Improving Risk Management

RIESGOS-CM

Análisis, Gestión y Aplicaciones

P2009/ESP-1685

Technical Report 2010.08

Lagrangean decomposition for large-scale two-stage stochastic mixed 0-1 problems

L.F. Escudero, M.A. Garín, G. Pérez and A. Unzueta

http://www.analisisderiesgos.org

Lagrangean decomposition for large-scale two-stage stochastic mixed 0-1 problems [1]

L.F. Escudero[1], M.A. Garín[2], G. Pérez[3] and A. Unzueta[4]

[1]Dpto. Estadística e Investigación Operativa
Universidad Rey Juan Carlos, Móstoles (Madrid), Spain
e-mail: laureano.escudero@urjc.es

[2]Dpto. de Economía Aplicada III
Universidad del País Vasco, Bilbao (Vizcaya), Spain
e-mail: mariaaraceli.garin@ehu.es

[3]Dpto. de Matemática Aplicada, Estadística e Investigación Operativa
Universidad del País Vasco, Leioa (Vizcaya), Spain
e-mail: gloria.perez@ehu.es

[4]Dpto. de Economía Aplicada III
Universidad del País Vasco, Bilbao (Vizcaya), Spain
e-mail: aitziber.unzueta@ehu.es

September 7, 2010

Abstract

In this paper we study solution methods for solving the dual problem corresponding to the Lagrangean Decomposition of two stage stochastic mixed 0-1 models. We represent the two stage stochastic mixed 0-1 problem by a splitting variable representation of the deterministic equivalent model, where 0-1 and continuous variables appear at any stage. Lagrangean Decomposition is proposed for satisfying both the integrality constraints for the 0-1 variables and the non-anticipativity constraints. We compare the performance of four iterative algorithms based on dual Lagrangean Decomposition schemes, as the Subgradient method, the Volume algorithm, the Progressive Hedging algorithm and the Dynamic Constrained Cutting Plane scheme. We test the conditions and properties of convergence for medium and large-scale dimension stochastic problems. Computational results are reported.

Keywords: Two-stage stochastic integer programming, Lagrangean decomposition, Subgradient method, Volume algorithm, Progressive Hedging algorithm and Dynamic Constrained Cutting plane scheme.

1 Introduction

We consider a general situation for two-stage stochastic mixed 0-1 problems, where the stochasticity in the parameters can appear anywhere in the model. The uncertainty is modeled via a finite set of scenarios $\omega = 1, ..., |\Omega|$, each with an associated probability of occurrence w^ω, $\omega \in \Omega$. Researchers have studied the properties and some solution approaches

[1]This research has been partially supported by the projects ECO2008-00777 ECON from the Ministry of Education and Science, Grupo de Investigación IT-347-10 from the Basque Government, grant FPU ECO-2006 from the Ministry of Education and Science, grants RM URJC-CM-2008-CET-3703 and RIESGOS CM from Comunidad de Madrid, and PLANIN MTM2009-14087-C04-01 from Ministry of Science and Innovation, Spain.

for such problems in the last decade, see [24, 33] for surveys on some mayor results in the area. The simplest form of two-stage stochastic integer programs contains first stage pure 0-1 variables and second stage continuous variables, see in [26] a branch-and-cut procedure based on Benders decomposition. A Branch-and-Fix Coordination BFC methodology is provided in [1, 2] for solving such a model in production planning applications, but the approach does not allow continuous first stage variables or 0-1 second stage variables. A generalization of the L-shaped method to deal with stochastic programs having 0-1 mixed-integer recourse variables and either pure continuous or pure first stage 0-1 variables is presented in [6]. A decomposition algorithms based on branch-and-cut generation for solving two-stage stochastic programs having first stage pure 0-1 variables and 0-1 mixed-integer recourse variables is proposed in [28, 35]. A similar branch-and-cut decomposition with a modified Benders decomposition method is developed in [34]. A branch-and-bound algorithm for problems having mixed-integer variables in both stages is designed in [7, 20]. However the approach of both papers focuses more on using Lagrangean relaxation to obtain good bounds, and less on branching and variable fixing. See also [38]. An algorithm for two-stage stochastic programs with mixed integers in *both* stages, and the continuous ones are in $[0, 1]$ is proposed in [37], but they only provide a limited computational experience. An interesting algorithm for solving a specific stochastic problem with uncertainty in the right hand side (rhs) by utilizing preprocessing, Lagrangean relaxation for obtaining lower bounds of the optimal solution and an ad-hoc heuristic is presented in [8] for obtaining good solutions in a multiperiod batch plant scheduling problem; the dimensions of the instances are very big. A hybrid algorithm for solving two-stage stochastic integer programs, where integer and continuous variables appear at in any stage is proposed in [39]. Two-stage stochastic integer programming arising in chemical bath scheduling are considered. Based on stage-decomposition, an evolucionary algorithm performs the search on the first-stage variables while the decoupled second-stage scenario problems are solved by a mixed integer linear optimization solver. We propose in [11, 13] a Branch and Fix Coordination based algorithm for obtaining the optimal solution of a general two-stage stochastic mixed 0-1 integer problem. Recently, a general algorithm for two-stage problems is described in [36].

In this paper we study solution methods for solving the dual problem corresponding to the Lagrangean Decomposition of two stage stochastic mixed 0-1 models. We represent the two-stage stochastic mixed 0-1 problem by a splitting variable representation of the Deterministic Equivalent Model, DEM, where 0-1 and continuous variables appear at any stage. Lagrangean Decomposition is proposed for satisfying both the integrality constraints for the 0-1 variables and the non-anticipativity constraints. We compare the performance of four solution methods based on dual Lagrangean Decomposition schemes, as the Subgradient method [18], the Volume algorithm [3], the Progressive Hedging algorithm [31] and the Dynamic Constrained Cutting Plane scheme [22]. See also [14]. At each iteration of these procedures a parametric mixed 0-1 problem is solved, and the parameters, i.e., the Lagrangean multipliers are updated by using the different methodologies. The performance of all these algorithms is very sensitive to the choice of given parameters as the step length, or the initial multiplier vector. We test the conditions and properties of convergence for medium and large-scale dimension stochastic problems. See in [23] a different approach for Lagrangeans updating.

A successful result may open the possibility for tightening the lower bounds of the solution at the candidate Twin Node Families in the exact Branch and Fix Coordination scheme [2]

for both two-stage and multistage types of problems, see [9, 12].

The remainder of the paper is organized as follows: Section 2 presents the two-stage stochastic mixed 0-1 problem, as well as the quadratic model that results from dualizing the nonanticipativity constraints. Section 3 summarizes the theoretical results on Lagrangean Decomposition. Section 4 presents the four methods mentioned above for updating the Lagrange multipliers. Section 5 reports the results of the computational experiment. Section 6 concludes.

2 Two-stage stochastic mixed 0-1 problem

Let us consider the two-stage stochastic mixed 0-1 problem, (MIP), in *compact* representation:

$$
\begin{array}{rlll}
(MIP): & z_{MIP} & = \min & c_1^T\delta + c_2^T x + E_\psi[(q_1^{\omega T}\gamma^\omega + q_2^{\omega T} y^\omega)] \\
& s.t. & & \\
& & & b_1 \leq A \begin{pmatrix} \delta \\ x \end{pmatrix} \leq b_2 \\
& & & h_1^\omega \leq T^\omega \begin{pmatrix} \delta \\ x \end{pmatrix} + W^\omega \begin{pmatrix} \gamma^\omega \\ y^\omega \end{pmatrix} \leq h_2^\omega, \quad \forall \omega \in \Omega \\
& & & x, y^\omega \geq 0, \quad \forall \omega \in \Omega \\
& & & \delta, \gamma^\omega \in \{0,1\}, \quad \forall \omega \in \Omega,
\end{array}
\tag{1}
$$

where c_1 and c_2 are known vectors of the objective function coefficients for the δ and x vector of variables in the first stage, respectively, b_1 and b_2 are the left and right hand side vectors for the first stage constraints, respectively, and A is the known matrix of coefficients for the first stage constraints. For each scenario ω, w^ω is the likelihood attributed to the scenario, and h_1^ω and h_2^ω are the left and right hand side vectors for the second stage constraints, respectively, and q_1^ω and q_2^ω are the objective function coefficients for the γ and y vector of variables, respectively, while T^ω and W^ω are the technology matrices under scenario ω, for $\omega \in \Omega$, where Ω is the set of scenarios to consider. Piecing together the stochastic components of the problem, we have a vector $\psi^\omega = (q_1^\omega, q_2^\omega, h_1^\omega, h_2^\omega, T^\omega, W^\omega)$. Finally, E_ψ represents the mathematical expectation with respect to ψ over the set of scenarios Ω.

