GRAVER BASES

Francisco Javier Blázquez Martínez

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Mathematics section,

Chair of discrete optimization

Project director: Dr. Friedrich Eisenbrand

Project supervisor: Jana Cslovjecsek

January 2021

To my brother Joaquín, for showing me what mathematics were during dinners at home.

To my parents, for teaching me beyond the scope of mathematics.

Acknowledgements

First and foremost I would like to express my sincere thanks to Jana Cslovjecsek for helping me throughout the whole project as well as to Professor Friedrich Eisenbrand for giving me the oportunity of doing this. It has been a very didactic experience that I really appreciate.

Moreover, this project is one of my finals steps for obtaining the Double degree in Mathematics-Computer Engineering at Complutense University of Madrid that, thanks to the help of many people, I have been able to carry out in this fantastic university that is the École Politechnique Fédéreale de Lausanne. I want to expressly thank Prof. Katzalin Olcoz and Prof. Daniel Chaver for all the facilities and help. I can't help but be grateful to the Complutense University and all the teachers I've had for everything I have learnt.

Finally I would like to thank my family for their unconditional support and also Luis Felipe Ramirez, for advising me so well in the important decisions these last years.

To all, thank you very much from the heart.

Contents

Chapter 1

Introduction

Hereafter, the underlying problem is the classical *Integer program* (IP), that we formulate in the following way:

$$(IP) \equiv max\{c^t x : Ax = b, l \leq x \leq u, x \in \mathbb{Z}^n\}$$

$A \in \mathbb{Z}^{mxn}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$, l and u lower and upper bounds for x

Despite the simplicity of its formulation, allowing only linear constraints and a linear objective function, it's well known the importance of IP. A large number of problems in diverse fields of the mathematics and algorithms (with an infinity of applications) admit an IP formulation. Unfortunately, it's also well known that IP is NP-Complete, what means that no efficient algorithm is likely to exist for solving the IP in the general case. This explains the great interest in restricted formulations of the problem and in certain resolution techniques (even when they can't be applied to the general IP). In the following sections we present the last techniques based on the **Graver bases** and its bounds as well as their application to the **N-Fold IP**, a restricted formulation of the IP which has won relevance in the last decades given its theoretical properties and its wide applications.

For this purpose, we first introduce the Graver basis of a given matrix, explore its properties, bounds, and how can these be applied for solving the general IP. We then study the N-Fold case and show, with the help of Graver bases, that the N-Fold IP can be solved in polynomial time. In the last sections we go further improving this polynomial complexity, obtaining two different efficient algorithms. One based on augmenting a feasible solution and another based on a proximity bound.

Chapter 2

Graver bases

Before introducing the concept of Graver basis of a matrix, we define a partial order $\sqsubseteq$ in $\mathbb{R}^n$ by $u \sqsubseteq v$ if $u_i \cdot v_i \geq 0$ and $|u_i| \leq |v_i|$ for all i. Note that the condition $u_i \cdot v_i \geq 0$ means that $\sqsubseteq$ can only compare vectors with the same sign componentwise. The Graver basis of a matrix is the set of minimal elements (for this order $\sqsubseteq$) in its integral kernel excluding zero. Formally:

Definition 2.1 (Graver basis). The Graver basis ($\mathcal{G}(A)$) of a given matrix $A \in \mathbb{Z}^{mxn}$ is defined as the set of $\sqsubseteq$-minimal elements in $\{z \in \mathbb{Z}^n : Az = 0, z \neq 0\}$.

Graver bases were initially defined as *universal integral test set* in [1] by Jack. E. Graver, in 1975. They often appear also defined in an equivalent way as the nonzero indecomposable elements in $ker(A)$. Indecomposable in the sense that they can not be expressed as the sum of two vectors with the same sign componentwise. It's easy to see the equivalence of both definitions.

Now that Graver bases are formally defined, we present their main properties in the form of propositions which will be the theoretical basis for the algorithms presented in the next sections.

Proposition 2.2. *For every matrix A, $\mathcal{G}(A)$ is a finite set.*

Proof. Dickson's lemma states that every subset of $\mathbb{N}^n$ has a finite number of minimal elements (with the order $\leq$ componentwise). It's easy to see that this implies that the integral kernel of A (excluding zero) has a finite number of $\sqsubseteq$-minimal elements in every orthant. As the elements in different orthants are not comparable we have that $\mathcal{G}(A)$ is the union of 2^n finite sets, concluding the proof. $\square$

Proposition 2.3. *Every integral element in ker(A) can be expressed as positive integral linear combination of sign compatible elements in $Gr(A)$.*

Proposition 2.4. *Given z in the feasible region of an IP, z is not optimum if and only if there exists $g \in Gr(A)$ s.t. $c^t g > 0$ and $l \leq z + g \leq u$*

Proof. Lets suppose first that a feasible point z is not an optimum, then $z^* - z$ belongs to $ker(A) \setminus \{0\}$. Thanks to the previous proposition we have $g_i \in G(A)$, $\alpha_i \geq 0$ s. t. $0 < c^t(z^* - z) = \sum \alpha_i c^t g_i$ and it's then clear that exists at least one $g_i \in G(A)$ verifying $c^t g_i > 0$ and respecting the bounds.

