

Truncated Gröbner Bases for Integer Programming

Rekha R. Thomas Robert Weismantel
Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB)
Heilbronner Str. 10, 10711 Berlin, Germany
[last_name]@zib-berlin.de

Abstract

In this paper we introduce a multivariate grading of the toric ideal associated with the integer program $\min\{cx : Ax = b, x \in \mathbb{N}^n\}$, and a truncated Buchberger algorithm to solve the program. In the case of $\max\{cx : Ax \leq b, x \leq u, x \in \mathbb{N}^n\}$ in which all data are non-negative, this algebraic method gives rise to a combinatorial algorithm presented in [16].

1 Introduction

We study integer programs of the form $\min\{cx : Ax = b, x \in \mathbb{N}^n\}$, denoted $IP_{A,b,c,=}$, where all data are integral and the program has a bounded feasible region. The cost function c is refined by the *lexicographic order* if necessary, to ensure that the objective function value cx creates a linear order on \mathbb{N}^n . In [4], Conti and Traverso describe a Gröbner basis algorithm that can be used to solve all programs $IP_{A,b,c,=}$ as b varies. Their algorithm requires the computation of the *reduced Gröbner basis* with respect to the refined cost c , of the *toric ideal* I_A associated with A . Gröbner basis algorithms for finding non-negative integer solutions to systems of linear equations were also given by Pottier [8], [9] and Ollivier [10].

A set $T \subseteq \mathbb{Z}^n$ is a *test set* for the family of integer programs $\{IP_{A,b,c,=}, \forall b\}$ if, for each non-optimal solution to a program in this family, there exists a vector $v \in T$ such that $x - v$ is again feasible for the same program and has an improved objective function value as compared to x . See [6], [11] and [12] for finite test sets in integer programming. The special nature of the algebraic algorithm in [4] allows a geometric interpretation which has been worked out in [15]. This geometric version recognizes the reduced Gröbner basis produced by the Conti-Traverso algorithm as a minimal test set for the above family of integer programs. These test sets can be computed in practice by using a computer algebra package like MACAULAY [1], or the software GRIN [7] which is a specialized implementation of Gröbner bases for integer programming.

It is often the case that the computation of the entire reduced Gröbner basis associated with the family of programs $\{IP_{A,b,c,=}, \forall b\}$, is very expensive or impossible. In practice, one is often interested in solving $IP_{A,b,c,=}$ for a fixed right hand side vector b which typically requires only a subset of the entire Gröbner basis. In this paper, we provide a *truncated Buchberger algorithm* for toric ideals that finds a sufficient test set for $IP_{A,b,c,=}$. This set is often a proper subset of the reduced Gröbner basis of I_A with respect to c . The algorithm follows from a *multivariate grading* induced by the matrix A , of the toric ideal I_A . This generalizes, in the case of toric ideals, the theory of truncated Gröbner bases for ideals that are homogeneous with respect to a grading given by a vector of non-negative integers (see Section 10.2 in [2]). We refer to [2] and [5] for the theory of Gröbner bases and to [13] for toric ideals, their Gröbner bases and connections to integer programming and convex polytopes.

This paper is organized as follows. In Section 2 we present the multivariate grading of I_A given by the matrix A and the truncated Buchberger algorithm to solve $IP_{A,b,c,=}$ for fixed b . We introduce a partial order \succeq on the monoid of all feasible right hand side vectors, and the truncated Buchberger algorithm, denoted *b-Buchberger*, produces a minimal test set for all programs $IP_{A,\beta,c,=}$ for which $\beta \preceq b$. The elements in the test sets obtained by varying the cost function c are the edge directions in the polytopes $\text{conv}\{x \in \mathbb{N}^n : Ax = \beta\}$ for $\beta \preceq b$.

In Section 3 we apply the above algebraic method to the program $\max\{cx : Ax \leq b, x \leq u, x \in \mathbb{N}^n\}$ with non-negative data, which allows for simplifications. In this case, a geometric interpretation of the truncated Buchberger algorithm gives a combinatorial algorithm in [16].

2 A truncated Buchberger algorithm for integer programs with a fixed right hand side

Let $IP_{A,b,c,=}$ denote the integer program $\min\{cx : Ax = b, x \in \mathbb{N}^n\}$ where $A = (a_1, \dots, a_n) \in \mathbb{Z}^{m \times n}$ is a matrix of rank m , $b \in \mathbb{Z}^m$ and $c \in \mathbb{R}^n$. The matrix A , right hand side vector b , and cost vector c will be fixed throughout this section. The program $IP_{A,b,c,=}$ is feasible if and only if b lies in the monoid $\mathcal{C}_{\mathbb{N}}(A) = \{\sum_{i=1}^n m_i a_i : m_i \in \mathbb{N}\}$. We assume that $\mathcal{C}(A) = \{\sum_{i=1}^n r_i a_i : r_i \in \mathbb{R}_+\}$ is a pointed cone and that $IP_{A,b,c,=}$ has a bounded feasible region. The objective function value cx is assumed to give a total order on \mathbb{N}^n , possibly after breaking ties using the lexicographic order.

The matrix A induces a monoid homomorphism $\pi_A : \mathbb{N}^n \rightarrow \mathbb{Z}^m$ given by $\pi_A(u) = Au$. This lifts to the homomorphism of monoid algebras $\hat{\pi}_A : k[x_1, \dots, x_n] \rightarrow k[t_1^{\pm 1}, \dots, t_m^{\pm 1}]$ where $x_j \mapsto t^{a_j} = t_1^{a_{1j}} \dots t_m^{a_{mj}}$. The toric ideal of A is the prime ideal $I_A = \text{kernel}(\hat{\pi}_A)$.

Lemma 2.1 *The toric ideal $I_A = \bigoplus_{\beta \in \mathcal{C}_{\mathbb{N}}(A)} I_A(\beta)$ where $I_A(\beta)$ is the k -vector space spanned by the binomials $\{x^u - x^v : Au = Av = \beta, u, v \in \mathbb{N}^n\}$.*

Proof. The toric ideal I_A is spanned as a k -vector space by the set of binomials $\{x^u - x^v : Au = Av, u, v \in \mathbb{N}^n\}$. The above decomposition is a grading of I_A where the component $I_A(\beta)$ is the k -vector space spanned by all those binomials $x^u - x^v$ for which $Au = Av = \beta \in \mathcal{C}_{\mathbb{N}}(A)$.

We call the polytope $P_{\beta}^I = \text{conv}\{x \in \mathbb{N}^n : Ax = \beta\}$, the β -fiber of π_A . Note that $x^u - x^v \in I_A(\beta)$ if and only if $u, v \in P_{\beta}^I \cap \mathbb{N}^n$. Hence, there exists a bijection between the fibers of π_A and the components in the above direct sum. Lemma 2.1 implies that $I_A = \langle x^u - x^v : Au = Av, u, v \in \mathbb{N}^n \rangle$. Hence, I_A always has a finite generating set that consists of binomials of the above form and consequently, every reduced Gröbner basis of I_A again consists of such binomials.

