Fast Computation of the Smith Normal Form of an Integer Matrix

Mark Giesbrecht†
Department of Computer Science
University of Manitoba
Winnipeg, Manitoba
Canada, R3T 3T6
mwg@cs.umanitoba.ca

Abstract

We present two new probabilistic algorithms for computing the Smith normal form of an $A \in \mathbb{Z}^{m\times n}$. The first requires an expected number of $O(m^2 n \cdot \mathrm{M}(m \log \|A\|))$ bit operations (ignoring logarithmic factors) and is of the Las Vegas type; that is, it never produces an incorrect answer. Here $\|A\| = \max_{ij} |A_{ij}|$ and $\mathrm{M}(l)$ bit operations are sufficient to multiply two l-bit integers ($\mathrm{M}(l) = l^2$ using standard arithmetic). This improves on the previously best known (deterministic) algorithm of Hafner and McCurley, which requires about $O(m^3 n \log \|A\| \cdot \mathrm{M}(m \log \|A\|))$ bit operations. We also present an even faster, more space efficient algorithm which requires an expected number of $O((m^3 n \log \|A\| + m^3 \log^2 \|A\|) \cdot \log(1/\epsilon))$ bit operations *using standard integer arithmetic.* This algorithm is of the Monte Carlo type: it returns the correct result with probability at least $1-\epsilon$ for a user specified tolerance $\epsilon > 0$. This algorithm also requires only $O(nm \log \|A\|)$ bits of storage, versus $O(nm^2 \log \|A\|)$ bits required by other known algorithms.

Introduction

It was proven by Smith (1861) that any $A \in \mathbb{Z}^{m\times n}$ is equivalent to a unique diagonal $S \in \mathbb{Z}^{m\times n}$ under unimodular transformations. That is, there exist unimodular $P \in \mathbb{Z}^{m\times m}$ and $Q \in \mathbb{Z}^{n\times n}$ (i.e., $\det P, \det Q = \pm 1$) such that

$$S = PAQ = \begin{pmatrix} s_1 & & & & \\ & \ddots & & \mathbf{0} & \\ & & s_r & & \\ & & & 0 & \\ & & & & \ddots \\ & \mathbf{0} & & & & 0 \end{pmatrix}$$
$$= \mathrm{diag}(s_1, \ldots, s_r, 0, \ldots, 0) \in \mathbb{Z}^{m\times n}$$

where $r = \mathrm{rank}(A)$ and $s_i \,|\, s_{i+1}$ for $1 \le i \le r-1$. S is called the *Smith normal form* of A and the non-zero diagonal elements of S the *invariant factors* of A.

†Research was supported in part by Natural Sciences and Engineering Research Council of Canada research grant OGP0155376

Appears in Proceedings of the International Symposium on Symbolic and Algebraic Computation, ISSAC'95

Computing the Smith normal form of an integer matrix is useful in many applications, including Diophantine analysis (see Newman 1972) and determining the canonical structure of Abelian groups. Recently, algorithms for the Smith normal form have been used to compute the structure of the class group of a number field (see Hafner & McCurley 1989, Buchmann 1988).

In this paper we present two new probabilistic algorithms to compute the Smith normal form of an integer matrix. Their costs are substantially smaller than the previously best known (deterministic) algorithm of Hafner & McCurley (1989,1991; see also Kannan & Bachem 1979, Chou & Collins 1982, Domich *et al.* 1987, Iliopolous 1989).

Suppose $A \in \mathbb{Z}^{m\times n}$ has rank r. Without loss of generality we assume throughout that $m \le n$ — the Smith normal form is invariant under transpose. In Section 1 we present a fast new probabilistic algorithm to compute the Smith normal form of A. It requires an expected number of $O\tilde{~}(m^2 n \cdot \mathrm{M}(m \log \|A\|))$ bit operations and is of the Las Vegas type, that is, it always produces the correct answer. For convenience we use "soft-Oh" notation: for any $f, g : \mathbb{R}^s \to \mathbb{R}$, $f = O\tilde{~}(g)$ if and only if $f = O(g \cdot \log^c g)$ for some constant $c > 0$. The deterministic algorithm of Hafner & McCurley (1991) requires $O\tilde{~}(m^3 n \log \|A\| \cdot \mathrm{M}(m \log \|A\|))$ bit operations in the worst case. Like many previous algorithms, ours computes modulo a multiple d of the non-zero invariant factors of A. It requires an expected number of $O(rmn \log\log d)$ operations in $\mathbb{Z}_d$. Such a d is easily found, but may contain $O(m(\log \|A\| + \log m))$ bits. Hafner & McCurley's (1991) algorithm also computes modulo d, but requires $O(rmn \log d)$ operations in $\mathbb{Z}_d$. Our algorithm is related to Hafner & McCurley's, but with a random, unimodular column operation before each phase of row reduction. The worst case cost of Hafner & McCurley's algorithm may well be encountered only on a small portion of inputs, but our randomization avoids it on all inputs with high probability.

In Section 2 we present an even faster probabilistic algorithm for the Smith normal form of $A \in \mathbb{Z}^{m\times n}$ which is considerably more space efficient as well. Define θ such that we can multiply two $r \times r$ matrices over a ring R with $O(r^\theta)$ operations in R; using standard matrix arithmetic $\theta = 3$, while the best know algorithm of Coppersmith & Winograd (1990) allows $\theta = 2.38$. This new algorithm to compute the Smith normal form of $A \in \mathbb{Z}^{m\times n}$ then requires an expected number of

$$O\tilde{~}((m^\theta n \log \|A\| + m^3 \log^2 \|A\|) \cdot \log(1/\epsilon))$$

bit operations *using standard integer arithmetic.* This algorithm is probabilistic of the Monte Carlo type: ϵ is a user specified parameter, and the output of the algorithm is guaranteed correct with probability greater than $1-\epsilon$. By comparison Hafner & McCurley's (1991) algorithm requires $O\tilde{}(m^5 n \log^3 \|A\|)$ bit operations using standard integer arithmetic (and $O\tilde{}(m^4 n \log^2 \|A\|)$ bit operations using currently impractical but asymptotically fast arithmetic). Moreover, this new algorithm requires additional storage for only $r = \mathrm{rank}(A)$ "large" integers, each with $O\tilde{}(r \log \|A\|)$ bits. These integers converge on the determinantal divisors of A as the algorithm proceeds. Previous algorithms, which work with dense $m \times n$ matrices in $\mathbb{Z}_d$, require $O(nm \log d)$ or $O\tilde{}(nm^2 \log \|A\|)$ bits of storage.

Our Monte Carlo algorithm is more akin to the methods of Kaltofen *et al.* (1987) for computing Smith normal forms of matrices of polynomials than to the modulo-determinant algorithms discussed. Hafner & McCurley (1991) conjectured that these methods might be adapted to work for integer matrices. Our algorithm demonstrates this, showing that we can find the Smith normal form of an integer matrix with about the same order of cost as is required to compute the determinant of that matrix. Additional memory required is on the order of the size of the input and output.

In neither the Las Vegas modulo-determinant algorithm of Section 1 nor the Monte Carlo algorithm of Section 2 do we compute the transition matrices to the Smith normal form. Because of the generally large sizes of the entries in the transition matrices, one cannot expect to compute them in the order of time our algorithms require. However, many applications of Smith normal form computations, such as determining the structure of the class group of a number field, do not require the transition matrices be computed.

Definitions and Notation.

For integers n and $k \le n$, define $\mathcal{C}_k^n = \{(c_1, \ldots, c_k) \in \mathbb{N}^k : 1 \le c_1 < \cdots < c_k \le n\}$. In a principal ideal ring P, with $B \in \mathsf{P}^{m\times n}$, $(b_1, \ldots, b_k) \in \mathcal{C}_k^m$ and $(c_1, \ldots, c_k) \in \mathcal{C}_k^n$ define the $(k \times k)$ *minor* of a B:

$$B\binom{b_1 \cdots b_k}{c_1 \cdots c_k} = \det \begin{pmatrix} B_{b_1 c_1} & B_{b_1 c_2} & \cdots & B_{b_1 c_k} \\ B_{b_2 c_1} & B_{b_2 c_2} & \cdots & B_{b_2 c_k} \\ \vdots & & & \vdots \\ B_{b_k c_1} & B_{b_k c_2} & \cdots & B_{b_k c_k} \end{pmatrix} \in \mathsf{P}.$$

The *kth determinantal divisor* $d_k \in \mathsf{P}$ *of* B is defined as the GCD of all $k \times k$ minors of B. A well known definition of the invariant factors of B is as a quotient of determinantal divisors of B. If B has Smith normal form $S = \mathrm{diag}(s_1, \ldots, s_r, 0, \ldots, 0) \in \mathsf{P}^{m\times n}$ then $s_1 = d_1$ and $s_i = d_i/d_{i-1}$ for $2 \le i \le r$. Thus if we can find all the determinantal divisors of B, we can easily recover the invariant factors of B.

1 A Las Vegas Algorithm for the Smith normal form

In this section we present a fast Las Vegas type probabilistic algorithm for computing the Smith normal form of an $A \in \mathbb{Z}^{m\times n}$ of rank r and $m \le n$. Like many previous algorithms (see Hafner & McCurley 1991), this algorithm computes in $\mathbb{Z}_d$, where d is a multiple of the non-zero invariant factors of A. Our new algorithm requires an expected $O(mnr \log\log d)$ operations in $\mathbb{Z}_d$, and *always* produces the correct answer. A suitable d with $\lceil \log_2 d \rceil = O(m(\log m + \log \|A\|))$ bits can be found quickly, yielding an algorithm which costs $O\tilde{}(m^2 n \cdot \mathrm{M}(m \log \|A\|))$ bit operations.

