_0, \ldots, c_{d-1})\mathbb{Z}$ and $S = \text{diag}(S_0, \ldots, S_{d-1})$. Use Algorithm 5.1 to compute an LLL-reduced basis $(b_1, \ldots, b_{d-1})_S$ for the lattice $L_S^\perp$.
4. For $i = 1, 2$, write $b_i = (a_{i,0}, \ldots, a_{i,d-2}, a_{i,d})$ and return the polynomial $f_i = a_{i,d}x^d + \sum_{j=0}^{d-2} a_{i,j}x^j$.

In the next section, parameter selection for Algorithm 5.2 is considered.

In Section 3.2, it was noted that a length $l$ geometric progressions can be used to generate degree $d$ polynomials for all $l/2 \leq d < l$. Given a geometric progression $c = [c_0, \ldots, c_{d+1}]$ defined by (5.1), it is therefore possible to generate polynomials of degree $d$ and $d+1$, for $d \geq 2$. Generating polynomials of degree less than $d$ should not be considered as $[c_0, \ldots, c_{d-1}]$ forms a rational GP. A degree $d + 1$ polynomial
\[ f = \sum_{i=0}^{d+1} a_i x^i \] with coefficient vector orthogonal to \( c \) will satisfy
\[
    f \left( \frac{m}{p} \right) \cdot p^{d+1} = \frac{kN}{a} (a_{d+1} m + a_d p).
\]

Hence, following the approach of Section 4.1 and choosing parameters so that \( f(mp^{-1}) \neq 0 \) is not sufficient to guarantee that \( f \) has degree equal to \( d + 1 \). Parameter selection is therefore more difficult and will not be addressed here.

5.1. Parameter selection for Algorithm 5.2. Throughout this section, notation from Algorithm 5.2 is retained. In addition, let \( c = [c_0, \ldots, c_d] \). Then the polynomials \( f_1 \) and \( f_2 \) satisfy
\[
    f_i \left( \frac{m}{p} \right) \cdot p^d = \frac{p^2}{a} (b_i, c) + \frac{a_i d k}{a} N = \frac{a_i d k}{a} N, \quad \text{for } i = 1, 2.
\]

Therefore, similar to Section 4.1, the parameters \( a \) and \( k \) can be utilised to aid the root properties of \( f_1 \) and \( f_2 \). Generating parameters for Algorithm 5.2 is significantly more difficult than for Algorithm 4.3. This problem has, in effect, been considered by Kleinjung [10] and Koo–Jo–Kwon [13, Section 4.2]. Therefore, the problem will not be discussed here. Instead, the problem of selecting parameters that minimise the coefficient norms of \( f_1 \) and \( f_2 \) is now considered.

Theorem 5.3 implies that
\[
    \det L_S \leq (S_0 \cdots S_{d-1}) \cdot \sqrt{\frac{c_0^2}{S_0^2} + \cdots + \frac{c_{d-1}^2}{S_{d-1}^2}} = s^{1 - \frac{d}{2}} \cdot \sqrt{\frac{c_0^2}{S_0^2} + \cdots + \frac{c_{d-1}^2}{S_{d-1}^2}}.
\]

By following the analysis of Section 4.1, the parameter space of Algorithm 5.2 can be restricted in such a way as to guarantee the degree of \( f_1 \) is equal to \( d \) and
\[
    \|f_1\|_{2,s} = O \left( \frac{a^{\frac{3d-4}{2d^2-3d+4}} \cdot p^{-\frac{d}{2d^2-3d+4}} \cdot k^{\frac{d}{2d^2-3d+4}} \cdot N^{\frac{d}{2d^2-3d+4}}} {a} \right).
\]

The restricted parameters then satisfy
\[
    m = \Theta \left( \left( \frac{kN}{a} \right)^{\frac{1}{2}} \right), \quad s = \Theta \left( \left( \frac{P}{a} \right)^{\frac{2}{2d^2-3d+4}} \right), \quad \sqrt{d} |m - \tilde{m}| \leq ps \leq m.
\]