Problem (1) is equivalent to the so-called *Deterministic Equivalent Model* that in the *splitting variable* representation is as follows,

$$
\begin{array}{rlll}
(MIP): & z_{MIP} & = \min & \sum_{\omega\in\Omega} w^\omega [c_1^T\delta^\omega + c_2^T x^\omega + q_1^{\omega T}\gamma^\omega + q_2^{\omega T} y^\omega] \\
& s.t. & & \\
& & & b_1 \leq A \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} \leq b_2, \quad \forall \omega \in \Omega \\
& & & h_1^\omega \leq T^\omega \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} + W^\omega \begin{pmatrix} \gamma^\omega \\ y^\omega \end{pmatrix} \leq h_2^\omega, \quad \forall \omega \in \Omega \\
& & & x^\omega \geq 0, \delta^\omega \in \{0,1\}, \quad \forall \omega \in \Omega \\
& & & y^\omega \geq 0, \gamma^\omega \in \{0,1\}, \quad \forall \omega \in \Omega \\
& & & x^\omega - x^{\omega+1} = 0, \quad \forall \omega = 1, ..., |\Omega| - 1 \\
& & & \delta^\omega - \delta^{\omega+1} = 0, \quad \forall \omega = 1, ..., |\Omega| - 1.
\end{array}
\tag{2}
$$

Observe that the relaxation of the nonanticipativity constraints

$$\begin{aligned} \delta^{\omega} - \delta^{\omega+1} &= 0, \quad \forall \omega = 1, ..., |\Omega| - 1 \\ x^{\omega} - x^{\omega+1} &= 0, \quad \forall \omega = 1, ..., |\Omega| - 1, \end{aligned}$$

from problem (2) results in $|\Omega|$ independent mixed 0-1 models. In order to avoid the use of non-signed vectors of Lagrangean multipliers in the dualization of equality constraints, we propose to express the nonanticipativity constraints in the following form

$$\delta^{\omega} - \delta^{\omega+1} \leq 0, \quad \forall \omega = 1, ..., |\Omega| - 1, \quad \delta^{|\Omega|} \leq \delta^{1} \tag{3}$$

$$x^{\omega} - x^{\omega+1} \leq 0, \quad \forall \omega = 1, ..., |\Omega| - 1, \quad x^{|\Omega|} \leq x^{1}. \tag{4}$$

We compare the elapsed time to obtain the optimal solution by the plain use of the optimization engine *COIN-OR*, for solving the (MIP) model and its linear relaxation over the compact representation (1) and splitting variable representation (2). This analysis represents a first step in the comparison of schemes based on dual Lagrangean problems, where several types of vectors of Lagrangean multipliers can be used.

3 Lagrangean decomposition

The Lagrangean Decomposition of the (MIP) problem (1) for a given nonnegative vector of weights $\mu = (\mu_{\delta}, \mu_x)$, is the mixed 0-1 minimization problem (5) in the vector of variables (δ, x, γ, y), where the optimal solution value is given by $z_{LD}(\mu)$. Let us denote this model as $(MIP_{LD}(\mu))$.

$$\begin{aligned} (MIP_{LD}(\mu)): \quad & z_{LD}(\mu) = \min \sum_{\omega \in \Omega} w^{\omega} [c_1^T \delta^{\omega} + c_2^T x^{\omega} + q_1^{\omega T} \gamma^{\omega} + q_2^{\omega T} y^{\omega}] \\ & + \sum_{\omega=1}^{|\Omega|-1} \mu_{\delta}^{\omega} (\delta^{\omega} - \delta^{\omega+1}) + \mu_{\delta}^{|\Omega|} (\delta^{|\Omega|} - \delta^{1}) + \\ & + \sum_{\omega=1}^{|\Omega|-1} \mu_{x}^{\omega} (x^{\omega} - x^{\omega+1}) + \mu_{x}^{|\Omega|} (x^{|\Omega|} - x^{1}) \\ s.t. \quad & \\ & b_1 \leq A \begin{pmatrix} \delta^{\omega} \\ x^{\omega} \end{pmatrix} \leq b_2, \quad \forall \omega \in \Omega \\ & h_1^{\omega} \leq T^{\omega} \begin{pmatrix} \delta^{\omega} \\ x^{\omega} \end{pmatrix} + W^{\omega} \begin{pmatrix} \gamma^{\omega} \\ y^{\omega} \end{pmatrix} \leq h_2^{\omega}, \quad \forall \omega \in \Omega \\ & x^{\omega} \geq 0, \delta^{\omega} \in \{0, 1\}, \quad \forall \omega \in \Omega \\ & y^{\omega} \geq 0, \gamma^{\omega} \in \{0, 1\}, \quad \forall \omega \in \Omega. \end{aligned} \tag{5}$$

Lagrangean Decomposition of (MIP) defined in (5), relative to the complicating nonanticipativity constraints for δ and x variables, (3)-(4), and non negative Lagrangean multipliers, μ_{δ} and μ_x, as vector of variables is a nonlinear minimization model with 0-1 variables. But also, it can be considered as a μ parametric mixed 0-1 problem.

Obviously, $(MIP_{LD}(\mu))$ is a relaxation of (MIP), since (i) the feasible set of $(MIP_{LD}(\mu))$ contains the feasible set of (MIP), and (ii) for any (δ, x, γ, y) feasible solution for (MIP) and any $\mu \geq 0$, it results that $z_{LD}(\mu) \leq z_{MIP}$. It follows that the optimal value $z_{LD}(\mu)$, which depends on μ is a lower bound on the optimal value of (MIP), z_{MIP}.

Definition 1 *The problem of finding the tightest Lagrangean lower bound on z_{MIP} is*

$$(MIP_{LD}): \quad z_{LD} = \max_{\mu \geq 0} z_{LD}(\mu).$$

It is called Lagrangean dual of (MIP) *relative to the (complicating) nonanticipativity constraints (3)-(4).*

It can be shown [17] that the Lagrangean Decomposition gives equal or stronger bounds of the solution value of the original problem than the Lagrangean relaxation of the constraints related to any of the scenarios to be decomposed. Then, let us concentrate our efforts in the study of Lagrangean Decomposition.

So, by LP duality, z_{LD} can be calculated from linear and mixed 0-1 programs. (MIP_{LD}) is a linear problem in the dual space of the lagrangean multipliers, whereas $(MIP_{LD}(\mu))$ is a μ-parametric mixed 0-1 problem in the vector of variables (δ, x, γ, y). Let $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ denote an optimal solution of $(MIP_{LD}(\mu))$ for some μ, i.e a Lagrangean solution. The following proposition shows when this solution is the optimal solution of (MIP).

Proposition 1 *1. If $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ is an optimal solution of $(MIP_{LD}(\mu))$ for some $\mu \geq 0$, then $z_{LD}(\mu) \leq z_{MIP}$.*

2. If in addition $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ is feasible for (MIP), then $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ is an optimal solution of (MIP) and $z_{MIP} = z_{LD}(\mu)$.

Proof:

1. Notice that, for some $\mu \geq 0$

$$\begin{aligned} z_{LD}(\mu) &= \min \sum_{\omega \in \Omega} w^\omega [c_1^T \delta^\omega + c_2^T x^\omega + q_1^{\omega T} \gamma^\omega + q_2^{\omega T} y^\omega] \\ &+ \sum_{\omega=1}^{|\Omega|-1} \mu_\delta^\omega (\delta^\omega - \delta^{\omega+1}) + \mu_\delta^{|\Omega|} (\delta^{|\Omega|} - \delta^1) + \\ &+ \sum_{\omega=1}^{|\Omega|-1} \mu_x^\omega (x^\omega - x^{\omega+1}) + \mu_x^{|\Omega|} (x^{|\Omega|} - x^1) \leq \\ &\leq \min \sum_{\omega \in \Omega} w^\omega [c_1^T \delta^\omega + c_2^T x^\omega + q_1^{\omega T} \gamma^\omega + q_2^{\omega T} y^\omega] = z_{MIP}. \end{aligned}$$

2. If $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ is feasible for (MIP), all the nonanticipativity constraints are satisfied and, then, complementary slackness holds automatically, i.e. $\mu_\delta^\omega (\delta^\omega - \delta^{\omega+1}) = 0$ and $\mu_x^\omega (x^\omega - x^{\omega+1}) = 0, \forall \omega$, thus $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ is an optimal solution of (MIP), with $z_{MIP} = z_{LD}(\mu)$.

Notice first that this is always a sufficient condition for optimality and, in case that the constraints that are dualized are equality constraints, it is also necessary. I.e., it is not possible for a feasible solution to be optimal for (MIP) if it does not satisfy complementary slackness.

The following result, from [15], gives a geometric interpretation of the Lagrangean dual problem in the primal space, see also [16, 17],

Proposition 2 *The Lagrangean dual (MIP_{LD}) is equivalent to the primal relaxation*

$$
\begin{array}{rcl}
z_{LD} & = & \min \sum_{\omega\in\Omega} w^\omega [c_1^T \delta^\omega + c_2^T x^\omega + q_1^{\omega T}\gamma^\omega + q_2^{\omega T} y^\omega] \\
s.t. & & \\
 & & \delta^\omega - \delta^{\omega+1} \leq 0, \quad \forall \omega = 1, ..., |\Omega| - 1, \quad \delta^{|\Omega|} \leq \delta^1 \\
 & & x^\omega - x^{\omega+1} \leq 0, \quad \forall \omega = 1, ..., |\Omega| - 1, \quad x^{|\Omega|} \leq x^1 \\
\delta^\omega, x^\omega, \gamma^\omega, y^\omega \in conv & \{ & b_1 \leq A \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} \leq b_2, \quad \forall \omega \in \Omega \\
 & & h_1^\omega \leq T^\omega \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} + W^\omega \begin{pmatrix} \gamma^\omega \\ y^\omega \end{pmatrix} \leq h_2^\omega, \quad \forall \omega \in \Omega \\
 & & x^\omega \geq 0, \delta^\omega \in \{0,1\}, \quad \forall \omega \in \Omega \\
 & & y^\omega \geq 0, \gamma^\omega \in \{0,1\}, \quad \forall \omega \in \Omega \\
 & \} &
\end{array}
$$

where conv$\{S\}$ denotes the convex hull of the set S.

We will see that the Lagrangean Decomposition bound z_{LD} is at least as good as the LP relaxation bound, z_{LP}, never worst.