The basis for modulo-determinant computation of the Smith normal form of A is as follows (see Hafner & McCurley 1989, Section 3). Let $A \in \mathbb{Z}^{m\times n}$ have rank r and Smith form $S = \mathrm{diag}(s_1, \ldots, s_r, 0, \ldots, 0) \in \mathbb{Z}^{m\times n}$. Let d be a multiple of $s_1 \cdots s_r$ and $\bar{A} = A \bmod d$. If $\bar{S} = \mathrm{diag}(\bar{s}_1, \ldots, \bar{s}_r, 0, \ldots, 0) \in \mathbb{Z}_d^{m\times n}$ is the Smith normal form of $\bar{A}$ over $\mathbb{Z}_d$, then $d_i = \gcd(\bar{s}_1 \cdots \bar{s}_i, d) \in \mathbb{Z}$ is the ith invariant factor of A for $1 \le i \le r$ and $s_1 = d_1$, $s_i = d_i/d_{i-1}$ for $2 \le i \le r$.

We initially count operations in $\mathbb{Z}_d$. For $a, b \in \mathbb{Z}_d$ we can find $a+b, a-b, ab \in \mathbb{Z}_d$, and determine if $a \mid b$ (i.e., $a = bc$ for some $c \in \mathbb{Z}_d$) with $O(\mathrm{M}(\log d) \log\log d)$ bit operations. In this same time we can also find

$$\gcd{}_d(a, b) = \left\{ \begin{array}{r} \gcd(d, \bar{a}, \bar{b}) \bmod d : \bar{a}, \bar{b} \in \mathbb{Z} \\ \bar{a} \equiv a \bmod d, \bar{b} \equiv b \bmod d \end{array} \right\} \in \mathbb{Z}_d,$$

the "least" principal generator of the ideal $(a, b) \subseteq \mathbb{Z}_d$.

Fact 1.1 (Hafner & McCurley 1991) *Let $\bar{A} \in \mathbb{Z}_d^{m\times n}$ be such that the first column is not all zeros. We can find a $\bar{P} \in \mathbb{Z}_d^{m\times m}$ with $\det \bar{P} = \pm 1$ such that $\bar{B} := \bar{P}\bar{A} \in \mathbb{Z}_d^{m\times n}$ satisfies $\bar{B}_{11} = \gcd_d(\bar{A}_{11}, \ldots, \bar{A}_{m1})$ and $\bar{B}_{i1} = 0$ for $2 \le i \le m$. The algorithm requires $O(nm)$ operations in $\mathbb{Z}_d$.*

Lemma 1.2 *Assume $\bar{A} \in \mathbb{Z}_d^{m\times n}$ and $\bar{v} = (1, v_2, \ldots, v_n) \in \mathbb{Z}_d^{n\times 1}$ is chosen randomly and uniformly from $\mathbb{Z}_d$. If $w = (w_1, \ldots, w_m)^t := \bar{A}\bar{v}$ then*

$$\begin{aligned} \mathrm{Prob}\{\gcd{}_d(\ w_1, \ldots, w_m) & \\ &= \gcd{}_d\{\bar{A}_{ij} : 1 \le i \le m, 1 \le j \le n\} \\ &\ge 1/(16 \log\log d). \end{aligned}$$

Proof Let p be a prime dividing d, $e = \min\{\mathrm{ord}_p(\bar{A}_{ij}) : 1 \le i \le m, 1 \le j \le n\}$ and k, l such that $\mathrm{ord}_p(\bar{A}_{kl}) = e$ (where $\mathrm{ord}_p(a) = \max\{i : p^i \mid a\}$ for $a \in \mathbb{Z}_d$). Let $x_2, \ldots, x_n$ be indeterminates and

$$f_k = \bar{A}_{k1} + \sum_{2 \le j \le n} x_j \bar{A}_{kj} \in \mathbb{Z}_d[x_2, \ldots, x_n].$$

Then $w_k = f_k(v_2, \ldots, v_n)$ and we can write $f_k = p^e g_k$, where $g_k \in \mathbb{Z}_d[x_2, \ldots, x_n] \setminus \{0\}$ and $g_k \not\equiv 0 \bmod p$. If each x_i is assigned a random v_i from $\mathbb{Z}_d$ for $2 \le i \le n$ then

$$\begin{aligned} \mathrm{Prob}\{\mathrm{ord}_p(w_k) = e\} &= \mathrm{Prob}\{g_k(v_2, \ldots, v_n) \not\equiv 0 \bmod p\} \\ &\ge 1 - 1/p \end{aligned}$$

by Corollary 1 of Schwartz (1980). Thus

$$\begin{aligned} \mathrm{Prob}\{\mathrm{ord}_p(\ \gcd{}_d(w_1, \ldots, w_m)) & \\ &= \mathrm{ord}_p(\gcd{}_d\{\bar{A}_{ij} : {\scriptstyle 1 \le i \le m \atop 1 \le j \le n}\})\} \\ &\ge 1 - 1/p. \end{aligned}$$

Since we choose $v_2, \ldots, v_n$ randomly and uniformly from $\mathbb{Z}_d$, the above probability is independent for each prime p dividing d. Thus,

$$\begin{aligned} \mathrm{Prob}\Big\{\ \gcd{}_d(w_1, \ldots, w_m) &= \gcd{}_d\{\bar{A}_{ij} : {\scriptstyle 1 \le i \le m \atop 1 \le j \le n}\}\Big\} \\ &= \prod_{p \mid d} \mathrm{Prob}\left\{ \begin{array}{l} \mathrm{ord}_p(\gcd_d(w_1, \ldots, w_m)) \\ \quad = \mathrm{ord}_p(\gcd_d\{\bar{A}_{ij} : {\scriptstyle 1 \le i \le m \atop 1 \le j \le n}\}) \end{array} \right\} \\ &\ge \prod_{p \mid d} (1 - 1/p) = \phi(d)/d, \end{aligned}$$

where $\phi(d)$ is Euler's totient function. By Rosser & Schoenfeld (1962), (3.41) and (3.42), $\phi(d)/d > 1/(1.8 \log\log d + 2.5036/\log\log d) > 1/(16 \log\log d)$ for all $d \geq 2$. $\square$

We now present a Las Vegas algorithm for the Smith normal form algorithm of an $\bar{A} \in \mathbb{Z}_d^{m\times n}$

Algorithm: `ModSmithForm`

Input: $\bar{A} \in \mathbb{Z}_d^{m\times n}$;

Output: the non-zero invariant factors (in $\mathbb{Z}_d$) of $\bar{A}$;

(1) If $\bar{A}$ is the zero matrix then quit;

(2) If $m = 1$ then output $\gcd_d(\bar{A}_{11}, \ldots, \bar{A}_{1n}) \in \mathbb{Z}_d$ and quit;

(3) Choose random $v_2, \ldots, v_n \in \mathbb{Z}_d$. If $\bar{A} = [\bar{C}_1|\cdots|\bar{C}_n]$, where $\bar{C}_i \in \mathbb{Z}_d^{m\times 1}$ is the ith column of $\bar{A}$, let $\bar{C}_1' := \bar{C}_1 + \sum_{2\leq i\leq n} v_i\bar{C}_i \in \mathbb{Z}_d^{m\times 1}$ and $\bar{A}' := [\bar{C}_1'|\bar{C}_2|\cdots|\bar{C}_n]$; if the first column of $\bar{A}'$ is all zeros, repeat (3);

(4) Find $\bar{P} \in \mathbb{Z}_d^{m\times m}$ with $\det \bar{P} = \pm 1$ such that $\bar{B} := \bar{P}\bar{A}'$ has $\bar{B}_{i1} = 0$ for $2 \leq i \leq m$;

(5) If $\bar{B}_{11} \mid \bar{B}_{ij}$ for $1 \leq i \leq m$, $1 \leq j \leq n$ then output $\bar{B}_{11}$ and call `ModSmithForm` on the submatrix $\bar{B}' \in \mathbb{Z}_d^{(m-1)\times(n-1)}$ of $\bar{B}$ formed by deleting the first row and first column, otherwise goto (3);

Theorem 1.3 *The algorithm* `ModSmithForm` *works correctly as specified. It requires an expected $O(mn\bar{r}\log\log d)$ operations in $\mathbb{Z}_d$, where $\bar{r} = \operatorname{rank}(\bar{A})$, the number of non-zero invariant factors of $\bar{A}$ in $\mathbb{Z}_d$.*

Proof In step (3) the transformation from $\bar{A}$ to $\bar{A}'$ is unimodular so the Smith form of $\bar{A}$ equals that of $\bar{A}'$. Since $\bar{B}_{11} = \gcd_d(\bar{A}_{11}', \ldots, \bar{A}_{m1}') \in \mathbb{Z}_d$, by Lemma 1.2 $\bar{B}_{11} \mid \bar{B}_{ij}$ in $\mathbb{Z}_d$ for $1 \leq i \leq m$, $1 \leq j \leq n$ with probability at least $1/16 \log\log d$. Thus, with this probability $\bar{B}_{11}$ is the first invariant factor of $\bar{A}$. If this is the case then $\bar{B}$ is column equivalent to $\begin{pmatrix} \bar{B}_{11} & 0 \\ 0 & \bar{B}' \end{pmatrix}$, and the remaining invariant factors of $\bar{A}$ are exactly the invariant factors of $\bar{B}'$. These are found by the recursion in step (5).

