

---

This is an electronic reprint of the original article.  
This reprint may differ from the original in pagination and typographic detail.

Oliveira, Fabricio; Christiansen, Jeffrey; Dandurand, Brian; Eberhard, Andrew

## Combining penalty-based and Gauss–Seidel methods for solving stochastic mixed-integer problems

*Published in:*  
International Transactions in Operational Research

DOI:  
[10.1111/itor.12525](https://doi.org/10.1111/itor.12525)

Published: 01/01/2020

*Document Version*  
Early version, also known as pre-print

*Please cite the original version:*  
Oliveira, F., Christiansen, J., Dandurand, B., & Eberhard, A. (2020). Combining penalty-based and Gauss–Seidel methods for solving stochastic mixed-integer problems. *International Transactions in Operational Research*, 27(1), 494-524. <https://doi.org/10.1111/itor.12525>

---

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.



## Combining Penalty-based and Gauss-Seidel Methods for solving Stochastic Mixed-Integer Problems

Journal:	<i>International Transactions in Operational Research</i>
Manuscript ID	ITOR-Oct-16-OA-0406.R1
Manuscript Type:	Original Article
Date Submitted by the Author:	n/a
Complete List of Authors:	Oliveira, Fabricio; RMIT University, School of Science Christiansen, Jeffrey; RMIT University, School of Science Dandurand, Brian; RMIT University, School of Science Eberhard, Andrew; RMIT University, School of Mathematical and Geospatial Sciences
Keywords:	Stochastic programming, Decomposition methods, Lagrangian duality, Penalty-based method, Gauss-Seidel method
Main Contributions:	Characterization of a family of penalty functions that achieve strong duality, Novel decomposition approach for stochastic mixed-integer programming, Combining penalty based and Gauss-Seidel methods to achieve parallelisation via separability into MIPs

Combining Penalty-based and Gauss-Seidel Methods for  
solving Stochastic Mixed-Integer Problems

F. Oliveira, J. Christiansen, B. Dandurand, A. Eberhard

Mathematical Sciences  
School of Science - RMIT University

Abstract

In this paper, we propose a novel decomposition approach for stochastic mixed-integer programming (SMIP) problems that is inspired by the combination of penalty-based Lagrangian and block Gauss-Seidel methods (PBGS). In this sense, PBGS is developed such that the inherent decomposable structure that SMIPs present can be exploited in a computationally efficient manner. The performance of the proposed method is compared with the Progressive Hedging method (PH), which also can be viewed as a Lagrangian-based method for obtaining solutions for SMIP. Numerical experiments performed using instances from the literature illustrate the efficiency of the proposed method in terms of computational performance and solution quality.

*Keywords:* Stochastic programming, Decomposition methods, Lagrangian duality, Penalty-based method, Gauss-Seidel method

1. Introduction

Inspired by recent advances and the increased availability of parallel computation resources, there has been a recent surge of methods which are capable of exploiting the structure of large-scale mathematical programming problems to achieve increased efficiency.

One relevant class of problems that can benefit from this paradigm is stochastic mixed-integer programming (SMIP) problems. The modelling framework for this class

\*Corresponding Author

of problems is versatile as it simultaneously allows for representation of integer-valued decisions and uncertainty in the input data. However, they are frequently challenging in terms of computational tractability due to their inherent NP-hard nature and their large-scale that arise from their scenario-based representations. Parallel computation is particularly appropriate for solving SMIPs, since the special structure of these problems (i.e. their partial separation by decision stage and by outcome scenario) makes them easier to decompose into smaller subproblems which may then be solved simultaneously.

The opportunities arising from approaching SMIP problems by means of decomposition has prompted the development of several different theoretical and algorithmic approaches. For example, the Integer L-Shaped method [24] employs Benders' decomposition to achieve stage-wise decomposition of SMIP problems. Other algorithms employ Lagrangian duality to achieve scenario-wise decomposition, such as the Dual Decomposition algorithm [11] which uses Lagrangian dual bounds in a branch-and-bound framework, or Progressive Hedging (PH) [31, 25, 35, 34] which applies an Alternating-Direction-type method to the augmented Lagrangian dual problem. Recent studies and applications of these methods include [1, 26, 21, 3, 17] and references therein.

All of the above methods are based on the concept of duality, and therefore they must consider the *duality gap* that may exist between the optimal values of the original (primal) problem and the dual problem. This duality gap is frequently nonzero in the context of non-convex problems, such as those with integer decision variables. If the duality gap for a particular problem is large, any algorithm based on that dual is unlikely to be effective [12].

Several possible approaches to modifying Lagrangian duality to deal with the duality gap which arises in non-convex problems have been considered in the literature. These approaches include  $l_1$ -like penalty functions [12], indicator augmenting functions [23], nonlinear Lagrangian functions [37], and semi-Lagrangian duality [4]. With the exception of nonlinear Lagrangian functions, these approaches have not yet been widely exploited in terms of experimental investigation and practical applications despite numerous theoretical developments available in literature.

In this paper, we propose an alternative approach to deal with SMIP problems that builds upon recent theoretical results from [9] and [15] showing that duality gaps can be

diminished with the use of finite-valued penalties for specific class of penalty functions. We show that one can obtain reasonable penalty functions using *positive bases* and that parallelisation can be obtained by the application of a block Gauss-Seidel approach. The combination of these two frameworks allows us to develop an efficient heuristics that is capable of providing solutions for large-scale SMIP problems. In terms of objective value quality and computational time, the developed approach is shown to be competitive with existing approaches such as PH, which, despite its heuristic nature in the context of SMIPs, has been relied upon as an efficient solution method (see, for example [32, 29, 34]). Furthermore, the theoretical basis for PH does not apply for problems containing integer variables. On the other hand there exists some supporting theory for the PBGS approach which we present in this paper. This partial theory has the potential to inform directions for the further improvement of heuristic methods of this kind.

This paper is structured as follows. In Section 2 we cover the technical background which will be used in the development of the proposed method. Section 3 describes the development of the penalty-based block Gauss-Seidel method, while in Section 4 we discuss computational aspects of the algorithm. Section 5 provides results of the numerical experiments performed. Finally, in Section 6 we provide conclusions and directions for further development of this research.

## 2. Technical Background

In the following developments, we consider two-stage stochastic mixed-integer programming problems of the form:

$$\zeta^{SIP} := \min_{x,y} c^\top x + \sum_{s \in S} p_s (q_s^\top y_s) \quad (1)$$

$$\text{s.t.: } x \in X \quad (2)$$

$$y_s \in Y_s(x), \forall s \in S, \quad (3)$$

where  $x \in \mathbb{R}^{n_x}, y \in \mathbb{R}^{n_y \times |S|}$  are decision variables,  $c \in \mathbb{R}^{n_x}$  and  $q \in \mathbb{R}^{n_y \times |S|}$  are input parameters, and  $p_s \in \mathbb{R}^{|S|}$  represents the scenario probabilities. Sets  $X \subset \mathbb{R}^{n_x}$  and  $Y_s(x) \subset \mathbb{R}^{n_y \times |S|}, \forall x \in X$ , define the feasible decision set and consist of integrality restrictions on  $x$  and  $y$ , and linear constraints in  $x$  for  $X$ , and in  $x$  and  $y$  for  $Y_s(x)$ .

To obtain a formulation that is amenable to decomposition, i.e., that can be exploited in terms of its block-angular structure,  $\zeta^{SIP}$  may be equivalently rewritten as:

$$\zeta^{SIP} := \min_{x,y,z} \sum_{s \in S} p_s (c^\top x_s + q_s^\top y_s) \quad (4)$$

$$\text{s.t.: } x_s - z = 0, \quad \forall s \in S \quad (5)$$

$$x_s \in X, \quad \forall s \in S \quad (6)$$

$$y_s \in Y_s(x_s), \quad \forall s \in S. \quad (7)$$

The set of constraints represented by (5) are referred to in the relevant literature as the non-anticipativity constraints (NAC), as they enforce consensus over the decision made before the observation of a given scenario  $s \in S$ . It is straightforward to see that these constraints prevent the problem from being solved by means of a decomposition approach that can exploit the otherwise block-angular structure of the problem. Hence, a natural way to approach this class of problems consists of relying on frameworks that are capable of considering relaxations for  $\zeta^{SIP}$  which do not include constraint (5).

### 2.1. Lagrangian Relaxation

A natural approach to solve this problem is to relax the NAC by means of Lagrangian relaxation. To achieve this, let  $\omega = (\omega_s)_{s \in S}$  be the Lagrangian multipliers associated with the constraints (5). Then, our Lagrangian relaxation can be formulated as the following problem:

$$\begin{aligned} \zeta^{LR}(\omega) &:= \min_{x,y,z} \sum_{s \in S} L_s(x_s, y_s, z, \omega_s) \\ \text{s.t.: } &x_s \in X, \quad \forall s \in S \\ &y_s \in Y_s(x_s), \quad \forall s \in S, \end{aligned}$$

where

$$L_s(x_s, y_s, z, \omega_s) := p_s (c^\top x_s + q_s^\top y_s) + \omega_s^\top (x_s - z). \quad (8)$$

In order to guarantee that  $\zeta^{LR}(\omega)$  has a bounded optimal solution, one must enforce that the dual feasibility condition  $\omega \in \Omega := \{\omega \mid \sum_{s \in S} p_s^\top \omega_s = 0\}$  holds. Under this assumption, (8) may be rewritten as

$$L_s(x_s, y_s, z, \omega_s) = p_s (c^\top x_s + q_s^\top y_s) + \omega_s^\top x_s, \quad (9)$$

which forces the  $z$  variable to vanish (note that the term  $-\omega_s^\top z$  was in fact the potential cause of the unboundedness of the dual relaxation  $\zeta^{LR}(\omega)$ ) and its removal yields complete separability for each  $s \in S$ .

It is well known that  $\zeta^{LR}(\omega) \leq \zeta^{SIP}$  for any  $\omega \in \Omega$ . The *Lagrangian dual problem* consists of finding the supremal value of  $\zeta^{LR}(\omega)$  over  $\omega \in \Omega$  to most closely approximate or bound  $\zeta^{SIP}$  from below, which in practice means solving the problem

$$\zeta^{LD} := \sup_{\omega \in \Omega} \zeta^{LR}(\omega). \quad (10)$$

In general only the weak duality condition  $\zeta^{LR}(\omega) \leq \zeta^{SIP}$  holds. When equality  $\zeta^{LD} = \zeta^{SIP}$  holds, we have strong duality. Due to the presence of integer restricted variables, the primal problem (4)-(7) is not convex, and strong duality (that is  $\zeta^{LD} = \zeta^{SIP}$ ) cannot be guaranteed. Instead, we typically have a duality gap, i.e.  $\zeta^{LD} < \zeta^{SIP}$ .

