ARTICLE IN PRESS

Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

# Influence of the normalization constraint on the integral simplex using decomposition

Samuel Rosat [a,c,*], Issmail Elhallaoui [a,c], François Soumis [a,c], Driss Chakour [b,c]

[a] Polytechnique Montréal, C.P. 6079, Succ. Centre-ville, Montreal, QC, Canada H3C 3A7

[b] École Polytechnique, Route de Saclay, 91128 Palaiseau, France

[c] GERAD, 3000, ch. de la Côte-Sainte-Catherine, Montreal, QC, Canada H3T 2A7

ARTICLE INFO

*Article history:*
Received 28 November 2014
Received in revised form 7 December 2015
Accepted 17 December 2015
Available online xxxx

*Keywords:*
0–1 programming
Set partitioning problem
Primal algorithms
Normalization constraint
Augmenting algorithms

ABSTRACT

Since its introduction in 1969, the set partitioning problem has received much attention, and the structure of its feasible domain has been studied in detail. In particular, there exists a decreasing sequence of integer feasible points that leads to the optimum, such that each solution is a vertex of the polytope of the linear relaxation and adjacent to the previous one. Several algorithms are based on this observation and aim to determine that sequence; one example is the integral simplex using decomposition (ISUD) of Zaghrouti et al. (2014). In ISUD, the next solution is often obtained by solving a linear program without using any branching strategy. We study the influence of the normalization-weight vector of this linear program on the integrality of the next solution. We extend and strengthen the decomposition theory in ISUD, prove theoretical properties of the generic and specific normalization constraints, and propose new normalization constraints that encourage integral solutions. Numerical tests on scheduling instances (with up to 500,000 variables) demonstrate the potential of our approach.



## 1. Introduction

Consider the set partitioning problem (SPP)

$$z^\star_{\text{SPP}} = \min_{\boldsymbol{x}} \left\{ \boldsymbol{c}^T \boldsymbol{x} \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{e},\ \boldsymbol{0} \le \boldsymbol{x} \le \boldsymbol{e},\ \boldsymbol{x} \text{ is integer} \right\} \tag{SPP}$$

where $\boldsymbol{A} \in \{0,1\}^{m\times n}$ is an $m \times n$ binary matrix, and $\boldsymbol{c} \in \mathbb{N}^n$ is the cost vector. The vector of all zeros (resp. ones) with dimension dictated by the context is denoted $\boldsymbol{0}$ (resp. $\boldsymbol{e}$). Without loss of generality, we assume that $\boldsymbol{A}$ is full rank, contains no zero rows or columns, and has no identical rows or columns. $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{e}$ are called the linear constraints and $\boldsymbol{0} \le \boldsymbol{x} \le \boldsymbol{e}$ the bound constraints. $\mathcal{F}_{\text{SPP}}$ denotes the set of all feasible solutions of SPP; $z^\star_{\text{SPP}}$ is called the optimal value (or optimum) of SPP; and any feasible solution $\boldsymbol{x}^\star \in \mathcal{F}_{\text{SPP}}$ such that $\boldsymbol{c}^T\boldsymbol{x}^\star = z^\star_{\text{SPP}}$ is called an optimal solution of SPP. Note that, given the bounds ($\boldsymbol{0}$ and $\boldsymbol{e}$) and the integrality constraints, $\mathcal{F}_{\text{SPP}}$ contains only 0–1 vectors, i.e., $\mathcal{F}_{\text{SPP}} \subseteq \{0,1\}^n$. Finally, the linear relaxation of SPP, denoted $\text{SPP}^{\text{RL}}$, is the linear program obtained by removing the integrality constraints of SPP. Its feasible domain and optimal value are respectively denoted $\mathcal{F}_{\text{SPP}^{\text{RL}}}$ and $z^\star_{\text{SPP}^{\text{RL}}}$. Note that all optimization programs will be written in their minimization form, but the ideas and results also apply to maximization scenarios.

* Corresponding author at: Polytechnique Montréal, C.P. 6079, Succ. Centre-ville, Montreal, QC, Canada H3C 3A7.
*E-mail addresses:* samuel.rosat@gerad.ca (S. Rosat), issmail.elhallaoui@gerad.ca (I. Elhallaoui), francois.soumis@gerad.ca (F. Soumis), driss.chakour@polytechnique.edu (D. Chakour).

http://dx.doi.org/10.1016/j.dam.2015.12.015
0166-218X/

Please cite this article in press as: S. Rosat, et al., Influence of the normalization constraint on the integral simplex using decomposition, Discrete Applied Mathematics (2016), http://dx.doi.org/10.1016/j.dam.2015.12.015

### 1.1. Integral simplex methods for the set partitioning problem

Since its introduction by Garfinkel and Nemhauser in 1969 [5], the set partitioning problem has received much attention because of its wide range of applications: vehicle and crew scheduling, clustering problems, etc. It often appears within column-generation frameworks for such problems.