By Lemma 1.2 we execute steps (3)-(5) an expected number of $16 \log\log d$ times in each recursive call. These steps require $O(nm)$ operations in $\mathbb{Z}_d$ by Fact 1.1 and there are at most $\bar{r}$ recursive calls. $\square$

Theorem 1.4 *Given an $A \in \mathbb{Z}^{m\times n}$ with $m \leq n$, we can compute the Smith normal form S of A with an expected $O(m^2 n \cdot \mathrm{M}(m(\log\|A\| + \log m)))$ bit operations. The algorithm is probabilistic of the Las Vegas type.*

Proof Compute $r = \operatorname{rank}(A)$ and a multiple d of the non-zero invariant factors of A. This can be done with

$$O(m^2 n \mathrm{M}(m(\log\|A\| + \log m)))$$

bit operations by Proposition 2.3 of Hafner & McCurley (1991). Next compute the Smith normal form

$$\bar{S} = \operatorname{diag}(\bar{s}_1, \ldots, \bar{s}_r, 0, \ldots, 0) \in \mathbb{Z}_d^{m\times n}$$

of $\bar{A} = A \bmod d$. This requires $O(m^2 n)$ operations in $\mathbb{Z}_d$ by Theorem 1.3. Note that it may be that $\bar{r} = \operatorname{rank}(\bar{A}) < r$ in which case $\bar{s}_i = 0$ for some i, and $s_i = d$. This only happens if $d = 1$, in which case $\bar{r} = 0$ and $S = \operatorname{diag}(1, \ldots, 1, 0, \ldots, 0) \in \mathbb{Z}^{m\times n}$ (with r ones on the diagonal), or if $d > 1$, $S = \operatorname{diag}(1, \ldots, 1, d, 0, \ldots, 0)$ and $\bar{r} = r - 1$. The algorithm works correctly in both these instances.

For $1 \leq i \leq r$, let $d_i = \gcd(\bar{s}_1 \cdots \bar{s}_i, d) \in \mathbb{Z}$ and $s_1 = d_1$, $s_i = d_i/d_{i-1}$ for $2 \leq i \leq r$. The output is then $S = \operatorname{diag}(s_1, \ldots, s_r, 0, \ldots, 0) \in \mathbb{Z}^{m\times n}$. $\square$

2 A faster, space efficient Smith form algorithm

In this section we present our new, faster and more space efficient Monte Carlo algorithm for the Smith normal form of an $A \in \mathbb{Z}^{m\times n}$ of rank r. The algorithm is related to that of Kaltofen *et al.* (1987) for computing Smith normal forms of matrices of polynomials over a field K. They show that for $A \in \mathsf{K}[x]^{n\times n}$, to compute the kth determinantal divisors d_k you need only consider the GCD of the leading minors of two random perturbations of A, and not of the $\binom{n}{k}^2$ minors of A indicated by the definition. Recall the *leading $k \times k$ minor* of any $B \in \mathbb{Z}^{m\times n}$ is $B\binom{1\ldots k}{1\ldots k}$. For randomly chosen $R, T, U, V \in \mathsf{K}^{n\times n}$ they prove that d_k equals the GCD of the leading $k\times k$ minors of RAT and UAV for *all* k $(1 \leq k \leq n)$ with high probability. They obtain a probabilistic algorithm in the parallel complexity class RNC^2 for the Smith normal form of a polynomial matrix.

Our new algorithm for the Smith normal form is based on a new characterization in Subsection 2.1 of the determinental divisors of A as the contents of some $f_1, \ldots, f_r \in \mathbb{Z}[x_1, \ldots, x_s]$, where $s = r(n+m)$ (recall that the content $\operatorname{cont}(f_i) \in \mathbb{Z}$ of f_i is the GCD of all the coefficients of f_i). While we cannot write down these $f_1, \ldots, f_r$ efficiently, we can evaluate them quickly, i.e., we provide a black box which computes $f_1(\vec{a}), \ldots, f_r(\vec{a}) \in \mathsf{R}$ for any $\vec{a} \in \mathsf{R}^s$, where R is a small extension ring of $\mathbb{Z}$. In Subsection 2.2 we show how to find the contents of a list of polynomials given by a black box, using only a small number of evaluations on "small" input points. The black box we need for the Smith normal form computation finds all the leading minors of a $B \in \mathsf{R}^{r\times r}$. In Subsection 2.3 we show how find these very quickly. Finally, in Subsection 2.4 we tie these results together to obtain a fast, space efficient, Monte Carlo algorithm for the Smith normal form of an integer matrix.

2.1 A new characterization of the determinental divisors

The following theorem gives a different and very useful characterization of the determinental divisors of a matrix as the content of the leading minors of a matrix of polynomials.

Theorem 2.1 *Let $A \in \mathbb{Z}^{m\times n}$ have rank r. Let $X = (X_{ij})$ be an $r \times m$ matrix and $Y = (Y_{lk})$ an $n \times r$ matrix of algebraically independent indeterminates in $\Lambda = \{X_{ij}, Y_{lk} : 1 \leq i, k \leq r; 1 \leq j \leq m; 1 \leq l \leq n\}$. Then the content of the kth leading minor $f_k \in \mathbb{Z}[\Lambda]$ of $B = XAY \in \mathbb{Z}[\Lambda]^{r\times r}$ equals the kth determinental divisor $d_k \in \mathbb{Z}$ of A, for $1 \leq k \leq r$.*

Proof Using the Binet-Cauchy formula (Gantmacher 1990, p. 9) we find

$$f_k = C\begin{pmatrix}1\ldots k\\1\ldots k\end{pmatrix} = \sum_{\substack{(b_1,\ldots,b_k)\in C_k^m\\(c_1,\ldots,c_k)\in C_k^n}} X\begin{pmatrix}1\ldots k\\b_1\ldots b_k\end{pmatrix} A\begin{pmatrix}b_1\ldots b_k\\c_1\ldots c_k\end{pmatrix} Y\begin{pmatrix}c_1\ldots c_k\\1\ldots k\end{pmatrix}.$$

for $1 \leq k \leq r$. The kth determinental divisor d_k of A by definition divides $A\binom{b_1\ldots b_k}{c_1\ldots c_k}$ for all $(b_1, \ldots, b_k) \in C_k^m$ and $(c_1, \ldots, c_k) \in C_k^n$, so $d_k \mid \operatorname{cont}(f_k)$.

Now we show $\mathrm{cont}(f_k) \mid d_k$. Assume to the contrary that there exists a prime p and integer e such that $p^e \mid \mathrm{cont}(f_k)$ but $p^e \nmid d_k$. Thus, there exists an $k \times k$ minor $\lambda \in \mathbb{Z}$ of A such that $p^e \nmid \lambda$. Since $k \leq r$, there exists a $P \in \mathbb{Z}^{r \times m}$ and $Q \in \mathbb{Z}^{n \times r}$ such that λ is the leading minor of $PAQ \in \mathbb{Z}^{r \times r}$, so $f_k(P, Q) = \lambda$. Therefore λ is divisible by the content of f_k, whence $p^e \mid \lambda$, a contradiction. □

We do not compute the leading minors of $XAY \in \mathbb{Z}[\Lambda]^{r \times r}$ explicitely, but instead compute them by a *black-box* subroutine. That is, we will assign $X := P \in \mathsf{R}^{r \times m}$ and $Y := Q \in \mathsf{R}^{n \times r}$ for some randomly chosen P and Q over an extension ring R of $\mathbb{Z}$. We then find the leading minors of $PAQ \in \mathsf{R}^{r \times r}$. If $f_k \in \mathbb{Z}[\Lambda]$ is the leading $k \times k$ minor of XAY, then the leading $k \times k$ minor of PAQ is $f_k(P, Q) \in \mathsf{R}$. The algorithm `FindContent` in the next subsection employs this black box subroutine to find the contents of $f_1, \ldots, f_r$.

2.2 Finding the content of black-box polynomials

Our goal in this subsection is to find the contents of $f_1, \ldots, f_r \in \mathbb{Z}[x_1, \ldots, x_s] \setminus \{0\}$, each with degree at most ν. This list of polynomials is given by a black box, that is, we do not have an explicit representation of each polynomial as a linear combination of monomials, but for an extension ring R of $\mathbb{Z}$ and $a_1, \ldots, a_s \in \mathsf{R}$ can compute

$$(f_1(a_1, \ldots, a_s), \ldots, f_r(a_1, \ldots, a_s)) \in \mathsf{R}^r$$

with one evaluation of the black box. We aim to find the contents with as few evaluations of our black box as possible, on the "smallest" points possible.

Informally, the idea is as follows. We maintain a vector $(c_1, \ldots, c_r) \in \mathbb{Z}^r$ which contains an "approximation" to $(\mathrm{cont}(f_1), \ldots, \mathrm{cont}(f_r))$. Initially we find a point $\vec{a} \in \mathbb{Z}^s$ such that if $(c_1^{(0)}, \ldots, c_r^{(0)}) := (f_1(\vec{a}), \ldots, f_r(\vec{a}))$, then $c_1^{(0)}, \ldots, c_r^{(0)} \neq 0$. If $f_i = d_i g_i$, where $d_i = \mathrm{cont}(f_i) \in \mathbb{Z}$ and $g_i \in \mathbb{Z}[x_1, \ldots, x_s]$ has content 1, then clearly $d_i \mid c_i^{(0)}$ for $1 \leq i \leq r$.