Corollary 1 *If $conv(S) = S$, where S is the set*

$$
\begin{array}{l}
b_1 \leq A \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} \leq b_2, \quad \forall \omega \in \Omega \\
h_1^\omega \leq T^\omega \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} + W^\omega \begin{pmatrix} \gamma^\omega \\ y^\omega \end{pmatrix} \leq h_2^\omega, \quad \forall \omega \in \Omega \\
x^\omega \geq 0, \delta^\omega \in \{0,1\}, \quad \forall \omega \in \Omega \\
y^\omega \geq 0, \gamma^\omega \in \{0,1\}, \quad \forall \omega \in \Omega
\end{array}
\tag{6}
$$

then $z_{LP} = z_{LD}$.

In that case, the Lagrangean Decomposition bound is equal to the LP bound and $z_{LP} = z_{LD} \leq z_{MIP}$. (It is said that the LD problem has the *integrality property*).

Corollary 2 *If $conv(S) \subset S$, where the set S is defined in (6), then $z_{LP} \leq z_{LD} \leq z_{MIP}$, and it may happen that the Lagrangean relaxation be strictly better than the LP bound.*

The interpretation of these two corollaries is that, unless (MIP_{LD}) does have the integrality property, it can yield an equal or stronger bound than the LP relaxation. If it has the integrality property then $z_{LP} = z_{LD} \leq z_{MIP}$. In the other case, $z_{LP} \leq z_{LD} \leq z_{MIP}$.

4 Obtaining Lagrangean duals

Our first proposal to test consists of solving (MIP) problem and its linear relaxation by the plain use of the optimization engine *COIN-OR*, see [21, 29], over the compact representation (1) and the splitting variable representation (2).

Our second proposal makes use of the expression of the Lagrangean dual z_{LD} as a maximization of the optimal solution values $z_{LD}(\mu)$ in μ. Then, before that, and for a given value of μ, we must solve the mixed $0-1$ problem (5) in $(\delta(\mu), x(\mu), \gamma(\mu), y(\mu))$ and obtaining the optimal solution value, $z_{LD}(\mu)$.

In this case we will compare several iterative methods for updating the value of the Lagrange multipliers and building the sequence $\{\mu^0, \mu^1, ..., \mu^k, ...\}$.

At each iteration k and given the current multiplier vector μ^k, the aim is to obtain the value $z_{LD}(\mu^k)$. Then, and by updating the Lagrangean multipliers μ in a finite number of iterations, the purpose is to obtain μ^* and $z_{LD}(\mu^*)$, where

$$\mu^* = \text{argmax}_\mu\{z_{LD}(\mu)\}. \tag{7}$$

4.0.1 Solving μ-parametric Lagrangean duals

As it has been shown above, the μ-parametric $(MIP_{LD}(\mu))$ problem (5) must be solved, where the parametric vector $(\mu) = (\mu_\delta, \mu_x)$ is given. Then, the second part of the objective function in $(MIP_{LD}(\mu))$ can be decomposed in the following form:

$$\begin{array}{rl}
& \sum_{\omega=1}^{|\Omega|-1} \mu_\delta^\omega(\delta^\omega - \delta^{\omega+1}) + \mu_\delta^{|\Omega|}(\delta^{|\Omega|} - \delta^1) + \sum_{\omega=1}^{|\Omega|-1} \mu_x^\omega(x^\omega - x^{\omega+1}) + \mu_x^{|\Omega|}(x^{|\Omega|} - x^1) = \\
= & \mu_\delta^1(\delta^1 - \delta^2) + \mu_\delta^2(\delta^2 - \delta^3) + \cdots + \mu_\delta^{|\Omega|-1}(\delta^{|\Omega|-1} - \delta^{|\Omega|}) + \mu_\delta^{|\Omega|}(\delta^{|\Omega|} - \delta^1) + \\
+ & \mu_x^1(x^1 - x^2) + \mu_x^2(x^2 - x^3) + \cdots + \mu_x^{|\Omega|-1}(x^{|\Omega|-1} - x^{|\Omega|}) + \mu_x^{|\Omega|}(x^{|\Omega|} - x^1) = \\
= & (\mu_\delta^1 - \mu_\delta^{|\Omega|})\delta^1 + (\mu_\delta^2 - \mu_\delta^1)\delta^2 + \cdots + (\mu_\delta^{|\Omega|-1} - \mu_\delta^{|\Omega|-2})\delta^{|\Omega|-1} + (\mu_\delta^{|\Omega|} - \mu_\delta^{|\Omega|-1})\delta^{|\Omega|} + \\
+ & (\mu_x^1 - \mu_x^{|\Omega|})x^1 + (\mu_x^2 - \mu_x^1)x^2 + \cdots + (\mu_x^{|\Omega|-1} - \mu_x^{|\Omega|-2})x^{|\Omega|-1} + (\mu_x^{|\Omega|} - \mu_x^{|\Omega|-1})x^{|\Omega|}.
\end{array} \tag{8}$$

So, the corresponding optimal value in problem $(MIP_{LD}(\mu))$ (5) can be decomposed as the sum of the optimal solutions of smaller subproblems (10)-(11), one for each scenario such that the optimal objective function value of $(MIP_{LD}(\mu))$ (5) can be calculated as:

$$z_{LD}(\mu) = \sum_{\omega=1}^{|\Omega|} z_{LD}(\mu^\omega), \tag{9}$$

where $z_{LD}(\mu^\omega)$ for $\omega = 2, ..., |\Omega|$, is the optimal solution of the following problem:

$$\begin{array}{rll}
z_{LD}(\mu^\omega) & = & \min[w^\omega c_1^T + (\mu_\delta^\omega - \mu_\delta^{\omega-1})]\delta^\omega + [w^\omega c_2^T + (\mu_x^\omega - \mu_x^{\omega-1})]x^\omega + w^\omega q_1^{\omega T}\gamma^\omega + w^\omega q_2^{\omega T} y^\omega \\
& & b_1 \leq A \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} \leq b_2 \\
& & h_1^\omega \leq T^\omega \begin{pmatrix} \delta^\omega \\ x^\omega \end{pmatrix} + W^\omega \begin{pmatrix} \gamma^\omega \\ y^\omega \end{pmatrix} \leq h_2^\omega \\
& & x^\omega \geq 0, \delta^\omega \in \{0,1\} \\
& & y^\omega \geq 0, \gamma^\omega \in \{0,1\}
\end{array} \tag{10}$$

and if $\omega = 1$, the corresponding optimization problem is given by:

$$\begin{aligned} z_{LD}(\mu^1) \quad = \quad & \min[w^1 c_1^T + (\mu_\delta^1 - \mu_\delta^{|\Omega|})]\delta^1 + [w^1 c_2^T + (\mu_x^1 - \mu_x^{|\Omega|})]x^1 + w^1 q_1^{1T}\gamma^1 + w^1 q_2^{2T} y^1 \\ & b_1 \leq A \begin{pmatrix} \delta^1 \\ x^1 \end{pmatrix} \leq b_2 \\ & h_1^1 \leq T^1 \begin{pmatrix} \delta^1 \\ x^1 \end{pmatrix} + W^1 \begin{pmatrix} \gamma^1 \\ y^1 \end{pmatrix} \leq h_2^1 \\ & x^1 \geq 0, \delta^1 \in \{0,1\} \\ & y^1 \geq 0, \gamma^1 \in \{0,1\}. \end{aligned} \tag{11}$$

4.1 Subgradient method

A number of methods have been proposed to solve the Lagrangean duals. They are either ad-hoc or general purpose, usually aiming at solving a generic nonsmooth convex optimization problem. One of the most popular approaches is the Subgradient method.

4.1.1 Scheme

The Subgradient method was proposed in [18] and then validated in [19]. See also [4] and [14], among many others. It is an iterative method in which at iteration k, given the current multiplier vector μ^k, a step is taken along a subgradient of $z_{LD}(\mu^k)$. Let $(\delta^{(k)}, x^{(k)}, \gamma^{(k)}, y^{(k)})$ be an optimal solution of $(MIP_{LD}(\mu^k))$. Then

$$s^k = \begin{pmatrix} (\delta^{(k)1} - \delta^{(k)2}) \\ \vdots \\ (\delta^{(k)|\Omega|-1} - \delta^{(k)|\Omega|}) \\ (\delta^{(k)|\Omega|} - \delta^{(k)1}) \\ (x^{(k)1} - x^{(k)2}) \\ \vdots \\ (x^{(k)|\Omega|-1} - x^{(k)|\Omega|}) \\ (x^{(k)|\Omega|} - x^{(k)1}) \end{pmatrix} \tag{12}$$

is a subgradient of $z_{LD}(\mu)$ at μ^k. If μ^* is an (unknown) optimal solution of (MIP_{LD}), the step $\mu^{k+1} - \mu^k$ is a nonnegative multiple of s^k, and then:

$$\mu^{k+1} - \mu^k = \beta^k \cdot s^k$$

The most commonly used expression for the step length β^k is as follows:

$$\beta^k = \frac{(z_{LD}(\mu^*) - z_{LD}(\mu^k))}{||s^k||^2}$$

where

$$\begin{aligned} ||s^k||^2 = & \sum_{\omega=1}^{|\Omega|-1} (\delta^{(k)\omega} - \delta^{(k)\omega+1})^2 + (\delta^{(k)|\Omega|} - \delta^{(k)1})^2 \\ & + \sum_{\omega=1}^{|\Omega|-1} (x^{(k)\omega} - x^{(k)\omega+1})^2 + (x^{(k)|\Omega|} - x^{(k)1})^2 \end{aligned}$$

so that

$$\mu^{k+1} = \mu^k + \frac{(z_{LD}(\mu^*) - z_{LD}(\mu^k))}{||s^k||^2} \cdot s^k$$