## 2.2. Augmented Lagrangian Approach

In the particular domain of mixed-integer problems such as SMIP problems, there has been renewed interest in the use of augmented Lagrangian approaches [10, 16, 18]. The augmented Lagrangian relaxation of  $\zeta^{SIP}$  which relaxes the NAC (5) is:

$$\zeta_\rho^{LR+}(\omega) := \min_{x,y,z} \sum_{s \in S} L_s(x_s, y_s, z, \omega_s) + \psi_\rho^s(x_s - z) \quad (11)$$

$$\text{s.t.: } x_s \in X, \forall s \in S \quad (12)$$

$$y_s \in Y_s(x_s), \forall s \in S, \quad (13)$$

where  $\omega = (\omega_s)_{s \in S} \in \Omega$  and  $\psi_\rho^s : \mathbb{R}^{n_x} \mapsto \mathbb{R}$  is an appropriate penalty function specific to scenario  $s$  that depends on the penalty parameter  $\rho$ . As in the ordinary Lagrangian relaxation,  $\omega \in \Omega$  implies that  $\sum_{s \in S} p_s^\top \omega_s = 0$  so as to ensure that the dual problem has a finite optimal value. The *augmented Lagrangian dual problem* is:

$$\zeta_\rho^{LD+} := \sup_{\omega \in \Omega} \zeta_\rho^{LR+}(\omega).$$

A common choice for the penalty function, in this context, is  $\psi_\rho^s(u_s) := \frac{\rho}{2} \|u_s\|^2$  for each  $s \in S$ , which provides smoothness to the original scenario-wise augmented Lagrangian dual function [30, 6, 7].

Recent results have shown that, for penalty functions satisfying some slight conditions, the augmented Lagrangian dual is capable of asymptotically achieving zero duality gap when the penalty term coefficient  $\rho$  is allowed to go to infinity [9, Prop. 3], [15, Prop. 2]. However, despite the theoretical relevance of this observation, it is not practically meaningful to deal with large-valued penalty parameters, in large part due to the associated numerical issues that arise.

Furthermore, [9, Cor. 1], [15, Thm. 4] demonstrates that it is possible to circumvent this drawback if the augmentation of the Lagrangian dual is made using a norm as the penalty function. In this case, the theory suggests that it is possible to attain strong duality for a finite value of  $\rho$ . This result is one of the major motivations for the developments to be presented next.

### 2.3. Semi-Lagrangian Duality

Semi-Lagrangian duality [4] is a variant of Lagrangian duality in which "difficult" equality constraints (e.g.  $Ax = b$ ) are reformulated as pairs of inequality constraints ( $Ax \leq b$  and  $Ax \geq b$ ). Lagrangian relaxation is then applied to one of the two sets of inequality constraints. Surrogate semi-Lagrangian duality [27, 22] is a variant which replaces one set of inequalities with its weighted sum ( $\lambda^\top Ax \leq \lambda^\top b$ , where  $\lambda \geq 0$  is a non-negative vector of the weights applied to each inequality) and then applies Lagrangian relaxation to the resulting single inequality. Special classes of problems can exhibit zero duality gap when utilising the semi-Lagrangian dual problem.

The semi-Lagrangian approach is effective when the semi-Lagrangian dual problem is more tractable than the original problem, even though some inequalities remain in the dual problem as explicit constraints. For problems to which semi-Lagrangian relaxation has been previously applied, such as the p-median problem [4] and the uncapacitated facility location problem [5], the semi-Lagrangian dual problem may be simplified by choosing appropriate dual variable values.

The method presented in this paper is similar to semi-Lagrangian duality in that it can be interpreted as a penalty-method analogue of applying Lagrangian duality to a reformulation of the original problem in which equalities are rewritten as inequalities. To attain this objective, let us first reformulate the problem (1)-(3) into the following



equivalent form:

$$\begin{aligned} \zeta^{SIP} : \min_{x,y,z} \sum_{s \in S} p_s(c^\top x_s + q_s^\top y_s) \\ \text{s.t.: } x_s - z \leq 0, \forall s \in S \end{aligned} \quad (14)$$

$$-(x_s - z) \leq 0, \forall s \in S \quad (15)$$

$$x_s \in X, \forall s \in S$$

$$y_s \in Y_s(x_s), \forall s \in S.$$

Unfortunately, in the context of the SMIP problems studied in this paper, the inequalities left as explicit constraints by semi-Lagrangian duality would not allow us to achieve our goal of scenario-wise decomposition.

The method presented here instead relaxes inequalities (14) and (15), so that penalty-based approaches can treat the deviations from the separate inequalities differently. Furthermore, the resulting dual problem is separable by scenario within the block Gauss-Seidel framework. As will be observed later, choosing penalty functions based on some positive bases can result in penalty terms analogous to the objective terms obtained through surrogate semi-Lagrangian duality.

#### 2.4. Desirable Properties of Penalty Functions

Our primary objective is to compute  $\zeta_\rho^{LD+}$  in a decomposed manner, which will require: *i)* the definition of suitable penalty functions  $\psi_\rho^s$  for each  $s \in S$ ; and *ii)* the application of a block Gauss-Seidel (GS)-based approach based on a decomposable structure.

One important result, originally proven for general mixed-integer programming (MIP) problems, which can be used in this case is Theorem 5 of [15], which is reproduced below (adapted to the context of SMIP problems).

**Theorem 1.** [15, Thm. 5] *Consider a feasible MIP problem given in (1)-(3) whose problem data is formed from rational entries and with its optimal value bounded. If  $\psi : \prod_{s \in S} \mathbb{R}^{n_x} \mapsto \mathbb{R}$  is a summed augmenting function  $\psi(u) := \sum_{s \in S} \psi_\rho^s(u_s)$  for problem (11)-(13) such that*

1.  $\psi(0) = 0$
2.  $\psi(u) \geq \delta > 0, \forall u \notin V$

$$3. \psi(u) \geq \gamma \|u\|_\infty, \forall u \in V$$

for some open neighbourhood  $V$  of 0, and positive scalars  $\delta, \gamma > 0$ , then there exists a finite  $\rho$  such that  $\zeta_\rho^{LD+} = \zeta_\rho^{LR+}(\omega_{LP}) = \zeta^{SIP}$ , for  $\omega_{LP}$  (an optimal multiplier of the linear programming relaxation of the NACs (5)).

**Proof.** Apply the general theorem [15, Thm. 5] to our problem (1)-(3). ■

**Remark 2.** In [9] other conditions that do not require the assumption of rationality of the data defining the problem are given that also ensure a limiting zero duality gap.

**Remark 3.** One may see with little difficulty that the proof of [15] does not rely on the setting of  $\omega = \omega_{LP}$ . Indeed one can show that for any  $\omega \in \Omega$  there still exists a finite penalty parameter such that Theorem 1 holds true. To see this, considering the notation of [15], one needs to replace  $\bar{\lambda}_{LP}$  with a generic (dual feasible)  $\lambda$  and  $z^{LP}$  in [15, Thm. 5] with the Lagrangian dual function  $z^{LR}(\lambda)$  defined earlier in [15]. Also, adjust the definition of  $\hat{\rho}$  in [15, Thm. 5] accordingly.

In [9] and later in [15] it has been observed that a zero duality gap is achievable for dual problems based on an augmented Lagrangian in MIP problems. In both papers, very general classes of augmenting functions were studied and consequently very little can be inferred as to what would be a practical penalty that one could use on a given problem. It is observed in [15] that the usual quadratic (squared Euclidean norm) penalty is probably not a practical choice for MIP. One would hope that an augmenting function would lead to a reformulation of the MIP that is not significantly worse to solve than the original problem, which would mean that augmenting functions should lead to a MIP reformulation.

Motivated by the aforementioned facts, we propose a class of augmenting functions based on the use of *positive basis* [14]. One special case of this class of penalty functions is given as follows. Given discrepancy vector  $u := (u_s)_{s \in S} \in \prod_{s \in S} \mathbb{R}^{n_x}$ , we define for each scenario  $s$  the penalty function

$$\psi_\rho^s(u_s) := \underline{\rho}_s^\top [u_s]^- + \bar{\rho}_s^\top [-u_s]^-,$$

where  $\rho = (\underline{\rho}_s, \bar{\rho}_s)_{s \in S} \in \mathbb{R}_{>0}^{2n_x|S|}$  and  $[v]^- := -\min\{0, v\}$  (performed component wise), where in this case  $v \in \mathbb{R}^{n_x}$ . Then we define

$$\psi_\rho(u) := \sum_{s \in S} \psi_\rho^s(u_s) = \sum_{s \in S} \underline{\rho}_s^\top [u_s]^- + \sum_{s \in S} \bar{\rho}_s^\top [-u_s]^- . \quad (16)$$

In the following developments we shall demonstrate that (16) satisfies the conditions of Theorem 1 and indeed lies in a special class of augmenting functions that form a practical set from which one can tailor-make an augmenting function for a given problem.

### 2.5. Positive Basis

A subset of reasonable augmenting functions may be defined by using a *positive basis*  $\{n_1, \dots, n_l\}$  to scalarise the deviations  $u \in \mathbb{R}^m$  (note that for our purposes,  $m = n_x |S|$ ). Such deviations can be associated, for example, with the satisfaction of linear inequalities, where we might have  $u = b - Ax$  given a constraint  $Ax \leq b$ .

**Definition 4.** We say a set of vectors  $\{n_1, \dots, n_l\}$ , where  $m+1 \leq l \leq 2m$ , is a *positive basis* for  $\mathbb{R}^m$  if and only if every  $u \in \mathbb{R}^m$  can be expressed as a non-negative combination of these vectors, i.e., there exists  $\alpha_i \geq 0$  for  $i = 1, \dots, l$  for which  $u = \sum_{i=1}^l \alpha_i n_i$ .

Notice that in this paper we adopt the original nomenclature *positive basis* proposed in [14], even though non-negative combinations are taken of its elements  $\{n_1, \dots, n_l\}$  to form each  $u \in \mathbb{R}^m$ . The following property of positive bases will be useful in the developments which follow.

**Theorem 5.** ([14] Theorem 3.1)  $\{n_1, \dots, n_l\}$  positively spans  $\mathbb{R}^m$  if and only if for every non-zero  $u$  there exists an index  $i$  such that  $u \cdot n_i > 0$ .

Let  $e_i$ ,  $i = 1, \dots, m$  represent the elementary unit vectors of  $\mathbb{R}^m$  with entry  $i$  set to one and all other entries set to zero. Examples of positive bases on  $\mathbb{R}^m$  include:

- The vertices of a  $m$ -simplex (generalised tetrahedron), centred at the origin.
- The set of vectors  $\{+e_i\}_{i=1}^m \cup \{\sum_{i=1}^m -e_i\}$
- The set of vectors  $\{\pm e_i\}_{i=1}^m$

## 2.6. Norm-Like Augmenting Functions

As noted in both [9] and [15], norms are viable augmenting functions with appealing theoretical support for overcoming duality gaps. However, norms have the disadvantage that they uniformly penalise constraint violations, which results in a loss of flexibility and precision in the fine-tuning of the penalisation. This limitation motivates the following development of asymmetrical but norm-like augmenting functions. The polyhedral norms  $\|\cdot\|_\infty$  and  $\|\cdot\|_1$  may be represented using the positive basis  $\{\pm e_i\}_{i=1}^m$  in the following ways:

$$\psi_\infty(u) := \|u\|_\infty = \max_{i=1,\dots,m} \{\pm e_i^\top u\}, \text{ and} \quad (17)$$

$$\psi_1(u) := \|u\|_1 = \sum_{i=1}^m \max\{+e_i^\top u, 0\} + \sum_{i=1}^m \max\{-e_i^\top u, 0\} \quad (18)$$

$$\text{or equivalently } \|u\|_1 = \sum_{i=1}^m \max\{\nu_i^\top u : \nu_i \in \{+e_i, -e_i\}\}. \quad (19)$$

The representation in (19) relies on each vector in the basis having a negative multiple which is also in the basis. The representations in (17) and (18) do not have this limitation, and may be generalised to any positive basis  $N := \{n_1, \dots, n_l\}$  as follows:

$$\psi_\infty^N(u) := \max_{i=1,\dots,l} \{n_i^\top u\} \text{ and} \quad (20)$$

$$\psi_1^N(u) := \sum_{i=1}^l \max\{n_i^\top u, 0\}. \quad (21)$$

The functions  $\psi_\infty^N$  and  $\psi_1^N$  are not necessarily norms, but do share some useful properties with norms. Specifically, these functions are positive homogeneous (which implies that they vanish at zero), continuous, strictly positive for all  $u \in \mathbb{R}^m \setminus \{0\}$ , finite valued, sub-additive, and coercive.