For the other implication is clear that $z + g$ is a feasible point which improves the objective function so z is not an optimum. □

2.1 Graver Basis greedy algorithm

General IP algorithm using Graver basis

1. From a feasible solution z_i
2. Find g^* optimum for the sub-problem:
   $max\{c^t g : g \in Gr(A), l \leq z_i + g \leq u\}$
   - $c^t g^* \leq 0 \implies z_i$ optimal solution.
   - $c^t g^* > 0 \implies g^*$ improvement direction, loop back to 1 with $z_{i+1} = z_i + \lambda \cdot g^*$ with the biggest λ respecting the bounds.

[References??]

The question that arises now is which is the complexity of this algorithm. [2] (Theorem 2.b) states that the number of augmentation steps is polynomial and, since the cost of each augmentation step is in the order of $|G(A)| \times n$, we have that this algorithm is polynomial. Also [3] states this (but the proof is more complicated).

This of course doesn't mean we have a polynomial algorithm for the general IP because the trick is that the Graver Basis is given as part of the input. The problem is of course computing it and, in most of the cases, its size is exponential in the dimension.

2.2 Graver Basis norm bounds

Proposition 2.5 (Graver basis bounds). *Given $A \in \mathbb{Z}^{mxn}$ and Δ an upper bound for the absolute value of each component of A, for every $g \in Gr(A)$:*

- $||g||_1 \leq m^{m/2}\Delta^m \cdot (n-m)$ *[Onn 2010]*
- $||g||_1 \leq (2m\Delta + 1)^m$ *[Eisenbrand,Hunkenschröder,Klein 2018]*

Unfortunately this bounds are both exponential. The second one has the advantage of being n-independent. In certain cases we can get a much tighter bound for the Graver Basis elements and this can help us to get a faster algorithm. The key ideas are the following points.

Bases of augmentation algorithm

- If not optimal, an element in Graver basis is an improvement direction.
- If Graver basis bounded, we can restrict our improvement direction search.

General IP algorithm using Graver basis norm bound

1. From a feasible solution z_i
2. Find g^* optimum for the sub-problem:
   $max\{c^t g : Ag = 0, l - z_i \leq g \leq u - z_i, g \in \mathbb{Z}^n, ||g||_1 \leq ||Gr(A)||\}$
   - $g^* = 0 \implies z_i$ optimal solution.
   - $g^* \neq 0 \implies g^*$ improvement direction, loop back to 1 with $z_{i+1} = z_i + \lambda \cdot g^*$ with the biggest λ respecting the bounds.

[Hemmecke, Onn, Romanchuk 2013]

The main advantage of this algorithm is that it doesn't require the explicit computation of the Graver Basis. However, the main drawback is that in general the bound for the graver bassis elements also increases exponentially with the dimension so this additional restriction to the problem won't be a help.

Chapter 3

N-Fold IP

A generalized N-Fold IP has constriction matrix A of the form ($A_i \in \mathbb{Z}^{rxt}$, $B_i \in \mathbb{Z}^{sxt}$):

$$N = \begin{pmatrix} A_1 & A_2 & \cdots & A_n \\ B_1 & 0 & \cdots & 0 \\ 0 & B_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B_n \end{pmatrix}$$

The N-Fold IP has a withe range of applications in ... (See that transportation problems etc admit N-Fold formulation!). It's also very interesting by itself by the theoretical properties it has.

3.1 N-Fold properties

Proposition 3.1. *Fix any pair of integer matrices $A \in \mathbb{Z}^{r\times q}$ and $B \in \mathbb{Z}^{s\times q}$. Then there is a polynomial time algorithm that, given n, computes the Graver basis of the N-Fold matrix $[A, B]^{(n)}$. In particular, the cardinality and the bit size of $G([A, B]^(n))$ are bounded by a polynomial function of n.*

Proposition 3.2. *Fix any pair of integer matrices $A \in \mathbb{Z}^{r\times q}$ and $B \in \mathbb{Z}^{s\times q}$. Then there is a polynomial time algorithm that, given n, objective vector $c \in \mathbb{Z}^{nq}$, and non-negative integer vector $x \in \mathbb{Z}^{nq}$, solves the generalized N-Fold integer programming problem in which x is feasible.*

Proposition 3.3. *Fix any pair of integer matrices $A \in \mathbb{Z}^{r\times q}$ and $B \in \mathbb{Z}^{s\times q}$. Then there is a polynomial time algorithm that, given n and demand vector $b \in \mathbb{Z}^{s+nr}$, either finds a feasible point $x \in \mathbb{N}^{nq}$ to the N-Fold IP of order n, or asserts that no feasible solution exists.*