Many algorithms have been developed to solve SPP. As is the case for generic integer linear programs, they can be classified into three main classes [10]: *dual fractional*, *dual integral*, and *primal* (or *augmentation*) methods. *Dual fractional* algorithms maintain optimality and linear-constraint feasibility at every iteration, and they stop when integrality is achieved. They are typically standard cutting plane procedures such as Gomory's algorithm [7]. The classical branch-and-bound scheme is also based on a dual-fractional approach, in particular for the determination of lower bounds. *Dual integral* methods maintain integrality and optimality, and they terminate once the primal linear constraints are satisfied. Letchford and Lodi [10] give a single example: another algorithm of Gomory [8]. Finally, *primal algorithms* maintain feasibility (and integrality) throughout the process and stop when optimality is reached. These are in fact descent algorithms for which the improving sequence $(\boldsymbol{x}_k)_{k=1\ldots K}$ satisfies the following conditions:

C1 $\boldsymbol{x}^k \in \mathcal{F}_{\text{SPP}}$;
C2 $\boldsymbol{x}^K$ is optimal;
C3 $\boldsymbol{c}^T\boldsymbol{x}^{k+1} < \boldsymbol{c}^T\boldsymbol{x}^k$.

A sequence satisfying the three conditions is called a sequence of *augmenting solutions* even in a minimization problem.

Given a current solution $\boldsymbol{x}^k \in \mathcal{F}_{\text{SPP}}$, primal algorithms are in fact based on the iterative solution of the *augmentation problem* defined as:

| | |
|---|---|
| **AUG** | Find an augmenting vector $\boldsymbol{y} \in \mathbb{N}^n$ s.t. $\boldsymbol{x}^{k+1} = (\boldsymbol{x}^k + \boldsymbol{y}) \in \mathcal{F}_{\text{SPP}}$ and $\boldsymbol{c}^T\boldsymbol{y} < 0$ or assert that $\boldsymbol{x}^k$ is optimal for SPP. |

Traditionally, papers on constraint aggregation and integral simplex algorithms deal with minimization problems, whereas most authors present generic primal algorithms for maximization problems. We therefore draw the reader's attention to the following: **to retain the usual classification, we call the improvement problem AUG, although it supplies a decreasing direction**. For further information on primal algorithms in a general context, see the review of Spille and Weismantel [19].

Two special features of SPP make it a particularly promising candidate for specialized primal methods. First, by definition, SPP is a 0–1 program, so every (integer) point of $\mathcal{F}_{\text{SPP}}$ is an extreme point (or vertex) of $\mathcal{F}_{\text{SPP}^{\text{RL}}}$. Thus, there exists a decreasing sequence of basic solutions satisfying conditions C1–C3. Second, as shown by Trubin et al. [21], SPP is *quasi-integral*, i.e., every edge of $Conv(\mathcal{F}_{\text{SPP}})$ is an edge of $\mathcal{F}_{\text{SPP}^{\text{RL}}}$. Thus, there exists a sequence of augmenting solutions such that the following condition holds:

C4 $\boldsymbol{x}^{k+1}$ is an adjacent vertex of $\boldsymbol{x}^k$ in $\mathcal{F}_{\text{SPP}^{\text{RL}}}$.

An augmentation that satisfies C4 is called an *integral augmentation*. Any sequence satisfying C1–C4 is a sequence of *augmenting adjacent solutions*, and an algorithm that yields such a sequence is an *integral simplex*. Every solution can be obtained from the previous one in the sequence by performing one or several simplex pivots in $\text{SPP}^{\text{RL}}$, hence the name. Such sequences were first introduced by Balas and Padberg [1]. Several other authors (Haus et al. [9], Thompson [20], Saxena [17]) have presented enumeration schemes that move from one integer solution to an adjacent one. An important recent contribution, in terms of both theory and applications, has been made by Rönnberg and Larsson [12–14]. However, none of these enumeration methods can guarantee a strict improvement (C3). They may perform degenerate pivots, in which there is no effective change in the solution (or the objective value) because the entering variable(s) takes the value zero. SPP tends to suffer severe degeneracy, so the computational time of these algorithms grows exponentially with the size of the instances.

Another pivot-based method proposed in the same spirit is the integral simplex using decomposition (ISUD) of Zaghrouti et al. [22]. It is one of the most promising recent developments, because it is based on linear programming techniques that take advantage of degeneracy and guarantee strict improvement at each iteration (see [3,11]). It follows an augmenting sequence of integer points leading to an optimal solution, such that each visited point is a vertex of $\mathcal{F}_{\text{SPP}^{\text{RL}}}$ adjacent to the previous one (integral simplex). To find the edge leading to the next point, one solves a linear program to select an augmenting direction for the current point from a cone of feasible directions. To ensure that this linear program is bounded (the directions could go to infinity), a normalization constraint is added and the optimization is performed on a section of the cone. The solution of the linear program, i.e., the direction proposed by the algorithm, depends strongly on the chosen normalization weights, and so does the likelihood that the next solution is integer. In their seminal paper, Zaghrouti et al. [22] consider all the weights to be equal to 1 in the normalization constraint; they therefore call it the *convexity* constraint. We extend the algorithm to the case of a generic normalization constraint, explore the theoretical properties of some specific constraints, and discuss the design of the normalization constraint based on our theoretical observations. We also report preliminary computational results that compare different normalization strategies and highlight their influence on the behavior of the algorithm.

Please cite this article in press as: S. Rosat, et al., Influence of the normalization constraint on the integral simplex using decomposition, Discrete Applied Mathematics (2016), http://dx.doi.org/10.1016/j.dam.2015.12.015