This formula unfortunately uses the unknown optimal value of (MIP_{LD}), $z_{LD}(\mu^*)$. An upper bound of this value, $\overline{z}_{LD}$, can be used (e.g., the value of the current feasible solution of the original problem $(MIP)(1)$), and if the objective function value does not improve in a given number iterations, thus one can try to reduce the difference $(\overline{z}_{LD} - z_{LD}(\mu^k))$, for example, by introducing from the start a positive factor $\alpha_0 \in (0, 2)$, in the Subgradient formula:

$$\mu^{k+1} = \mu^k + \alpha_k \cdot \frac{(\overline{z}_{LD} - z_{LD}(\mu^k))}{||s^k||^2} \cdot s^k$$

and reducing the scalar α_k when there is no improvement for too long, see Section 4.5. See in [30] that $z_{LD}(\mu) \longrightarrow z_{LD}$ if $\beta^k \longrightarrow 0$ and $\sum_{k=0}^{\infty} \alpha_k \longrightarrow \infty$. In practice, $\mu^0 = 0$ is very frequent, although sometimes better results can be obtained by considering that μ^0 is given by the duals of nonanticipativity constraints in the LP relaxation of the original problem (1), i.e. the simplex multipliers, see Section 5. Usually α_k is decreased after a certain number of iterations without an improvement. There are proofs of convergence for different choices of the step length, see [30] but, as we know, these properties have not been tested for large-scale dimension problems.

4.1.2 Procedure

Let $\overline{z}_{LD}$ be an upper bound of the optimal solution value of (MIP) (1), and $z_{LD}(\mu^k)$ the optimal solution value of $(MIP_{LD}(\mu^k))$ (5), computed at iteration k as the sum given in (9). Let α_k be a real parameter, see Section 4.5 and μ^0 the initial multiplier vector. The implementation of the Subgradient method is as follows:

Step 0: We start with a vector μ^0, and solve $|\Omega|$ problems (10)-(11) to obtain $(\delta^{(0)}, x^{(0)}, \gamma^{(0)}, y^{(0)})$ and $z_{LD}(\mu^0)$ as the sum given in (9). Set $k := 0$.

Step 1: Compute $s^k = \begin{pmatrix} (\delta^{(k)1} - \delta^{(k)2}) \\ \vdots \\ (\delta^{(k)|\Omega|-1} - \delta^{(k)|\Omega|}) \\ (\delta^{(k)|\Omega|} - \delta^{(k)1}) \\ (x^{(k)1} - x^{(k)2}) \\ \vdots \\ (x^{(k)|\Omega|-1} - x^{(k)|\Omega|}) \\ (x^{(k)|\Omega|} - x^{(k)1}) \end{pmatrix}$, check the stopping criterion (19) and if it is not satisfied, set $\mu^{k+1} := \mu^k + \alpha_k \cdot \frac{(\overline{z}_{LD} - z_{LD}(\mu^k))}{||s^k||^2} \cdot s^k$

Solve the $|\Omega|$ problems (10)-(11) with μ^{k+1}, and let $(\delta^{(k+1)}, x^{(k+1)}, \gamma^{(k+1)}, y^{(k+1)})$ and $z_{LD}(\mu^{k+1})$ be the optimal solution and optimal solution value of (5) that have been obtained, respectively.

Set $k := k + 1$ and go to Step 1.

Although there are many cases in the literature in which this procedure has been very efficient, also there exist problems in which the Subgradient algorithm presents an oscillatory behavior for updating the Lagrange multipliers. Based in our computational experience, we have observe that the procedure is very sensitive to the choice of given parameters related to the step length, such that the parameter α_k or the initial multiplier vector μ^0.

4.2 The Volume algorithm

We present a version of the Volume algorithm given in [3]. This procedure only updates the multipliers when an improvement in the incumbent solution of the Lagrangean problem, $z_{LD}(\mu)$, is produced. Additionally, the feasible solution is replaced by a convex combination of solutions obtained in previous iterations.

Remind that problem $(MIP_{LD}(\mu))$ (5) is decomposed as a sum of $|\Omega|$ lower dimensional problems (10)-(11), one for each scenario. And the corresponding optimal solution value is obtained as the sum given in (9).

4.2.1 Procedure

Let $\overline{z}_{LD}$ be an upper bound of the optimal solution, and α_k and f_k two real parameters related to the step length, where $\alpha_0 \in (0,2)$ and $f_k \in (0,1)$, see Section 4.5. Let μ^0 be the initial multiplier vector.

Step 0: We start with a multiplier vector μ^0, and solve the $|\Omega|$ problems (10)-(11) to obtain $(\delta^{(0)}, x^{(0)}, \gamma^{(0)}, y^{(0)})$ and $z_{LD}(\mu^0)$ as the sum given in (9). Set:

$$\begin{aligned}(\overline{\delta}, \overline{x}, \overline{\gamma}, \overline{y}) &= (\delta^{(0)}, x^{(0)}, \gamma^{(0)}, y^{(0)}),\\ \overline{\mu} &= \mu^0,\\ \overline{z}(\overline{\mu}) &= z_{LD}(\overline{\mu}) = \sum_{\omega=1}^{|\Omega|} z_{LD}(\overline{\mu}^{\omega}),\end{aligned}$$

and $k := 1$.

Step 1: Compute $\overline{s}^k = \begin{pmatrix} (\overline{\delta}^1 - \overline{\delta}^2) \\ \vdots \\ (\overline{\delta}^{|\Omega|-1} - \overline{\delta}^{|\Omega|}) \\ (\overline{\delta}^{|\Omega|} - \overline{\delta}^1) \\ (\overline{x}^1 - \overline{x}^2) \\ \vdots \\ (\overline{x}^{|\Omega|-1} - \overline{x}^{|\Omega|}) \\ (\overline{x}^{|\Omega|} - \overline{x}^1) \end{pmatrix}$, check the stopping criterion (19) and if it not satisfied, set

$$\mu^k := \overline{\mu} + \alpha_k \cdot \frac{(\overline{z}_{LD} - \overline{z}(\overline{\mu}))}{||\overline{s}^k||^2} \cdot \overline{s}^k.$$

Solve the $|\Omega|$ problems (10)-(11) with μ^k, and let $(\delta^{(k)}, x^{(k)}, \gamma^{(k)}, y^{(k)})$ and $z_{LD}(\mu^k)$ be the optimal solution and optimal solution value of (5) that have been obtained, respectively. Then, $(\overline{\delta}, \overline{x}, \overline{\gamma}, \overline{y})$ is updated as

$$(\overline{\delta}, \overline{x}, \overline{\gamma}, \overline{y}) := f_k \cdot (\delta^{(k)}, x^{(k)}, \gamma^{(k)}, y^{(k)}) + (1 - f_k) \cdot (\overline{\delta}, \overline{x}, \overline{\gamma}, \overline{y}).$$

Step 2: If $z_{LD}(\mu^k) > \overline{z}(\overline{\mu})$ update $\overline{\mu}$ and $\overline{z}(\overline{\mu})$ as

$$\overline{\mu} := \mu^k, \qquad \overline{z}(\overline{\mu}) := z_{LD}(\mu^k).$$

Set $k := k + 1$ and go to Step 1.

4.3 Progressive Hedging algorithm

The basic features of the Progressive Hedging Algorithm, see [31, 38], for updating the Lagrange multipliers of the nonanticipativity constraints are as follows: let $(\delta^{(k)}, x^{(k)}, \gamma^{(k)}, y^{(k)})$ be an optimal solution of problem $(MIP_{LD}(\mu^k))$ (5), at iteration k. A non necessarily new feasible solution can be defined as: $\hat{\delta} = \sum_{\omega \in \Omega} w^\omega \delta^{(k)\omega}$ and $\hat{x} = \sum_{\omega \in \Omega} w^\omega x^{(k)\omega}$. These expressions represent an estimation as the mean over the set of scenarios of the optimal solutions $\delta^{(k)}, x^{(k)}$ of model(5), obtained at iteration k.

The updating of the Lagrange multipliers is as follows,

$$\mu^{k+1} = \mu^k + \beta^k \cdot \hat{s}^k, \tag{13}$$

where

$$\hat{s}^k = \begin{pmatrix} (\delta^{(k)1} - \hat{\delta}^{(k)}) \\ (\delta^{(k)2} - \hat{\delta}^{(k)}) \\ \vdots \\ (\delta^{(k)|\Omega|} - \hat{\delta}^{(k)}) \\ (x^{(k)1} - \hat{x}^{(k)}) \\ (x^{(k)2} - \hat{x}^{(k)}) \\ \vdots \\ (x^{(k)|\Omega|} - \hat{x}^{(k)}) \end{pmatrix} \tag{14}$$

and $\hat{\delta}^{(k)}$ and $\hat{x}^{(k)}$ are the approximated (estimated) values of the δ and x variables, respectively, for each scenario at iteration k.

4.3.1 Procedure

Let $\overline{z}_{LD}$ be an upper bound of the optimal solution, and α_k a real parameter related to the step length, see Section 4.5. Let μ^0 be the initial multiplier vector.

Step 0: Start with a given Lagrange multipliers vector, say, μ^0 and solve the $|\Omega|$ problems (10)-(11) to obtain $(\delta^{(0)}, x^{(0)}, \gamma^{(0)}, y^{(0)})$ and $z_{LD}(\mu^0)$ as the sum given in (9). Set $k := 0$.

