Computers & Operations Research 36 (2009) 2590 – 2600

Contents lists available at ScienceDirect

Computers & Operations Research

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

A general algorithm for solving two-stage stochastic mixed 0–1 first-stage problems

L.F. Escudero[a], M.A. Garín[b,*], M. Merino[c], G. Pérez[c]

[a]Dpto. de Estadística e Investigación Operativa, Universidad Rey Juan Carlos, Móstoles, Madrid, Spain
[b]Dpto. de Economía Aplicada III, Fac. CC. Económicas y Empresariares, UPV/EHU, Universidad del País Vasco, Avd. Lehendakari Aguirre 83, 48015 Bilbao, Vizcaya, Spain
[c]Dpto. de Matemática Aplicada, Estadística e Investigación Operativa, Universidad del País Vasco, Leioa, Vizcaya, Spain

ARTICLE INFO

Available online 6 December 2008

Keywords:
Two-stage stochastic integer programming
Benders decomposition
Nonanticipativity constraints
Splitting variables
Twin node family
Branch-and-fix coordination

ABSTRACT

We present an algorithmic approach for solving large-scale two-stage stochastic problems having mixed 0–1 first stage variables. The constraints in the first stage of the deterministic equivalent model have 0–1 variables and continuous variables, while the constraints in the second stage have only continuous. The approach uses the *twin node family* concept within the algorithmic framework, the so-called *branch-and-fix coordination*, in order to satisfy the *nonanticipativity* constraints. At the same time we consider a scenario cluster Benders decomposition scheme for solving large-scale *LP* submodels given at each *TNF* integer set. Some computational results are presented to demonstrate the efficiency of the proposed approach.

1. Introduction

Very frequently, mainly for optimization problems with a given time horizon to be exploited, some coefficients in the objective function, in the right-hand side (rhs) vector and, to a lesser extent in the constraint matrix, are not known with certainty when decisions are to be made. These circumstances allow the use of stochastic programs with recourse.

Computation in stochastic programs with recourse has focused on two-stage problems, since they reflect the simplest mode of interplaying decision and information. The uncertain parameters are random variables on some probability space $(\Omega', \mathscr{A}', \mathscr{P}')$, with Ω', $\mathscr{A}'$, and $\mathscr{P}'$, respectively, denoting the set of all outcomes, a collection of random variables, and the assigned probabilities. Without the loss of generality, a finite number of scenarios, Ω, is often considered based on some discretization of the realization of $\omega' \in \Omega'$, each with an associated probability of occurrence w^ω, $\omega \in \Omega$.

In the general formulation of a two-stage program, decisions on the first and second stage variables have to be made stage-wise. First-stage variables are selected before observing the realization of uncertain parameters. After having decided on first stage and having observed each realization of uncertain parameters, the second stage (or recourse) decision has to be made. The first stage corresponds to decisions to be made without anticipating of some of the problem data, i.e., first-stage variables take the same value in each scenario (*nonanticipativity* constraints).

When a finite number of scenarios is considered, a general two-stage program can be expressed in terms of the first-stage decision variables being equivalent to a large, dual block-angular programming problem, introduced in Wets [1] and known as *deterministic equivalent model* (DEM).

Sometimes a general two-stage linear problem adds the condition that some variables, in either the first stage or the second stage should be integer. In many practical situations the restrictions are in fact, that the variables must be binary, i.e., they can only take the value 0 or 1.

The simplest form of two-stage stochastic integer programs contains first-stage pure 0–1 variables and second stage continuous variables. Laporte and Louveaux [2] apply a branch-and-cut procedure for such problems, based on the Benders decomposition (BD) method, see Benders [3]. Alonso-Ayuso et al. [4], provide an efficient branch-and-fix coordination (BFC) methodology for solving two types of stochastic 0–1 problems, namely, mixed 0–1 problems for two-stage environments, where the first stage has only 0–1 variables, and pure 0–1 problems for multistage environments, where uncertainty appears only in the objective function coefficients and in the *rhs*. This methodology is used by Alonso-Ayuso et al. [5,6] for solving such a model in production planning applications.

