Basis Reduction, and the Complexity of Branch-and-Bound

Gábor Pataki and Mustafa Tural *
Department of Statistics and Operations Research, UNC Chapel Hill

Abstract

The classical branch-and-bound algorithm for the integer feasibility problem

$$\text{Find}\, x \in Q \cap \mathbb{Z}^n, \text{ with } Q = \left\{ x \mid \begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} \le \begin{pmatrix} A \\ I \end{pmatrix} x \le \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \right\} \quad (1)$$

has exponential worst case complexity. We prove that it is surprisingly efficient on reformulations of (1), in which the columns of the constraint matrix are short, and near orthogonal, i.e. a reduced basis of the generated lattice.

The analysis builds on Furst and Kannan's work on the subset sum problem. It also uses an upper bound on the size of the branch-and-bound tree which depends on the norms of the Gram-Schmidt vectors of the constraint matrix.

We show that when the entries of A are from $\{1, \dots, M\}$ for a large enough M, branch-and-bound solves almost all reformulated instances at the rootnode, and explore practical aspects of this result. We compute numerical values of M which guarantee that 90, and 99 percent of the reformulated problems solve at the root: these turn out to be surprisingly small when the problem size is moderate.

A computational study also confirms that random integer programs become easier, as the coefficients grow.

1 Introduction and main results

The Integer Programming (IP) feasibility problem asks whether a polyhedron Q contains an integral point. Branch-and-bound, which we abbreviate

*Department of Statistics and Operations Research, UNC Chapel Hill, **gabor@unc.edu, tural@email.unc.edu**

as B&B is a classical solution method. It starts with Q as the sole subproblem (node). In a general step, one chooses a subproblem Q', a variable x_i, and creates nodes $Q' \cap \{x | x_i = \gamma\}$, where γ ranges over all possible integer values of x_i. We repeat this until all subproblems are shown to be empty, or we find an integral point in one of them.

B&B (and its version used to solve optimization problems) enhanced by cutting planes is a dependable algorithm implemented in most commercial software packages. However, instances in [10, 4, 9, 13] show that it is theoretically inefficient: it can take an exponential number of subproblems to prove the infeasibility of simple knapsack problems. While B&B is inefficient in the worst case, Cornuéjols et al. in [6] developed useful computational tools to give an early estimate on the size of the B&B tree in practice.

Since IP feasibility is NP-complete, one can ask for polynomiality of a solution method only in fixed dimension. All algorithms that achieve such complexity rely on advanced techniques. The algorithms of Lenstra [17] and Kannan [11] first round the polyhedron (i.e. apply a transformation to make it have a spherical appearance), then use basis reduction to reduce the problem to a provably small number of smaller dimensional subproblems. On the subproblems the algorithms are applied recursively, e.g. rounding is done again. Generalized basis reduction, proposed by Lovász and Scarf in [18] avoids rounding, but needs to solve a sequence of linear programs to create the subproblems.

There is a simpler way to use basis reduction in integer programming: preprocessing (1) to create an instance with short and near orthogonal columns in the constraint matrix, then simply feeding it to an IP solver. We describe two such methods that were proposed recently. We assume that A is an integral matrix with m rows, and n columns, and the w_i and ℓ_i are integral vectors.

The rangespace reformulation of (1) proposed by Krishnamoorthy and Pataki in [13] is

$$\text{Find}\, y \in Q_R \cap \mathbb{Z}^n, \text{ with } Q_R = \left\{ y \,|\, \begin{pmatrix} \ell_1 \\ \ell_2 \end{pmatrix} \leq \begin{pmatrix} A \\ I \end{pmatrix} U y \leq \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \right\}, \quad (2)$$

where U is a unimodular matrix computed to make the columns of the constraint matrix a reduced basis of the generated lattice.