Step 1: Compute $\hat{s}^k = \begin{pmatrix} (\delta^{(k)1} - \hat{\delta}^{(k)}) \\ (\delta^{(k)2} - \hat{\delta}^{(k)}) \\ \vdots \\ (\delta^{(k)|\Omega|} - \hat{\delta}^{(k)}) \\ (x^{(k)1} - \hat{x}^{(k)}) \\ (x^{(k)2} - \hat{x}^{(k)}) \\ \vdots \\ (x^{(k)|\Omega|} - \hat{x}^{(k)}) \end{pmatrix}$, check the stopping criterion (19) and if it not satisfied, set

$$\mu^{k+1} := \mu^k + \alpha_k \cdot \frac{(\overline{z}_{LD} - z_{LD}(\mu^k))}{||\hat{s}^k||^2} \cdot \hat{s}^k, \tag{15}$$

Solve the $|\Omega|$ problems (10)-(11) with μ^{k+1}, and let $(\delta^{(k+1)}, x^{(k+1)}, \gamma^{(k+1)}, y^{(k+1)})$ and $z_{LD}(\mu^{k+1})$ be the optimal solution that have been obtained. Compute the estimations of $\hat{\delta}^{k+1}$ and $\hat{\delta}^{k+1}$, given by

$$\begin{aligned} \hat{\delta}^{(k+1)} &= \sum_{\omega \in \Omega} w^\omega \delta^{(k+1)\omega} \\ \hat{x}^{(k+1)} &= \sum_{\omega \in \Omega} w^\omega x^{(k+1)\omega} \end{aligned}$$

respectively.

Set $k := k + 1$ and goto step 1.

4.4 Dynamic Constrained Cutting Plane method

The Dynamic Constrained Cutting Plane method is closely related to the use of trust-region algorithms for the solution of the dual problem, see [14, 22].

Let $z_{LD}(\mu^i)$ denote the objective function value of the model $MIP_{LD}(\mu)$ (5) attained at iteration i, for the given value μ^i. A truncation of the Taylor series expansion of the function $z_{LD}(\mu)$ around the point μ^i can be expressed as

$$\overline{z}_{LD}(\mu^i) = z_{LD}(\mu^i) + \sum_{\omega \in \Omega} (\mu^\omega - \mu^{i,\omega}) s^i, \tag{16}$$

where $\mu^{i,\omega}$ denotes the Lagrangean element of μ vector at iteration i under scenario ω, and s^i is the subgradient of $z_{LD}(\mu)$ at μ^i.

The updated value, say, μ^k of the Lagrange multipliers at iteration k can be obtained as the μ solution of the model,

$$\begin{aligned} \overline{z}_{LD}(\mu^k) &= \max_{\mu \in C^k(\mu)} z \\ z &\leq \overline{z}_{LD}(\mu^i), \quad \forall i \in I, \end{aligned}$$

i.e.

$$\begin{aligned} \overline{z}_{LD}(\mu^k) &= \max_{\mu \in C^k(\mu)} z \\ z &\leq z_{LD}(\mu^i) + \sum_{\omega \in \Omega} (\mu^\omega - \mu^{i,\omega}) s^i, \quad \forall i \in I, \end{aligned} \tag{17}$$

where $z_{LD}(\mu^i)$ is the optimal objective function value of model (5) at iteration i, and s^i is the subgradient of $z_{LD}(\mu)$ at μ^i, for $i = 0, ..., |I|$. I is the set of cutting planes to use, such that $|I| = min\{k, \hat{n}\}$, where k denote the current iteration and $\hat{n}$ denotes the maximum number of cutting planes, i.e. the maximum number of constraints in (17). $C^k(\mu)$ denotes the feasible region of the Lagrange multipliers at iteration k. This set is defined by a lower bound and an upper bound, in the following sense

$$C^k(\mu) = \{\mu, \underline{\mu}^k \leq \mu \leq \overline{\mu}^k\}.$$

So, it is dynamically adjusted at each iteration. Let μ_j^k be the jth component of the multiplier vector obtained as optimal solution of (17) at iteration k. Given the problem structure, it is noted that either $\mu_j^k = \overline{\mu}_j^k$ or $\mu_j^k = \underline{\mu}_j^k$.

Then, at each iteration the new feasible set $C^{k+1}(\mu)$ is defined around the optimal multiplier μ^k obtained at iteration k as the optimal solution of the problem (17). The updating is as follows:

$$\underline{\mu}_j^{k+1} = \mu_j^k - \alpha_k \cdot \beta^k \cdot |s_j^k|, \text{ and } \overline{\mu}_j^{k+1} = \mu_j^k + \alpha_k \cdot \beta^k \cdot |s_j^k|,$$

and the step length β^k and the parameter α_k are computed as in the Subgradient method.

Notice that the number of constraints of model (17) grows with the number of iterations. To avoid the excessive growth, instead of considering all the cuts, only some of them are used. Let $d^i \geq 0$, for $i \in I$ denote the difference between the ith hyperplane evaluated at the current multiplier vector, $z_{LD}(\mu^i) + \sum_{\omega \in \Omega} (\mu^{k,\omega} - \mu^{i,\omega}) s^i$, and the current objective function value $z_{LD}(\mu^k)$, such that

$$d^i = z_{LD}(\mu^i) + \sum_{\omega \in \Omega} (\mu^{k,\omega} - \mu^{i,\omega}) s^i - z_{LD}(\mu^k), \quad \forall i \in I \tag{18}$$

It should be noted that the residual d^i is always positive, since the cutting plane reconstruction of the dual function overestimates the actual dual function. Then, the most distant hyperplanes are deleted from set I. Let k be the current iteration, such that if $k \leq \hat{n}$ then all the cutting planes are considered in model (17), i.e. $|I| = k + 1$. If $k > \hat{n}$, the most distant hyperplane, $\iota = argmax_{i \in I} d_i$, is deleted from I, so the cutting plane related to iteration ι is not considering in model (17).

Although this algorithm is typically stopped when the multiplier vector difference between two consecutive iterations is below a pre-specified threshold, a small difference between an upper bound and a lower bound of the dual optimum is also an appropriate stopping criterion, see Subsection 4.5.

4.4.1 Procedure

Let $\overline{z}_{LD}$ be an upper bound of the optimal solution value of (MIP) (1), and $z_{LD}(\mu^k)$ the optimal solution value of $(MIP_{LD}(\mu^k))$ (5), computed at iteration k as the sum given in (9). Let α_k be a real parameter and μ^0 the initial multiplier vector. Let also $\hat{n}$ be a natural parameter.

Step 0: We start with a vector μ^0, and solve the $|\Omega|$ problems (10)-(11) to obtain $(\delta^{(0)}, x^{(0)}, \gamma^{(0)}, y^{(0)})$ and $z_{LD}(\mu^0)$ as the sum given in (9). Set $k := 0$.

Step 1: Compute $s^k = \begin{pmatrix} (\delta^{(k)1} - \delta^{(k)2}) \\ \vdots \\ (\delta^{(k)|\Omega|-1} - \delta^{(k)|\Omega|}) \\ (\delta^{(k)|\Omega|} - \delta^{(k)1}) \\ (x^{(k)1} - x^{(k)2}) \\ \vdots \\ (x^{(k)|\Omega|-1} - x^{(k)|\Omega|}) \\ (x^{(k)|\Omega|} - x^{(k)1}) \end{pmatrix}$, check the stopping criterion (19) and if it is satisfied, stop.

Step 2: Define the feasible region of the Lagrangean multipliers $C^{k+1}(\mu)$,

$$\underline{\mu}_j^{k+1} = \mu_j^k - \alpha_k \cdot \beta^k \cdot |s_j^k|, \quad \text{and} \quad \overline{\mu}_j^{k+1} = \mu_j^k + \alpha_k \cdot \beta^k \cdot |s_j^k|$$

where

$$\beta^k = \frac{(\overline{z}_{LD} - z_{LD}(\mu^k))}{||s^k||^2}$$

and α_k is defined in Section 4.5.

Solve model (17) for calculating the new vector μ^{k+1}.

If $k > \hat{n}$, compute d_i as

$$d^i = z_{LD}(\mu^i) + \sum_{\omega \in \Omega} (\mu^{k,\omega} - \mu^{i,\omega}) s^i - z_{LD}(\mu^k), \quad \forall i \in I$$

and compute $\iota = argmax_{i \in I} d_i$. The cut ι is deleted from I.

Step 3: Solve the $|\Omega|$ problems (10)-(11) with μ^{k+1}, and let $(\delta^{(k+1)}, x^{(k+1)}, \gamma^{(k+1)}, y^{(k+1)})$ and $z_{LD}(\mu^{k+1})$ be the optimal solution and optimal solution value of (5) that have been obtained, respectively.

Set $k := k + 1$ and go to Step 1.

4.5 Choice of the parameters α_k and f_k, and the stopping criterion

The performance of all the procedures reported in Section 5 is very sensitive to the choice of the given parameters α_k and f_k, see some implementation details in [3]. Following the notation given in that paper, we consider three types of iterations for setting the value of α_k. Each time that we do not find an improvement, $z_{LD}(\mu^k) \leq z_{LD}(\mu^{k-1})$, we call this iteration *red*. In other case, $z_{LD}(\mu^k) > z_{LD}(\mu^{k-1})$, we compute d^k as follows:

1. In the Subgradient and Dynamic Constrained Cutting Plane algorithms, this value is defined as $d^k = (s^k)^t \cdot s^{k-1}$, where s^k is the subgradient vector calculated in (12).