Convergence of the algorithm is measured in terms of the differences in the orders of primes dividing c_i and d_i. Entering iteration $j \geq 1$, we choose a "random" $\vec{a} \in \mathbb{Z}^s$ and evaluate the black box to obtain $(b_1, \ldots, b_r) := f(\vec{a})$. Now let $(c_1^{(j)}, \ldots, c_r^{(j)}) := (\gcd(b_1, c_1^{(j-1)}), \ldots, \gcd(b_r, c_r^{(j-1)}))$. Certainly $d_i \mid c_i^{(j)}$ and $c_i^{(j)} \mid c_i^{(j-1)}$ for $1 \leq i \leq r$. Also, for each i and each prime $p \mid d_i$, $\mathrm{ord}_p(c_i^{(j-1)}) \geq \mathrm{ord}_p(c_i^{(j)}) \geq \mathrm{ord}_p(\mathrm{cont}(f_i))$ and $(c_1^{(j)}, \ldots, c_r^{(j)})$ will (hopefully) "converge" on $(d_1, \ldots, d_r)$. Informally this follows since the probability $g_i(\vec{a}) \equiv 0 \bmod p$ is usually low, so the probability $\mathrm{ord}_p(d_i) = \mathrm{ord}_p(c_i^{(j)})$ is quite high. Iterating this processes allows us to make the probability of error arbitrarily small.

Unfortunately, this method does not necessarily work for small primes p, and the order of such a p in c_i may converge slowly, or even not at all, to the order of p in d_i. However, by working in a specially constructed extension ring R of $\mathbb{Z}$ we are able to prove sufficiently fast convergence. Consider, for example, finding the content of the $f(x) = 2x^2 - 2x \in \mathbb{Z}[x]$. It is easily shown that $4 \mid f(a)$ for all $a \in \mathbb{Z}$, so we cannot identify the content by computing GCD's of evaluations of f at integers. However, in $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$, where $\Gamma = x^2 + x + 1 \in \mathbb{Z}[x]$, we can choose $a = (x \bmod \Gamma)$, so $f(a) = (4x - 2 \bmod \Gamma) \in \mathsf{R}$. GCD's are not well defined in R, so we take GCD's of the *contents* of the evaluations at points in R treated as polynomials (reduced modulo Γ) in $\mathbb{Z}[x]$. In this rather trivial case, the content of $-4x - 2$ is 2, as is the content of f. We prove that for any polynomial f, the corresponding sequence of GCD's of contents of evaluations of f in $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$ converges quickly on the content of f with high probability, for an appropriately selected Γ.

In our applications of this algorithm, the black box we are given may not work on all inputs. We allow for the existence of a $\chi \in \mathbb{Z}[x_1, \ldots, x_s]$ of degree $O(\nu^2)$ such that for $\vec{a} \in \mathsf{R}^s$, the black box is only guaranteed to find $(f_1(\vec{a}), \ldots, f_r(\vec{a}))$ when $\chi(\vec{a}) \in \overline{\mathsf{R}}^*$. Here $\overline{\mathsf{R}} = \mathbb{Q} \otimes \mathsf{R}$ and $\overline{\mathsf{R}}^*$ is the set of units in $\overline{\mathsf{R}}$. When $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$, the black box may report "failure" but cannot produce an incorrect answer.

A variant of Schwartz's Lemma

To prove that the algorithm sketched above to find the contents of black-box polynomials converges we will require a variant of Corollary 1 of Schwartz (1980), "Schwartz's Lemma". This bounds from above the probability that a random point, with coordinates chosen randomly and uniformly from a finite set $\mathcal{V}$, is a non-zero of a polynomial. We require a version of this in which coordinates are not chosen uniformly from $\mathcal{V}$, but where we only have an upper bound on the probability of choosing any one element of $\mathcal{V}$.

Lemma 2.2 *Assume $f \in \mathsf{D}[x_1, \ldots, x_k]$ is non-zero, D an integral domain, and $\mathcal{V}$ a finite subset of D. Suppose elements $a_1, \ldots, a_k$ are randomly chosen from $\mathcal{V}$ such that each a_i is assigned any one element of $\mathcal{V}$ with probability at most ϱ. Then $\mathrm{Prob}\{f(a_1, \ldots, a_k) = 0 : a_1, \ldots, a_k \in \mathcal{V}\} \leq \varrho \deg f$.*

Proof Follows almost identically to Corollary 1 of Schwartz (1980). □

Building a rough extension of $\mathbb{Z}$

Unfortunately Lemma 2.2 above, like Schwartz's (1980) Corollary 1, is of little use when the size of $\mathcal{V}$ is less than $\deg f$. In particular, when $\mathsf{D} = \mathbb{Z}_p$, $\#\mathcal{V} \leq p$ and $\tau \geq 1/p$. When $\deg f \geq p$, Lemma 2.2 is trivial. Our solution will be to constuct a *rough* extension ring R of $\mathbb{Z}$, one such that $\mathsf{R} \bmod p$ contains a large finite field for each of selected set of primes p. We show that the degree of R over $\mathbb{Z}$ can be kept surprisingly small.

Let $\eta \geq 1$ and $p_1, \ldots, p_\kappa$ be primes with $2 \leq p_1, \ldots, p_\kappa \leq \tau$ for some $\tau \geq 2$. We next show how to construct a monic, squarefree $\Gamma \in \mathbb{Z}[x]$ of small height and degree $\gamma = O(\eta \log \kappa)$ such that for each i $(1 \leq i \leq \kappa)$, $\Gamma \bmod p_i$ has a factor of degree greater than η in $\mathbb{Z}_p[x]$ (i.e., the factorization of Γ is "rough" modulo each prime p_i). The *height* $\|\Gamma\|$ of $\Gamma = c_0 + c_1 x + \cdots + c_\gamma x^\gamma \in \mathbb{Z}[x]$ is defined as $\max_{0 \leq i \leq \gamma} |c_i|$. Notice that the degree of Γ is *logarithmic* in the number of primes. This will be important in our application, where κ may be relatively large.

For any fixed prime p and $n \in \mathbb{N}$, define

$$\mathcal{M}_p(n) = \{g \in \mathbb{Z}_p[x] : g \text{ monic and } \deg g = n\},$$

$$\mathcal{I}_p(n) = \left\{ \begin{array}{c} g \in \mathbb{Z}_p[x] : g \text{ monic and irreducible} \\ \text{and } \deg g = n \end{array} \right\}.$$

We first give a lower bound on

$$R_p(\eta) = \#\{f \in \mathcal{M}_p(2\eta) : \exists k > \eta,\ g \in \mathcal{I}_p(k) \text{ such that } g \mid f\},$$

the number of monic polynomials in $\mathbb{Z}_p[x]$ of degree 2η which have an irreducible factor in $\mathbb{Z}_p[x]$ of degree greater than η.

Lemma 2.3 *For a prime p and integer $\eta \geq 1$ we have $R_p(\eta) > p^{2\eta}/3$.*

Proof Since any $f \in \mathcal{M}_p(2\eta)$ can have at most one irreducible factor with degree greater than η, we obtain the formula

$$R_p(\eta) = \sum_{\eta < i \le 2\eta} p^{2\eta - i} N_p(i)$$

where $N_p(i) = \#\mathcal{I}_p(i)$. Using the lower bound on $N_p(i)$ given by Lidl & Niederreiter(1983, Exercise 3.27) and Euler's summation formula, we get

$$R_p(\eta) > p^{2\eta}\left(\sum_{\eta<i\le 2\eta}\frac{1}{i} - \frac{p}{p-1}\sum_{\eta<i\le 2\eta}\frac{1}{ip^{i/2}}\right) > p^{2\eta}\cdot s(p,\eta),$$

where

$$s(p,\eta) = \log 2 - \frac{1}{2\eta} - \frac{p}{p-1}\cdot\frac{1}{\eta}\cdot\frac{\sqrt{p}}{\sqrt{p}-1}\cdot\frac{1}{p^{\eta/2+1/2}}.$$

The function $s(p,\eta)$ is strictly increasing in both p and η and is greater than $1/3$ with the exceptions $p=2$, $2 \le \eta \le 4$, and $p=3$, $\eta=2$. Excepting these cases, $R_p(\eta) > s(p,\eta) > p^{2\eta}/3$. It is easily checked, using the exact formula $N_p(i) = (1/i)\cdot\sum_{d\,|\,i}\mu(d)p^{i/d}$ for $i \ge 1$ (where μ is the Mobius function), that indeed $R_p(\eta) > p^{\eta}/3$ in the exceptional cases as well. □

Let $\mathcal{V} := \{g \in \mathbb{Z}[x]\colon g \text{ monic}, \deg g = 2\eta, \text{ and } \|g\| \le 2\eta\tau\}$. If we choose f randomly and uniformly from $\mathcal{V}$, f falls into a particular residue class in $\mathcal{M}_p(2\eta)$ with probability at least $(\lfloor(4\eta\tau+1)/p\rfloor/(4\eta\tau+1))^{2\eta} \ge (1/p - 1/(4\eta\tau+1))^{2\eta}$. Thus by Lemma 2.3 the probability that $f \bmod p$ has an irreducible factor modulo p of degree greater than η is at least $(1/p - 1/(4\eta\tau+1))^{2\eta}\cdot p^{2\eta}/3 > 1/6$.

The following algorithm constructs a $\Gamma \in \mathbb{Z}[x]$ as required.