The nullspace reformulation of Aardal, Hurkens, and Lenstra proposed in [2], and further studied in [1] is applicable, when $w_1 = \ell_1$. It is

$$\text{Find}\, y \in Q_N \cap \mathbb{Z}^{n-m}, \text{ with } Q_N = \{\, y \,|\, \ell_2 - x_0 \,\leq\, By \,\leq\, w_2 - x_0 \,\}, \quad (3)$$

where $x_0 \in \mathbb{Z}^n$ satisfies $Ax_0 = \ell_1$, and the columns of B are a reduced basis of the lattice $\{ x \in \mathbb{Z}^n \mid Ax = 0 \}$.

We analyze the use of Lenstra-Lenstra-Lovász (LLL) [16], and reciprocal Korkhine-Zolotarev (RKZ) reduced bases [14] in the reformulations. The former is computable in polynomial time for varying n. An RKZ reduced basis is polynomial time computable, only when the dimension is fixed, but it has stronger properties. When Q_R is computed using RKZ reduction, we call it the RKZ-rangespace reformulation of Q, and abusing notation we also call (2) the RKZ-rangespace reformulation of (1). Similarly we talk about an RKZ-nullspace, LLL-rangespace, and LLL-nullspace reformulation.

Example 1. *The polyhedron*

$$\begin{array}{rcl} 207 \leq & 41x_1 + 38x_2 & \leq 217 \\ 0 \leq & x_1, x_2 & \leq 10 \end{array} \tag{4}$$

is shown on the first picture of Figure 1. It is long and thin, and defines an infeasible, and relatively difficult integer feasibility problem for B&B, as branching on either x_1 or x_2 yields 6 subproblems. Lenstra's and Kannan's algorithms would first transform this polyhedron to make it more spherical; generalized basis reduction would solve a sequence of linear programs to find the direction $x_1 + x_2$ along which the polyhedron is thin.

The LLL-rangespace reformulation is

$$\begin{array}{rcl} 207 \leq & -3x_1 + 8x_2 & \leq 217 \\ 0 \leq & -x_1 - 10x_2 & \leq 10 \\ 0 \leq & x_1 + 11x_2 & \leq 10 \end{array} \tag{5}$$

shown on the second picture of Figure 1: now branching on y_2 proves integer infeasibility. (A similar example was given in [13]).

The reformulation methods are easier to describe, than, say Lenstra's algorithm, and are also successful in practice in solving several classes of hard integer programs: see [2, 1, 13].

However, they seem difficult to analyze in general. Krishnamoorthy and Pataki in [13] studied knapsack problems with a constraint vector a having a given decomposition $a = \lambda p + r$, with p and r integral vectors, and λ an integer, large compared to $\| p \|$ and $\| r \|$. They showed that branching on the constraint px in Q (which creates a small number of subproblems, as λ is large), is equivalent to branching on the last variable in Q_R and Q_N.

A result one may hope for is proving polynomiality of B&B on the reformulations of (1) when the dimension is fixed. While this seems difficult,

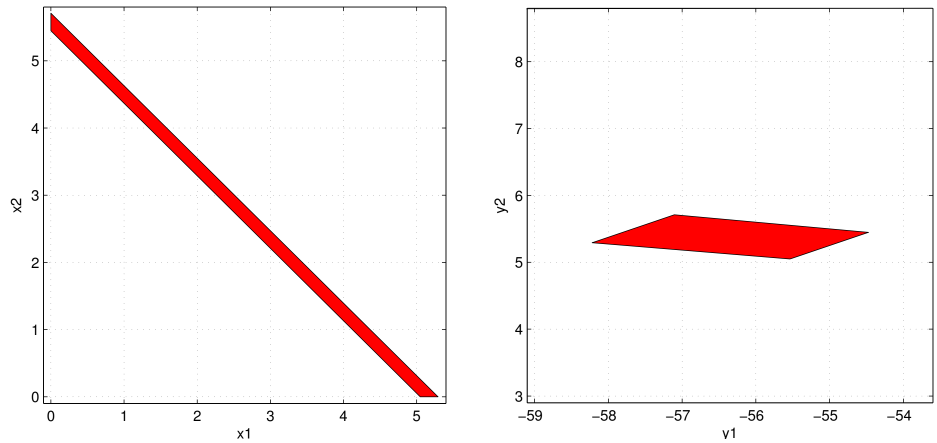

Figure 1: The polyhedron of Example 1 before and after the reformulation

we give a different, and perhaps even more surprising complexity analysis. It is in the spirit of Furst and Kannan's work in [8] on subset sum problems and builds on a generalization of their Lemma 1 to bound the fraction of integral matrices for which the shortest vectors of certain corresponding lattices are short. We also use an upper bound on the size of the B&B tree, which depends on the norms of the Gram-Schmidt vectors of the constraint matrix. We introduce necessary notation, and state our results, then give a comparison with [8].