The proposed augmenting function  $\psi_\rho(u)$  given in (16) may be represented in the form of (21), using the positive basis  $N_\rho = \{\bar{\rho}_{s,i} e_{i+(s-1)n_x} \mid s \in S, i \in \{1, \dots, n_x\}\} \cup \{-\underline{\rho}_{s,i} e_{i+(s-1)n_x} \mid s \in S, i \in \{1, \dots, n_x\}\}$ , as follows:

$$\begin{aligned}
 \psi_1^{N_\rho}(u) &= \sum_{s \in S} \sum_{i=1, \dots, n_x} \bar{\rho}_{s,i} \max\{0, u_{s,i}\} + \sum_{s \in S} \sum_{i=1, \dots, n_x} \underline{\rho}_{s,i} \max\{0, -u_{s,i}\} \\
 &= \left( \sum_{s \in S} \underline{\rho}_s^\top [u_s]^- + \sum_{s \in S} \bar{\rho}_s^\top [-u_s]^- \right) \\
 &= \sum_{s \in S} \psi_\rho^s(u) = \psi_\rho(u).
 \end{aligned} \tag{22}$$

**Lemma 6.** *If two functions  $\psi_A : \mathbb{R}^m \rightarrow \mathbb{R}$  and  $\psi_B : \mathbb{R}^m \rightarrow \mathbb{R}$  are positive homogeneous, continuous, and strictly positive for all  $u \neq 0$  then there exists a finite  $\gamma > 0$  such that*

$$\psi_A(u) \geq \gamma \psi_B(u) \quad \text{for all } u \in \mathbb{R}^m.$$

**Proof.** Since they are positive homogeneous,  $\psi_A$  and  $\psi_B$  vanish at zero and so the required property trivially holds with equality at  $u = 0$ . To obtain the required inequality for nonzero  $u$ , set  $V = \{u : \|u\| = 1\}$  (where  $\|\cdot\|$  is any norm) and take  $\alpha = \min_{u \in V} \psi_A(u)$  and  $\beta = \max_{u \in V} \psi_B(u)$ . Since  $\psi_A$  and  $\psi_B$  are continuous and defined on the closed and bounded set  $V$ , by the Extreme Value Theorem these extrema exist and are attained by their respective functions. Since these functions are strictly positive and finite valued on  $V$ , and they attain their extrema,  $\alpha$  and  $\beta$  both strictly positive and finite.

For any point  $u \in \mathbb{R}^m \setminus \{0\}$ ,  $\|u\|$  is strictly positive and the point  $\frac{u}{\|u\|}$  is in  $V$ . Therefore, by the positive homogeneity of  $\psi_A(u)$  and  $\psi_B(u)$  we have

$$\psi_A(u) = \|u\| \psi_A\left(\frac{u}{\|u\|}\right) \geq \alpha \|u\|$$

and

$$\beta \|u\| \geq \|u\| \psi_B\left(\frac{u}{\|u\|}\right) = \psi_B(u).$$

Let  $\gamma = \alpha/\beta$ . Since  $\alpha$  and  $\beta$  are strictly positive and finite,  $\gamma$  is also strictly positive and finite. The required inequality follows:

$$\psi_A(u) \geq \alpha \|u\| = \frac{\alpha}{\beta} \beta \|u\| = \gamma \beta \|u\| \geq \gamma \psi_B(u).$$

■

**Proposition 7.** *For any positive basis  $N$ , the augmenting functions  $\psi_\infty^N$  and  $\psi_1^N$  given in (20) and (21) respectively satisfy the conditions given in Theorem 1.*

**Proof.**

Let  $V = B_\varepsilon^\infty(0)$  be an open ball in the infinity norm with radius  $\varepsilon > 0$  centred at the origin. This is an appropriate open neighbourhood of 0 for the purposes of Conditions 2 and 3 of Theorem 1.

**Condition 1:**  $\psi(0) = 0$ .

If  $u = 0$  then  $n_i^\top u = 0$  and therefore  $\psi_\infty^N(u) = 0$  and  $\psi_1^N(u) = 0$ , as required.

**Condition 2:**  $\psi(u) \geq \delta > 0, \forall u \notin V$  for some positive scalar  $\delta$ .

Using Theorem 5, for any  $u \neq 0$  we have some  $i$  such that  $n_i^\top u > 0$  and hence  $\psi_\infty^N(u) > 0$ .

Now define

$$\delta := \min_u \{ \max_{i=1,\dots,l} \{ n_i^\top u \} \mid \|u\|_\infty = \varepsilon \} > 0, \quad (23)$$

where  $\delta > 0$  follows from the compactness of the  $\varepsilon$ -ball, the continuity of  $u \mapsto \max_{i=1,\dots,l} \{ n_i^\top u \}$ , and Theorem 5. For any  $u \notin V$ , the point  $v := \varepsilon \frac{u}{\|u\|_\infty}$  is in  $V$  and hence  $\psi_\infty^N(v) \geq \delta > 0$ .

Using the positive homogeneity property we have

$$\begin{aligned} \frac{\varepsilon}{\|u\|_\infty} \psi_\infty^N(u) &\geq \delta > 0 \\ \text{and so } \psi_\infty^N(u) &\geq \delta \frac{\|u\|_\infty}{\varepsilon} \geq \delta > 0, \end{aligned}$$

using the fact that  $u \notin V$  means  $\|u\|_\infty \geq \varepsilon$ . This is the required inequality for  $\psi_\infty^N$ .

Apply Lemma 6 to deduce that there exists a  $\eta > 0$  such that:

$$\psi_1^N(u) \geq \eta \psi_\infty^N(u) \geq \eta \delta > 0 \text{ for all } u \notin V.$$

$\eta \delta$  is also a positive scalar and so this is the required inequality for  $\psi_1^N$ .

**Condition 3:**  $\psi(u) \geq \gamma \|u\|_\infty, \forall u \in V$  for some positive scalar  $\gamma$ .

The property holds trivially for  $u = 0$ . For any  $u \in V \setminus \{0\}$ , the point  $v := \varepsilon \frac{u}{\|u\|_\infty}$  is in  $V$  and using the same  $\delta$  as defined in (23) we have

$$\begin{aligned} \frac{\varepsilon}{\|u\|_\infty} \psi_\infty^N(u) &\geq \delta > 0 \\ \text{and so } \psi_\infty^N(u) &\geq \delta \frac{\|u\|_\infty}{\varepsilon} \geq \frac{\delta}{\varepsilon} \|u\|_\infty > 0, \end{aligned}$$

and so we may place  $\gamma := \frac{\delta}{\varepsilon} > 0$ . This is the required inequality for  $\psi_{\infty}^N$ .

As above, apply Lemma 6 to deduce that there exists a  $\eta > 0$  such that:

$$\psi_1^N(u) \geq \eta \psi_{\infty}^N(u) \geq \eta \gamma \|u\|_{\infty} > 0.$$

$\eta \gamma$  is also a positive scalar and so this is the required inequality for  $\psi_1^N$ . ■

**Corollary 8.** *Assume that  $\zeta^{SIP}$  is feasible, its optimal value is finite, and the data which defines it is rational. Then, for each  $\omega \in \Omega$ , there is a finite  $\rho > 0$  for which we have*

$$\zeta_{\rho}^{LD+} = \zeta_{\rho}^{LR+}(\omega) = \zeta^{SIP} \quad (24)$$

with the augmenting function taking the form 20 or (21). In particular, this applies to our proposed augmenting function (16).

**Proof.** Equalities (24) follow directly from Theorem 1, Remark 3, and Proposition 7. The last claim follows from the observation that (16) may be represented as a function of the form of (21), as demonstrated in (22). ■

**Remark 9.** *Consider a positive basis  $N = \{n_1, \dots, n_l\}$ . Each of the functions  $g_i(u) = \max\{n_i^{\top} u, 0\}$  is non-negative, positive homogeneous and finite valued, and these properties are preserved if multiple  $g_i$ s are summed, or their maximum is taken. By Theorem 5, for any non-zero  $u$  there exists an index  $i \in \{1, \dots, l\}$  such that  $g_i(u)$  is strictly positive. Therefore, if every one of the  $g_i$  functions is combined using a combination of summation and/or maximisation, the resulting function  $g(u)$  will be strictly positive for all non-zero  $u$ . Applying Lemma 6 to bound  $g$  below by a positive multiple of  $\psi_{\infty}^N$  (as  $\psi_1^N$  was treated in Proposition 7) shows that this function  $g(u)$  satisfies the conditions of Theorem 1, and as such will close the duality gap if used as an augmenting function.*

Remark 9 implies that we can construct not only  $\psi_{\infty}^N$  and  $\psi_1^N$  but also a wide variety of other augmenting functions from any given positive basis, depending on the order in which the maximisation and summation operations are applied to the  $g_i$  functions. Furthermore, the sum or maximum of any two augmenting functions which satisfy the conditions of Theorem 1 will itself satisfy the conditions of Theorem 1, which yields further flexibility.

**Remark 10.** *By using the positive basis  $\{+e_i\}_{i=1}^m \cup \{\sum_{i=1}^m -e_i\}$  or similar to define an augmenting function, we can obtain penalty terms analogous to the Lagrangian terms obtained through surrogate semi-Lagrangian relaxation.*

### 3. Developing a penalty-based block Gauss-Seidel method

To exploit the potential for decomposability that this formulation presents, we consider a block Gauss-Seidel (GS) method approach. In Section 3.1, we present a classical framework for GS methods as applied to nonlinear optimisation problems, and in Section 3.2, we show how it can be adapted to obtain solutions for SMIP.

#### 3.1. A block Gauss-Seidel method

We consider the general problem given by

$$\begin{aligned} \min_{x,z} f(x,z) \\ \text{s.t.: } x \in X, z \in Z. \end{aligned} \tag{25}$$

We assume that  $f$  is convex, but not necessarily differentiable. The sets  $X$  and  $Z$  are closed, but not necessarily convex. The assumption that  $x \in X$  and  $z \in Z$  are taken from disjoint sets is adequate for our purposes, although block GS approaches have been studied in a more general setting where  $(x, z)$  is taken from a set  $K \subset \mathbb{R}^{n_x+n_z}$ . (See [36] for a treatment of the case where the constraint set is disjoint, and [19] for the case where the constraint set may not be disjoint and developments are based on biconvexity assumptions.) GS methods solve problem (25) by separating it into two simpler problems. Given an iterate  $(x^k, z^k)$ , problem (25) is solved with respect to  $x$  for fixed  $z = z^k$ , yielding a new  $x$ -iterate  $x^{k+1}$ . Then, problem (25) is solved with respect to  $z$  for fixed  $x = x^{k+1}$ , yielding a new  $z$ -iterate  $z^{k+1}$ . In Algorithm 1, a formal listing of a block GS method applied to problem (25) is given.