The Conti-Traverso algorithm to solve all programs of the form $IP_{A,b,c,=}$ as b varies, involves the following two main steps:

Step 1. Compute the reduced Gröbner basis \mathcal{G}_c of the toric ideal I_A with respect to c .

Step 2. For a specified right hand side vector b , compute the *normal form* modulo \mathcal{G}_c (remainder on division by elements in \mathcal{G}_c), of the monomial x^u , where u is any feasible solution of $IP_{A,b,c,=}$. The exponent vector of this normal form is the unique optimum of $IP_{A,b,c,=}$.

The reduced Gröbner basis \mathcal{G}_c computed in Step 1 is a test set for all integer programs $IP_{A,b,c,=}$ such that $b \in \mathcal{C}_{\mathbb{N}}(A)$. For fixed b , the set \mathcal{G}_c often contains a number of elements that are not used in Step 2 of the algorithm. We describe a modification of the Buchberger algorithm for I_A , inspired by Lemma 2.1, to provide a test set for $IP_{A,b,c,=}$. This set maybe considerably smaller (depending on b) than the reduced Gröbner basis \mathcal{G}_c .

Let M denote the set of all monomials in $k[x] = k[x_1, \dots, x_n]$ where k is a field. The monoids M and \mathbb{N}^n are isomorphic via the usual identification of a monomial x^u with its exponent vector.

Under this identification, the monoid homomorphism π_A induces a multivariate grading of M and hence $k[x]$, where the π_A -degree of x^u denoted $\pi_A(x^u) = \pi_A(u) = Au \in \mathcal{C}_{\mathbf{N}}(A)$. Let $M(f)$ denote the monomials in a polynomial $f \in k[x]$.

Definition 2.2 *A polynomial $0 \neq f \in k[x]$ is said to be π_A -homogeneous if $\pi_A(s) = \pi_A(t)$ for all monomials $s, t \in M(f)$. The π_A -degree of such a homogeneous polynomial f , denoted $\pi_A(f)$, equals the π_A -degree of any monomial in $M(f)$.*

Note that a polynomial $f \in I_A$ is π_A -homogeneous if and only if the exponent vectors of all monomials in $M(f)$ lie in the same fiber of π_A .

Lemma 2.3 *The toric ideal I_A is homogeneous with respect to the grading π_A .*

Proof. Every binomial $x^u - x^v \in I_A$ is π_A -homogeneous since $Au = Av$. Let f_β denote the sum of all monomials of π_A -degree β , in a non-zero polynomial $f \in I_A$. The decomposition of I_A in Lemma 2.1 implies that $f_\beta \in I_A(\beta) \subset I_A$ for all $\beta \in \mathcal{C}_{\mathbf{N}}(A)$. Hence I_A is a homogeneous ideal with respect to this grading.

From now on we use the word homogeneous to mean π_A -homogeneous. The above multivariate grading induced by the matrix A , generalizes the usual grading of ideals by a vector of non-negative integers. An ideal that is homogeneous with respect to grading by a vector allows a natural truncation of the Buchberger algorithm that is compatible with the grading. We generalize this concept for the toric ideal I_A , which has been shown to be homogeneous with respect to the above multivariate grading. Our exposition follows Section 10.2 in [2].

Associated with the monoid $\mathcal{C}_{\mathbf{N}}(A)$ there is a ‘‘natural’’ partial order \succeq such that for $b_1, b_2 \in \mathcal{C}_{\mathbf{N}}(A)$, $b_1 \succeq b_2$ if and only if $b_1 - b_2 \in \mathcal{C}_{\mathbf{N}}(A)$. Notice that when $\mathcal{C}_{\mathbf{N}}(A) = \mathbf{N}^m$, the partial order \succeq coincides with the componentwise partial order \geq , where $b_1 \geq b_2$ if $b_1 - b_2 \geq 0$. Let $in_c(f)$ denote the leading monomial of $f \in k[x]$ with respect to the refined cost function c .

Lemma 2.4 *The following properties hold for the partial order \succeq and the grading π_A :*

- (i) *If x^u divides x^v , then $\pi_A(x^u) \preceq \pi_A(x^v)$.*
- (ii) *Let $f, g \in I_A$ be homogeneous polynomials such that $\pi_A(f) = \pi_A(g)$ and $f, g, f + g \neq 0$. Then $f + g$ is again homogeneous with $\pi_A(f + g) = \pi_A(f)$.*
- (iii) *Let $0 \neq f, g \in I_A$ be homogeneous polynomials. Then fg is homogeneous with $\pi_A(fg) = \pi_A(f) + \pi_A(g)$.*
- (iv) *Let $0 \neq f, p \in I_A$ be homogeneous binomials and g be obtained by reducing f by p with respect to c . Then $\pi_A(f) \succeq \pi_A(p)$ and g is a homogeneous binomial with $\pi_A(g) = \pi_A(f)$.*

Proof. We prove just (i) and (iv) since (ii) and (iii) follow from the definitions.

(i) If x^u divides x^v then $v = u + w$ for some $w \in \mathbf{N}^n$. Therefore, $Av - Au = Aw \in \mathcal{C}_{\mathbf{N}}(A)$ which implies that $\pi_A(x^u) \preceq \pi_A(x^v)$.

(iv) Since p reduces f , $in_c(p)$ divides some term of f . Using (i) and the homogeneity of f and p , $\pi_A(f) \succeq \pi_A(p)$. The binomial g is again homogeneous with $\pi_A(g) = \pi_A(f)$ since reduction by a homogeneous binomial keeps the exponent vectors of g on the same fiber as those of f .

We are now ready to describe a truncated Buchberger algorithm for I_A called *b-Buchberger*, that produces a test set for $IP_{A,\beta,c,=}$ for all $\beta \preceq b$. We denote the normal form of a binomial g , modulo a set of binomials G and cost function c , as $norm_{\{G,c\}}(g)$. The S-binomial of two binomials g_1 and g_2 , with respect to c , is denoted $S\text{-bin}_c(g_1, g_2)$.

The b-Buchberger algorithm for toric ideals

Input: A finite homogeneous binomial basis F of I_A and the refined cost vector c .

Output: A truncated (with respect to b) Gröbner basis of I_A with monomial order given by c .

$$i = -1$$

$$G_0 = F$$

Repeat

$$i = i + 1$$

$$G_{i+1} = G_i \cup (\{ \text{norm} f_{\{G_i, e\}}(S\text{-bin}_c(g_1, g_2)) : g_1, g_2 \in G_i, \pi_A(S\text{-bin}_c(g_1, g_2)) \preceq b \} \setminus \{0\})$$

Until $G_{i+1} = G_i$.