When a statement is true for all, but at most a fraction of $1/2^n$ of the elements of a set S, we say that it is true for *almost all* elements. The value of n will be clear from the context. *Reverse B&B* is B&B branching on the variables in reverse order starting with the one of highest index. We assume $w_2 > \ell_2$, and for simplicity of stating the results we also assume $n \geq 5$. For a positive integer M we denote by $G_{m,n}(M)$ the set of matrices with m rows, and n columns, and the entries drawn from $\{1, \dots, M\}$. For matrices (and vectors) A and B, we write $(A; B)$ for $\begin{pmatrix} A \\ B \end{pmatrix}$. If B&B generates at most one node at each level of the tree, we say that it solves an integer feasibility problem at the rootnode.

If Q is a polyhedron, and z is an integral vector, then the width of Q along z is

$$\text{width}(z, Q) = \max \{ \langle z, x \rangle \mid x \in Q \} - \min \{ \langle z, x \rangle \mid x \in Q \} . \tag{6}$$

The main results of the paper follow.

Theorem 1. *There are positive constants $d_1 \leq 2$, and $d_2 \leq 6$ such that the following hold.*

(1) If

$$M > (d_1 n \,\|\,(w_1; w_2) - (\ell_1; \ell_2)\,\|)^{n/m+1}, \tag{7}$$

then for almost all $A \in G_{m,n}(M)$ reverse B&B solves the RKZ-rangespace reformulation of (1) at the rootnode.

(2) If

$$M > (d_2(n-m) \,\|\, w_2 - \ell_2 \,\|)^{n/m}, \tag{8}$$

then for almost all $A \in G_{m,n}(M)$ reverse B&B solves the RKZ-nullspace reformulation of (1) at the rootnode. □

The proofs also show that when M obeys the above bounds, then Q has at most one element for almost all $A \in G_{m,n}(M)$. When n/m is fixed, and the problems are binary, the magnitude of M required is a polynomial in n.

Theorem 2. *The conclusions of Theorem 1 hold for the LLL-reformulations, if the bounds on M are*

$$(2^{(n+4)/2} \,\|\,(w_1; w_2) - (\ell_1; \ell_2)\,\|)^{n/m+1}, \text{ and } (2^{(n-m+4)/2} \,\|\, w_2 - \ell_2 \,\|)^{n/m},$$

respectively. □

Furst and Kannan, based on Lagarias' and Odlyzko's [15] and Frieze's [7] work show that the subset sum problem is solvable in polynomial time using a simple iterative method for almost all weight vectors in $\{1, \ldots, M\}^n$, and all right hand sides, when M is sufficiently large, and a reduced basis of the orthogonal lattice of the weight vector is available. The lower bound on M is $2^{cn \log n}$, when the basis is RKZ reduced, and 2^{dn^2}, when it is LLL reduced. Here c and d are positive constants.

Theorems 1 and 2 generalize the solvability results from subset sum problems to bounded integer programs; also, we prove them via branch-and-bound, an algorithm considered inefficient from the theoretical point of view.

A practitioner of integer programming may ask for the value of Theorems 1 and 2. Proposition 1 and a computational study put these results into a more practical perspective. Proposition 1 shows that when m and n are not too large, already fairly small values of M guarantee that the RKZ-nullspace reformulation (which has the smallest bound on M) of the majority of binary integer programs get solved at the rootnode.