2. In Volume algorithm, this value is defined as $d^k = (s^k)^t \cdot \overline{s}^k$, where $\overline{s}^k$ is the subgradient vector calculated in the Step 1 of the corresponding procedure.

3. In Progressive Hedging algorithm, this value is defined as $d^k = (s^k)^t \cdot \hat{s}^k$, where $\hat{s}^k$ is the subgradient vector calculated in the Step 1 of the corresponding procedure.

In all the methods, if $d^k < 0$ it means that a longer step in the direction of s^k would have given a smaller value for $z_{LD}(\mu^k)$. In this case, we call this iteration *yellow*. If $d^k \geq 0$ we call this iteration *green*. At each green iteration we multiply α_k by 1.1. After each sequence of $\#red$ consecutive *red* iterations we multiply α_k by 0.66. The best value for $\#red$ in our computational experimentation has been 1. When $\#red$ larger than 1, sometimes the procedure does not converge. We have tested all the cases for different values of $\alpha_0 \in (0, 2)$, and finally we have chosen $\alpha_0 = 0.1$.

Moreover, the parameter f_k in the Volume algorithm is set to a fixed value for a number of iterations and it is decreased afterwards. Let s^k and $\overline{s}^k$ be defined as in (12) and in the Step 1 of the Volume procedure, and let f_{max} be an upper bound of f_k. Then, we can compute f_{opt} as the value that minimizes $||f \cdot s^k + (1-f) \cdot \overline{s}||$. It is easy to verify that this value is $f_{opt} = \dfrac{\sum_{i=1}^{2|\Omega|} \overline{s}_i^k (s_i^k - \overline{s}_i^k)}{\sum_{i=1}^{2|\Omega|} (\overline{s}_i^k - s_i^k)(s_i^k - \overline{s}_i^k)}$. If $f_{opt} < 0$, set $f_k = \frac{1}{10} \cdot f_{max}$. Otherwise, set $f_k = \min\{f_{max}, f_{opt}\}$. In our computational experimentation we have used $f_{max} = f_0 = 0.1$ and we have decreased its value near the end.

The stopping criterion common to the four procedures is as follows:

$$\frac{|\sum_{\omega \in \Omega} w^\omega [c_1^T \delta^{(k)\omega} + c_2^T x^{(k)\omega} + q_1^{\omega T} \gamma^{(k)\omega} + q_2^{\omega T} y^{(k)\omega}] - z_{LD}(\mu^k)|}{|z_{LD}(\mu^k)|} < \epsilon_z$$

and (19)

$$\frac{\sum_{i=1}^{|\Omega|} |s_{i\delta}|}{|\Omega| \cdot n_\delta} < \epsilon_\delta \quad \text{and} \quad \frac{\sum_{i=1}^{|\Omega|} |s_{ix}|}{|\Omega| \cdot n_x} < \epsilon_x$$

where $|\Omega| \cdot n_\delta$ and $|\Omega| \cdot n_x$ are the number of nonanticipativity constraints for the δ and x vector of variables, respectively. $s_{1\delta}$ and s_{1x} denote the deviations for the corresponding δ and x rows of vector s, say $\overline{s}$, or $\hat{s}$, respectively, and ϵ_z, ϵ_δ and ϵ_x are given tolerances. In particular, we have considered, $\epsilon_z = 0.008$, $\epsilon_\delta = 0.01$ and $\epsilon_x = 0.1$. Another stopping criterion that we have considered is that if the incumbent solution, $z_{LD}(\mu^k)$, does not improve (given a tolerance $\epsilon = 0.0001$) after a sequence of ten consecutive iterations, stop.

Finally, in the Dynamic Constrained Cutting Plane procedure, the maximum number of cutting planes has been fixed to $\hat{n} = 5$, for all the instances.

5 Computational experience

We report the results of the computational experience obtained while optimizing the two-stage stochastic mixed 0-1 model (1) over some randomly generated instances. Our algorithmic approach has been implemented in a C++ experimental code, which uses the optimization engine *COIN-OR*, see [21, 29], to solve the linear and mixed 0-1 related models. The computations

were carried out on a Workstation Sun FIRE v245, under Solaris System 1.0, having cpu speed of 1.5GHz and 4Gb of RAM.

The first part of the testbed has small-medium sized instances, while the second has larger, harder instances, significantly bigger instances than those normally reported in the literature, e.g., [37]. The structure of the DEM considered is inspired in model (38) of [37]. The vectors of the objective function coefficients, $c_1, c_2, (q_1^\omega), (q_2^\omega)$ from model (1) for the test cases, were generated using uniform distributions over $[-2.5,\ -1.5]$, $[-2.5+k_1,\ -1.5+k_1]$, with $k_1 \in \{0, 1\}$, $[-30,\ -10]$ and $[-30,\ -10]$, respectively. The left hand side vector b_1 was fixed to 0 and $(h_1^\omega) = -(h_2^\omega)$. The rhs vectors b_2 and (h_2^ω) were generated using uniform distributions over $[a * |\Omega|,\ |\Omega| * [a + k(n_\delta + n_x)]]$ and $[a,\ a + k(n_\delta + n_x + n_\gamma + n_y)]$, respectively, where $a = 1$ and $k = 1$. n_δ, and n_x are the number of 0-1 and continuous first stage variables, respectively, and n_γ, and n_y are the corresponding number of 0-1 and continuous second stage variables. The matrix of coefficients for the first stage constraints, A, and the technology matrices for the second stage constraints, T^ω and W^ω, were generated using uniform distributions over $[0,\ 2 * |\Omega|]$, $[0,\ 0.3]$ and $[0,\ k_2]$, with $k_2 \in \{8, 4.5, 2.8\}$ respectively.

Tables 1 shows the dimensions of the instances in the compact and in splitting variable representations. The headings are as follows: m, number of constraints; n_δ, number 0-1 first stage variables; n_x, number of continuous first stage variables; n_γ, number 0-1 second stage variables; n_y, number of continuous second stage variables; nel, number of nonzero coefficients in the constraint matrix; and $dens$, constraint matrix density %. We have considered $|\Omega| = 32$, 64 and 128 scenarios. Notice that the number of second stage variables is the same in both representations.

Table 1. Model dimensions

	Compact representation							Splitting variable representation					
Case	m	n_δ	n_x	n_γ	n_y	nel	$dens$	m	n_δ	n_x	nel	$dens$	$\|\Omega\|$
P1	136	4	4	128	128	2112	5.88	640	128	128	4608	1.41	32
P2	148	10	10	128	128	3984	9.75	1408	320	320	17664	1.40	32
P3	202	10	10	640	288	9608	5.02	1152	320	320	17088	0.95	32
P4	394	50	50	640	640	47080	8.66	7424	3200	3200	122880	0.22	64
P5	394	50	50	1280	640	50920	6.40	7424	3200	3200	126720	0.21	64
P6	520	4	4	512	512	8256	1.54	2560	512	512	18432	0.35	128
P7	516	3	7	768	512	10280	1.54	2304	384	896	17920	0.30	128
P8	532	10	10	512	512	14736	2.65	5632	1280	1280	70656	0.35	128
P9	1290	10	10	1280	1280	51400	1.54	5120	1280	1280	81920	0.31	128
P10	532	75	50	512	512	70596	11.55	19072	9600	6400	420096	0.13	128
P11	778	60	70	6400	1280	147220	2.42	18688	7680	8960	345600	0.08	128
P12	712	100	100	512	512	146496	16.81	51712	12800	12800	5277696	0.38	128
P13	778	120	160	6400	1280	263920	4.26	37888	15360	20480	691200	0.04	128

For completeness, Table 2 shows the results obtained by plain use of the optimization engine *COIN-OR*, for solving the (MIP) model and its linear relaxation in the compact and in splitting variable representations. The headings are as follows: z_{LP}, solution value of the LP relaxation of the original problem (1); z_{MIP}, solution value of the original problem (1); GAP, optimality gap defined as $\frac{|z_{MIP} - z_{LP}|}{|z_{LP}|}\%$; T^c_{LP} and T^s_{LP}, elapsed time (s.) for obtaining the LP solution in the compact and splitting variable representations, respectively; T^c_{MIP} and T^s_{MIP}, total elapsed time (s.) to obtain the optimal solution to the original problem by plain use of the optimization engine in $COIN - OR$ in the compact and splitting variable

representations, respectively. And finally, $\overline{z}_{LD}$ denotes the initial upper bound considered at each instance for the four procedures that have been experimented with.

Table 2. Linear and integer solution

Case	z_{LP}	z_{MIP}	GAP	T^c_{LP}	T^s_{LP}	T^c_{MIP}	T^s_{MIP}	$\overline{z}_{LD}$
P1	-81.1408	-80.4820	0.81	0.06	0.11	27.14	365.29	-73.03
P2	-100.4230	-99.8996	0.52	0.11	0.27	95.36	8158.55	-90.38
P3	-548.0340	-547.8620	0.03	0.03	0.05	2.15	5.87	-274.02
P4	-409.4240	-409.3940	0.01	0.11	0.41	0.45	2.85	-204.71
P5	-779.3410	-779.3050	0.00	0.11	0.33	2.03	9.66	-389.67
P6	-1.2568	-0.8032	36.09	0.34	0.60	3.95	23.17	-0.13
P7	-218.9400	-218.9260	0.01	0.07	0.09	0.14	0.35	-109.47
P8	-8.2157	-1.6361	80.09	0.38	0.59	1.64	18.81	-0.82
P9	-418.3500	-418.2420	0.03	0.14	0.23	41.75	156.77	-209.17
P10	-1.6943	-0.9180	45.82	1.89	2.70	5.90	64.99	-0.17
P11	-1391.1200	-1391.0900	0.00	0.34	1.06	207.71	1145.41	-695.56
P12	-16.8730	-3.4486	79.56	4.05	16.96	4.05	1271.34	-1.69
P13	-1762.0300	-1762.0300	0.00	0.58	1.88	4.36	16.25	-881.01

Notice that the GAP is specially small in some of the cases, for example $P4$, $P5$, $P7$, $P11$, and $P13$ whereas it is larger in other cases, as for example $P6$, $P8$, $P10$ or $P12$. In the cases in which the GAP is larger we have considered the upper bound, $\overline{z}_{LD}$ as a 10% of the LP solution value, to ensure that the upper bound of the (MIP) model be higher than the optimal solution value. In particular, Table 3 shows the values of the upper bound.