---

**Algorithm 1** A block GS method

---

```

1: initialise  $(x^0, z^0) \in X \times Z$ 
2: for  $k = 1, \dots, k_{\max}$  do
3:    $x^k \leftarrow \operatorname{argmin}_x \{f(x, z^{k-1}) : x \in X\}$ 
4:    $z^k \leftarrow \operatorname{argmin}_z \{f(x^k, z) : z \in Z\}$ 
5:    $k \leftarrow k + 1$ 
6: end for
7: return  $(x^{k_{\max}}, z^{k_{\max}})$ 

```

---

The sequence  $\{(x^k, z^k)\}$  generated by iterations of Algorithm 1 has limit points when  $X$  and  $Z$  are compact. When  $f$  is furthermore continuous and bounded from below over  $X \times Z$ , the limit points  $(x^*, z^*) \in X \times Z$  are easily shown to be *partial minima* [36]; that is, it holds that

$$f(x^*, z^*) \leq f(x, z^*), \quad \forall x \in X, \quad (26)$$

$$f(x^*, z^*) \leq f(x^*, z), \quad \forall z \in Z. \quad (27)$$

This claim is formally stated in Proposition 11, whose proof is implicit from the developments of [36] and is given here for the sake of completeness.

**Proposition 11.** *For problem (25), let  $f$  be continuous and bounded from below, and let  $X$  and  $Z$  be compact. Then the limit points  $(x^*, z^*)$  of the sequence  $\{(x^k, z^k)\}$  generated by iterations of Algorithm 1 are partial minima.*

**Proof.** We have by construction that  $f(x^k, z^k) \leq f(x^k, z)$  for all  $z \in Z$ , and by the continuity of  $f$ , we have the second requirement  $f(x^*, z^*) \leq f(x^*, z)$ ,  $\forall z \in Z$  for partial optimality. To establish the first requirement (26), assume for sake of contradiction that there is an  $\bar{x} \in X$  for which  $f(x^*, z^*) > f(\bar{x}, z^*)$ . Due to the continuity of  $f$ , we have, for some infinite subsequence index set  $\mathcal{K}$  such that  $\lim_{k \rightarrow \infty, k \in \mathcal{K}} (x^k, z^k) = (x^*, z^*)$ , the existence of  $\gamma > 0$  such that  $f(x^k, z^k) - f(\bar{x}, z^k) > \gamma > 0$ . Thus,  $f(x^k, z^k) > f(\bar{x}, z^k) + \gamma \geq f(x^{k+1}, z^k) + \gamma \geq f(x^{k+1}, z^{k+1}) + \gamma$ , which would imply that  $\lim_{k \rightarrow \infty} f(x^k, z^k) = -\infty$  since  $\mathcal{K}$  is an infinite index set and  $f(x^k, z^k)$  is monotonically non-increasing in the original sequence, so that  $f$  is unbounded from below, a contradiction. Therefore,  $(x^*, z^*)$  must be a partial minimum for problem (25). ■

**Remark 12.** For practical purposes, we might approximate the satisfaction of (26) and (27) through the  $\epsilon \geq 0$  parameterised termination criterion

$$f(x^k, z^k) - f(x^{k+1}, z^{k+1}) \leq \epsilon. \quad (28)$$

In the setting where  $f$  is convex and differentiable,  $X$  and  $Z$  are nonempty, closed and convex, and  $(x, z) \mapsto f(x, z)$  is inf-compact, it is well-known (see, for example, [7, 20, 33]) that the limit points  $(x^*, z^*)$  are optimal for problem (25). However, in the more general setting where  $f$  is non-differentiable and/or  $X$  and  $Z$  are non-convex, it is well-known that a partial minimum need not be a global, or even a local, minimum. In what follows, we provide a few small examples to illustrate this suboptimal stabilisation, and to motivate heuristic features of our developed algorithm that can mitigate this unfortunate tendency.

**Examples:**

1. Let problem (25) be specified so that  $f(x, z) : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$  is defined to be  $f(x, z) = 7x^2 + 10xz + 7z^2$ , and let  $X = Z = \{-2, -1, 0, 1, 2\}$ . For  $(x^0, z^0) = (2, -2)$ , the application of Algorithm 1 leads immediately to the one limit point  $(\bar{x}, \bar{z}) = (1, -1)$ . We have  $f(1, -1) = 4$ , but  $f(0, 0) = 0$ , so  $(\bar{x}, \bar{z}) = (1, -1)$  is not optimal. Note here that  $f$  is convex and continuously differentiable, but the constraint set  $X \times Z$  is nonconvex due to the integer restriction, and this is the reason that the limit point was not guaranteed to be optimal.
2. Let problem (25) be specified so that  $f(x, z) : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$  is defined to be  $f(x, z) = -2x - z + \rho|x - z|$ ,  $X = [-2, 3]$ , and  $Z = [0, 5]$ . For  $\rho \in [0, 1)$ , the optimal solution is  $(x^*, z^*) = (3, 5)$ . For  $\rho = 1$ , the optimal solutions are taken from  $(x^*, z^*) \in \{3\} \times [3, 5]$ , and for  $\rho > 1$ , the optimal solution is  $(x^*, z^*) = (3, 3)$ .
  - (a) When applying the GS approach of Algorithm 1 with  $\rho \in (0, 1)$ , the resulting sequence stabilises after one iteration at the optimum  $(x^*, z^*) = (3, 5)$  for any feasible starting point.
  - (b) For  $\rho = 1$  with  $z^0 \geq 3$ , we have after half an iteration  $(x^1, z^0) = (3, z^0)$  which is an optimum solution, and the remaining updates stay at some optimal solution  $(x^*, z^*) \in \{3\} \times [3, 5]$ . For  $\rho = 1$  with starting point  $z^0 < 3$ , we have  $x^1 = 3$  and  $z^1 \in [3, 5]$  and so stabilisation at an optimal solution also occurs.

- (c) For  $\rho > 1$  with  $z^0 \geq 3$ , we have  $(x^1, z^1) = (3, 3)$ , which is optimal. However, for  $\rho > 1$  with  $z^0 < 3$ , we have  $x^1 = z^0$  and  $z^1 = z^0$ , so that stabilisation occurs at  $(\bar{x}, \bar{z}) = (z^0, z^0)$ , which is not optimal.

3. Let problem (25) be specified so that  $f(x, z) : \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3 \mapsto \mathbb{R}$  is defined to be

$$f(x, z) = 2x_{1,1} - 1x_{1,2} - 2x_{1,3} - 2x_{2,1} - 1x_{2,2} + 2x_{2,3} + \rho \sum_{i=1,2} \sum_{j=1,2,3} |x_{i,j} - z_j|,$$

and let  $X$  and  $Z$  be defined so that

$$X = \left\{ (x)_{i,j} : \sum_{j=1}^3 x_{i,j} \leq 1 \text{ for } i = 1, 2; x_{i,j} \in \{0, 1\} \text{ for } i = 1, 2, j = 1, 2, 3 \right\},$$

and  $Z = \{0, 1\}^3$ . For  $\rho \rightarrow \infty$  (simulating the enforcement of constraints  $x_{i,j} = z_j$  for  $i = 1, 2$  and  $j = 1, 2, 3$ ) we have the optimal solution

$$(x^*, z^*) = ((0, 1, 0), (0, 1, 0)), (0, 1, 0).$$

If such constraints are altogether ignored ( $\rho = 0$ ), then the optimal  $x$ -component is  $x^* = ((0, 0, 1), (1, 0, 0))$ . This behaviour would only change at the threshold  $\rho = 1$ . For  $\rho > 1$ , the optimal solution would be  $(x^*, z^*) = ((0, 1, 0), (0, 1, 0)), (0, 1, 0)$ .

- (a) Now we consider what happens when the GS approach of Algorithm 1 is applied. Let  $z^0 = (0, 0, 0)$ . Starting with a small penalty such as  $\rho = 0.5$ , we have

$$x^1 = ((0, 0, 1), (1, 0, 0)) \quad \text{and} \quad z^1 \in \{(0, 0, 0), (0, 0, 1), (1, 0, 0), (1, 0, 1)\},$$

where there is more than one way to choose  $z^1$ . If, for example, we make it a policy to choose  $z$  by some bitwise lexicographical rule, then we choose  $z^1 = (0, 0, 0)$ . Keeping this same penalty  $\rho = 0.5$ , we find that stabilisation has occurred, where  $x^k = x^1$  and  $z^k = z^1$  for  $k \geq 1$ . If we increase the penalty value to  $\rho = 2$  for iteration  $k = 2$ , then we have the stabilisation  $x^1 = ((0, 0, 0), (0, 0, 0))$  and  $z^1 = (0, 0, 0)$ , which is suboptimal (and  $\rho = 2$  is the threshold for this change in stabilisation to occur).

If, instead, the  $z$  update is chosen by a reverse-lexicographic rule, so that  $z^1 = (1, 0, 1)$ , then we have immediate stabilisation with

$$(x^k, z^k) = ((0, 0, 1), (1, 0, 0), (1, 0, 1))$$

for all  $k \geq 1$  for all  $\rho > 0$ . (Notice that no matter how large the penalty is, consensus is not achieved in the GS setting. That is, without additional restriction on how  $z$  is updated, the optimal  $z$  update may be chosen to always correspond to a consensus solution that is infeasible for both scenarios. In practice, we would need a rule to insure that the  $z$  update is chosen to satisfy  $\sum_{j=1}^3 z_j \leq 1$  to match with the constraints in the  $x$  update subproblems.)

- (b) The shortcomings of the above GS approach motivate the introduction of more precision in how the consensus discrepancies are penalised, where  $f$  is redefined to be

$$f(x, z) = 2x_{1,1} - 1x_{1,2} - 2x_{1,3} - 2x_{2,1} - 1x_{2,2} + 2x_{2,3} + \sum_{i=1,2} \sum_{j=1,2,3} \rho_{i,j} |x_{i,j} - z_j|.$$

That is, instead of one scalar  $\rho$ , we have term-specific  $\rho_{i,j} > 0$  for each  $i = 1, 2$  and  $j = 1, 2, 3$ . We start as before with  $z^0 = (0, 0, 0)$ , and let  $\rho_{i,j} = 0.5$  for each  $i = 1, 2$  and  $j = 1, 2, 3$ . Assuming lexicographic rule in choosing  $z$ , we have as before

$$x^1 = ((0, 0, 1), (1, 0, 0)) \quad \text{and} \quad z^1 = (0, 0, 0),$$

and this is stable if the penalty does not change. Now increase  $\rho_{1,3} = \rho_{2,1} = 1$ , and we have

$$x^2 = ((0, 0, 1), (1, 0, 0)) \quad \text{and} \quad z^2 = (1, 0, 1),$$

and this is stable if the penalty does not change. Increasing  $\rho_{1,1} = \rho_{2,3} = 1$ , we have again

$$x^3 = ((0, 0, 1), (1, 0, 0)) \quad \text{and} \quad z^3 = (0, 0, 0),$$

and this is stable. But once we again increase  $\rho_{1,3} = \rho_{2,1} = 2$ , we have

$$x^4 = ((0, 1, 0), (0, 1, 0)) \quad \text{and} \quad z^4 = (0, 1, 0),$$

which is optimal for the original problem.