Proposition 1. *Suppose that m and n are chosen according to Table 1, and M is as shown in the third column. Then for at least 90% of $A \in G_{m,n}(M)$,*

n	m	M for 90 %	M for 99 %
30	20	31	35
50	20	1846	2071
50	30	93	100
60	30	410	443
70	40	193	205

Table 1: Values of M to make sure that the RKZ-nullspace reformulation of 90 or 99 % of the instances of type (9) solve at the rootnode

and all b right hand sides, reverse B&B solves the RKZ-nullspace reformulation of

$$\begin{aligned} Ax &= b \\ x &\in \{0,1\}^n \end{aligned} \tag{9}$$

at the rootnode. The same is true for 99% of $A \in G_{m,n}(M)$, if M is as shown in the fourth column. □

Note that 2^{n-m} is the best upper bound one can give on the number of nodes when B&B is run on the original formulation (9); also, randomly generated IPs with $n - m = 30$ are nontrivial even for commercial solvers.

According to Theorems 1 and 2, random integer programs with coefficients drawn from $\{1, \ldots, M\}$ should get easier, as M grows. Our computational study confirms this somewhat counterintuitive hypothesis on the family of marketshare problems of Cornuéjols and Dawande in [5]. The original formulations are notoriously difficult for commercial solvers, while the nullspace reformulations are much easier to solve as shown by Aardal et al. in [1].

We generated twelve 5 by 40 matrices with entries drawn from $\{1, \ldots, M\}$ with $M = 100, 1000,$ and 10000 (this is 36 matrices overall), set $b = \lfloor Ae/2 \rfloor$, where e is the vector of all ones, and constructed the instances of type (9), and

$$\begin{aligned} b - e \leq \quad Ax \quad &\leq b \\ x \in \{0,1\}^n&. \end{aligned} \tag{10}$$

The latter of these are a relaxed version, which correspond to trying to find an almost-equal market split.

Table 2 shows the average number of nodes that the commercial IP solver CPLEX 9.0 took to solve the rangespace reformulation of the inequality- and the nullspace reformulation of the equality constrained problems.

M	EQUALITY	INEQUALITY
100	17531.92	38884.92
1000	1254.42	22899.67
10000	200.83	1975.67

Table 2: Average number of B&B nodes to solve the inequality- and equality-constrained marketshare problems

Since RKZ reformulation is not implemented in any software that we know of, we used the Korkhine-Zolotarev (KZ) reduction routine from the NTL library [20]. KZ reduced bases (see e.g. [11, 12]) are also LLL reduced, but have stronger properties, and yield better reformulations, as shown in [13]. For brevity we only report the number of B&B nodes, and not the actual computing times.

All equality constrained instances turned out to be infeasible, except two, corresponding to $M = 100$. Among the inequality constrained problems there were fifteen feasible ones: all twelve with $M = 100$, and three with $M = 1000$. Since infeasible problems tend to be harder, this explains the more moderate decrease in difficulty as we go from $M = 100$ to $M = 1000$.

Table 2 confirms the theoretical findings of the paper: the reformulations of random integer programs become easier as the size of the coefficients grow.

In Section 2 we introduce further necessary notation, and give the proof of Theorems 1 and 2. The proof of Proposition 1 is given in Appendix A.

2 Further notation, and proofs

A lattice is a set of the form

$$L = \mathbb{L}(B) = \{ Bx \,|\, x \in \mathbb{Z}^r \}, \tag{11}$$

where B is a real matrix with r independent columns, called a *basis* of L, and r is called the *rank* of L.

The euclidean norm of a shortest nonzero vector in L is denoted by $\lambda_1(L)$, and Hermite's constant is

$$C_j = \sup \left\{ \lambda_1(L)^2/(\det L)^{2/j} \,|\, L \text{ is a lattice of rank } j \right\}. \tag{12}$$

We define

$$\gamma_i = \max \{ C_1, \ldots, C_i \}. \tag{13}$$

A matrix A defines two lattices that we are interested in:

$$L_R(A) = \mathbb{L}(A;I),\ L_N(A) = \{x \in \mathbb{Z}^n | Ax = 0\}, \tag{14}$$

where we recall that $(A;I)$ is the matrix obtained by stacking A on top of I.