The only difference between the usual Buchberger algorithm and the b -Buchberger algorithm described above is that the latter only considers those S -binomials with $\pi_A(S\text{-bin}_c(g_1, g_2)) \preceq b$. Following this notation, we may denote the usual Buchberger algorithm as ∞ -Buchberger. Let b -Buchberger(F, c) denote the output of the algorithm b -Buchberger with starting basis F and cost function c . The algorithm terminates in finitely many steps since the Buchberger algorithm does so. Let $I_A[b] = \bigoplus_{\beta \preceq b} I_A(\beta)$. From Lemma 2.4 and the definition of the b -Buchberger algorithm, we get the following proposition.

Proposition 2.5 *Let F be a finite generating set for I_A that consists of homogeneous binomials and $\mathcal{G}_c(b) = b$ -Buchberger(F, c). Then the following hold:*

(i) *Every $g \in \mathcal{G}_c(b)$ is homogeneous and $\pi_A(g) \preceq b$ for all $g \in \mathcal{G}_c(b) \setminus F$.*

(ii) *For all $g_1, g_2 \in \mathcal{G}_c(b)$ with $\pi_A(S\text{-bin}_c(g_1, g_2)) \preceq b$, $S\text{-bin}_c(g_1, g_2)$ reduces to 0 modulo $\mathcal{G}_c(b)$.*

Let $\mathcal{RG}_c(b)$ denote the set obtained by performing the following two operations on $\mathcal{G}_c(b)$:

(i) remove an element if its initial term is divisible by the initial term of another element in $\mathcal{G}_c(b)$, (ii) reduce the resulting minimal set modulo itself. Both $\mathcal{G}_c(b)$ and $\mathcal{RG}_c(b)$ are called b -Gröbner bases of I_A with respect to c .

Notice that only those binomials in the starting basis F with $\pi_A(f) \preceq b$ play a role in the algorithm b -Buchberger. If there exists a binomial $f \in F$ such that $\pi_A(f)$ is not less than or equal to b with respect to the partial order \preceq , then it may be checked that the S -binomial formed by f and any other binomial will also inherit this property and hence will not be considered by b -Buchberger. These “passive” elements in F are carried along simply to preserve the generated ideal. By the same observation, the set $\mathcal{RG}_c(b)$ defined above is not a unique set associated with A, c and b . However, a b -Gröbner basis of I_A generates I_A .

Suppose we now collect those elements in $\mathcal{RG}_c(b)$ of π_A -degree less than or equal to b with respect to \succeq . Let $T_c(b) = \{g \in \mathcal{RG}_c(b) : \pi_A(g) \preceq b\}$. Recall that the reduced Gröbner basis of I_A with respect to c was denoted as \mathcal{G}_c .

Proposition 2.6 *The set $T_c(b) = \mathcal{G}_c \cap I_A[b]$.*

Proof. Since $\mathcal{RG}_c(b)$ generates I_A , it follows that ∞ -Buchberger($\mathcal{RG}_c(b), c$) is a Gröbner basis of I_A with respect to c , containing the set $T_c(b)$. We first show that during the run of this algorithm, no binomial g is created such that $\text{in}_c(g)$ divides either the leading or trailing term of an element in $T_c(b)$. Suppose such a g is created and it is the first such. By Lemma 2.4 (i), $\pi_A(g) = d \preceq b$ and hence, g is the normal form of an S -binomial $S\text{-bin}_c(g_1, g_2)$ of π_A -degree d . This implies that $\pi_A(g_1), \pi_A(g_2) \preceq d \preceq b$. Therefore, $g_1, g_2 \in T_c(b)$ by assumption. But then by Proposition 2.5 (ii) and the definition of $\mathcal{RG}_c(b)$, $S\text{-bin}_c(g_1, g_2)$ reduces to zero modulo $T_c(b)$. Hence, no such g is created. After the operations of making this Gröbner basis minimal and reduced we obtain the reduced Gröbner basis \mathcal{G}_c . By the above argument, no element of $T_c(b)$ is altered during this process and no new binomial g with $\pi_A(g) \preceq b$ is created.

The above proposition also proves that the set $T_c(b)$ is unique, although it may not be a generating set for I_A . We denote by $\text{in}_c(T_c(b))$ the set of all initial terms with respect to c , of the binomials in $T_c(b)$, and by $\text{in}_c(I_A)$ the initial ideal of I_A with respect to c .

Theorem 2.7 *The set $T_c(b) = \mathcal{G}_c \cap I_A[b]$ has the following properties:*

- (i) *Every monomial $s \in \text{in}_c(I_A)$ such that $\pi_A(s) \preceq b$ is divisible by some $t \in \text{in}_c(T_c(b))$.*
- (ii) *Every $0 \neq f \in I_A[b]$ reduces to zero modulo $T_c(b)$.*
- (iii) *Every homogeneous $f \in k[x]$ with $\pi_A(f) \preceq b$ has a unique normal form modulo $T_c(b)$.*

Proof. (i) Suppose there exists some monomial $s \in \text{in}_c(I_A)$ with $\pi_A(s) \preceq b$ that is not divisible by any $t \in \text{in}_c(T_c(b))$. By Lemma 2.4 (i) and Proposition 2.6, it follows that there does not exist any $t \in \text{in}_c(\mathcal{G}_c)$ that divides s . This contradicts that \mathcal{G}_c is a Gröbner basis of I_A with respect to c . Using (i) and Proposition 2.5, we get (ii) and (iii). ■

Corollary 2.8 *The set $T_c(b)$ is a minimal test set for all integer programs $IP_{A,\beta,c,=}$ with $\beta \preceq b$.*

Proof. Let u be a feasible non-optimal solution to $IP_{A,\beta,c,=}$ for some $\beta \preceq b$ for which the optimal solution is v . By Theorem 2.7, the binomial $x^u - x^v \in I_A[b]$ reduces to zero modulo $T_c(b)$ where $x^v \notin \text{in}_c(T_c(b))$. Hence, the unique normal form of x^u modulo $T_c(b)$ is x^v . This set is minimal by Proposition 2.6 since otherwise \mathcal{G}_c would not be a minimal test set for the family of programs $\{IP_{A,b,c,=}, \forall b\}$.