The last example suggests that there may be no fixed ideal penalty in a GS setting that will lead to both a closing of the duality gap and avoiding the nonoptimal stationarity

due to GS iterations. The penalty must vary in a manner that takes the component-wise consensus status into consideration. Any sensible heuristic for varying the penalties would have all penalties start small (but nonzero), and increase carefully, in a "fine-tuned" manner so as to "suggest" a temporary fixing of certain components of  $x$  to the current fixed values of the corresponding components of  $z$ . The strength of suggestion for each component is always relative to the other components as the magnitude of each component-wise penalty is relative to the magnitude of the other component-wise penalties.

An approach based on such an idea where some subset of variables is subject to "suggested" fixing with strength of suggestion determined by the penalties would be of a "soft" combinatorial nature. This is in contrast with a "hard" combinatorial approach that might be based on the idea of choosing some subset of integer variables at each iteration and simply fixing each one to some constant feasible value while conducting a minimisation over the unfixed variables. The algorithm to be presented later is of a soft combinatorial nature.

### 3.2. Adapting block Gauss-Seidel method to solve SMIPs using Penalty functions

In this section we present how block GS method can be used to obtain solutions for SMIP problems. The approach will rely on the delayed calculation of variable  $z$ , which will in turn allow us to obtain a decomposed version of the problem. To do such, let us first explicitly state  $\zeta_{\rho}^{LR+}(\omega)$  as

$$\begin{aligned} \zeta_{\rho}^{LR+}(\omega) : \min_{x,y,z} & \sum_{s \in S} p_s(c^{\top} x_s + q_s^{\top} y_s) + \omega_s^{\top} x_s \\ & + \sum_{s \in S} \underline{\rho}_s^{\top} [x_s - z]^{-} + \sum_{s \in S} \bar{\rho}_s^{\top} [z - x_s]^{-} \end{aligned} \quad (29)$$

$$\text{s.t.: } x_s \in X, \forall s \in S \quad (30)$$

$$y_s \in Y_s(x_s), \forall s \in S. \quad (31)$$

The following proposition will become useful in the following derivations.

**Proposition 13.** *For problem (29)-(31) with any  $\omega \in \Omega$ , there exists a finite  $\rho^*$  such that  $\zeta_{\rho^*}^{LR+}(\omega) = \zeta^{SIP}$ .*

**Proof.** The penalty terms in (29) result from the evaluation of  $\psi_\rho((x_s - z)_{s \in S})$  with  $\psi_\rho$  as defined in (16). Thus, the result follows immediately from Corollary 8. ■

Proposition 13 enables us to make the choice of  $\omega = 0$ . Notice that any other choice of  $\omega \in \Omega$  could also be used, without compromising the development of the algorithm. Setting  $\omega = 0$  leads to

$$\zeta_\rho^{LR+}(0) : \min_{x,y,z} \sum_{s \in S} p_s(c^\top x_s + q_s^\top y_s) + \sum_{s \in S} \rho_s^\top [x_s - z]^- + \sum_{s \in S} \bar{\rho}_s^\top [z - x_s]^- \quad (32)$$

$$\text{s.t.: } x_s \in X, \forall s \in S \quad (33)$$

$$y_s \in Y_s(x_s), \forall s \in S. \quad (34)$$

The block GS method for solving  $\zeta_\rho^{LR+}(0)$  proceeds as follows. Let

$$\phi^\rho(x, y, z, \rho) := \sum_{s \in S} \phi_s^\rho(x_s, y_s, z, \rho_s),$$

where

$$\phi_s^\rho(x_s, y_s, z, \rho_s) := p_s(c^\top x_s + q_s^\top y_s) + \rho_s^\top [x_s - z]^- + \bar{\rho}_s^\top [z - x_s]^-.$$

For a given  $\rho^k = (\rho_s^k, \bar{\rho}_s^k)_{s \in S}$  and an initial  $z^{0,0}$ , the proposed method will iterate between the solution of the following  $l = 0, 1, \dots, l_{\max}$  subproblems:

$$(x^{k,l+1}, y^{k,l+1})_{s \in S} \leftarrow \operatorname{argmin}_{x,y} \phi^\rho(x, y, z^{k,l}, \rho^k) \quad (35)$$

$$\text{s.t.: } x_s \in X, \forall s \in S \quad (36)$$

$$y_s \in Y_s(x_s), \forall s \in S, \quad (37)$$

and

$$z^{k,l+1} \leftarrow \operatorname{argmin}_z \phi^\rho(x^{k,l+1}, y^{k,l+1}, z, \rho^k), \quad (38)$$

followed by  $l = l + 1$  and successive repetition until *partial convergence* is approximately achieved in the sense of (28). In this context, *partial convergence* can be interpreted as having

$$\phi^\rho(x^{k,l}, y^{k,l}, z^{k,l}, \rho^k) - \phi^\rho(x^{k,l+1}, y^{k,l+1}, z^{k,l+1}, \rho^k) \leq \epsilon,$$

given a threshold  $\epsilon \geq 0$ .

At last, if the current primal infeasibility level, given by a residual measure such as  $\|x^{k,l} - z^{k,l}\|_2^2$ , is not acceptable for a  $\epsilon$  threshold, the set of penalties  $\rho^k$  are then updated to  $\rho^{k+1}$  and the process is repeated for iteration  $k + 1$ .

#### 4. Computational Implementation Aspects

Two remarkable features can be exploited in the design of an algorithm based on this idea. First, scenario-wise separability is straightforwardly achieved in the calculation of  $(x^{k,l+1}, y^{k,l+1})_{s \in S}$ . This means that instead of solving one large mixed-integer linear programming (MILP) problem in this update step, we can solve several small MILP problems instead, which is typically more efficient due to the exponential nature of the branch-and-cut-based methods used to solve them.

To formulate the  $(x^{k,l}, y^{k,l})_{s \in S}$ -update (35)-(37), it is necessary to explicitly represent the function  $[\cdot]^-$ . To do so, we consider an equivalent reformulation of the problem given by

$$\begin{aligned} \phi_s^\rho(x_s^{k,l+1}, y_s^{k,l+1}, z^{k,l}, \rho_s) = & \min_{x, y, \underline{w}, \overline{w}} p_s(c^\top x_s + q_s^\top y_s) + \underline{\rho}_s^k \underline{w}_s + \overline{\rho}_s^k \overline{w}_s \\ \text{s.t.: } & \underline{w}_s \geq 0, \quad \underline{w}_s \geq z^{k,l} - x_s \\ & \overline{w}_s \geq 0, \quad \overline{w}_s \geq x_s - z^{k,l} \\ & x_s \in X, \quad y_s \in Y_s(x_s). \end{aligned}$$

Second, the calculation of

$$z^{k,l+1} \in \operatorname{argmin}_z \phi^\rho(x^{k,l+1}, y^{k,l+1}, z, \rho^k) \quad (39)$$

may be performed by computing

$$z^{k,l+1} \in \operatorname{argmin}_z \zeta^\rho(x^{k,l+1}, z, \rho^k),$$

where the penalty function  $(x, z) \mapsto \zeta^\rho(x, z, \rho)$  is defined by

$$\zeta^\rho(x, z, \rho) := \psi_\rho((x_s - z)_{s \in S}) = \sum_{s \in S} \left( \underline{\rho}_s^\top [x_s - z]^- + \overline{\rho}_s^\top [z - x_s]^- \right).$$

The last displayed problem can be solved using the following equivalent mathematical programming formulation:

$$\zeta^\rho(x^{k,l+1}, z^{k,l+1}, \rho^k) = \min_{z, \underline{w}, \bar{w}} \sum_{s \in S} (\rho_s^k)^\top \underline{w}_s + (\bar{\rho}_s^k)^\top \bar{w}_s \quad (40)$$

$$\text{s.t.: } \underline{w}_s \geq 0, \forall s \in S, \quad \underline{w}_s \geq z - x_s^{k,l+1}, \forall s \in S \quad (41)$$

$$\bar{w}_s \geq 0, \forall s \in S, \quad \bar{w}_s \geq x_s^{k,l+1} - z, \forall s \in S. \quad (42)$$

When the  $x$  components are all restricted to take binary values, it is possible to show that the calculation of  $z^{k,l+1}$  can be performed in the following closed form where each component of  $z^{k,l+1}$  always takes binary value. In that case, its optimal solution is given by

$$z_i^{k,l+1} = \begin{cases} 1, & \text{if } \sum_{s \in S} (1 - x_{s,i}^{k,l+1}) \rho_s^k < \sum_{s \in S} x_{s,i}^{k,l+1} \bar{\rho}_s^k \\ 0, & \text{if } \sum_{s \in S} (1 - x_{s,i}^{k,l+1}) \rho_s^k > \sum_{s \in S} x_{s,i}^{k,l+1} \bar{\rho}_s^k \\ \text{either 0 or 1,} & \text{otherwise} \end{cases}, \quad i = 1, \dots, n_x. \quad (43)$$

The cases in which we have a tie might require "flipping a coin" for deciding on the value for  $z^{k,l+1}$ , as it becomes a case of multiple minima. The existence of multiple minima can be better understood from the following explicit form of the solution for the general case given in Proposition 14, from which (43) is a special case. In the following proposition, we assume  $Z$  is a closed convex set, so that no explicit integrality constraints are enforced.

**Proposition 14.** *Suppose a set of scenario dependent solutions  $(x_s)_{s \in S}$ , where  $x_s = (x_{s,i})_{i=1, \dots, n_x}$ , are given and  $z := (z_i)_{i=1, \dots, n_x}$ . For each  $i \in \{1, \dots, n_x\}$  define*

$$I^+(z_i) := \{s \in S \mid x_{s,i} > z_i\}$$

$$I^-(z_i) := \{s \in S \mid x_{s,i} < z_i\}$$

$$I^0(z_i) := \{s \in S \mid x_{s,i} = z_i\}$$

*Then  $z_i$  solves problem (39) given fixed  $(x_s)_{s \in S}$  if and only if*

$$\sum_{s \in I^+(z_i)} \bar{\rho}_{s,i} - \sum_{s \in I^-(z_i)} \rho_{s,i} \in \left[ - \sum_{s \in I^0(z_i)} \bar{\rho}_{s,i}, \sum_{s \in I^0(z_i)} \rho_{s,i} \right]. \quad (44)$$



**Proof.** The index  $s$  term of the penalty function  $\zeta^\rho$  may be written as

$$\begin{aligned} & \zeta_s^\rho((x_s)_{s \in S}, z, \rho) \\ &:= \sum_{i=1}^{n_x} \left[ \sum_{s \in I^+(z_i)} \bar{\rho}_{s,i} \max\{0, x_{s,i} - z_i\} + \sum_{s \in I^-(z_i)} \underline{\rho}_{s,i} \max\{0, z_i - x_{s,i}\} \right]. \end{aligned}$$

As this is separable in the variables  $(z_1, \dots, z_{n_x})$ , its subdifferential is defined as the cross product of intervals, one for each component  $i$ . Thus, the necessary and sufficient condition

$$0 \in \partial \zeta_s^\rho((x_s)_{s \in S}, z, \rho),$$

can be equivalently stated as

$$0 \in \partial_{z_i} \zeta_s^\rho((x_s)_{s \in S}, z_i, \rho),$$

for each  $i = 1, \dots, n_x$ , which is given by:

$$\begin{aligned} 0 &\in \sum_{s \in I^-(z_i)} \underline{\rho}_{s,i} - \sum_{s \in I^+(z_i)} \bar{\rho}_{s,i} + \sum_{s \in I^0(z_i)} [-\bar{\rho}_{s,i}, \underline{\rho}_{s,i}] \\ &= \sum_{s \in I^-(z_i)} \underline{\rho}_{s,i} - \sum_{s \in I^+(z_i)} \bar{\rho}_{s,i} + \left[ -\sum_{s \in I^0(z_i)} \bar{\rho}_{s,i}, \sum_{s \in I^0(z_i)} \underline{\rho}_{s,i} \right], \end{aligned}$$

which in turn is equivalent to (44). ■

We now consider how to update the penalty parameters  $\rho^k$ . A simple strategy is

$$\begin{aligned} \underline{\rho}_s^{k+1} &= \underline{\rho}_s^k + \gamma [x_s^{k,l} - z_i^{k,l}]^- \\ \bar{\rho}_s^{k+1} &= \bar{\rho}_s^k + \gamma [z_i^{k,l} - x_s^{k,l}]^-. \end{aligned}$$

By doing so, we are reinforcing the penalties associated with the respective discrepancies.