Given independent vectors $b_1, \ldots, b_r$, the vectors $b_1^*, \ldots, b_r^*$ form the Gram-Schmidt orthogonalization of $b_1, \ldots, b_r$, if $b_1^* = b_1$, and b_i^* is the projection of b_i onto the orthogonal complement of the subspace spanned by $b_1, \ldots, b_{i-1}$ for $i \geq 2$.

We do not define LLL and RKZ reducedness formally, only collect their properties that we will use. If $b_1, \ldots, b_r$ form an RKZ reduced basis of the lattice L with Gram-Schmidt orthogonalization $b_1^*, \ldots, b_r^*$, then by [14]

$$\| b_i^* \| \geq \lambda_1(L)/C_i \tag{15}$$

holds. If they are an LLL-reduced basis, then by [16]

$$\| b_i^* \| \geq \lambda_1(L)/2^{(i-1)/2}. \tag{16}$$

Lemma 1. *Let P be a polyhedron*

$$P = \{y \in \mathbb{R}^r \,|\, \ell \leq By \leq w\}, \tag{17}$$

and $b_1^*, \ldots, b_r^*$ the Gram-Schmidt orthogonalization of the columns of B. When reverse B&B is applied to P, the number of nodes on the level of y_i is at most

$$\left(\left\lfloor \frac{\| w - \ell \|}{\| b_i^* \|} \right\rfloor + 1 \right) \ldots \left(\left\lfloor \frac{\| w - \ell \|}{\| b_r^* \|} \right\rfloor + 1 \right). \tag{18}$$

Proof First we show

$$\mathrm{width}(e_r, P) \leq \| w - \ell \| \,/\, \| b_r^* \|. \tag{19}$$

Let $x_{r,1}$ and $x_{r,2}$ denote the maximum and the minimum of x_r over P. Writing $\bar{B}$ for the matrix composed of the first $r-1$ columns of B, and b_r for the last column, it holds that there is $x_1, x_2 \in \mathbb{R}^{r-1}$ such that $\bar{B}x_1 + b_r x_{r,1}$ and $\bar{B}x_2 + b_r x_{r,2}$ are in P. So

$$\begin{aligned} \| w - \ell \| &\geq \| (\bar{B}x_1 + b_r x_{r,1}) - (\bar{B}x_2 + b_r x_{r,2}) \| = \| \bar{B}(x_1 - x_2) + b_r(x_{r,1} - x_{r,2}) \| \\ &\geq \| b_r^* \| \, |x_{r,1} - x_{r,2}| = \| b_r^* \| \, \mathrm{width}(e_r, P) \end{aligned}$$

holds, and so does (19).

After branching on $e_r, \ldots, e_{i+1}$, each subproblem is defined by a matrix formed of the first i columns of B, and bound vectors ℓ_i and w_i, which are translates of ℓ and w by the same vector. Hence the above proof implies that the width along e_i in each of these subproblems is at most

$$\| w - \ell \| \,/\, \| b_i^* \|, \tag{20}$$

and this completes the proof. □

Our Lemma 2 builds on Furst and Kannan's Lemma 1 in [8], with inequality (22) also being a direct generalization.

Lemma 2. *For a positive integer k, let ϵ_R and ϵ_N be the fraction of $A \in G_{m,n}(M)$ with $\lambda_1(L_R(A)) \leq k$, and $\lambda_1(L_N(A)) \leq k$, respectively. Then*

$$\epsilon_R \leq \frac{(2k+1)^{n+m}}{M^m}, \tag{21}$$

and

$$\epsilon_N \leq \frac{(2k+1)^{n}}{M^m}. \tag{22}$$

Proof We first prove (22). For v, a fixed nonzero vector in $\mathbb{Z}^n$, consider the equation

$$Av = 0. \tag{23}$$

There are at most $M^{m(n-1)}$ matrices in $G_{m,n}(M)$ that satisfy (23): if the components of $n-1$ columns of A are fixed, then the components of the column corresponding to a nonzero entry of v are determined from (23). The number of vectors v in $\mathbb{Z}^n$ with $\| v \| \leq k$ is at most $(2k+1)^n$, and the number of matrices in $G_{m,n}(M)$ is M^{mn}. Therefore