Table 3. Initial upper bound

Case	P1	P2	P3	P4	P5			
$\overline{z}_{LD}$	$0.9 \cdot z_{LP}$	$0.9 \cdot z_{LP}$	$0.5 \cdot z_{LP}$	$0.5 \cdot z_{LP}$	$0.5 \cdot z_{LP}$			
Case	P6	P7	P8	P9	P10	P11	P12	P13
$\overline{z}_{LD}$	$0.1 \cdot z_{LP}$	$0.5 \cdot z_{LP}$	$0.1 \cdot z_{LP}$	$0.5 \cdot z_{LP}$	$0.1 \cdot z_{LP}$	$0.5 \cdot z_{LP}$	$0.1 \cdot z_{LP}$	$0.5 \cdot z_{LP}$

As it can be seen in Table 2, the splitting variables representation requires much more elapsed time than the compact representation for solving the original integer model and its LP relaxation. So, more computational effort is needed to obtain the upper bound, $\overline{z}_{LD}$, and the Lagrangean initial multipliers, μ^0, when this vector is considered as the simplex multipliers of the nonanticipativity constraints.

Tables 4 and 5 show the main results of our computational experimentation. In these tables we can observe and compare the efficiency of the four procedures for each choice of the initial values of the Lagrangean parameters. Based on this choice, we obtain the Lagrangean bound with each procedure, and present the elapsed time (in seconds) and the number of iterations to obtain the corresponding bound.

The headings are as follows: z_{SUB} optimal Lagrangean bound calculated by the Subgradient method, T_S elapsed time (in seconds) and ite_S number of iterations to compute z_{SUB}, respectively; z_{VOL} optimal Lagrangean bound calculated by the Volume algorithm, T_V elapsed time (in seconds) and ite_V number of iterations to compute z_{VOL}, respectively; z_{PH}

optimal Lagrangean bound calculated by the Progressive Hedging algorithm, T_P elapsed time (in seconds) and ite_P number of iterations to compute z_{PH}, respectively; and finally, z_{DC-CP} optimal Lagrangean bound calculated by the Dynamic Constrained Cutting Plane algorithm, T_D elapsed time (in seconds) and ite_D number of iterations to compute z_{DC-CP}, respectively.

Table 4. Lagrangean bounds ($\mu_0 = 0$)

Case	z_{SUB}	T_S	ite_S	z_{VOL}	T_V	ite_V	z_{PH}	T_P	ite_P	z_{DC-CP}	T_D	ite_D
P1	-80.5198	48.56	69	-80.4905	72.30	103	-80.4854	114.58	161	-81.3907	82.14	178
P2	-99.9302	45.22	52	-99.9273	29.67	36	-99.9066	57.58	67	-101.6100	160.39	240
P3	-547.8920	5.81	26	-547.8620	3.91	17	-547.9540	31.36	150	-547.8870	8.19	36
P4	-409.4180	42.68	35	-409.4200	38.68	33	-409.4130	363.31	209	-411.0390	273.05	175
P5	-779.3330	28.55	49	-779.3050	7.88	13	-779.3250	137.85	238	-779.3130	27.64	45
P6	-0.8417	11.45	7	-0.8073	58.56	41	-0.8187	11.33	7	-1.1278	119.18	94
P7	-218.9260	0.42	0	-218.9260	0.43	0	-218.9260	0.43	0	-218.9260	0.42	0
P8	-1.6897	42.33	22	-1.6461	309.55	160	-1.8123	15.91	8	-4.2338	289.70	187
P9	-418.3090	19.53	27	-418.2670	9.64	13	-418.2820	232.20	336	-418.5160	133.02	183
P10	-0.9180	2.42	0	-0.9180	2.41	0	-0.9180	2.33	0	-0.9180	2.34	0
P11	-1391.1000	63.78	40	-1391.11	36.09	23	-1391.1200	637.87	407	-1391.2400	283.71	168
P12	-3.5950	17714.70	166	-4.5455	1012.00	17	-4.0849	1464.03	22	-16.6972	23197.70	500
P13	-1762.0300	2.20	0	-1762.0300	2.22	0	-1762.0300	2.23	0	-1762.0300	2.21	0

Table 5. Lagrangean bounds (μ_0 =simplex multipliers)

Case	z_{SUB}	T_S	ite_S	z_{VOL}	T_V	ite_V	z_{PH}	T_P	ite_P	z_{DC-CP}	T_D	ite_D
P1	-80.5082	43.94	62	-80.6996	15.59	31	-80.4863	101.28	146	-81.6599	113.94	228
P2	-99.9118	27.15	31	-100.0130	23.79	31	-99.9069	62.37	76	-101.3810	214.59	339
P3	-547.8870	6.73	31	-547.8840	1.74	7	-547.9040	32.88	156	-547.9350	7.65	34
P4	-409.4190	45.65	34	-409.4240	5.24	4	-409.4200	257.64	165	-411.5400	437.57	245
P5	-779.3320	0.54	0	-779.3320	0.54	0	-779.3320	0.53	0	-779.3320	0.53	0
P6	-0.8128	5.91	3	-0.8803	36.74	25	-0.8129	5.86	3	-0.8166	204.70	159
P7	-218.9260	0.41	0	-218.9260	0.43	0	-218.9260	0.42	0	-218.9260	0.41	0
P8	-1.8086	19.49	10	-1.6451	325.07	173	-1.9588	10.49	5	-4.5399	245.84	170
P9	-418.3780	63.43	90	-418.2640	3.43	4	-418.2960	230.17	337	-418.2420	53.32	74
P10	-0.9180	2.30	0	-0.9180	2.31	0	-0.9180	2.31	0	-0.9180	2.31	0
P11	-1391.1900	132.13	83	-1391.1200	62.98	42	-1391.1100	707.31	454	-1391.3900	325.79	198
P12	-4.1695	22220.80	195	-4.6556	880.07	15	-4.0140	25096.10	215	-13.8782	15877.20	327
P13	-1762.0300	2.13	0	-1762.0300	2.15	0	-1762.0300	2.19	0	-1762.0300	2.18	0

Although the elapsed time needed to solve the Lagrangean Decomposition problems with any of the four methods is always higher than the elapsed time needed to obtain the LP bound, the Lagrangean Decomposition bounds are better in general, except for the bounds obtained by the Dynamic Constrained Cutting Plane method in cases $P1$, $P2$, $P4$ and $P11$.

Moreover, the Lagrangean Decomposition bounds coincide with the optimal solution value to the (MIP) problem in some of the cases. In particular, in cases $P7$, $P10$ and $P13$ the optimal solution value is achieved with the four methods in both types of initial vectors of the Lagrangean multipliers. Notice also, that in these cases, the four methods stop after solving the first Lagrangean dual at iteration zero, since the stopping criterion is satisfied. In cases $P3$, $P5$, $P7$ and $P10$ and $P13$ the Lagrangean Decomposition bound obtained with the Volume algorithm taking the zero as initial vector values of Lagrangean multipliers, coincides with the optimal solution value of the (MIP) problem. And, finally, in case $P9$ the Lagrangean Decomposition bound obtained with the Dynamic Constrained Cutting Plane method coincides

with the (MIP) solution value while taking the simplex multipliers as initial vector values of Lagrangean multipliers.

Comparing the performance of the four methods we can say that Dynamic Constrained Cutting Plane method is the less robust. It works specially well in the case $P9$, obtaining the optimal solution to the (MIP) problem but in general, it needs more iterations than the other three methods, specially the Volume and Subgradient algorithms, to reach a worse Lagrangean Decomposition bound. Notice also, that it provides a worse Lagrangean Decomposition bound than the LP solution value for the cases $P1$, $P2$, $P4$ and $P11$. In all of these cases, the procedure is stopped after a sequence of ten consecutive iterations without improving the incumbent objective function value.

If we compare the performance of the Progressive Hedging algorithm against the Subgradient method and the Volume algorithm, we observe that the first one needs more iterations to obtain a similar Lagrangean Decomposition bound value, in most of the cases.

We can also observe in Tables 4 and 5 that the goodness of the Lagrangean Decomposition bounds are very similar. However, it can not be concluded that using the simplex multipliers as initial Lagrangean multipliers improves the Lagrangean Decomposition solution value. We can consider some specific cases, for example Subgradient method in cases $P5$ and $P9$, where the conclusions are not very definitive. On one hand, the Subgradient method in case $P9$ obtains a better Lagrangean Decomposition solution considering the initial Lagrangean multipliers equal to zero, needing also less number of iterations. On the other hand, the same algorithm in case $P5$, provides a better Lagrangean Decomposition bound with the initial Lagrangean multiplier vector equal to the simplex multipliers, also with zero iterations.