The algorithm b -Buchberger described above considers an S -binomial $g = x^u - x^v$ for reduction if and only if $\pi_A(g) = Au = Av \preceq b$. This amounts to checking feasibility of the system $\{x \in \mathbb{N}^n : Ax = b - Au\}$ which is as hard as solving the original integer program $IP_{A,b,c,=}$. Therefore, in order to implement b -Buchberger in practice, we propose two relaxations of the above check. Consider the S -binomial $g = x^u - x^v \in I_A$ for reduction if:

- (i) $b - Au \in \mathcal{C}(A)$ where $\mathcal{C}(A) = \{Ax : x \in \mathbb{R}_+^n\}$. I.e., check feasibility of the linear programming relaxation of the original check.
- (ii) $b - Au \in \mathcal{C}(A) \cap \mathbb{Z}A$ where $\mathbb{Z}A = \{Az : z \in \mathbb{Z}^n\}$. This is a relaxation of the original check since in general, $\mathcal{C}_{\mathbb{N}}(A)$ is strictly contained in $\mathcal{C}(A) \cap \mathbb{Z}A$.

Both the above relaxations consider all S -binomials that were considered by the original algorithm and possibly more. Hence, the output of the algorithm b -Buchberger with these relaxed checks for S -binomials will still provide a test set (not necessarily minimal) for all programs $IP_{A,\beta,c,=}$ with $\beta \preceq b$.

The truncated Buchberger algorithm can be sped up by applying Buchberger's first and second criteria to remove unnecessary S -binomials during the run of the algorithm. The first criterion allows S -binomial $S\text{-bin}_c(g_1, g_2)$ to be discarded if, $\text{in}_c(g_1)$ and $\text{in}_c(g_2)$ are relatively prime. This condition is not affected by any truncation of the Buchberger algorithm. The second criterion states that the S -binomial $S\text{-bin}_c(g_1, g_2)$ can be discarded if there exists a binomial f in the current partial basis such that $S\text{-bin}_c(g_1, f)$ and $S\text{-bin}_c(f, g_2)$ have been already considered by the algorithm and $\text{in}_c(f)$ divides $\text{lcm}(\text{in}_c(g_1), \text{in}_c(g_2))$. We show that this criterion is also unaffected by the truncation. If there exists f such that $\text{in}_c(f)$ divides $\text{lcm}(\text{in}_c(g_1), \text{in}_c(g_2))$, then $\text{lcm}(\text{in}_c(g_1), \text{in}_c(f))$ and $\text{lcm}(\text{in}_c(f), \text{in}_c(g_2))$ both divide $\text{lcm}(\text{in}_c(g_1), \text{in}_c(g_2))$. Therefore, if $\pi_A(S\text{-bin}_c(g_1, g_2)) \preceq b$, by Lemma 2.4 (i), $\pi_A(S\text{-bin}_c(g_1, f)) \preceq b$ and $\pi_A(S\text{-bin}_c(f, g_2)) \preceq b$. Hence, $S\text{-bin}_c(g_1, f)$ and $S\text{-bin}_c(f, g_2)$ are not victims of the truncation and Buchberger's second criterion can also be applied to remove unnecessary S -pairs during the run of the truncated Buchberger algorithm.

We remark that the theory of a truncated Buchberger algorithm and Gröbner basis in the context of a multivariate grading induced by an integer matrix A , will hold for any ideal that is homogeneous with respect to this grading. The above results can be generalized to this situation.

We now examine the geometry of the elements in the set $T_c(b)$. The set $UGB_A = \cup_c \mathcal{G}_c$ is a well defined unique finite set called the *universal Gröbner basis* of A (see [14]). This is a *universal test set* associated with A since it contains a test set for programs of the form

$IP_{A,b,c,=}$ with any cost function c and right hand side vector b . On similar lines we define the set $UGB_A(b) = \cup_c T_c(b)$ which we call the *universal b -Gröbner basis* of A . Clearly, $UGB_A(b)$ is a universal test set for all integer programs $IP_{A,\beta,c,=}$ with $\beta \preceq b$.

Lemma 2.9 *The set $UGB_A(b) = UGB_A \cap I_A[b]$.*

Proof. By Proposition 2.6, $UGB_A(b) = \cup_c T_c(b) = \cup_c (\mathcal{G}_c \cap I_A[b]) = UGB_A \cap I_A[b]$.

The above lemma implies that the set $UGB_A(b)$ is both unique and finite. Recall that the b -fiber of π_A is the polytope $P_b^I = \text{conv}\{x \in \mathbb{N}^n : Ax = b\}$. The following theorem gives a geometric characterization of elements in UGB_A . A vector $v \in \mathbb{Z}^n$ is said to be *primitive* if the g.c.d. of its components is one.

Theorem 2.10 (Theorem 5.1 in [14]) *A binomial $x^\alpha - x^\beta \in UGB_A$ if and only if the vector $\alpha - \beta$ is primitive and the line segment $[\alpha, \beta]$ is an edge of the $A\alpha$ -fiber of π_A .*

Corollary 2.11 *A binomial $x^\alpha - x^\beta \in UGB_A(b)$ if and only if the vector $\alpha - \beta$ is primitive and the line segment $[\alpha, \beta]$ is an edge of the $A\alpha$ -fiber of π_A where $A\alpha \preceq b$.*

The Graver basis of A , introduced in [6], is a universal test set associated with A that contains the set UGB_A . For the binomial $g = x^u - x^v \in I_A$, we call the Av -fiber of π_A , the *fiber of g* . By Theorem 2.10, the elements in the Graver basis of A that are in UGB_A are precisely those binomials that are edges in their fibers. Therefore, any algorithm to compute the Graver basis of A can be extended to compute UGB_A .

We briefly describe the algorithm to compute the Graver basis of A presented in [14] and show how it maybe modified to compute the universal b -Gröbner basis $UGB_A(b)$. Consider the $(m+n) \times 2n$ -matrix $\Lambda(A) = \begin{pmatrix} A & \mathbf{0} \\ I_n & I_n \end{pmatrix}$, called the *Lawrence lifting* of A , where $\mathbf{0}$ is an $m \times n$ matrix of zeros and I_n is the identity matrix of size n . It may be checked that $\text{kernel}_{\mathbb{Z}}(\Lambda(A)) = \{(u, -u) : u \in \text{kernel}_{\mathbb{Z}}(A)\}$ and hence the toric ideal $I_{\Lambda(A)} = \langle x^p y^q - x^q y^p : p, q \in \mathbb{N}^n, Ap = Aq \rangle \subseteq k[x_1, \dots, x_n, y_1, \dots, y_n]$. The Lawrence lifting $\Lambda(A)$ has the property that *any* reduced Gröbner basis of $I_{\Lambda(A)}$ coincides with the Graver basis of $\Lambda(A)$ and $UGB_{\Lambda(A)}$. See Theorem 4.1 in [14] for a proof. This along with the above discussion gives the following algorithm (Algorithm 4.3 in [14]) to compute the Graver basis of A .