$$\epsilon_N \leq \frac{(2k+1)^n M^{m(n-1)}}{M^{mn}} = \frac{(2k+1)^n}{M^m}.$$

For (21), note that $(v_1; v_2) \in \mathbb{Z}^{m+n}$ is a nonzero vector in $L_R(A)$, iff $v_2 \neq 0$, and

$$Av_2 = v_1. \tag{24}$$

An argument like the one in the proof of (22) shows that for fixed $(v_1; v_2) \in \mathbb{Z}^{m+n}$ with $v_2 \neq 0$, there are at most $M^{m(n-1)}$ matrices in $G_{m,n}(M)$ that satisfy (24). The number of vectors in $\mathbb{Z}^{n+m}$ with norm at most k is at most $(2k+1)^{n+m}$, so

$$\epsilon_R \leq \frac{(2k+1)^{n+m} M^{m(n-1)}}{M^{mn}} = \frac{(2k+1)^{n+m}}{M^m}.$$

□

Proof of Theorems 1 and 2 For part (1) in Theorem 1, let $b_1^*, \ldots, b_n^*$ be the Gram-Schmidt orthogonalization of the columns of $(A; I)U$. Lemma 1 implies that the number of nodes generated by reverse B&B applied to Q_R is at most one, if

$$\| b_i^* \| > \| (w_1; w_2) - (\ell_1; \ell_2) \| \tag{25}$$

for $i = 1, \ldots, n$. Since the columns of $(A; I)U$ form an RKZ reduced basis of $L_R(A)$, (15) implies

$$\| b_i^* \| \geq \lambda_1(L_R(A))/C_i, \tag{26}$$

so (25) holds, when

$$\lambda_1(L_R(A)) > C_i \, \| (w_1; w_2) - (\ell_1; \ell_2) \| \tag{27}$$

does for $i = 1, \ldots, n$, which is in turn implied by

$$\lambda_1(L_R(A)) > \gamma_n \, \| (w_1; w_2) - (\ell_1; \ell_2) \| \, . \tag{28}$$

By Lemma 2 (28) is true for all, but at most a fraction of ϵ_R of $A \in G_{m,n}(M)$ if

$$M > \frac{(\lfloor 2\gamma_n \, \| (w_1; w_2) - (\ell_1; \ell_2) \| + 1 \rfloor)^{(m+n)/m}}{\epsilon_R^{1/m}}, \tag{29}$$

and using the known estimate $\gamma_n \leq 1 + n/4$ (see for instance [19]) , setting $\epsilon_R = 1/2^n$, and doing some algebra yields the required result.

The proof of part (2) of Theorem 1 is along the same lines: now $b_1^*, \ldots, b_{n-m}^*$ is the Gram-Schmidt orthogonalization of the columns of B, which is an RKZ reduced basis of $L_N(A)$. Lemma 1, and the reducedness of B implies that the number of nodes generated by reverse B&B applied to Q_N is at most one, if

$$\lambda_1(L_N(A)) > C_{n-m} \, \| w_2 - \ell_2 \|, \tag{30}$$

and by Lemma 2 (30) is true for all, but at most a fraction of ϵ_N of $A \in G_{m,n}(M)$ if

$$M > \frac{(\lfloor 2\gamma_{n-m} \, \| w_2 - \ell_2 \| + 1 \rfloor)^{n/m}}{\epsilon_N^{1/m}}. \tag{31}$$

Then simple algebra completes the proof.

The proof of Theorem 2 is an almost verbatim copy, now using the estimate (16) to lower bound $\| b_i^* \|$. □

References

[1] Karen Aardal, Robert E. Bixby, Cor A. J. Hurkens, Arjen K. Lenstra, and Job W. Smeltink. Market split and basis reduction: Towards a solution of the Cornuéjols-Dawande instances. *INFORMS Journal on Computing*, 12(3):192–202, 2000.

[2] Karen Aardal, Cor A. J. Hurkens, and Arjen K. Lenstra. Solving a system of linear Diophantine equations with lower and upper bounds on the variables. *Mathematics of Operations Research*, 25(3):427–442, 2000.