Ahmed et al. [7] develop a branch-and-bound solution approach for stochastic programs having a fixed technology matrix and general first stage and pure integer recourse variables. Carøe and Tind [8] generalize the BD to deal with stochastic programs having 0–1 mixed-integer recourse variables and either pure continuous or pure first-stage 0–1 variables.

* Corresponding author. Tel.: +34 94 601 3734; fax: +34 94 601 3754.
E-mail addresses: laureano.escudero@urjc.es (L.F. Escudero), mariaaraceli.garin@ehu.es (M.A. Garín), maria.merino@ehu.es (M. Merino), gloria.perez@ehu.es (G. Pérez).

0305-0548/$ - see front matter
doi:10.1016/j.cor.2008.11.011

When the first stage contains pure 0–1 variables, finite termination is readily justified by adopting search procedures that branch over the 0–1 first-stage variables. Sherali and Fraticelli [9], Sen and Higle [10], and Ntaimo and Sen [11] propose 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. Sen and Sherali [12] and Sherali and Zhu [13] propose a similar branch-and-cut propose a decomposition approach, where a modified BD method is developed. Carøe and Schultz [14] and Hemmecke and Schultz [15] design a branch-and-bound algorithm for problems having mixed-integer variables in both stages. However their approach focuses more on using Lagrangian relaxation to obtain good bounds, and less on branching and variable fixing. They obtain lower and upper bounds but they are not seeking the optimal solution. They can estimate the feasible solutions to be within a percentage of the optimum. Moreover, randomness occurs only in the rhs of the second-stage problem. With this same stochastic structure, Takriti and Birge [16] also use Lagrangean relaxation but instead of updating the Lagrange multipliers via traditional methods, they use the progressive hedging algorithm of Rockafellar and Wets [17].

In this paper we study general two-stage stochastic mixed 0–1 problems, where 0–1 variables and continuous variables have nonzero elements in the first stage and there are continuous variables in the second stage. The continuous variables do not need to be bounded. Furthermore, the stochasticity, discretized into a finite set of scenarios, can appear anywhere in the model. We propose an algorithmic approach based on a specialization of the BFC scheme. Notice that the original *BFC* only considers 0–1 variables in the first-stage constraints, the set of variables above where the branching procedure is developed. The relaxation of the nonanticipativity constraints of the first-stage variables in the DEM allows for the independent solution of the so-called mixed 0–1 scenario *cluster*-related problems. The nonanticipativity constraints of the first-stage 0–1 variables are satisfied by using a scheme that is based on the *twin node family* (*TNF*), concept introduced in Alonso-Ayuso et al. [4]. The scheme is specifically designed for coordinating the node branching selection and pruning and the 0–1 variable branching selection and fixing at each *branch-and-fix* (*BF*) tree.

As previously mentioned, we consider 0–1 and continuous first-stage variables, such that the BFC scheme just forces us to satisfy *nonanticipativity* for the 0–1 variables. In addition, the proposed approach considers the *compact* representation of the *DEM* at each *TNF integer set*. By fixing the 0–1 variables to the nodes values, the *DEM* has only continuous variables. In order to satisfy the nonanticipativity constraints also on the first-stage continuous variables, we need to solve two linear submodels of the *DEM*, for the given *TNF* integer set. The optimal values of those submodels, guarantee that our approach finds the optimal solution or determines infeasibility in the original two-stage stochastic instance. In order to increase the efficiency of our approach for solving large-scale instances we exploit the remaining model's structure, such that a BD is used to solve linear submodels in several steps of the procedure. The conditions for pruning a *TNF* are also stated. Another contribution of the paper is the decomposition of the set of scenarios in clusters, not only for the general scheme of the *BFC–TSMIP* procedure but also for the BD. Some computational experience is reported to compare the quality of the solution obtained by our approach and the solution obtained by solving the *DEM* by the plain use of a state-of-the-art optimization engine. The proposed approach compares favorably.