Algorithm: `BuildRoughExtension`

Input: $\eta \in \mathbb{Z}$ and primes $2 \le p_1, \ldots, p_\kappa \le \tau$;

Output: a squarefree, monic $\Gamma \in \mathbb{Z}[x]$ such that for each i $(1 \le i \le \kappa)$, $\Gamma \bmod p_i$ has an irreducible factor in $\mathbb{Z}_{p_i}[x]$ of degree greater than η.

(1) Repeat
(2) Let $\mathcal{P} := \{1,\ldots,\kappa\}$; $H := \{\}$;
(3) For $i := 1$ to $4 + 6\log\kappa$ while $\mathcal{P} \ne \{\}$ do
(4) Choose a random $h_i \in \mathcal{V}$;
(5) For $j \in \mathcal{P}$ do
(6) If $h_i \bmod p_j \in \mathbb{Z}_{p_j}[x]$ has an irreducible factor modulo p_j of degree greater than η Then $\mathcal{P} := \mathcal{P}\setminus\{j\}$; $H := H \cup \{h_i\}$;
End For;
End For;
Until $\mathcal{P} = \{\}$;
Return $\Gamma = \prod_{h\in H} h \in \mathbb{Z}[x]$;

Theorem 2.4 *The algorithm* `BuildRoughExtension` *always produces the correct results as described and requires an expected number of $O((\eta^3 + \eta^2\log\tau)\cdot\kappa\log^2\tau\log\kappa)$ bit operations. The output $\Gamma \in \mathbb{Z}[x]$ has degree at most $2\eta(4 + 6\log\kappa) = O(\eta\log\kappa)$ and $\|\Gamma\| \le (4\eta^2\tau)^{\log\kappa}$.*

Proof We first examine the probability that the algorithm successfully finds a Γ in an iteration of the outer loop, or equivalently, finds an $H \subseteq \mathcal{V}$ such that for each i, there exists an $h \in H$ such that $h \bmod p_i$ has an irreducible factor in $\mathbb{Z}_p[x]$ of degree greater than η. For fixed j, the probability that $h_i \bmod p_j$ has no factor in $\mathcal{I}_{p_j}(r)$ for some $r > \eta$, for *all* $1 \le i \le l$ is less than $(5/6)^l$ by Lemma 2.3. The probability this is true for all j is less than $\kappa\cdot(5/6)^l < 1/2$ by our choice of $l = 4 + 6\log\kappa \ge -\log(2\kappa)/\log(5/6)$.

For each random choice of $h_i \in \mathcal{V}$ the inner loop of steps (5)–(6) can be accomplished with an expected number of $O((\eta^3 + \eta^2\log\tau)\log^2\tau\cdot\kappa)$ bit operations using Berlekamp's (1970) factoring algorithm, and this loop is executed $4 + 6\log\kappa$ times per iterations of the outer loop.

For any $h_1, h_2 \in \mathbb{Z}[x]$, $\|h_1h_2\| \le \min(\deg h_1, \deg h_2)\cdot\|h_1\|\|h_2\|$. Since Γ is the product of $O(\log\kappa)$ polynomials of degree 2η and height at most $2\eta\tau$, it follows that $\|\Gamma\| \le (4\eta^2\tau)^{\log\kappa}$. □

We define $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$, an extension ring of R where $\Gamma \in \mathbb{Z}[x]$ is monic of degree γ and is constructed using `BuildRoughExtension` on some η and primes $p_1,\ldots,p_\kappa$. The ring $\mathsf{R}_{p_i} = \mathsf{R} \bmod p_i$ contains a copy of $\mathrm{GF}(p_i^{\xi_i})$ for some $\xi_i > \eta$ for $1 \le i \le \kappa$. We represent an $a \in \mathsf{R}$ by its *least degree residue* $\hat{a} \in \mathbb{Z}[x]$ with $a \equiv \hat{a} \bmod \Gamma$ and $\deg\hat{a} < \gamma$. The notion of height can be extended to R by $\|a\| = \|\hat{a}\|$.

The probability of correctly finding the content

We return the the problem of obtaining the content $d_i \in \mathbb{Z}$ of $f_i \in \mathbb{Z}[x_1,\ldots,x_s]$ where $\deg f_i \le \nu$, for $1 \le i \le r$. As discussed above we choose points randomly from a subset $\mathcal{W}$ of a rough extension ring R of $\mathbb{Z}$ at which to evaluate $f_1,\ldots,f_r$. Let $\Gamma \in \mathbb{Z}[x]$ be monic of degree $\gamma > 1$ and $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$. Let $\beta \ge 5$ be an integer with $2\beta \ge \nu$, and let $L = \{-\beta,\ldots,\beta\} \subseteq \mathbb{Z}$. Let $\eta \le \deg\Gamma$ be an integer and $\mathcal{W} = \{h \bmod \Gamma : h \in L[x], \deg h < \eta\} \subseteq \mathsf{R}$. We will further specify our choices of Γ, γ, β and η in the sequel.

To determine the order of a prime p in d_i, assume $f_i = d_ig_i$ for some $g_i \in \mathbb{Z}[x_1,\ldots,x_s]$ of content 1. For $\vec{a} \in \mathsf{R}^s$, $b_i := f_i(\vec{a}) = d_ig_i(\vec{a}) \in \mathsf{R}$. Define $\overline{\mathrm{cont}}(b_i) = \mathrm{cont}(\bar{b}_i)$, where $\bar{b}_i \in \mathbb{Z}[x]$, $\deg\bar{b}_i < \deg\Gamma$ and $\bar{b}_i \equiv b_i \bmod \Gamma$. We have $\overline{\mathrm{cont}}(b_i) = d_i\,\overline{\mathrm{cont}}(g_i(\vec{a}))$ so $\mathrm{ord}_p(d_i) \le \mathrm{ord}_p(\overline{\mathrm{cont}}(b_i))$, with equality when $g_i(\vec{a}) \not\equiv 0 \bmod p$.

In the next two lemmas we examine the probability that, for a $g \in \mathbb{Z}[x_1,\ldots,x_s]$ with content 1, a prime p, and a randomly and uniformly selected point $\vec{a} \in \mathcal{W}^s$, that $g(\vec{a}) \equiv 0 \bmod p$. Two difficulties must be overcome. First, we make random choices uniformly from $\mathcal{W}$, but these are not generally uniform choices from $\mathcal{W} \bmod p$. Second, Corollary 1 of Schwartz (1980) is not useful for small primes ($\le \deg g$), and we must employ the properties of a rough extension ring R constructed with `BuildRoughExtension`.

Lemma 2.5 *Let $p > 2\beta$ be prime and $g \in \mathbb{Z}[x_1,\ldots,x_s]$ with $\deg g \le \nu$ and cont$(g) = 1$ ($2\beta \ge \nu$ as above). Suppose $\vec{a}$ is chosen randomly and uniformly from $\mathcal{W}^s$. Then* $\mathrm{Prob}\,\{g(\vec{a}) \equiv 0 \bmod p\} \le \nu/(2\beta)$.

Proof Assume $\vec{a} = (\bar{a}_1,\ldots,\bar{a}_s) \in \mathcal{W}^s$ where $\bar{a}_i \equiv a_i \bmod \Gamma$ and $a_i = \sum_{0\le j<\eta} a_{ij}x^j \in \mathbb{Z}[x]$ for $a_{ij} \in L$. Then $g(\vec{a}) \equiv 0 \bmod p \iff g(a_1,\ldots,a_s) \equiv 0 \bmod (\Gamma, p)$. Assume for now that the a_{ij}'s are independent indeterminates over $\mathbb{Q}$. Let $\Lambda = \{a_{ij} : 1 \le i \le s, 0 \le j < \eta\}$, and define $\hat{g} = g(a_1,\ldots,a_s) \in \mathbb{Z}[\Lambda][x]$. Consider the division of $\hat{g}$ by Γ in $\mathbb{Q}(\Lambda)[x]$ to obtain remainder $\rho = \sum_{0\le k<\gamma}\rho_k(\Lambda)x^k \in \mathbb{Z}[\Lambda][x]$, where $\rho_k \in \mathbb{Z}[\Lambda]$ has degree at most ν for $0 \le k < \gamma$. Now $g \not\equiv 0 \bmod p$ and $p > \deg g$, so there exists a $\vec{b} \in \mathbb{Z}^s$ such that $g(\vec{b}) \not\equiv 0 \bmod p$. Since $g(\vec{b}) \in \mathbb{Z}$, $\rho_0 \not\equiv 0 \bmod p$.