[3] Hans Frederik Blichfeldt. A new principle in the geometry of numbers, with some applications. *Transactions of the American Mathematical Society*, 15(3):227–235, 1914.

[4] Vašek Chvátal. Hard knapsack problems. *Operations Research*, 28(6):1402–1411, 1980.

[5] Gérard Cornuéjols and Milind Dawande. A class of hard small 0–1 programs. *INFORMS Journal on Computing*, 11(2):205–210, 1999.

[6] Gérard Cornuéjols, Miroslav Karamanov, and Yanjun Li. Early estimates of the size of branch-and-bound trees. *INFORMS Journal on Computing*, 18(1):86–96, 2006.

[7] Alan Frieze. On the Lagarias-Odlyzko algorithm for the subset sum problem. *SIAM Journal on Computing*, 15:536–540, 1986.

[8] Merrick Furst and Ravi Kannan. Succinct certificates for almost all subset sum problems. *SIAM Journal on Computing*, 18:550 – 558, 1989.

[9] Zonghao Gu, George L. Nemhauser, and Martin W. P. Savelsbergh. Lifted cover inequalities for 0–1 integer programs: Complexity. *INFORMS J. on Computing*, 11:117–123, 1998.

[10] Robert G. Jeroslow. Trivial integer programs unsolvable by branch-and-bound . *Mathematical Programming*, 6:105–109, 1974.

[11] Ravi Kannan. Minkowski's convex body theorem and integer programming. *Mathematics of Operations Research*, 12(3):415–440, 1987.

[12] A. Korkine and G. Zolotarev. Sur les formes quadratiques. *Mathematische Annalen*, 6:366–389, 1873.

[13] Bala Krishnamoorthy and Gábor Pataki. Column basis reduction and decomposable knapsack problems. *Discrete Optimization*, 6:242–270, 2009.

[14] Jeffrey C. Lagarias, Hendrik W. Lenstra, and Claus P. Schnorr. Korkine-Zolotarev bases and successive minina of a lattice and its reciprocal lattice. *Combinatorica*, 10(4):333–348, 1990.

[15] Jeffrey C. Lagarias and Andrew M. Odlyzko. Solving low-density subset sum problems. *Journal of ACM*, 32:229–246, 1985.

[16] Arjen K. Lenstra, Hendrik W. Lenstra, Jr., and László Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:515–534, 1982.

[17] Hendrik W. Lenstra, Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8:538–548, 1983.

[18] László Lovász and Herbert E. Scarf. The generalized basis reduction algorithm. *Mathematics of Operations Research*, 17:751–764, 1992.

[19] Jacques Martinet. *Perfect Lattices in Euclidean Spaces*. Springer-Verlag, Berlin, 2003.

[20] Victor Shoup. NTL: A Number Theory Library, 1990. http://www.shoup.net.

A Appendix: Proof of Proposition 1

Let $N(n,k)$ denote the number of integral points in the n-dimensional ball of radius k. In the previous proofs we used $(2k+1)^n$ as an upper bound for $N(n,k)$. The proof of Part (2) of Theorem 1 actually implies that when

$$M > \frac{(N(n, \lceil \gamma_{n-m} \, \| w_2 - \ell_2 \| \rceil))^{1/m}}{\epsilon_N^{1/m}}, \tag{32}$$

then for all, but at most a fraction of ϵ_N of $A \in G_{m,n}(M)$ reverse B&B solves the nullspace reformulation of (9) at the rootnode.

We use Blichfeldt's upper bound

$$C_i \leq \frac{2}{\pi} \Gamma \left(\frac{i+4}{2} \right)^{2/i}, \tag{33}$$

from [3] to bound γ_{n-m} in (32), dynamic programming to exactly find the values of $N(n,k)$, and the values $\epsilon_N = 0.1$, and $\epsilon_N = 0.01$ to obtain Table 1.

We note that in general $N(n,k)$ is hard to compute, or find good upper bounds for; however for small values of n and k a simple dynamic programming algorithm finds the exact value quickly. □