Graver basis

Francisco Javier Blázquez Martínez

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Mathematics section,

Chair of discrete optimization

Project director: Dr. Friedrich Eisenbrand

Project supervisor: Jana Cslovjecsek

December 2020

Acknowledgements

I want to thank the Complutense University and all the teachers I've had these last four years for introducing me in these fantastic worlds that are the Mathematics and the Computer Sciences. I especially want to thank Daniel Chaver, David Atienza and Katzalin Olcoz for helping me in the process of coming to this fantastic university, the École Politechnique Fédéral de Lausanne, to finish my studies.

Of course I want to thank Professor Friedrich Eisenbrand for trusting me for this project and Jana Cslovjecsek for helping me during the whole process.

Contents

Chapter 1

Introduction

The underlying problem we try to solve is the clasical *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 its simplicity, it's well known the importance of IP. Several problems in diverse fields of the mathematics and algorithms admit an IP equivalent formulation (examples?). Unfortunately, IP is NP-Complete. This means that there is no efficient (polynomial) algorithm for solving IP in the general case (say general techniques and complexity?) and, therefore, knowing their importance and the lack of a general efficient algorithm for their resolution, there has been a great interest in restricted formulations of the problem and their resolution techniques.

In this project we present the concept of **Graver Basis** and its applications for solving the IP with, of course, the theoretical justification of this based on its properties. We apply this to a concrete IP formulation, the N-Fold IP and prove that it leads to a polynomial and efficient algorithm for this case.

Chapter 2

Graver Basis

Definition 2.1. Two vectors $u, v \in \mathbb{R}^n$ are said to be **sign compatible** if $u_i \cdot v_i \geq 0$ for all $i \in \{1, ..., n\}$.

Definition 2.2. A vector $u \in ker(A)$ is **indecomposable** if it is not the sum of two sign compatible and non zero elements in $ker(A)$.

Definition 2.3 (Graver basis). The Graver Basis of a given matrix $A \in \mathbb{Z}^{mxn}$ is defined as the set of integral indecomposable elements in the kernel of A.
(Initially defined as *universal integral test set* in [Graver 1975])

Proposition 2.4. *For every matrix A, $Gr(A)$ is a finite set.*

Proposition 2.5. *Every integral element in ker(A) can be expressed as positive integral linear combination of elements in $Gr(A)$.*

Proposition 2.6. *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 $l \leq z + g \leq u$*

Proposition 2.7 (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]*

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]

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

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

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

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

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

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

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

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

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