Clearly, the parameter \( p \) should be chosen as large as possible. By enforcing \( p = \Theta(m/s) \),
\[
    \|f_1\|_{2,s} = O \left( \frac{a^{\frac{2(2d-3)}{2d^2-3d+6}} \cdot k^{\frac{d^2-4d+6}{2d^2-3d+6}} \cdot N^{\frac{d^2-4d+6}{2d^2-3d+6}}} {a} \right),
\]

where \( s = \Theta((kN/a^{d+1})/(2d)/(d^2-3d+6)) \). Similar to Section 4.1 if the degree of \( f_2 \) is not equal to \( d \), then a second degree \( d \) polynomial can be found by considering linear combinations of the polynomials \( f_1, f_2 \) and \( px - m \). Finally, by setting \( a = O(1) \) and \( k = O(1) \), it follows that Algorithm 5.2 can be used to obtain a pair of degree \( d \) polynomials with
\[
    \|f_i\|_{2,s} = O \left( N^{(1/d)(d^2-4d+6)/(d^2-3d+6)} \right), \quad \text{for } i = 1, 2.
\]

This yields polynomials of size \( O(N^{1/6}) \), for \( d = 3 \); and \( O(N^{3/20}) \), for \( d = 4 \). The exponent for \( d = 3 \) is optimal as a result of Corollary 2.3.
Acknowledgements

The author would like to thank Dr Victor Scharaschkin for many helpful discussions and suggestions throughout the preparation of this paper.

References

[1] A.C. Aitken. Determinants and Matrices. Oliver and Boyd, Edinburgh, ninth edition, 1956.
[2] Kazumaro Aoki, Jens Franke, Thorsten Kleinjung, Arjen K. Lenstra, and Dag Arne Osvik. A kilobit special number field sieve factorization. In Advances in cryptology—ASIACRYPT 2007, volume 4833 of Lecture Notes in Comput. Sci., pages 1–12. Springer, Berlin, 2007.
[3] J. P. Buhler, H. W. Lenstra, Jr., and Carl Pomerance. Factoring integers with the number field sieve. In The development of the number field sieve, volume 1554 of Lecture Notes in Math., pages 50–94. Springer, Berlin, 1993.
[4] J. W. S. Cassels. An introduction to the geometry of numbers. Springer-Verlag, Berlin, 1971.
[5] Henri Cohen. A course in computational algebraic number theory, volume 138 of Graduate Texts in Mathematics. Springer-Verlag, Berlin, 1993.
[6] Richard Crandall and Carl Pomerance. Prime numbers: A computational perspective. Springer, New York, second edition, 2005.
[7] Alicia Dickenstein and Ioannis Z. Emiris, editors. Solving polynomial equations, volume 14 of Algorithms and Computation in Mathematics. Springer-Verlag, Berlin, 2005. Foundations, algorithms, and applications.
[8] Marije Elkenbracht-Huizing. An implementation of the number field sieve. Experiment. Math., 5(3):231–253, 1996.
[9] George Havas, Bohdan S. Majewski, and Keith R. Matthews. Extended GCD and Hermite normal form algorithms via lattice basis reduction. Experiment. Math., 7(2):125–136, 1998.
[10] Thorsten Kleinjung. Polynomial selection. Slides presented at the CADO workshop, Nancy, France, 2008. 30 pages, available at http://cado.gforge.inria.fr/workshop/slides/kleinjung.pdf.
[11] Thorsten Kleinjung. On polynomial selection for the general number field sieve. Math. Comp., 75(256):2037–2047 (electronic), 2006.
[12] Thorsten Kleinjung, Kazumaro Aoki, Jens Franke, Arjen Lenstra, Emmanuel Thomé, 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. In CRYPTO 2010 Advances in Cryptology - CRYPTO 2010 (Santa Barbara, USA, 2010), T. Rabin, Ed., volume 6223 of Lecture Notes in Comput. Sci., pages 333–350. Springer-Verlag, Berlin, 2010.
[13] Namhun Koo, Gooc Hwa Jo, and Soonhak Kwon. On nonlinear polynomial selection and geometric progression (mod n) for number field sieve. Cryptology ePrint Archive, Report 2011/292, 2011. http://eprint.iacr.org/2011/292.pdf.
[14] A. K. Lenstra and H. W. Lenstra, Jr., editors. The development of the number field sieve, volume 1554 of Lecture Notes in Mathematics. Springer-Verlag, Berlin, 1993.
[15] A. K. Lenstra, H. W. Lenstra, Jr., M. S. Manasse, and J. M. Pollard. The number field sieve. In The development of the number field sieve, volume 1554 of Lecture Notes in Math., pages 11–42. Springer, Berlin, 1993.
[16] Arjen K. Lenstra, Jr. Hendrik W. Lenstra, and László Lovász. Factoring polynomials with rational coefficients. Math. Ann., 261(4):515–534, 1982.
[17] Hendrik W. Lenstra, Jr. Lattices. In Algorithmic number theory: lattices, number fields, curves and cryptography, volume 44 of Math. Sci. Res. Inst. Publ., pages 127–181. Cambridge Univ. Press, Cambridge, 2008.
[18] Jacques Martinet. Perfect lattices in Euclidean spaces, volume 327 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]. Springer-Verlag, Berlin, 2003.
[19] Alexander May. New RSA Vulnerabilities Using Lattice Reduction Methods. PhD thesis, University of Paderborn, 2003.
[20] Peter L. Montgomery. Searching for higher-degree polynomials for the general number field sieve. http://www.ipam.ucla.edu/publications/scws1/scws1_6223.ppt, 2006. PowerPoint presentation, 34 pages.