The remainder of the paper is organized as follows. Section 2 presents the mixed 0–1 *DEM*. Section 3 presents the *TNF* based *BFC* algorithmic framework of the proposed approach for problem solving. Section 4 presents the models to be solved for obtaining the LP optimal solution at the *TNF integer sets*. Section 5 introduces the *BFC* implementation. An illustrative case is included in Section 6. Section 7 reports on the computational results. Section 8 concludes.

2. Mixed 0–1 DEM

Let us consider the general two-stage deterministic mixed 0–1 model, having 0–1 and continuous first-stage variables

$$
\begin{aligned}
\min \quad & c_1^{\mathrm{T}}\delta + c_2^{\mathrm{T}}x + q^{\mathrm{T}}y \\
\text{s.t.} \quad & b_1 \leqslant A\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant b_2, \\
& h_{01} \leqslant T_0\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant h_{02}, \\
& h_1 \leqslant T\begin{pmatrix}\delta\\ x\end{pmatrix} + Wy \leqslant h_2, \\
& x,\ y \geqslant 0, \\
& \delta \in \{0,1\}^n,
\end{aligned} \tag{1}
$$

where c_1, c_2 and q are the vectors of the objective function coefficients for the first-stage variables vectors δ and x, and the second stage variables vector y, respectively; and b_1, b_2, and h_{01} and h_{02} are the left-hand side (lhs) and rhs vectors for the first stage of constraint block, respectively. Additionally, A and T_0, are the matrices of the first-stage constraint block; T and W are the constraints matrices of the second stage constraint block and h_1 and h_2 are the corresponding lhs and rhs, respectively. In a general purpose problem (1), the lhs b_1, h_{01} or h_2 can take the value $-\infty$, the rhs b_2, h_{02} or h_2 can take the value ∞ , or we may find the case where some of the constraints satisfy with equality. The vector δ has n 0–1 variables, and the vectors x and y have continuous variables.

Let us assume that in the general stochastic setting, some of the coefficients in the vectors q, h_{01}, h_{02}, h_1, h_2 and the matrices T_0, T, W are uncertain, but the uncertainty is represented by the scenarios ω from the finite set, say, Ω, each with an associated probability of occurrence w^ω, $\omega \in \Omega$. Observe that the vectors h_{01}, h_{02} and the matrix T_0, can include uncertain coefficients, but all of them correspond to the block of first-stage constraints. So, the stochastic version of the two stage problem (1) can be represented by the so-called DEM that in the *compact* representation has the following structure:

$$
\begin{aligned}
z_{MIP} = \min \quad & c_1^{\mathrm{T}}\delta + c_2^{\mathrm{T}}x + \sum_{\omega\in\Omega} w^\omega q^{\omega\mathrm{T}} y^\omega \\
\text{s.t.} \quad & b_1 \leqslant A\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant b_2, \\
& h_{01}^\omega \leqslant T_0^\omega\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant h_{02}^\omega, \quad \omega\in\Omega, \\
& h_1^\omega \leqslant T^\omega\begin{pmatrix}\delta\\ x\end{pmatrix} + W^\omega y^\omega \leqslant h_2^\omega, \quad \omega\in\Omega, \\
& x, y^\omega \geqslant 0, \quad \omega\in\Omega, \\
& \delta \in \{0,1\}^n.
\end{aligned} \tag{2}
$$

The *compact* representation of *DEM* (2) can be transformed into a *splitting variable* representation, such that the variables vectors δ and x are replaced with δ^ω and x^ω, respectively, $\forall \omega\in\Omega$. So, there is a model for each scenario $\omega\in\Omega$, but they are linked by the so-called *nonanticipativity* constraints

$$\delta^\omega - \delta^{\omega'} = 0, \tag{3}$$

$$x^\omega - x^{\omega'} = 0, \tag{4}$$

$\forall\omega,\omega'\in\Omega:\omega\neq\omega'$. Notice that constraints (3) and (4) force the equality of the values of the first-stage variables.