Again assuming $\vec{a}$ is randomly selected from $\mathcal{W}^s$, a necessary condition for $g(\vec{a}) \equiv 0 \bmod p$ is that $\rho_0(\vec{a}) \equiv 0 \bmod p$, whence $\text{Prob}\,\{g(\vec{a}) \equiv 0 \bmod p\} \leq \text{Prob}\,\{\rho_0(\vec{a}) \equiv 0 \bmod p\} \leq \nu/(2\beta+1)$ by Corollary 1 of Schwartz (1980). $\square$

Lemma 2.6 *Let $\beta \geq 5$ and $p \leq 2\beta$ prime, and assume $\Gamma \bmod p$ has an irreducible factor $\Upsilon \in \mathbb{Z}_p[x]$ of degree greater than η. Let $g \in \mathbb{Z}[x_1, \ldots, x_s]$ have degree $\leq \nu$ with cont$(g) = 1$. For a randomly and uniformly chosen $\vec{a} \in \mathcal{W}^s$,*

$$\text{Prob}\{g(\vec{a}) \equiv 0 \bmod p\} \leq \nu \cdot (3/5)^\eta.$$

Proof A randomly chosen $a \in \mathcal{W}$ lies in a particular residue class of $\mathcal{W} \bmod p$ with probability at most

$$\left(\left\lceil \frac{2\beta+1}{p} \right\rceil \cdot \frac{1}{2\beta+1}\right)^\eta \leq \left(\frac{1}{p} + \frac{1}{2\beta+1}\right)^\eta \leq (3/5)^\eta$$

for $\beta \geq 5$ and $p \leq 2\beta$. Since $\deg \Upsilon \geq \eta$ (and each $a \in \mathcal{W}$ has $\deg a < \eta$), the probability that a is in a particular residue class of $\mathcal{W} \bmod (p, \Upsilon)$ is also at most $(3/5)^\eta$. Applying Lemma 2.2, $\text{Prob}\{g(\vec{a}) \equiv 0 \bmod (p, \Upsilon)\} \leq \nu \cdot (3/5)^\eta$, whence $\text{Prob}\{g(\vec{a}) \equiv 0 \bmod p\} \leq \nu \cdot (3/5)^\eta$. $\square$

Define $\overline{\mathsf{R}} = \mathbb{Q} \otimes \mathsf{R} = \mathbb{Q}[x]/(\Gamma) \supseteq \mathsf{R}$, with units $\overline{\mathsf{R}}^* = \{a \bmod \Gamma : a \in \mathbb{Q}[x], \gcd(a, \Gamma) = 1\}$. We allow that our black box may not work when $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$. The following lemma demonstrates this seldom happens.

Lemma 2.7 *Let $\chi \in \mathbb{Z}[x_1, \ldots, x_s] \setminus \{0\}$. For a randomly and uniformly chosen $\vec{a} \in \mathcal{W}^s$, $\text{Prob}\{\chi(\vec{a}) \notin \overline{\mathsf{R}}^*\} \leq \deg(\chi) \cdot \deg(\Gamma)/(2\beta)$.*

Proof Assume $\vec{a} = (\bar{a}_1, \ldots, \bar{a}_s) \in \mathcal{W}^s$ where $\bar{a}_i = a_i \bmod \Gamma$ and $a_i = \sum_{0 \leq j < \eta} a_{ij} x^j \in \mathbb{Z}[x]$ for $1 \leq i \leq s$. Then $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$ if and only if $\gcd(\chi(a_1, \ldots, a_s), \Gamma) \neq 1$ in $\mathbb{Q}[x]$, which is true if and only if the resultant $\text{res}(\chi(a_1, \ldots, a_s), \Gamma)$ is zero.

Now assume the a_{ij}'s above are algebraically independent indeterminates over $\mathbb{Q}$ and let $\Lambda = \{a_{ij} : 1 \leq i \leq s, 0 \leq j < \eta\}$. Then $a_i = \sum_{0 \leq j < \eta} a_{ij} x^j \in \mathbb{Z}[\Lambda][x]$ and $\text{res}(\chi(a_1, \ldots, a_s), \Gamma) \in \mathbb{Z}[\Lambda]$ has degree at most $\deg(\chi) \cdot \deg(\Gamma)$. If the a_{ij}'s are assigned uniformly and randomly values from L, then by Schwartz's (1980) Corollary 1, the probability that $\text{res}(\chi(a_1, \ldots, a_s), \Gamma) = 0$, and hence the probability that $\chi(a_1, \ldots, a_s) \notin \overline{\mathsf{R}}^*$, is at most $\deg(\chi) \cdot \deg(\Gamma)/(2\beta+1)$. $\square$

Algorithm: `FindContent`

Input: $r \geq 2$, $\epsilon > 0$ and a black box which on input $\vec{a} \in \mathsf{R}^s$ evaluates $(f_1(\vec{a}), \ldots, f_r(\vec{a})) \in \mathsf{R}^r$, where $f_1, \ldots, f_r \in \mathbb{Z}[x_1, \ldots, x_s]$, $\deg f_i = \nu$, and R an extension ring of $\mathbb{Z}$; We allow for the existence of a $\chi \in \mathbb{Z}[x_1, \ldots, x_s] \setminus \{0\}$ of degree $O(\nu^2)$ such that if $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$ the black box may report "failure" and is not evaluated.

Output: $(\text{cont}(f_1), \ldots, \text{cont}(f_r)) \in \mathbb{Z}^r$, correct with probability $\geq 1 - \epsilon$;

(1) Let $\beta := \max(r\nu + \deg \chi, \nu \cdot \deg(\chi) \log(\deg(\chi)), \nu^2, 5)$;

(2) Choose a random $\vec{a} \in \{-\beta, \ldots, \beta\}^s$;
let $(c_1^{(0)}, \ldots, c_r^{(0)}) := (f_1(\vec{a}), \ldots, f_r(\vec{a})) \in \mathbb{Z}^r$;
if any of $c_1^{(0)}, \ldots, c_r^{(0)} = 0$, repeat (2);

(3) Find $p_1, \ldots, p_\kappa \in \mathbb{N}$, all the primes $\leq 2\beta$ which divide $\text{lcm}(c_1^{(0)}, \ldots, c_r^{(0)})$;

(4) Using `BuildRoughExtension` on $p_1, \ldots, p_\kappa$ and $\eta = 4 \log \nu$ construct $\Gamma \in \mathbb{Z}[x]$, such that for $1 \leq i \leq \kappa$, $\Gamma \bmod p_i$ has an irreducible factor of degree greater than η in $\mathbb{Z}_p[x]$. Let $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$ and $\mathcal{W} = \{h \bmod \Gamma : h \in L[x], \deg h < \eta\} \subseteq \mathsf{R}$.

(5) For $i := 1$ to $l = \log(1/\epsilon) + \log r + \log\log \max_i |c_i^{(0)}|$ do

(6) Choose random $\vec{a} \in \mathcal{W}^s$;

(7) Let $(b_1, \ldots, b_r) := (f_1(\vec{a}), \ldots, f_r(\vec{a})) \in \mathsf{R}^r$;
If the black box fails to find $(b_1, \ldots, b_r)$ on $\vec{a}$, so $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$, this choice of $\vec{a}$ may be ignored, and execution continued with next i at step (5);

(8) Let $(\bar{b}_1, \ldots, \bar{b}_r) := (\overline{\text{cont}}(b_1), \ldots, \overline{\text{cont}}(b_r)) \in \mathbb{Z}^r$;

(9) Let $(c_1^{(i)}, \ldots, c_r^{(i)}) :=$
$(\gcd(c_1^{(i-1)}, \bar{b}_1), \ldots, \gcd(c_r^{(i-1)}, \bar{b}_r)) \in \mathbb{Z}^r$;

End For;

(10) Output $(c_1^{(l)}, \ldots, c_r^{(l)})$;

Theorem 2.8 *The algorithm `FindContent` works correctly as described and produces the correct answer with probability at least $1 - \epsilon$. An expected number of $O((\log(1/\epsilon) + \log r + \log\log s + \log\log \max_i \|f_i\|)/\log \nu)$ evaluations of the black box are needed. These evaluations are in the ring $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$, where $\deg \Gamma = O(\log(\nu)\log(r))$ and $\log \|\Gamma\| = O((\log r + \log \nu)^2)$. The arguments to the black box from R^s have height $(r\nu)^{O(1)}$.*

`FindContent` requires $O\tilde{\ }(r(\nu + \log \max_i \|f_i\|)^2 \cdot \log(1/\epsilon))$ additional bit operations (using standard arithmetic) and $O\tilde{\ }(s + r(\nu + \log \max_i \|f_i\|))$ bits of additional storage.

Proof Step (2) finds a non-zero multiple c_i of d_i for $1 \leq i \leq r$. For a randomly chosen $\vec{a} \in \{-\beta, \ldots, \beta\}$

$$\begin{aligned}
&\text{Prob}\{\ \chi(\vec{a}) \neq 0, f_i(\vec{a}) \neq 0 \text{ for all } 1 \leq i \leq r\} \\
&\quad = \text{Prob}\{(\chi \cdot f_1 \cdots f_r)(\vec{a}) \neq 0\} \\
&\quad \geq 1 - (r\nu + \deg \chi)/(2(r\nu + \deg \chi)) \geq 1/2
\end{aligned}$$

by Corollary 1 of Schwartz (1980). Thus we expect to evaluate the black box two times on points of height $O(\beta)$ in step (2). On completion of (2) we have

$$\begin{aligned}
|c_i^{(0)}| = |f_i(\bar{a})| &\leq \|f_i\| \sum_{0 \leq j \leq \nu} \binom{j+s-1}{j} \beta^j \\
&= O(\|f_i\| \cdot \beta^\nu (\nu + s)^\nu),
\end{aligned}$$

since there are at most $\binom{j+s-1}{j}$ monomials of degree j in f_i.