Algorithm to compute the Graver basis of A .

1. Compute the reduced Gröbner basis \mathcal{G} of $I_{\Lambda(A)}$ with respect to any term order.
2. The Graver basis of A consists of all binomials $x^p - x^q$ such that $x^p y^q - x^q y^p$ appears in \mathcal{G} .

Applying the decomposition in Lemma 2.1 to $I_{\Lambda(A)}$ we see that the component $I_{\Lambda(A)}(\beta, \beta')$ is the k -vector space spanned by all binomials of the form $\{x^p y^q - x^q y^p : Ap = Aq = \beta, p + q = \beta', p, q \in \mathbb{N}^n\}$. This implies that $x^p - x^q \in I_A(\beta)$ if and only if $x^p y^q - x^q y^p \in I_{\Lambda(A)}(\beta, \beta')$ for some $\beta' \in \mathbb{N}^n$. We define the *b -Graver basis* of A to be all those elements g in the Graver basis of A with $\pi_A(g) \preceq b$. Let $(b, *)$ -Buchberger be the truncated Buchberger algorithm on $I_{\Lambda(A)}$ that only considers those S -binomials $x^p y^q - x^q y^p$ such that $Ap \preceq b$. Note that $\pi_{\Lambda(A)}(x^p y^q - x^q y^p) = (Ap, p + q) \in \mathcal{C}_{\mathbb{N}}(A) \oplus \mathbb{N}^n$ and $(b, *)$ -Buchberger only checks the first m components of $\pi_{\Lambda(A)}(x^p y^q - x^q y^p)$ in order to decide whether this S -binomial should be considered for reduction or not. An algorithm to compute the b -Graver basis of A is then immediate.

Algorithm 2.12. How to compute the b -Graver basis of A .

1. Compute the test set $T_{\succ}(b, *)$ of $\Lambda(A)$ with respect to any term order \succ .
2. The b -Graver basis of A consists of all binomials $x^p - x^q$ such that $x^p y^q - x^q y^p \in T_{\succ}(b, *)$.

Proof of correctness of Algorithm 2.12. By Proposition 2.6, $T_{\succ}(b, *) = \{x^p y^q - x^q y^p \in \mathcal{G}_{\succ} : Ap = Aq \preceq b\}$ where \mathcal{G}_{\succ} is the reduced Gröbner basis of $I_{\Lambda(A)}$ with respect to \succ . By the above discussion, \mathcal{G}_{\succ} is also the Graver basis of $\Lambda(A)$ and a binomial $x^p - x^q$ is in the b -Graver basis of A if and only if $x^p y^q - x^q y^p$ is in the Graver basis of $\Lambda(A)$ and $Ap = Aq \preceq b$.

Algorithm 2.12 and Corollary 2.11 give the following algorithm to compute $UGB_A(b)$.

Algorithm 2.13. How to compute the universal b -Gröbner basis $UGB_A(b)$.

1. Compute the b -Graver basis of A using Algorithm 2.12.
2. A binomial $x^p - x^q$ in the b -Graver basis of A is in $UGB_A(b)$ if and only if $[p, q]$ is an edge in the fiber of $x^p - x^q$.

3 A special case

In this section we specialize the theory developed in the previous section to integer programs of the form $\max\{cx : Ax \leq b, x \leq u, x \in \mathbb{N}^n\}$, denoted $IP_{A,b,c,u,\leq}$, where all data are non-negative and integral. We show that a geometric interpretation of this specialization gives a combinatorial algorithm for $IP_{A,b,c,u,\leq}$, presented in [16]. As before, we may assume that c has been refined to create a total order on \mathbb{N}^n . The non-negativity of the data ensures that $IP_{A,b,c,u,\leq}$ is bounded with respect to every cost function. The vector of upper bounds $u \in \mathbb{N}^n$ and as before, A, b, c and u will be fixed throughout this section.

In order to apply the results of the previous section to $IP_{A,b,c,u,\leq}$, we add slack vectors and transform the problem to $\max\{cx + 0s + 0r : Ax + I_m s = b, I_n x + I_n r = u, x \in \mathbb{N}^n, s \in \mathbb{N}^m, r \in \mathbb{N}^n\}$ which we denote as $IP_{A',(b,u),c',=}$. Here I_p denotes the identity matrix of size p and s and r are slack vectors of the sizes specified. The matrix $A' = \begin{bmatrix} A & I_m & 0 \\ I_n & 0 & I_n \end{bmatrix}$ is in $\mathbb{N}^{(m+n) \times (2n+m)}$,

right hand side vector $(b, u) \in \mathbb{N}^{m+n}$ and cost vector $c' = (c, 0, 0) \in \mathbb{N}^{2n+m}$. The monoid $\mathcal{C}_{\mathbb{N}}(A') = \mathbb{N}^{m+n}$ and the partial order \succeq is just the componentwise partial order \geq on \mathbb{N}^{m+n} .

The associated monoid homomorphism $\pi_{A'} : \mathbb{N}^{2n+m} \rightarrow \mathbb{N}^{m+n}$ takes $(x, s, r) \mapsto A'(x, s, r)$. The toric ideal $I_{A'} = \text{kernel}(\hat{\pi}_{A'})$ where $\hat{\pi}_{A'} : k[x, s, r] \rightarrow k[t, z]$ such that $x_j \mapsto t^{a_j} z_j$, $s_i \mapsto t_i$ and $r_j \mapsto z_j$ for $j = 1, \dots, n$ and $i = 1, \dots, m$. Let J denote the polynomial ideal in $k[t, z, x, s, r]$ given by $J = \langle x_j - t^{a_j} z_j, s_i - t_i, r_j - z_j, j = 1, \dots, n, i = 1, \dots, m \rangle$. It follows from Theorem 2 in Section 3.3 of [5] that $I_{A'} = J \cap k[x, s, r]$.

Lemma 3.1 *The toric ideal $I_{A'} = \langle x_j - s^{a_j} r_j, j = 1, \dots, n \rangle \subseteq k[x, s, r]$.*

Proof. The set $\mathcal{G}_{>} = \{\underline{x_j} - s^{a_j} r_j, \underline{t_i} - s_i, \underline{z_j} - r_j, j = 1, \dots, n, i = 1, \dots, m\}$ with the underlined terms as leading terms, is the reduced Gröbner basis of J with respect to any elimination order $>$ such that $t, z, x > s, r$. Hence $I_{A'} = J \cap k[x, s, r] = \langle \mathcal{G}_{>} \cap k[x, s, r] \rangle$.