Recalling that  $[v]^- := -\min\{0, v\}$ , we have namely that for each  $i = 1, \dots, n_x$ :

$$\begin{aligned} \underline{\rho}_{s,i}^{k+1} &= \begin{cases} \underline{\rho}_{s,i}^k + \gamma (z_i^{k,l} - x_{s,i}^{k,l}), & \text{if } x_{s,i}^{k,l} < z_i^{k,l} \\ \underline{\rho}_{s,i}^k, & \text{if } x_{s,i}^{k,l} \geq z_i^{k,l} \end{cases} \\ \bar{\rho}_{s,i}^{k+1} &= \begin{cases} \bar{\rho}_{s,i}^k + \gamma (x_{s,i}^{k,l} - z_i^{k,l}), & \text{if } z_i^{k,l} < x_{s,i}^{k,l} \\ \bar{\rho}_{s,i}^k, & \text{if } z_i^{k,l} \geq x_{s,i}^{k,l} \end{cases} \end{aligned}$$

**Remark 15.** The update in  $\rho^{k+1}$  has the effect of changing the left hand side of (44) at the next iteration by the amount:

$$\Delta_i^{k+1} := \gamma \left[ \sum_{s \in I^+(z_i^k)} [z_i^k - x_{s,i}^k]^- - \sum_{s \in I^-(z_i^k)} [x_{s,i}^k - z_i^k]^- \right], \quad (45)$$

for each  $i = 1, \dots, n_x$ . If the addition of this factor ensures the sum in left hand side of (44) at iteration  $k+1$  exits the interval  $\left[ -\sum_{s \in I^0(z_i)} \bar{\rho}_{s,i}, \sum_{s \in I^0(z_i)} \underline{\rho}_{s,i} \right]$  associated with the prior choice of  $z_i^k = x_{s,i}^k$  then we would be forced to choose new consensus values  $z_i^k$  in order to re-establish the satisfaction of the optimality condition (44). In doing so, a reassignment of the index sets  $I^+(z_i^k)$ ,  $I^-(z_i^k)$ , and  $I^0(z_i^k)$  is effected. As intuition would suggest, the optimality condition (44) is more easily satisfied when  $s \in I^0(z_i^k)$  for large  $\underline{\rho}_{s,i}$  and  $\bar{\rho}_{s,i}$ , as this makes the target interval  $\left[ -\sum_{s \in I^0(z_i)} \bar{\rho}_{s,i}, \sum_{s \in I^0(z_i)} \underline{\rho}_{s,i} \right]$  larger.

To effect a gradual increase in the terms  $\Delta^k$  in an attempt to improve convergence with the satisfaction of the NAC, we considered an increasing multiplier factor to  $\psi_\rho$  given by  $\beta^{(k-1)} - 1$  (where  $\beta \in (1, 2]$  for practical purposes and  $(k-1)$  represents an exponent and not an iteration index). In other words, we consider the objective at a given iteration  $k$  as being

$$\phi^{\rho,k}(x_s, y_s, z, \omega) := \sum_{s \in S} p_s (c^\top x_s + q_s^\top y_s) + (\beta^{(k-1)} - 1) \left[ \sum_{s \in S} \underline{\rho}_s^\top [x_s - z]^- + \sum_{s \in S} \bar{\rho}_s^\top [z - x_s]^- \right].$$

Combining what have been exposed so far, one first algorithmic approach consists of the following setting presented in Algorithm 2.

**Algorithm 2** Penalty-based block Gauss-Siedel (PGBS) method for SMIP

---

```

1: initialise  $\rho^0 = (\underline{\rho}^0, \bar{\rho}^0), \hat{z}^0, \epsilon, \gamma, \beta, l_{\max}, k_{\max}$ 
2: for  $s \in S$  do
3:    $\hat{x}_s^0 \leftarrow \operatorname{argmin}_{x,y} \{ \phi^{\rho^0,1}(x_s, y_s, \hat{z}^0, \rho^0) : x_s \in X, y_s \in Y_s(x_s) \}$ 
4: end for
5: for  $k = 1, \dots, k_{\max}$  do
6:    $x^{k,0} \leftarrow \hat{x}^{k-1}$ 
7:    $z^{k,0} \leftarrow \hat{z}^{k-1}$ 
8:   for  $l = 1, \dots, l_{\max}$  do
9:     for  $s \in S$  do
10:       $(x_s^{k,l}, y_s^{k,l}) \leftarrow \operatorname{argmin}_{x,y} \{ \phi^{\rho,k}(x_s, y_s, z^{k,l-1}, \rho^k) : x_s \in X, y_s \in Y_s(x_s) \}$ 
11:    end for
12:     $z^{k,l} \leftarrow \operatorname{argmin}_z \phi^{\rho,k}(x^{k,l}, y^{k,l}, z, \rho^k)$ 
13:     $\Gamma \leftarrow \phi^{\rho,k}(x^{k,l-1}, y^{k,l-1}, z^{k,l-1}, \rho^k) - \phi^{\rho,k}(x^{k,l}, y^{k,l}, z^{k,l}, \rho^k)$ 
14:    if  $\Gamma \leq \epsilon$  or  $l = l_{\max}$  then
15:       $(\hat{x}_s^k, \hat{y}_s^k) \leftarrow (x_s^{k,l}, y_s^{k,l})$  for all  $s \in S$ 
16:       $\hat{z}^k \leftarrow z^{k,l}$ 
17:      break
18:    end if
19:     $l \leftarrow l + 1$ 
20:  end for
21:  if  $\|\hat{x}^k - \hat{z}^k\|_2^2 \leq \epsilon$  or  $k = k_{\max}$  then
22:    return  $((\hat{x}_s^k, \hat{y}_s^k)_{s \in S}, \hat{z}^k)$ 
23:  else
24:     $\underline{\rho}_s^k = \underline{\rho}_s^{k-1} + \gamma[\hat{x}_s^k - \hat{z}^k]^-$  for all  $s \in S$ 
25:     $\bar{\rho}_s^k = \bar{\rho}_s^{k-1} + \gamma[\hat{z}^k - \hat{x}_s^k]^-$  for all  $s \in S$ 
26:  end if
27:   $k \leftarrow k + 1$ 
28: end for

```

---

## 5. Experimental setting

In this section we describe the computational experiments performed to assess the performance of the proposed approach. To evaluate the performance of the proposed method, we tested its efficacy on three distinct classes of problems from literature, namely the capacitated facility location problems (CAP) from [8], the dynamic capacity allocation problems (DCAP) available in [2], and the server location under uncertainty problems (SSLP) first introduced in [28]. To provide a more solid base of comparison, 50 random instances of two problems from each class were generated.

The CAP problems are two-stage SMIP problems with pure binary first- and second-stage variables arising in the context of network design problems. We selected the instances coded as 101 and 111 in [8], considering random samples of 100 scenarios from a list of 5000 scenarios available.

The DCAP problems are two-stage SMIP problems arising in dynamic capacity acquisition and allocation under uncertainty. All problem instances have mixed-integer first-stage variables and pure binary second-stage variables. We selected the instances coded as 233 and 342 (which encodes the number of resources, tasks, and periods, respectively), considering random samples of 100 scenarios from the original 500 available.

The SSLP problems are two-stage SMIP problems arising in server location under uncertainty. The problems have pure binary first-stage variables and mixed-binary second-stage variables. We considered the instances coded as 5-50 and 10-50 (which encode the number of servers and the number of clients, respectively) with 100 scenarios that were randomly generated according to the guidelines provided in [28].

To compare and benchmark the performance of the proposed approach against a known quantity we have implemented the Progressive Hedging (PH) algorithm, which was originally proposed by [31] and, as previously discussed, has been widely used as an heuristic approach to solve SMIP problems. The PH algorithm is stated in Algorithm 3 for the sake of completeness. In this algorithm,

$$L_{\rho}^s(x_s, y_s, z, \omega_s) := (c + \omega_s)^{\top} x_s + q_s^{\top} y_s + \frac{\rho}{2} \|x_s - z\|_2^2,$$

which means that Line 8 comprises the solution of  $|S|$  mixed-integer quadratic subproblems at each iteration  $k$ . The analogous step in PGBS algorithm requires us to solve

typically less difficult MIP problems instead.

Another advantage of PGBS in the context of SMIP problems is that  $z^{k,l+1}$  tends to (in most cases) satisfy the integrality constraints of the problem. This is in contrast with the consensus value computed with the averaging of PH (Line 10 in Algorithm 3), which tends to steer the consensus value away from integral values. In the case of binary variables, the PH averaging computation of the consensus  $z^{k,l+1}$  is especially prone to producing many fractional valued components which can lead to episodic cycling in binary values set in the assignment of scenario specific variables.

One important difference between PGBS and PH is that in PGBS we do not update the dual variables  $\omega$ , as it is performed in PH (Line 14 in Algorithm 3). In principle, one could do such in PGBS as well. However, preliminary experiments with the subgradient method to update the dual solutions  $\omega$  did not provide significant improvement in the PGBS performance. We believe that this observation is related to the asymmetrical way we update the penalty terms which, in effect, subsumes the role usually taken by the linear multiplier term. Consequently, in Algorithm 2 we use  $\omega = 0$  and do not update the dual solutions.