Assume for now we choose $\vec{a}$ randomly in step (6) from $\mathcal{W}$ and do not eliminate the cases when $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$ in step (7). Fix an i between 1 and r. For a prime $p > 2\beta$ dividing d_i,

$$\begin{aligned}
\text{Prob}\{\text{ord}_p(\bar{b}_i) \neq \text{ord}_p(d_i)\} &= \text{Prob}\,\{g_i(\vec{a}) \equiv 0 \bmod p\} \\
&\leq \nu/(2\beta) \leq \nu/(2\nu^2) \leq 1/\nu
\end{aligned}$$

by Lemma 2.5. For a prime $p \leq 2\beta$ dividing d_i

$$\begin{aligned}
&\text{Prob}\{\text{ord}_p(\bar{b}_i) \neq \text{ord}_p(d_i)\} = \text{Prob}\{g_i(\vec{a}) \equiv 0 \bmod p\} \\
&\leq \nu \cdot (3/5)^\eta \leq \nu \cdot (3/5)^{4 \log \nu} \leq 1/\nu
\end{aligned}$$

by Lemma 2.6. Thus, for any prime p, $\text{Prob}\{\text{ord}_p(\bar{b}_i) \neq \text{ord}_p(d_i)\} \leq 1/\nu$.

The probability that $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$, and hence the probability that the black box might not work, is at most

$$\deg(\chi)\deg(\Gamma)/(2\beta) = 4\deg(\chi)\cdot\log(\nu)\cdot(4+6\log(\beta))/\beta$$
$$\leq 81\log^2(\nu)/\nu,$$

using Lemma 2.7 and the fact that $\kappa \leq \beta$ for $\beta \geq 5$. Thus, for a fixed i and prime p dividing $c_i^{(0)}$, the probability that $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$ *or* $\text{ord}_p(\bar{b}_i) \neq \text{ord}_p(d_i)$ is at most $1/\nu + 81\log^2(\nu)/\nu \leq 82\log^2(\nu)/\nu$.

Let $\omega(c_i^{(0)})$ denote the number of distinct primes dividing $c_i^{(0)}$. The probability that after l iterations of the loop there exists an i and a prime p dividing $c_i^{(0)}$ such that for all l randomly chosen $\vec{a} \in \mathsf{R}^s$ we have $\chi(\vec{a}) \notin \overline{\mathsf{R}}^*$ or $\text{ord}_p(c_i^{(l)}) \neq \text{ord}_p(d_i)$ is at most

$$\sum_{1\leq i\leq r} \omega(\ c_i^{(0)})\cdot(82\log^2(\nu)/\nu)^l$$
$$\leq r\log_2\max|c_i^{(0)}|(82\log^2(\nu)/\nu)^l \leq \epsilon$$

by our choice of l. Thus, the probability that the algorithm produces the correct answer is at least $1-\epsilon$, as required. The number of evaluations of the black box is $l = O((\log(1/\epsilon) + \log r + \log\log s + \log\log\max_i \|f_i\|)/\log\nu)$.

Note that $\deg\Gamma = O(\eta\log\kappa) = O(\log(\nu)\log(r))$ and $\log\|\Gamma\| = O((\log r + \log\nu + \log\deg\chi)^2)$ by Lemma 2.4. Arguments $(a_1,\ldots,a_s) \in \mathcal{W}^s$ to the black box satisfy $\|a_i\| \leq \beta = (r\nu\deg\chi)^{O(1)}$ for $1 \leq i \leq s$.

The cost (in addition to the cost of the black-box evaluations) is dominated by the loop in steps (5)-(9). For $1 \leq i \leq r$, $\log\|b_i\| = O(\log\|f_i\| + \nu\log(\nu+s) + \nu\log\beta + \nu\gamma\log\|\Gamma\|)$, so the cost of l iterations of steps (8)-(9) is $O\tilde{\ }(lr(\nu+\log\|f_i\|)^2)$ bit operations using standard arithmetic. At any time we store $O(r)$ elements of R with $O\tilde{\ }(\nu+\log\max_i\|f_i\|)$ bits each and s elements with $O(\log r + \log\nu)$ bits each. □

2.3 Finding the leading minors of a matrix over R

We now show how to find all the leading minors of a $B \in \mathsf{R}^{r\times r}$. Again $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$ is an extension ring of $\mathbb{Z}$, where $\Gamma \in \mathbb{Z}[x]$ is monic of degree γ, and $\overline{\mathsf{R}} = \mathbb{Q}[x]/(\Gamma)$. For a prime p, let $\mathsf{R}_p = \mathsf{R} \bmod p$. We attempt to determine the leading minors of B by finding matrices $L, U \in \mathsf{R}_p^{r\times r}$ such that L is lower triangular with ones on the diagonal, U is upper triangular, and $B \equiv LU \bmod p$ — an LU-decomposition of $B \bmod p$. If such a decomposition exists, the kth leading minor d_k of B satisfies $d_k \equiv \prod_{1\leq i\leq k} U_{ii} \bmod p$. We can recover each d_k by Chinese remaindering with sufficiently many primes.

In what follows, let E be any commutative ring.

Lemma 2.9 *Let $B \in \mathsf{E}^{r\times r}$ such that all leading minors are in E^*. Then there exists a lower triangular matrix $L \in \mathsf{E}^{r\times r}$ with ones on the diagonal, and an upper triangular matrix $U \in \mathsf{E}^{r\times r}$, such that $B = LU$.*

Proof The proof is essentially the same as for the existence of an LU-decomposition over $\mathbb{R}$ without pivoting (see Golub & Van Loan 1983, Section 4.2). □

Lemma 2.10 *Let $B \in \mathsf{E}^{r\times r}$. Then we can either find an upper triangular $U \in \mathsf{E}^{r\times r}$, and a lower triangular $L \in \mathsf{E}^{r\times r}$ with ones on its diagonal, such that $B = LU$, or report that one of the leading minors is not in E^*, using $O(r^\theta)$ operations in E.*

Proof This follows from an application of the asymptotically fast LUP-decomposition algorithm of Aho *et al.* (1974, section 6.4). This returns the factorization $B = LUP$, for L, U as above and P an $n\times n$ permutation matrix. When all leading minors of B lie in E^*, it is easily proven by induction on the number of rows in B that the returned P is an identity matrix. Simply run the LUP decomposition algorithm and if at any stage the returned $P \neq I$ report that one of the leading minors is not in E^*. Aho *et al.*'s (1974) algorithm requires $O(r^\theta)$ operations in E. □

We again assume that $B \in \mathsf{R}^{r\times r}$ and define $\|B\| = \max\{\|B_{ij}\| : 1 \leq i,j \leq r\}$. We require a simple upper bound for $\|\det B\|$ in terms of $\|B\|$, $\|\Gamma\|$, and $\gamma = \deg\Gamma$.

Lemma 2.11 *Let $\Gamma \in \mathbb{Z}[x]$ be monic of degree γ and $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$. For $B \in \mathsf{R}^{r\times r}$ we have $\|\det B\| < 2^{r\gamma}(r\gamma)^r \cdot \|B\|^r\|\Gamma\|^{r\gamma}$.*

Proof Let $\hat{B} \in \mathbb{Z}[x]^{r\times r}$ be such that $\deg\hat{B}_{ij} \leq \gamma - 1$ and $\hat{B} \equiv B \bmod \Gamma$. Then $\det\hat{B}$ is a sum of $r!$ polynomials of degree at most $r(\gamma-1)$, each of is a product of r elements of $\hat{B}$ and so has height at most $\gamma^{r-1}\|B\|^r$. (see Giesbrecht 1993, Theorem 1.5 for heights of products and division with remainder of integer polynomials). Doing division with remainder by Γ we find $\|\det B\| \leq 2^{r\gamma}(r\gamma)^r \cdot \|B\|^r\|\Gamma\|^{r\gamma}$. □

Theorem 2.12 *Let $B \in \mathsf{R}^{r\times r}$. We can construct an algorithm which either returns all the leading minors of B or reports that one of them is not a unit in R. The algorithm requires $O\tilde{\ }(r^{\theta+1}\cdot(\gamma^2\log\|B\| + \gamma^4\log\|\Gamma\|))$ bit operations using standard integer and polynomial arithmetic. The algorithm is probabilistic of the Las Vegas type and requires space for $O\tilde{\ }(r^2(\log\|B\| + \gamma\log\|\Gamma\|))$ bits.*

Proof Let $h \in \mathbb{Z}[x]$ with $h \equiv \det(B) \bmod \Gamma$ with $\deg h < \gamma$. By Lemma 2.11, $\|h\| < \|B\|^r\|\Gamma\|^{r\gamma}(r\gamma)^r 2^{r\gamma}$.

Assume for now that all the leading minors of B lie in $\overline{\mathsf{R}}^*$ so $B = LU$, where $L \in \overline{\mathsf{R}}^{r\times r}$ is lower triangular with ones on the diagonal and $U \in \overline{\mathsf{R}}^{r\times r}$ is upper triangular. The kth leading minor of B is then $m_k = \prod_{1\leq i\leq k} U_{ii} \in \mathsf{R}$. For a prime p, we can compute $m_1 \bmod p, \ldots, m_r \bmod p$ when each or these is in R_p^*. This is true if and only if $\text{res}(h,\Gamma) \not\equiv 0 \bmod p$. Applying Hadamard's bound to the Sylvester matrix of Γ and h, we find

$$|\text{res}(h,\Gamma)| \leq \gamma^\gamma\|h\|^\gamma\|\Gamma\|^\gamma \leq \xi := \gamma^\gamma\|B\|^{r\gamma}\|\Gamma\|^{r\gamma^2}(r\gamma)^{r\gamma}2^{r\gamma^2}.$$

Let $\mu = \|B\|^r\|\Gamma\|^{r\gamma}(r\gamma)^r 2^{r\gamma}$ — by Lemma 2.11, μ is greater than the height of any minor of B. Let z be such that the product of the z smallest primes is greater than $2\mu\xi$, and let $\mathcal{P}$ be the set containing the smallest z primes. It must be the case that the primes in $\mathcal{P}$ which do not divide $\text{res}(h,\Gamma)$ have product greater than twice the height of any leading minor of B. If the kth minor $m_k \in \mathsf{R}$ of B is not in $\overline{\mathsf{R}}^*$ then $(m_k \bmod p) \notin \mathsf{R}_p^*$ for any prime p.