As in [14] and [15], we can think of the binomial $y^\alpha - y^\beta$ in a toric ideal $I_A \subseteq k[y] = k[y_1, \dots, y_n]$, with no common factors in the two terms, as the vector $\alpha - \beta \in \mathbb{Z}^n$ or alternatively as the line segment $[\alpha, \beta] \subset \mathbb{R}^n$. In the case of the programs $IP_{A',(b,u),c',=}$ under consideration, we modify the usual interpretation so that a binomial $x^\alpha s^\beta r^\gamma - x^\delta s^\mu r^\lambda$ in $I_{A'}$ with no common factors in the two terms, is identified with the vector $\alpha - \delta \in \mathbb{Z}^n$ or the line segment $[\alpha, \delta] \subset \mathbb{R}^n$ by ignoring the slack components. Conversely, there exists a unique way in which a vector in \mathbb{Z}^n can be interpreted as a binomial in $I_{A'}$. Given $v \in \mathbb{Z}^n$, we first write it uniquely as $v = v^+ - v^-$ where $v^+, v^- \in \mathbb{N}^n$. The binomial associated with $[v^+, v^-]$ is then defined as $\text{bin}(v) = x^{v^+} s^{(Av)^-} r^{v^-} - x^{v^-} s^{(Av)^+} r^{v^+}$. It can be seen that the two terms in $\text{bin}(v)$ have no common factors and that the above choice of slack exponents is the smallest possible that will ensure

$\text{bin}(v) \in I_{A'}$. Given two vectors $v, w \in \mathbb{Z}^n$ and the refined cost function c , $S\text{-bin}_c(\text{bin}(v), \text{bin}(w))$ equals $(\text{bin}(v - w))$ possibly up to sign and multiplication by a monomial.

The (b, u) -fiber of $\pi_{A'}$ is the polytope $P_{(b,u)}^I = \text{conv}\{(x, s, r) \in \mathbb{N}^{2n+m} : Ax + I_m s = b, I_n x + I_n r = u\}$ and let $Q_{(b,u)}^I = \text{conv}\{x \in \mathbb{N}^n : Ax \leq b, x \leq u\}$. Under the above interpretation of binomials in $I_{A'}$, the generators of $I_{A'}$ in Lemma 3.1 are the n unit vectors in \mathbb{R}^n .

Observation 3.2 *There exists a connected undirected graph in every polytope $Q_{(b,u)}^I$ for $b \in \mathbb{N}^m$ and $u \in \mathbb{N}^n$, where the nodes are the lattice points in $Q_{(b,u)}^I$ and edges are translations of the unit vectors in \mathbb{R}^n .*

The above observation follows from the non-negativity of the data since one can construct a path from every lattice point $x \in Q_{(b,u)}^I$ to the origin by consecutively subtracting unit vectors and keeping all intermediate points in $Q_{(b,u)}^I$. The observation also follows from a general fact about generating sets for toric ideals: a set of binomials $\{y^{\alpha_i} - y^{\beta_i}, A\alpha_i = A\beta_i, \alpha_i, \beta_i \in \mathbb{N}^n, i = 1, \dots, p\}$ generates the toric ideal $I_A \subseteq k[y]$ if and only if in every fiber of π_A , we can build a connected (undirected) graph in which nodes are the lattice points in the fiber and edges are translations of the segments $[\alpha_i, \beta_i]$. The argument is completed by noting the bijection between lattice points in $P_{(b,u)}^I$ and $Q_{(b,u)}^I$.

We now show that a number of algebraic operations required in Section 2 can be reduced to easy checks on vectors, for the programs $IP_{A,b,c,u,\leq}$. As in the previous section, $\pi_{A'}$ defines a multivariate grading of $I_{A'}$ under which the degree of $\text{bin}(v)$ is $\pi_{A'}(\text{bin}(v)) = \begin{pmatrix} Av^- + (Av)^+ \\ v^+ + v^- \end{pmatrix}$.

However, $Av^- + (Av)^+ = \max\{Av^+, Av^-\}$ where \max computes the componentwise maximum of vectors. The (b, u) -Buchberger algorithm considers the S -binomial $\text{bin}(v)$ for reduction if and only if $\pi_{A'}(\text{bin}(v)) \leq (b, u)$. This yields the following lemma.

Lemma 3.3 *An S -binomial of the form $\text{bin}(v)$ will be considered for reduction by the algorithm (b, u) -Buchberger if and only if $Av^+ \leq b$, $Av^- \leq b$ and $0 \leq v^+, v^- \leq u$.*

In this section we will assume that all S -binomials considered are of the form $\text{bin}(v)$, i.e., the common terms in the two monomials have been removed. This is not required for the truncated Buchberger algorithm described in the previous section. We do this here in order to be able to store a binomial without ambiguity, as a vector equal to the difference of its exponent vectors. A vector $v \in \mathbb{Z}^n$ satisfies $Av^+ \leq b$, $Av^- \leq b$ and $0 \leq v^+, v^- \leq u$ if and only if v is the difference of two feasible solutions of $IP_{A,b,c,u,\leq}$. Therefore, the algorithm (b, u) -Buchberger considers an S -binomial $\text{bin}(v)$ only if, v is the difference of two feasible solutions of $IP_{A,b,c,u,\leq}$. As remarked earlier, for a general integer matrix A and right hand side vector b , checking whether the π_A -degree of an S -binomial is less than or equal to b with respect to the partial order \succeq , amounts to checking feasibility of an integer program. In the case of the programs $IP_{A',(b,u),c',=}$ studied here, this check reduces to the above easy check on the vectors v^+ and v^- . This allows the algorithm (b, u) -Buchberger to be implemented without relaxations.

Since $IP_{A',(b,u),c',=}$ is a maximization problem, if $cv > 0$, the leading term of the binomial $\text{bin}(v)$ with respect to c is $x^{v^-} s^{(Av)^+} r^{v^+}$ which is the monomial corresponding to v^- . Therefore, the binomial $\text{bin}(v)$ reduces the leading term of the binomial $\text{bin}(w)$, where $cw > 0$, if $x^{v^-} s^{(Av)^+} r^{v^+}$ divides $x^{w^-} s^{(Aw)^+} r^{w^+}$. We may write this down as an operation between the vectors v and w . For a vector $d \in \mathbb{Z}^n$, let $d^c = d$ if $cd > 0$ and $d^c = -d$ otherwise. If v is written without a superscript, we assume $cv > 0$.

Definition 3.4 [16] A vector $w \neq 0$ can be reduced by v if $v^+ \leq w^+$, $v^- \leq w^-$ and $(Av)^+ \leq (Aw)^+$. If the above conditions are satisfied, we obtain $(w - v)^c$ by reducing w by v .