Comparing both tables, the total elapsed time to compute the Lagrangean Decomposition solution for the four procedures in some of the cases is higher in Table 5, i.e. when the initial vector of Lagrangean multipliers is taken as the simplex multipliers, but in some other cases this time is higher in Table 4. However, in general the number of iterations is higher in Table 4, specially for the Subgradient and Volume algorithms.

Finally, with respect to the speed of convergence we have observed that the performance of the four methods is totally dependent on the initial upper bound, $\overline{z}_{LD}$, on the initial step size parameter, α_0, and on the parameter which determines the increase or decrease of this step size, along the iterations, i.e., the sequence of iterations without improvement, $\#red$. When we have used an upper bound $\overline{z}_{LD}$ or an initial parameter value of α_0 too small, the step size along the iterations has become too short, and then, more iterations are needed to obtain a small improvement, in anyway. Two possible consequences can appear in this situation. The first one is that the algorithm could reach the maximum number of iterations, without providing a good bound. And the second one, the corresponding method stops, since there is not a considerable improvement in a given number of iterations, what means that the second stopping criterion is satisfied. So, it is necesary to update the step size parameter along the iterations in an optimal way and give an appropriate value to the upper bound and to the initial step size parameter, to ensure the convergence and the good performance of this type of iterative methods.

If the parameter $\#red$ is too high, some of the the algorithms can present a divergent behaviour. They go well at the beginning, the Lagarangean Decomposition approach is improving during some steps until it reach to a point where they start to diverge. This behaviour has

been observed specially in the Progressing Hedging algorithm and the Dynamic Constrained Cutting Plane procedure. To avoid it, the parameter $\#red$ has been considered equal to one in all the procedures.

The choice of these parameters related to the convergence of the procedures is independent of the initial Lagrangean multipliers vector that is considered, either in order to obtain a good Lagrangean Decomposition solution or in order to compare the goodness of the bounds that it provides.

6 Conclusions

Four iterative procedures for solving the μ-parametric dual problems corresponding to the Lagrangean Decomposition of the two stage stochastic mixed 0-1 models have been studied. The models have been introduced by using the splitting variable representation of the deterministic equivalent model, where $0-1$ and continuous variables appear at any stage. The Lagrangean Decomposition has been proposed to satisfy both the integrality constraints for the 0-1 variables and the non-anticipativity constraints. We have compared the performance of four iterative procedure schemes based on dual Lagrangean problems, as the Subgradient method, the Volume algorithm, the Progressive Hedging algorithm and the Dynamic Constrained Cutting Plane method, in terms of the goodness of the Lagrangean Decomposition solution, the number of iterations and the total elapsed time to obtain them. At each iteration of the procedures a μ-parametric mixed 0-1 problem is solved, and the vector of parameters, μ, i.e., the Lagrangean multipliers are updated by using the different methodologies. The performance of all these algorithms has turned out to be very sensitive to the choice of given parameters as the upper bound, the initial step size parameter and the parameter which determines the increase or decrease of this step size along the iterations. We have tested the conditions and properties of convergence for a set of medium and large-scale dimension problems. Based on the testbed that we have used in our experimentation, the Volume algorithm and the traditional Subgradient method are the procedures of choice. As a future work, we are planning to use these procedures for tightening the lower bound of the submodels for actives Twin Node Families in the Branch-and-Fix phase of the algorithmic procedure for solving multistage stochastic mixed 0-1 problems by using our Branch-and-Fix coordination approach, see [9, 12].

References

[1] A. Alonso-Ayuso, L.F. Escudero, A. Garín, M.T. Ortuño and G. Pérez. An approach for strategic supply chain planning based on stochastic 0–1 programming. *Journal of Global Optimization*, 26:97-124, 2003.

[2] A. Alonso-Ayuso, L.F. Escudero and M.T. Ortuño. BFC, a Branch-and-Fix Coordination algorithmic framework for solving some types of stochastic pure and mixed 0-1 programs. *European Journal of Operational Research*, 151:503–519, 2003.

[3] F. Barahona and R. Anbil. The Volume algorithm: Producing primal solutions with a subgradient methhod. *Mathematical Programming*, 87:385–399, 2000.

[4] D.P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982.

[5] J.R. Birge and F.V. Louveaux. *Introduction to Stochastic Programming*. Springer, 1997.

[6] C.C. Carøe and J. Tind. L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming*, 83:451–464, 1998.

[7] C.C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24:37–45, 1999.

[8] S. Engell, A. Märkert, G. Sand and R. Schultz. Agregated scheduling of a multiproduct batch plant by two-stage stochastic integer programming. Optimization and Engineering, 5: 335-359, 2004.

[9] L.F. Escudero On a mixture of the fix-and-relax coordination and Lagrangian substitution schemes for multistage stochastic mixed integer programming. *TOP*, 17:5–29, 2009.

[10] L.F. Escudero, A. Garín, M. Merino and G. Pérez. A two-stage stochastic integer programming approach as a mixture of Branch-and-Fix Coordination and Benders Decomposition schemes. *Annals of Operations Research*, 152:395–420, 2007.

[11] L.F. Escudero, A. Garín, M. Merino and G. Pérez. A general algorithm for solving two-stage stochastic mixed 0-1 first stage problems. *Computers and Operations Research*, 36:2590-2600, 2009.

[12] L.F. Escudero, A. Garín, M. Merino and G. Pérez. On BFC-MSMIP strategies for scenario cluster partitioning, Twin Node Family branching selection and bounding for multistage stochastic mixed integer programming. *Computers and Operations Research*, 37:738-753, 2010.

[13] L.F. Escudero, A. Garín, M. Merino and G. Pérez. An exact algorithm for solving large-scale two-stage stochastic mixed integer problems: some theoretical and experimental aspects *European Journal of Operational Research*, 204:105-116, 2010.

[14] D. Li and X. Sun. *Nonlinear Integer Programming*. Springer, 2006.

[15] A.M. Geoffrion. Lagrangean relaxation for integer programming. *Mathematical Programming Studies*, 2:82-114, 1974.

[16] M. Guignard, Lagrangean relaxation. *TOP*, 11:151–228, 2003.

[17] M. Guignard and S. Kim. Lagrangean decomposition. A model yielding stronger Lagrangean bounds. *Mathematical Programming*, 39:215-228, 1987.

[18] M. Held and R.M. Karp. The traveling salesman problem and minimum spanning trees: part II. *Mathematical Programming*, 1:6-25, 1971.

[19] M. Held, P. Wolfe and H. Crowder. Validation of subgradient optimization. *Mathematical Programming*, 6:62-88, 1974.

[20] R. Hemmecke and R. Schultz. Decomposition methods for two-stage stochastic Integer Programs. In M. Grötschel, S.O. Krumke and J. Rambau, editors. *Online Optimization of Large Scale Systems*. Springer, 601–622, 2001.

[21] INFORMS. COIN-OR: COmputational INfrastructure for Operations Research. www.coin-or.org 2008.

[22] N. Jimenez Redondo and A.J. Conejo. Short-term hydro-thermal coordination by Lagrangean relaxation: solution of the dual problem. *IEEE Transactions on Power Systems*, 14:89–95, 1997.

[23] K.C. Kiwiel. Proximity control in bundle methods for convex nondifferentiable minimization. *Mathematical Programming*, 46:15–122, 1990.

[24] Klein Haneveld, W. and van der Vlerk Kang, M. Stochastic integer programming: General models and algorithms. Annals of Operations Research 1999;85; 39-57.

[25] G. Laporte and F.V. Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13:133–142, 1993.

[26] G. Laporte and F.V. Louveaux. An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research*, 50:415-423, 2002.

[27] J.M. Mulvey and A. Ruszczynski. A diagonal quadratic approximation method for large-scale linear programs. *Operations Research Letters*, 12:205–221, 1992.

[28] L. Ntaimo and S. Sen. The million variable 'march' for stochastic combinatorial optimization. *Journal of Global Optimization* 32:385–400, 2005.

[29] G. Pérez and M.A. Garín. On downloading and using $COIN - OR$ for solving linear/integer optimization problems. *Working paper series Biltoki. DT.2010.05*, 2010.

[30] B.T. Polyak. *Introduction to Optimization Software*. 1987.

[31] R.T. Rockafellar and R.J-B Wets. Scenario and policy aggregation in optimisation under uncertainty. *Mathematics of Operations Research*, 16:119–147, 1991.

[32] A. Ruszczynski. An augmented Lagrangian decomposition for block diagonal programming problems. *Operations Research Letters*, 8:287–294, 1989.

[33] R. Schultz. Stochastic programming with integer variables. *Mathematical Programming*, Ser. B 97:285-309, 2003.

[34] S. Sen and H.D. Sherali. Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. *Mathematical Programming*, Series A 106:203–223, 2006.

[35] S. Sen and J.L. Higle. The C3 theorem and a D2 algorithm for large scale stochastic mixed-integer programming: Set convexification. *Mathematical Programming*, Series A 104:1–20, 2005.

[36] H.D. Sherali and J.C. Smith. Two-stage hierarchical multiple risk problems: Models and algorithms. *Mathematical Programming*, Series A 120:403-427, 2009.

[37] H.D. Sherali and X. Zhu. On solving discrete two stage stochastic programs having mixed-integer first and second stage variables. *Mathematical Programming*, Series A 108:597–611, 2006.

[38] S. Takriti and J.R. Birge. Lagrangean solution techniques and bounds for loosely coupled mixed-integer stochastic programs. *Operations Research*, 48:91–98, 2000.

[39] J. Till, G. Sand, M. Urselmann and S. Engell. A hybrid evolutionary algorithm for solving two-stage stochastic integer programs in chemical batch scheduling. *Computers & Chemical Engineering*, 31: 630-647, 2007.