**Algorithm 3** Progressive Hedging for SMIP

---

```

1: initialise  $\rho, (\omega_s^0)_{s \in S}, \epsilon, k_{\max}$ 
2: for  $s \in S$  do
3:    $x_s^0 \leftarrow \operatorname{argmin}_{x,y} \{c^\top x_s + q_s^\top y_s : x_s \in X, y_s \in Y_s(x_s)\}$ 
4: end for
5:  $z^0 \leftarrow \sum_s p_s x_s^0$ 
6: for  $k = 1, \dots, k_{\max}$  do
7:   for  $s \in S$  do
8:      $(x_s^k, y_s^k) \leftarrow \operatorname{argmin}_{x,y} \{L_\rho^s(x_s, y_s, z^k, \omega^{k-1}) : x_s \in X, y_s \in Y_s(x_s)\}$ 
9:   end for
10:   $z^k \leftarrow \sum_s p_s x_s^k$ 
11:  if  $\|x^k - z^{k-1}\|_2^2 \leq \epsilon$  or  $k = k_{\max}$  then
12:    return  $((x_s^k, y_s^k)_{s \in S}, z^k)$ 
13:  else
14:     $\omega_s^k \leftarrow \omega_s^{k-1} + \rho(x_s^k - z^k), \forall s \in S$ 
15:  end if
16:   $k \leftarrow k + 1$ 
17: end for

```

---

In the PGBS experiments, the parameters were chosen from  $\beta \in \{1.11, 1.25\}$ , and  $\gamma \in \{0.5\rho^0, \rho^0\}$ . Three different initial values for  $\rho^0$  were used in both the PGBS and PH experiments. In the Progressive Hedging algorithm, dual multipliers were initialised as 0 and the penalty parameter ( $\rho$ ) was set to  $\rho = \rho^0$ . The parameter  $z^{0,0}$  has been initialised according to the solution  $x_s^0$  (from Line 3 in Algorithm 2 and Line 3 in Algorithm 3). For Algorithm 2, we initialised  $z^{0,0}$  as being

$$z_i^{0,0} = \left\lceil \sum_{s \in S} p_s x_{i,s}^0 \right\rceil, \forall i = 1 \dots, n_x \quad (46)$$

for all components restricted to be integer variables, where  $\lceil \cdot \rceil$  denotes rounding operation. For the components without integrality restrictions, we have dropped the rounding operator. In case of PH (Algorithm 3), all components were calculated by dropping the rounding operator.

As both CAP and SSLP problems have pure binary first-stage variables, we have used (43) to perform the step depicted in Line 12 of Algorithm 2. For DCAP, we relied on solving (40)-(42) explicitly.

A time limit of 1000 seconds and termination condition of  $\epsilon = 10^{-3}$  was used for both methods. A total of 300 ( $3 \times 2 \times 50$ ) instances were solved with three parameters choices for PH (different choices of  $\rho^0$ ) and 12 combinations of parameter choices for PBGS (different choices of  $\rho^0$ ,  $\beta$ , and  $\gamma$ ). Parameters  $\rho^0$  were set based on early experiments (in case of CAP) and considering values known to be (near) optimal from other studies that have applied PH to instances of these problems (in case of DCAP and SSLP, e.g., [21] and [17]). All methods have been implemented using AIMMS 4.14 and subproblems have been solved using CPLEX 12.6.2 with its standard configurations. All instances generated, as well as numerical results obtained, can be obtained upon request to the authors.

### 5.1. Numerical results

A summary of the computational results is presented in Figures 1 to 3, which depicts the average computational time and objective value difference for the 50 instances considered for both PH and PBGS in all parameter settings that have been tested.

The blue bars indicate the average wall clock times for both methods. We highlight that the instances in which PH terminated due to the time limit of 1000s have been removed from the average calculations, these being treated as outliers. The green line shows the average objective value relative difference, which is calculated as

$$\frac{1}{N} \sum_{i=1}^N \frac{z_{PBGS}^i - z_{PH}^i}{z_{PH}^i},$$

where  $z_{PBGS}^i$  and  $z_{PH}^i$  are the objective function values obtained for the solutions returned by PBGS and PH for instance  $i$ , respectively, and  $N$  is the total number of instances considered for average value calculations. To obtain  $z_{PBGS}^i$  and  $z_{PH}^i$ , we used the last solution returned by both methods and evaluated it a posteriori. For the cases in which PH returned solutions that were infeasible in regard to integrality restrictions (typically those obtained when the algorithm stopped due to the time criterion), rounding has been performed to recover a feasible solution to be evaluated when applicable.

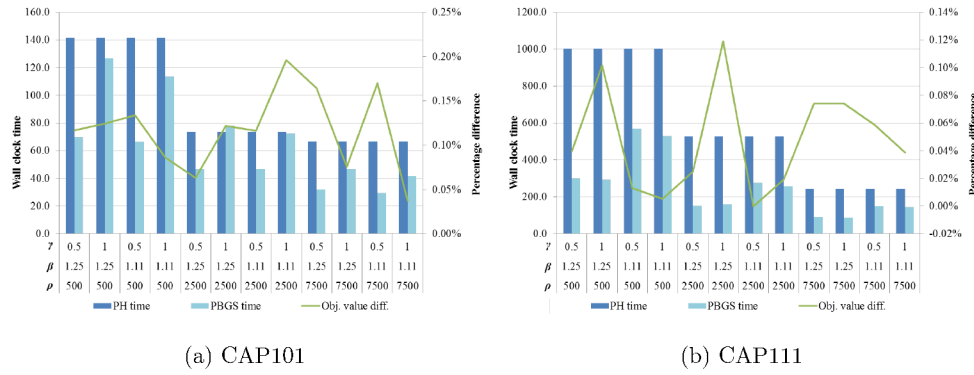


Figure 1: Results for CAP Problems

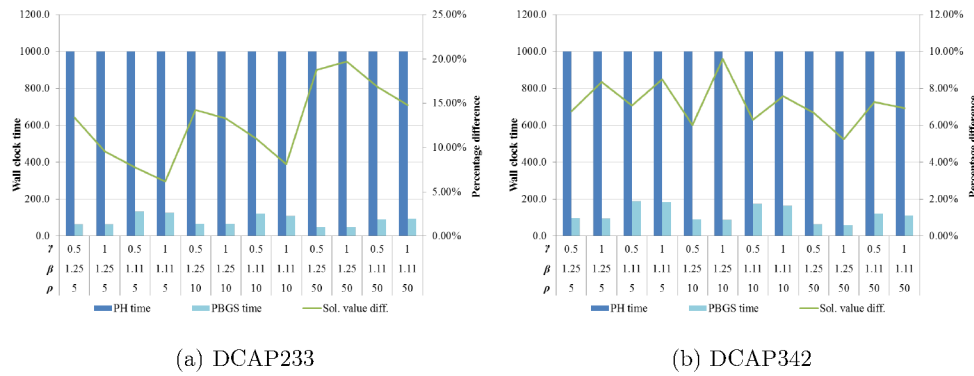


Figure 2: Results for DCAP Problems

For the CAP instances, all configurations tested with PBGS and PH presented similar values for the objective function, and in most configurations PBGS presented better performance in terms of computational time. For the DCAP instances, in all cases PH terminated due to the time limit of 1000 seconds. For these problems, a comparison in terms of objective function shows that the differences between the objective function value of the solutions found by PBGS and PH are more pronounced. A similar behaviour can be observed in the SSLP instances, in which PBGS outperforms PH in terms of solution times in most cases while providing solutions that are, in the worst case, 0.5% worse for SSLP5-50 and 5% worse for SSLP10-50. In the Appendix we present a detailed summary



of the statistics for each of the problems, including the fraction of the runs in which PH was not able to converge within the specified time limit. Overall, PBGS seems to be able to obtain comparably good solutions however presenting more reliable convergence behaviour.

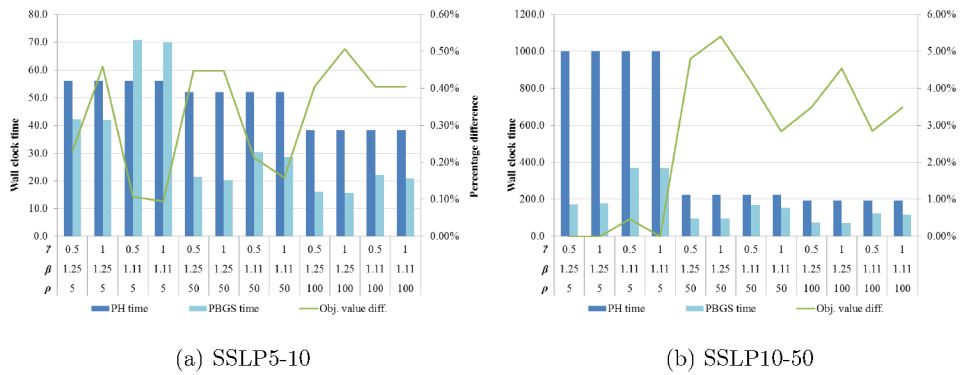


Figure 3: Results for SSLP Problems

## 6. Conclusions

In this paper we have presented an alternative approach for solving stochastic mixed-integer problems based on the combination of penalty-based and block Gauss-Seidel methods. The motivation of such arises from recent theoretical results that motivates the consideration of Lagrangian-based methods under alternative perspectives to approach such problems.

The computational experiments performed suggest that there is potential for exploiting this framework as it allows the development of a competitive approach in terms of computational efficiency. It is worth highlighting that the methodology developed is readily amenable to parallelisation, which is a key point for dealing with large-scale SMIPs. Moreover, as it is the case for PH, the methodology can be straightforwardly adapted to multi-stage problems. It can also potentially benefit from many modern enhancement for PH available in literature, such as improved techniques to set values for  $\rho$  [35]) and scenario aggregation techniques [13, 17].

Further developments of this research could be classified under two distinct stand-

points. Under a theoretical perspective, suitable alternative extensions of the block Gauss-Seidel approach into non-smooth non-separable problems are worth investigation. A better understanding of how to fine-tune the updates of the penalty coefficients would improve the likelihood (or perhaps even guarantee!) that the block Gauss-Seidel iterations do not display suboptimal stationarity. This would improve the trend of the objective values computed by the main algorithm. In terms of practical considerations, it would be of interest to evaluate the performance of the proposed approach with alternative versions of PH that rely on heuristically improved ways of setting the penalty parameter  $\rho$  to provide a more broad comparison of performance between the two algorithms. It would be also of interest to evaluate the performance of the proposed approach in contexts other than SMIPs, its extension to the multi-stage case, and compare its performance with other algorithms that are tailored to problems of similar structure, such as presented in [1], for example.

### Acknowledgements

The authors would like to acknowledge the support provided by the Australian Research Council (ARC) grant ARC DP140100985.

### References

- [1] S. AHMED, *A scenario decomposition algorithm for 0–1 stochastic programs*, Operations Research Letters, 41 (2013), pp. 565–569.
- [2] S. AHMED AND R. GARCIA, *Dynamic capacity acquisition and assignment under uncertainty*, Annals of Operations Research, 124 (2003), pp. 267–283.
- [3] G. ANGULO, S. AHMED, AND S. S. DEY, *Improving the integer L-shaped method*, INFORMS Journal on Computing, 28 (2016), pp. 483–490.
- [4] C. BELTRAN, C. TADONKI, AND J.-P. VIAL, *Solving the p-median problem with a semi-Lagrangian relaxation*, Comput. Optim. Appl., 35 (2006), pp. 239–260.
- [5] C. BELTRAN-ROYO, J.-P. VIAL, AND A. ALONSO-AYUSO, *Semi-Lagrangian relaxation applied to the uncapacitated facility location problem*, Comput. Optim. Appl., 51 (2012), pp. 387–409.
- [6] D. BERTSEKAS, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, 1982.
- [7] ———, *Nonlinear Programming*, Athena Scientific, 1999.
- [8] M. BODUR, S. DASH, O. GÜNLÜK, AND J. LUEDTKE, *Strengthened benders cuts for stochastic integer programs with continuous recourse*, tech. rep., Technical Report. Optimization Online 2014-03-4263, 2014.

[9] N. L. BOLAND AND A. C. EBERHARD, *On the augmented Lagrangian dual for integer programming*, Math. Program., 150 (2015), pp. 491–509.