By the above definition, v reduces w if the leading term of $\text{bin}(v)$ divides the leading term of $\text{bin}(w)$. In the usual theory of Gröbner bases, the binomial $\text{bin}(v)$ reduces $\text{bin}(w)$ if the leading term of $\text{bin}(v)$ divides either term of $\text{bin}(w)$. By the above definition, if the leading term of $\text{bin}(v)$ divides the trailing term of $\text{bin}(w)$, we would have to say that v reduces $-w$. If v reduces w , the following conditions hold:

- (1) if x and $x + w$ are feasible for $IP_{A,b,c,u,\leq}$, then so is $x + v$.
- (2) $|v|_1 \leq |w|_1$, with equality only if $v = w$ and $|w - v|_1 < |w|_1$.
- (3) $c(x + v) > cx$.

On the same lines, we may think of the reduction of a homogeneous binomial in $I_{A'}$ by a set of homogeneous binomials in $I_{A'}$ as an operation on vectors.

Reducing a vector $w \in \mathbb{Z}^n$ by a set of vectors $B \subseteq \mathbb{Z}^n$. [16]

This algorithm computes the normal form \bar{w}^B of a vector $w \in \mathbb{Z}^n$ by a set of vectors $B \subseteq \mathbb{Z}^n$.

- (1) Input $B \subseteq (\mathbb{Z}^n)^c$, $w \in \mathbb{Z}^n$.
- (2) As long as possible, find $v \in B$ such that $r \in \{w, -w\}$ can be reduced by v , and replace r by $r - v$.
- (3) Output $\bar{w}^B := r^c$.

The specializations of the algebra to the case of $IP_{A',(b,u),c',=}$ described above, allow the algorithm (b, u) -Buchberger to be described combinatorially. This is precisely Algorithm 3.7 in [16] which we reproduce below. A combinatorial proof of the correctness of this algorithm can be found in [16].

A combinatorial (b, u) -Buchberger algorithm for $IP_{A,b,c,u,\leq}$

- (1) Set $B_{old} := \emptyset$, $B := \{e_i : i = 1, \dots, n\}$
- (2) While $B_{old} \neq B$ repeat the following :
 - (2.1) Set $B_{old} := B$
 - (2.2) For all pairs of vectors $v, v' \in B_{old}$ such that $cv < cv'$ perform the following steps:
 - (2.2.1) If $Av^+ \leq b$, $Av^- \leq b$, $0 \leq v^+, v^- \leq u$, set $w = v' - v$.
 - (2.2.2) Compute $r := \bar{w}^B$.
 - (2.2.3) Set $B := B \cup \{r\}$.

Theorem 3.5 The output of the combinatorial (b, u) -Buchberger algorithm is a minimal test set for all programs $IP_{A,b',c,u',\leq}$ for which $b' \leq b$ and $u' \leq u$.

The set of generators of the toric ideal $I_{A'}$ that is used as input to the (b, u) -Buchberger algorithm is the set $\{x_j - s^{a_j} r_j, j = 1, \dots, n\}$. The $\pi_{A'}$ -degree of $x_j - s^{a_j} r_j$ is (a_j, e_j) for $j = 1, \dots, n$. We may assume without loss of generality that $a_j \leq b$ and $u_j \geq 1$ for $j = 1, \dots, n$ since otherwise, we could have removed column j from the matrix A . Therefore, all generators of $I_{A'}$ take part in the algorithm (b, u) -Buchberger.

The following example shows that truncated Gröbner bases can be considerably smaller than the entire reduced Gröbner basis associated with an integer program.

Example 3.6 Consider the knapsack problem in two variables

$$\max \{\mu x + \lambda y : \mu x + \lambda y \leq b, \quad x, y \geq 0 \text{ and integral}\},$$

where $\lambda > 0$ is an odd integer, $\mu = \frac{\lambda+1}{2}$ and b is some natural number. Then

$$(-1, 1), (-3, 2), (-5, 3), \dots, (-\lambda, \mu)$$

is a set of irreducible vectors with

$$\mu - 1 = -\mu + \lambda > \mu - 2 = -3\mu + 2\lambda > \mu - 3 = -5\mu + 3\lambda > \dots > 0 = -\lambda\mu + \mu\lambda.$$

Hence, this sequence of vectors must be contained in any test set for the family of integer programs as b varies (see [17] for further details).

Setting $b = k\lambda$ for a fixed natural number k , it is clear that the b -Gröbner basis contains only the first k elements in the above sequence of vectors and hence, is of significantly smaller size than the entire Gröbner basis. It should be clear that for higher dimensional knapsack problems even more drastic examples can be constructed.

In the remainder of this section we examine the geometry of the elements in the test sets produced by (b, u) -Buchberger. We denote by $UGB_A(b, u)$, the universal (b, u) -Gröbner basis of $I_{A'}$. As before, let $Q_{(b', u')}^I = \text{conv}\{x \in \mathbb{N}^n : Ax \leq b', x \leq u'\}$ and $P_{(b', u')}^I = \text{conv}\{(x, s, r) \in \mathbb{N}^{2n+m} : Ax + I_m s = b', I_n x + I_n r = u'\}$. If $[u, v]$ is an edge of a polytope P where u and v are adjacent vertices of P , we say that $(u - v)$ (up to sign) is an edge direction of P . By Corollary 2.11, we know that the elements in $UGB_A(b)$, thought of as vectors in \mathbb{Z}^{2n+m} , are the primitive edge directions in the polytopes $P_{(b', u')}^I$ for $b' \leq b$ and $u' \leq u$. Since all interpretations so far were done in n -space, we think of $UGB_A(b, u)$ as a subset of \mathbb{Z}^n and give an elementary combinatorial proof of the following fact.

Proposition 3.7 *The set $UGB_A(b, u)$ consists of all primitive edge directions in the polytopes $Q_{(b', u')}^I$ with $b' \leq b$ and $u' \leq u$.*

Proof. We start by showing that every primitive edge direction among the polytopes $Q_{(b', u')}^I$, with $b' \leq b$, $u' \leq u$ must be contained in $UGB_A(b, u)$. Let $e = y - z$ be a primitive edge direction where z and y are adjacent vertices of $Q_{(b', u')}^I$ for some $b' \leq b$ and $u' \leq u$. Let c be a cost function such that $cy > cz > cp$, for all $p \in \{\mathbb{N}^n \cap Q_{(b', u')}^I\} \setminus \{y, z\}$. Such a c exists since $[z, y]$ is an edge of $Q_{(b', u')}^I$. Therefore, the only vector that can be added to z to get an improved solution is e . Therefore, $T_c(b, u)$ and hence $UGB_A(b, u)$ must contain $e = y - z$.