[21] Peter L. Montgomery. Small geometric progressions modulo n. Unpublished note of 2 pages, December 1993.

[22] Brian Murphy and Richard P. Brent. On quadratic polynomials for the number field sieve. In Computing theory '98 (Perth), volume 20 of Aust. Comput. Sci. Commun., pages 199–213. Springer, Singapore, 1998.

[23] Brian A. Murphy. Modelling the yield of number field sieve polynomials. In Algorithmic number theory (Portland, OR, 1998), volume 1423 of Lecture Notes in Comput. Sci., pages 137–150. Springer, Berlin, 1998.

[24] Brian A. Murphy. Polynomial selection for the number field sieve integer factorisation algorithm. PhD thesis, Australian National University, July 1999.

[25] Phong Nguyen and Jacques Stern. Merkle-Hellman revisited: a cryptanalysis of the Qu-Vanstone cryptosystem based on group factorizations. In Advances in cryptology—CRYPTO '97 (Santa Barbara, CA, 1997), volume 1294 of Lecture Notes in Comput. Sci., pages 198–212. Springer, Berlin, 1997.

[26] Phong Q. Nguyễn and Damien Stehlé. Floating-point LLL revisited. In Advances in cryptology—EUROCRYPT 2005, volume 3494 of Lecture Notes in Comput. Sci., pages 215–233. Springer, Berlin, 2005.

[27] Phong Q. Nguyen and Damien Stehlé. An LLL algorithm with quadratic complexity. SIAM J. Comput., 39(3):874–903, 2009.

[28] Phong Q. Nguyen and Damien Stehlé. Low-dimensional lattice basis reduction revisited. ACM Trans. Algorithms, 5(4):Art. 46, 48, 2009.

[29] Michael E. Pohst. A modification of the LLL reduction algorithm. J. Symbolic Comput., 4(1):123–127, 1987.

[30] Thomas Prest and Paul Zimmermann. Non-linear polynomial selection for the number field sieve. Research report, CARAMEL - INRIA Nancy - Grand Est / LORIA - INRIA - CNRS : UMR7503 - Université Henri Poincaré - Nancy I - Université Nancy II - Institut National Polytechnique de Lorraine, 2010. http://hal.inria.fr/inria-00540483/PDF/polyselect.pdf.

[31] Alexander Schrijver. Theory of linear and integer programming. Wiley-Interscience Series in Discrete Mathematics. John Wiley & Sons Ltd., Chichester, 1986. A Wiley-Interscience Publication.

[32] Robert D. Silverman. Optimal parameterization of SNFS. J. Math. Cryptol., 1(2):105–124, 2007.

[33] Wilberd Van Der Kallen. Complexity of the Havas, Majewski, Matthews LLL Hermite normal form algorithm. J. Symbolic Comput., 30(3):329–337, 2000.

[34] Ronnie S. Williams, Jr. Cubic polynomials in the number field sieve. Master’s thesis, Texas Tech University, May 2010.

School of Mathematics and Physics, The University of Queensland, Brisbane, QLD 4072, Australia
E-mail address: ncoxon@maths.uq.edu.au