We proceed by computing an LU-decomposition of B modulo each $p \in \mathcal{P}$. We either obtain the leading minors of $B \bmod p$ in $\mathsf{R} \bmod p$ or report that one of the leading minors of $B \bmod p$ is a non-unit in $\mathsf{R} \bmod p$. Let $\mathcal{Q} \subseteq \mathcal{P}$ be the primes p modulo which each minor of $B \bmod p$ is a unit in $\mathsf{R} \bmod p$ – we can computed an LU-decomposition of $B \bmod p$. If $\prod_{p\in\mathcal{Q}} p \geq \mu$ then we can recover all the leading minors of B by Chinese remaindering, and we return these. Otherwise, we report that one of the leading minors of B is not a unit in $\overline{\mathsf{R}}$.

To analyse the cost of this algorithm we note

$$\begin{aligned} z &= \pi\left(\min_y\{e^{\vartheta(y)} \geq \mu\xi\}\right) \\ &= O\left(\frac{r\log\|B\| + r\gamma^2\log\|\Gamma\| + r\gamma\log(r\gamma)}{\log\log\|B\| + \log\log\|\Gamma\| + \log r + \log\gamma}\right) \end{aligned}$$

where $\vartheta(y)$ is the Chebyshev theta function (the log of the product of all primes less than y), and $\pi(y)$ is the prime number function (the number of primes less than y). It is well known that $\vartheta(y) = O(\log y)$ and $\pi(y) = y/\log(y)$. All primes in $\mathcal{P}$ have $l = O(\log\log\|B\| + \log\log\|\Gamma\| + \log r + \log\gamma)$ bits. For a prime $p \in \mathcal{P}$ we can do an operation in $\mathsf{R} \bmod p$ with $O(\gamma^2 l^2)$ bit operations, and we can find the leading minors of $(B \bmod p) \in (\mathsf{R} \bmod p)^{r\times r}$ with $O(r^\theta\gamma^2 l^2)$ bit operations. To do this for all z primes requires $O(zr^\theta\gamma^2 l^2)$ or $O\tilde{~}(r^{\theta+1}\cdot(\gamma^2\log\|B\| + \gamma^4\log\|\Gamma\|))$ bit operations. The Chinese remaindering needed to recover the leading minors in R can also be done in this time. The space required for the output is dominant, and this is $O(r\log\mu)$ or $O\tilde{~}(r^2(\log\|B\| + \gamma\log\|\Gamma\|))$ □

2.4 Monte Carlo computation of the Smith normal form

In this subsection we present our new Monte Carlo algorithm for computing the Smith normal form of an integer matrix. It is based on a reduction to finding the contents of a list of black box polynomials, as discussed above.

We require an asymptotically fast algorithm for finding the rank of an $A \in \mathbb{Z}^{m\times n}$:

Lemma 2.13 *Let $A \in \mathbb{Z}^{m\times n}$ with $m \leq n$. We can compute $r = \operatorname{rank}(A)$ with $O(m^\theta n\log\|A\|)$ bit operations using standard integer arithmetic.*

Proof We use a standard homomorphic imaging scheme. Compute a number z such that $b = \prod_{p\leq z}^{p \text{ prime}} p > m^{m/2}\|A\|^m$. By Hadamard's bound every minor of A is less than b. Suppose $c \in \mathbb{Z}$ is an $r\times r$ minor of A. For each prime $p \leq z = O(m(\log m + \log\|A\|))$ find the rank of $A \bmod p$. Ibarra *et al.* (1982) show this can be accomplished with $O(m^{\theta-1}n\log^2 p)$ bit operations. At least one prime $p \leq z$ does not divide c, and $\operatorname{rank}(A \bmod p)$ is maximal and equal to $r = \operatorname{rank}(A)$. □

Theorem 2.14 *Let $A \in \mathbb{Z}^{m\times n}$ with $m \leq n$, and ϵ a positive constant. We can construct a Monte Carlo type probabilistic algorithm to compute the Smith normal form of A which requires an expected number of $O\tilde{~}((m^\theta n\log\|A\| + m^3\log^2\|A\|)\cdot\log(1/\epsilon))$ bit operations using standard integer and polynomial arithmetic. It returns the correct answer with probability at least $1-\epsilon$, and requires $O\tilde{~}(nm\log\|A\|)$ bits of storage.*

Proof By Theorem 2.1 we need only find the content d_k of the kth leading minor $f_k \in \mathbb{Z}[\Lambda]$ of $XAY \in \mathbb{Z}[\Lambda]^{r\times r}$ for $1 \leq k \leq r$, where Λ, $X \in \mathbb{Z}^{r\times m}$, and $Y \in \mathbb{Z}[\Lambda]^{n\times r}$ are as in Theorem 2.1. By Theorem 2.8, the algorithm `FindContent` does just this. The height of f_i is $O(\|A\|^i i^i)$ for $1 \leq i \leq r$.

The algorithm works in an extension ring $\mathsf{R} = \mathbb{Z}[x]/(\Gamma)$ of $\mathbb{Z}$, for a monic $\Gamma \in \mathbb{Z}[x]$ of degree $O(\log^2 r)$ with $\log\|\Gamma\| = O(\log^2 r)$. The black-box we use is from Theorem 2.12 and finds the leading minors of $B = PAQ$ for $P \in \mathsf{R}^{r\times m}$ and $Q \in \mathsf{R}^{n\times r}$. The arguments to the black box have height $\|B\| = nm\|A\|\cdot r^{O(1)}$. Thus, we can find all the leading minors of B with $O\tilde{~}(r^{\theta-2}mn\log\|A\| + r^{\theta+1}\log\|A\|)$ bit operations and space for $O\tilde{~}(r^2\log\|A\|)$ bits.

By Theorem 2.8 we require $O((\log(1/\epsilon) + \log\log n + \log\log\|A\|)/\log r)$ evaluations of the black-box plus an additional $O\tilde{~}(r^3\log^2\|A\|)$ bit operations. The total cost is an expected number of $O\tilde{~}((m^\theta n\log\|A\| + r^3\log^2\|A\|)\cdot\log(1/\epsilon))$ bit operations. □

References

A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms.* Addison-Wesley (Reading MA), 1974.

E. R. Berlekamp. Factoring polynomials over large finite fields. *Math. Comp.* **24**, pp. 713–735, 1970.

J. Buchmann. A subexponential algorithm for the determination of class groups and regulators of algebraic number fields. In *Séminaire de théorie des nombres*, Paris, 1988.

T. J. Chou and G. E. Collins. Algorithms for the solution of systems of linear Diophantine equations. *SIAM J. of Computing* **11**, pp. 687–708, 1982.

D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symb. Comp.* **9**, pp. 251–280, 1990.

P. Domich, R. Kannan, and L. Trotter. Hermite normal form computation using modulo determinant arithmetic. *Math. Operations Research* **12**, pp. 50–59, 1987.

F. R. Gantmacher. *The Theory of Matrices, Vol. I.* Chelsea Publishing Co. (New York NY), 1990.

M. Giesbrecht. *Nearly Optimal Algorithms for Canonical Matrix Forms.* PhD thesis, University of Toronto, 1993. 196 pp.

G. Golub and C. Van Loan. *Matrix Computations.* Johns Hopkins University Press (Baltimore, USA), 1983.

J. L. Hafner and K. S. McCurley. A rigorous subexponential algorithm for computation of class groups. *J. Amer. Math. Soc.* **2**, pp. 837–850, 1989.

J. L. Hafner and K. S. McCurley. Asymptotically fast triangulization of matrices over rings. *SIAM J. of Computing* **20**(6), pp. 1068–1083, 1991.

O. Ibarra, S. Moran, and R. Hui. A generalization of the fast LUP matrix decomposition algorithm and application. *J. of Algorithms* **3**, pp. 45–56, 1982.

C. Iliopolous. Worst-case complexity bounds on algorithms for computing the canonical structure of finite abelian groups and the Hermite and Smith normal forms of an integer matrix. *SIAM J. Computing* **18**, pp. 658–669, 1989.

E. Kaltofen, M. S. Krishnamoorthy, and B. D. Saunders. Fast parallel computation of Hermite and Smith forms of polynomial matrices. *SIAM J. Algebraic and Discrete Methods* **8**, pp. 683–690, 1987.

R. Kannan and A. Bachem. Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. *SIAM J. Comp.* **8**, pp. 499–507, 1979.

R. Lidl and H. Niederreiter. *Finite Fields*, vol. 20 of *Encyclopedia of Mathematics and its Applications*. Addison-Wesley (Reading MA), 1983.

M. Newman. *Integral Matrices*. Academic Press (New York), 1972.

J. B. Rosser and L. Schoenfeld. Approximate formulas for some functions of prime numbers. *Ill. J. Math.* **6**, pp. 64–94, 1962.

J. T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *J. Assoc. Computing Machinery* **27**, pp. 701–717, 1980.

H. J. S. Smith. On systems of linear indeterminate equations and congruences. *Philos. Trans. Royal Soc. London* **151**, pp. 293–326, 1861.