Next we consider the reverse inclusion. Let $v \in UGB_A(b, u)$ and c be a cost function such that $v \in T_c(b, u)$ with $cv^+ > cv^-$. Define $u' := v^+ + v^-$ and $b' = Av^- + (Av)^+$. Since $(b', u') = \pi_{A'}(\text{bin}(v)) \leq (b, u)$ we have $u' \leq u$ and $b' \leq b$. The vector $v = v^+ - v^-$ is primitive since it belongs to $UGB_A(b, u)$. We will show that $[v^+, v^-]$ is an edge of $Q_{(b', u')}^I$. Notice that v^+ and $v^- \in Q_{(b', u')}^I \cap \mathbb{N}^n$ since, $0 \leq v^+, v^- \leq v^- + v^+ = u'$, $Av^- \leq Av^- + (Av)^+ = b'$, and $Av^+ = Av^- + (Av^+ - Av^-) = Av^- + Av = Av^- + (Av)^+ - (Av)^- \leq Av^- + (Av)^+ = b'$.

For any $z \in Q_{(b', u')}^I \cap \mathbb{N}^n$ distinct from v^+ and v^- , we have $(z - v^-) \geq -v^-$ since $z \geq 0$. Also, $z \leq v^+ + v^-$ implies $(z - v^-) \leq v^+$. Therefore $(z - v^-)^+ \leq v^+$ and $(z - v^-)^- \leq v^-$. Moreover, $Az \leq b'$ implies that $Az = Av^- + A(z - v^-) = Av^- + (A(z - v^-))^+ - (A(z - v^-))^- \leq Av^- + (Av)^+ = b'$. The last relation implies that $(A(z - v^-))^+ \leq (Av)^+$ since $(A(z - v^-))^+$ and $(A(z - v^-))^-$ have disjoint supports. Putting these arguments together we see that every $z \in Q_{(b', u')}^I$ satisfies the conditions $(z - v^-)^+ \leq v^+$, $(z - v^-)^- \leq v^-$ and $(A(z - v^-))^+ \leq (Av)^+$. Therefore, if $c(z - v^-) > 0$ then $z - v^-$ reduces v and v cannot be in $T_c(b, u)$ which is a contradiction. Therefore, $cz < cv^-$. Therefore, $cz < cv^- < cv^+$ for every $z \in Q_{(b', u')}^I$ distinct from v^+ and v^- .

Now we show that v^+ and v^- are vertices of $Q_{(b', u')}^I$. Suppose that v^+ is not a vertex of $Q_{(b', u')}^I$. Then $v^+ = \sum_{w \in W} \lambda_w w + \lambda_0 v^-$ where W is a subset of vertices in $Q_{(b', u')}^I$ and

$\lambda_0 + \sum_{w \in W} \lambda_w = 1$, $\lambda_w \geq 0$ for all $w \in W$ and $\lambda_0 \geq 0$. Clearly, $\lambda_0 = 0$ since v^- and v^+ have disjoint supports. Therefore, $v^+ = \sum_{w \in W} \lambda_w w$ which is impossible because $cw < cv^+$ for all $w \in W$. This implies that v^+ is a vertex of $Q_{(b',u')}^I$. Similarly v^- is a vertex of $Q_{(b',u')}^I$.

It remains to be shown that the vertices v^+ and v^- are adjacent. Suppose the converse is true, then there exists a point z on the line connecting v^+ and v^- that can be written as a convex combination of vertices in $Q_{(b',u')}^I$ different from v^+ and v^- . I.e., $z = \mu v^+ + \sigma v^-$ with $\mu + \sigma = 1$, $\mu, \sigma > 0$ has a representation as $z = \sum_{w \in W} \lambda_w w$, $\sum \lambda_w = 1$, $\lambda_w \geq 0$, with W being a subset of vertices in $Q_{(b',u')}^I$ not containing v^+ and v^- . Again, we obtain a contradiction, since $cz > cv^- > cw$ for every $w \in W$. This completes the proof.

Acknowledgements. We thank Karin Gatermann and Bernd Sturmfels for helpful discussions and comments. This work was done while the first author was visiting ZIB.

References

- [1] D. Bayer & M. Stillman: MACAULAY: A computer algebra system for algebraic geometry, Available by anonymous ftp from zariski.harvard.edu.
- [2] T. Becker & V. Weispfenning: *Gröbner Bases : A Computational Approach to Commutative Algebra*, Springer Verlag, New York, 1993.
- [3] B. Buchberger: Gröbner bases: an algorithmic method in polynomial ideal theory, in N.K. Bose (ed.), *Multidimensional Systems Theory*, D. Reidel Publications, 1985, 184-232.
- [4] P. Conti & C. Traverso: Buchberger algorithm and integer programming, Proceedings AAEECC-9 (New Orleans), Springer LNCS **539**, 1991, 130-139.
- [5] D. A. Cox, J. B. Little & D. O'Shea: *Ideals, Varieties, and Algorithms : An Introduction to Computational Algebraic Geometry and Commutative Algebra*, Undergraduate Texts in Mathematics, Springer-Verlag, New York, 1992.
- [6] J. E. Graver: On the foundations of linear and integer programming I, *Mathematical Programming* **8**, 1975, 207-226.
- [7] S. Hosten & B. Sturmfels: GRIN : An implementation of Gröbner bases for integer programming, Available by anonymous ftp from ftp.orie.cornell.edu.
- [8] L. Pottier: Minimal solutions of linear diophantine systems : bounds and algorithms, Proceedings RTA (Como), Springer Verlag, LNCS **488**, 1991.
- [9] L. Pottier: Sub-groups of \mathbb{Z}^n , standard bases and linear diophantine systems, Research Report 1510, I.N.R.I.A. Sophia Antipolis, 1991.
- [10] F. Ollivier: Canonical bases : relations with standard bases, finiteness conditions and applications to tame automorphisms, Proceedings MEGA-90, Birkhauser, Progress in Mathematics, 1991, 379-400.
- [11] H. E. Scarf: Neighborhood systems with production sets with indivisibilities, *Econometrica* **54**, 1986, 507-532.
- [12] A. Schrijver: *Theory of Linear and Integer Programming*, Wiley-Interscience Series in Discrete Mathematics and Optimization, New York, 1986.

- [13] B. Sturmfels: *Gröbner Bases and Convex Polytopes*, American Mathematical Society, 1996, to appear.
- [14] B. Sturmfels & R. R. Thomas: Variation of cost functions in integer programming, Technical Report, School of Operations Research, Cornell University, 1994.
- [15] R. R. Thomas: A geometric Buchberger algorithm for integer programming, Mathematics of Operations Research, to appear.
- [16] R. Urbaniak & R. Weismantel & G. Ziegler: A variant of Buchberger's algorithm for integer programming, preprint SC 94-29, Konrad-Zuse-Zentrum Berlin, 19 pages.
- [17] R. Weismantel: Hilbert bases and the facets of special knapsack polytopes, preprint SC 94-19, Konrad-Zuse-Zentrum Berlin, 25 pages.