Theorem 3.4 (N-Fold IP is polynomially solvable). *Fix any pair of integer matrices A, B of compatible sizes. Then there is a polynomial time algorithm that solves the generalized n-fold integer programming problem on any input n, b, c.*

3.2 N-Fold augmentation algorithm

Lemma 3.5 (Steinitz Lemma). *Let $v_1, ..., v_n$ be vectors with $||v_i|| \leq \Delta$ for $i = 1, ..., n$. If $\sum_{i=1}^{n} v_i = 0$, then there is a reordering $\pi \in S_n$ such that for each $k \in \{1, ..., n\}$ the partial sum $p_k := \sum_{i=1}^{k} v_{\pi(i)}$ satisfies $||p_k|| \leq n\Delta$.*

It's possible (using Steinitz Lemma) to obtain a much tighter bound for the norm of the elements in the Graver basis than the ones mentioned before. This implies a restriction in the space of search for the improvement direction in the augmentation algorithm making it much faster.

Lemma 3.6 (N-Fold Graver basis bound). *For all $g \in Gr(N)$ $||g||_1 \leq L_B(2r\Delta L_B + 1)^r =: L_A$ where $L_B = (2s\Delta + 1)^s$*

Lemma 3.7 (N-Fold augmentation algorithm complexity). *The N-Fold IP can be solved in time $(nt)^2 log^2(nt) \cdot \varphi(rs\Delta)^{O(r^2s+rs^2)} + LP$*

[Eisenbrand, Hunkenschröder, Klein 2018]

3.3 N-Fold via LP rounding

N-Fold resolution via RLR

- **N-Fold RLR complexity**
  The N-Fold IP restricted linear relaxation problem can be solved in time

$$O(nt \cdot log^2(nt) \cdot \varphi p(r)(s\Delta)^{O(s^2)})$$

- **N-Fold RLR to optimum complexity**
  Given an optimal vertex of an N-Fold RLR, the N-Fold IP can be solved in time

$$O(nt \cdot (rs\Delta)^{O(r^2 s + s^2)})$$

[Cslovjecsek, Eisenbrand, Weismantel 2020]

N-Fold from RLR to optimum

- **N-Fold proximity to RLR**
  Let x^* be an optimal vertex solution of a N-Fold RLR, then there exists an optimal solution z^* for the N-Fold IP verifying:

$$||z^* - x^*||_1 \leq (rs\Delta)^{O(rs)}$$

[Cslovjecsek, Eisenbrand, Weismantel 2020]

Facts for N-Fold complexity

- $|S_l| \leq (rs\Delta)^{O(r^2 s)}$
- $|V| + |E| \leq O(n(rs\Delta)^{O(r^2 s)})$
- The edge IP can be computed in time $t((r+s)\Delta)^{O(r+s)^2}$
- Longest path problem in a acyclic digraph can be solved in linear time.

N-Fold complexity

- **N-Fold complexity**
  The N-Fold IP can be solved in time $nt(rs\Delta)^{O(r^2 s + s^2)} + RLR$

[Cslovjecsek, Eisenbrand, Weismantel 2020]

Appendix A

Graver basis computation with 4ti2

Appendix B

IP resolution with Graver Basis example

Bibliography

[1] Jack E. Graver. On the foundations of linear and integer linear programming i. 1975.

[2] Raymond Hemmecke, Shmuel Onn, and Robert Weismantel. A polynomial oracle-time algorithm for convex integer minimization. 2009.

[3] Jesús A. De Loera, Raymond Hemmecke, Shmuel Onn, and Robert Weismantel. N-fold integer programming. 2006.

[4] Bernd Sturmfels. Algebraic recipes for integer programming. 2003.

[5] Elisabeth Finhold and Raymond Hemmecke. Lower bounds on the graver complexity of m-fold matrices. 2013.

[6] Friedrich Eisenbrand, Christoph Hunkenschröder, and Kim-Manuel Klein. Faster algorithms for integer programs with block structure. 2018.

[7] Jana Cslovjecsek, Friedrich Eisenbrand, and Robert Weismantel. N-fold integer programming via lp rounding. 2020.

[8] Raymond Hemmecke, Shmuel Onn, and Lyubov Romanchuk. N-fold integer programming in cubic time. 2011.

[9] Friedrich Eisenbrand and Robert Weismantel. Proximity results and faster algorithms for integer programming using the steinitz lemma. 2019.

[10] Raymond Hemmecke. Exploiting symmetries in the computation of graver bases. 2004.

[11] Elisabeth Finhold and Raymond Hemmecke. Lower bounds on the graver complexity of m -fold matrices. 2013.

[12] Shmuel Onn. Convex discrete optimization. 2007.

[13] N. Alon and K. A. Berman. Regular hypergraphs, gordon's lemma, steinitz' lemma and invariant theory. 1986.