[10] R. S. BURACHIK, A. N. IUSEM, AND J. G. MELO, *The exact penalty map for nonsmooth and nonconvex optimization*, Optimization, 64 (2015), pp. 717–738.

[11] C. C. CARØE AND R. SCHULTZ, *Dual decomposition in stochastic integer programming*, Oper. Res. Lett., 24 (1999), pp. 37–45.

[12] Y. CHEN AND M. CHEN, *Extended duality for nonlinear programming*, Computational Optimization and Applications, 47 (2010), pp. 33–59.

[13] T. G. CRAINIC, M. HEWITT, AND W. REI, *Scenario grouping in a progressive hedging-based meta-heuristic for stochastic network design*, Computers & Operations Research, 43 (2014), pp. 90–99.

[14] C. DAVIS, *Theory of positive linear dependence*, American Journal of Mathematics, 76 (1954), pp. 733–746.

[15] M. J. FEIZOLLAHI, S. AHMED, AND A. SUN, *Exact augmented lagrangian duality for mixed integer linear programming*, Mathematical Programming, 161 (2017), pp. 365–387.

[16] M. J. FEIZOLLAHI, M. COSTLEY, S. AHMED, AND S. GRIJALVA, *Large-scale decentralized unit commitment*, International Journal of Electrical Power & Energy Systems, 73 (2015), pp. 97–106.

[17] D. GADE, G. HACKEBEIL, S. M. RYAN, J.-P. WATSON, R. J.-B. WETS, AND D. L. WOODRUFF, *Obtaining lower bounds from the progressive hedging algorithm for stochastic mixed-integer programs*, Mathematical Programming, 157 (2016), pp. 47–67.

[18] B. GEISSLER, A. MORSI, L. SCHEWE, AND M. SCHMIDT, *Solving power-constrained gas transportation problems using an MIP-based alternating direction method*, Computers & Chemical Engineering, 82 (2015), pp. 303–317.

[19] J. GORSKI, F. PFEUFFER, AND K. KLAMROTH, *Biconvex sets and optimization with biconvex functions: a survey and extensions*, Mathematical Methods of Operations Research, 66 (2007), pp. 373–407.

[20] L. GRIPPO AND M. SCIANDRONE, *On the convergence of the block nonlinear Gauss-Seidel method under convex constraints*, Operations Research Letters, 26 (2000), pp. 127–136.

[21] G. GUO, G. HACKEBEIL, S. M. RYAN, J.-P. WATSON, AND D. L. WOODRUFF, *Integration of progressive hedging and dual decomposition in stochastic integer programs*, Oper. Res. Lett., 43 (2015), pp. 311–316.

[22] K. JÖRNSTEN AND A. KLOSE, *An improved Lagrangian relaxation and dual ascent approach to facility location problems*, Computational Management Science, (2015). Article in Press.

[23] C. LALITHA, *A new augmented lagrangian approach to duality and exact penalization*, Journal of Global Optimization, 46 (2010), pp. 233–245.

[24] G. LAPORTE AND F. V. LOUVEAUX, *The integer L-shaped method for stochastic integer programs with complete recourse*, Oper. Res. Lett., 13 (1993), pp. 133–142.

[25] A. LØKKETANGEN AND D. L. WOODRUFF, *Progressive hedging and tabu search applied to mixed integer (0,1) multistage stochastic programming*, Journal of Heuristics, 2 (1996), pp. 111–128.

[26] M. LUBIN, K. MARTIN, C. G. PETRA, AND B. SANDIKÇI, *On parallelizing dual decomposition in*

- stochastic integer programming, *Oper. Res. Lett.*, 41 (2013), pp. 252–258.
- [27] E. MONABBATI, *An application of a Lagrangian-type relaxation for the uncapacitated facility location problem*, *Jpn. J. Ind. Appl. Math.*, 31 (2014), pp. 483–499.
- [28] L. NTAIMO AND S. SEN, *The million-variable march for stochastic combinatorial optimization*, *Journal of Global Optimization*, 32 (2005), pp. 385–400.
- [29] G. PERBOLI, L. GOBBATO, AND F. MAGGIONI, *A progressive hedging method for the multi-path travelling salesman problem with stochastic travel times*, *IMA Journal of Management Mathematics*, 28 (2017), pp. 65–86.
- [30] R. ROCKAFELLAR, *Monotone operators and the proximal point algorithm*, *SIAM Journal on Control and Optimization*, 14 (1976), pp. 877–898.
- [31] R. T. ROCKAFELLAR AND R. J.-B. WETS, *Scenarios and policy aggregation in optimization under uncertainty*, *Math. Oper. Res.*, 16 (1991), pp. 119–147.
- [32] S. M. RYAN, R. J.-B. WETS, D. L. WOODRUFF, C. SILVA-MONROY, AND J.-P. WATSON, *Toward scalable, parallel progressive hedging for stochastic unit commitment*, in 2013 IEEE Power & Energy Society General Meeting, IEEE, 2013, pp. 1–5.
- [33] P. TSENG, *Convergence of a block coordinate descent method for nondifferentiable minimization*, *Journal of Optimization Theory and Applications*, 109 (2001), pp. 475–494.
- [34] F. B. VELIZ, J.-P. WATSON, A. WEINTRAUB, R. J.-B. WETS, AND D. L. WOODRUFF, *Stochastic optimization models in forest planning: A progressive hedging solution approach*, *Annals of Operations Research*, 232 (2015), pp. 259–274.
- [35] J.-P. WATSON AND D. L. WOODRUFF, *Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems*, *Computational Management Science*, 8 (2011), pp. 355–370.
- [36] R. E. WENDELL AND A. P. HURTER, *Minimization of a non-separable objective function subject to disjoint constraints*, *Operations Research*, 24 (1976), pp. 643–657.
- [37] X. Q. YANG AND X. X. HUANG, *A nonlinear Lagrangian approach to constrained optimization problems*, *SIAM Journal on Optimization*, 11 (2001), pp. 1119–1144.

Appendix A. Additional computational results

In this Appendix, we present a detailed summary of the computational results obtained. In Tables A.1 to A.6, row “Obj. diff.” presents the average value (“Average”) and standard deviation (“St. dev.”) for the relative difference of the objective function value for the solutions obtained with PBGS and PH (with feasibility restored by rounding whenever PH terminated due to the time limit of 1000s). Row “Speed-up” calculates the relative speed-up that PBGS presented in comparison to PH in terms of wall clock time (values greater than 1 mean that PGBS was faster). Finally, row “PG conv. fraction” displays the fraction of instances in which PH converged before reaching the specified time limit.

		$\rho$	500				2500				7500			
		$\beta$	1.25		1.11		1.25		1.11		1.25		1.11	
		$\gamma$	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1
Obj. diff.	Average		0.12%	0.12%	0.13%	0.09%	0.06%	0.12%	0.12%	0.20%	0.16%	0.08%	0.17%	0.04%
	St. dev.		0.13%	0.15%	0.18%	0.14%	0.10%	0.14%	0.14%	0.20%	0.14%	0.10%	0.14%	0.07%
Speed-up	Average		2.02	1.12	2.14	1.24	1.63	0.93	1.57	1.01	2.12	1.44	2.28	1.60
	St. dev.		0.47	0.29	0.62	0.31	2.31	1.04	1.91	1.12	0.82	0.55	0.86	0.53
PH conv. Fraction.			96.0%				92.0%				94.0%			

Table A.1: CAP101

37

		500				2500				7500			
		1.25		1.11		1.25		1.11		1.25		1.11	
		$\rho$	$\beta$	$\gamma$		$\rho$	$\beta$	$\gamma$		$\rho$	$\beta$	$\gamma$	
		0.5	1	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1
Obj. diff.	Average	0.04%	0.10%	0.01%	0.01%	0.02%	0.12%	0.00%	0.02%	0.07%	0.07%	0.06%	0.04%
	St. dev.	0.06%	0.11%	0.03%	0.01%	0.16%	0.31%	0.17%	0.16%	0.05%	0.05%	0.06%	0.05%
Speed-up	Average	3.95	3.67	2.06	1.87	2.98	2.93	1.60	1.73	1.97	2.06	1.19	1.25
	St. dev.	0.27	0.01	0.11	0.37	1.28	1.31	0.51	0.61	0.76	0.81	0.48	0.51
PH conv. Fraction.		4.0%				86.0%				92.0%			

Table A.2: CAP111

		$\rho$	5				10				50			
		$\beta$	1.25	1.11		1.25		1.11		1.25	1.11			
		$\gamma$	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1
Obj. diff.	Average	13.41%	9.56%	7.78%	6.16%	14.22%	13.28%	11.04%	8.10%	18.75%	19.73%	16.86%	14.80%	
	St. dev.	2.18%	2.44%	2.26%	2.53%	2.35%	2.88%	2.78%	2.65%	2.06%	2.33%	2.21%	2.57%	
Speed-up	Average	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	
	St. dev.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	
PH conv.	Fraction.	0.0%				0.0%				0.0%				

Table A.3: DCAP233

		$\rho$	5				10				50			
		$\beta$	1.25		1.11		1.25		1.11		1.25		1.11	
		$\gamma$	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1
Obj. diff.	Average	6.77%	8.36%	7.08%	8.51%	6.01%	9.61%	6.31%	7.59%	6.69%	5.24%	7.27%	6.92%	
	St. dev.	4.78%	4.45%	4.47%	4.39%	3.26%	5.36%	5.07%	3.70%	3.82%	2.59%	2.90%	3.67%	
Speed-up	Average	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	
	St. dev.	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	
PH conv. Fraction.		0.0%				0.0%				0.0%				

Table A.4: DCAP342



		$\rho$	5				50				100			
		$\beta$	1.25		1.11		1.25		1.11		1.25		1.11	
		$\gamma$	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1
Obj. diff.	Average	0.23%	0.46%	0.11%	0.10%	0.45%	0.45%	0.21%	0.16%	0.40%	0.51%	0.40%	0.40%	
	St. dev.	0.76%	1.14%	0.49%	0.32%	1.08%	1.08%	0.64%	0.53%	1.07%	1.25%	1.07%	1.07%	
Speed-up	Average	1.29	1.29	0.76	0.81	1.32	1.40	0.93	1.03	1.12	1.21	0.82	0.86	
	St. dev.	0.60	0.56	0.30	0.31	0.91	0.96	0.61	0.67	0.68	0.82	0.58	0.49	
PH conv. Fraction.			100.0%				98.0%				98.0%			

Table A.5: SSLP5-50

40

		$\rho$	5				50				100			
		$\beta$	1.25		1.11		1.25		1.11		1.25		1.11	
		$\gamma$	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1	0.5	1
Obj. diff.	Average	0.5	0.00%	0.00%	0.47%	0.00%	4.80%	5.41%	4.19%	2.84%	3.50%	4.54%	2.85%	3.50%
	St. dev.	0.0%	0.0%	0.5%	0.0%	12.5%	12.9%	12.0%	10.5%	8.8%	9.9%	8.0%	8.8%	
Speed-up	Average	7.95	7.86	2.95	4.07	2.27	2.35	1.35	1.41	1.69	1.72	1.03	1.10	
	St. dev.	2.14	2.08	0.36	1.13	1.11	1.29	0.78	0.74	1.04	1.05	0.69	0.75	
PH conv. Fraction.			4.0%				96.0%				90.0%			

Table A.6: SSLP10-50