Then, the *splitting variable* representation is as follows:

$$(MIP): z_{MIP} = \min \sum_{\omega\in\Omega} w^{\omega}(c_1^{\mathrm{T}}\delta^{\omega} + c_2^{\mathrm{T}}x^{\omega} + q^{\omega\mathrm{T}}y^{\omega})$$

$$\text{s.t.}\quad b_1 \leqslant A\begin{pmatrix}\delta^{\omega}\\ x^{\omega}\end{pmatrix} \leqslant b_2,\quad \omega\in\Omega,$$

$$h_{01}^{\omega} \leqslant T_0^{\omega}\begin{pmatrix}\delta^{\omega}\\ x^{\omega}\end{pmatrix} \leqslant h_{02}^{\omega},\quad \omega\in\Omega,$$

$$h_1^{\omega} \leqslant T^{\omega}\begin{pmatrix}\delta^{\omega}\\ x^{\omega}\end{pmatrix} + W^{\omega}y^{\omega} \leqslant h_2^{\omega},\quad \omega\in\Omega,$$

$$x^{\omega}, y^{\omega} \geqslant 0,\quad \omega\in\Omega,$$

$$\delta^{\omega}\in\{0,1\}^n,\quad \omega\in\Omega,$$

$$x^{\omega} - x^{\omega'} = 0\quad \forall\omega,\omega'\in\Omega:\omega\neq\omega',$$

$$\delta^{\omega} - \delta^{\omega'} = 0\quad \forall\omega,\omega'\in\Omega:\omega\neq\omega'. \tag{5}$$

Notice that the dualization (or the relaxation) of constraints (3) and (4) from model *MIP* (5) results in $|\Omega|$ independent mixed 0–1 models. For solving the original model, we propose the use of a *BFC* scheme for each of the scenario-related models to ensure the integrality condition on the δ-variables, such that the *nonanticipativity* constraints (3) are satisfied while selecting the branching nodes and the branching variables. To this end, the so-called *TNF* concept is used. Additionally, the proposed approach optimizes the linear submodel that results from model *MIP* (5) at each *TNF* integer set, so that the *nonanticipativity* constraints (4) are also satisfied, see below.

3. BFC algorithmic framework

The scenario-related model for $\omega\in\Omega$ that results from ignoring the *nonanticipativity* constraints (3) and (4) in model *MIP* (5) can be expressed as

$$(MIP^{\omega}): z_{MIP}^{\omega} = \min w^{\omega}(c_1^{\mathrm{T}}\delta^{\omega} + c_2^{\mathrm{T}}x^{\omega} + q^{\omega\mathrm{T}}y^{\omega})$$

$$\text{s.t.}\quad b_1 \leqslant A\begin{pmatrix}\delta^{\omega}\\ x^{\omega}\end{pmatrix} \leqslant b_2,$$

$$h_{01}^{\omega} \leqslant T_0^{\omega}\begin{pmatrix}\delta^{\omega}\\ x^{\omega}\end{pmatrix} \leqslant h_{02}^{\omega},$$

$$h_1^{\omega} \leqslant T^{\omega}\begin{pmatrix}\delta^{\omega}\\ x^{\omega}\end{pmatrix} + W^{\omega}y^{\omega} \leqslant h_2^{\omega},$$

$$x^{\omega}, y^{\omega} \geqslant 0,$$

$$\delta^{\omega}\in\{0,1\}^n. \tag{6}$$

Instead of obtaining independently the optimal solution to the programs MIP^{ω} (6), we propose a specialization of the *BFC* approach.

It is specially designed to coordinate the selection of the branching node and branching variable for each scenario-related *BF* tree, such that the relaxed constraints (3) are satisfied when fixing the appropriate variables to either 1 or 0.

The approach proceeds by branching on the δ-variables, coordinating the execution of the linear submodels under the scenarios. In minimization problems, as those defined in this work, it computes a chain of lower bounds, say $\underline{Z}_i$, where $\underline{Z}_i = \sum_{\omega=1}^{|\Omega|} z_i^{\omega}$, and z_i^{ω} denotes the solution to the linear relaxation of MIP^{ω} model (6), where the first i δ-variables have been fixed to 0 or 1, and where $\underline{Z}_0$ denotes the solution value of the model associated with the root node, say, $i=0$, that can be calculated by solving the linear relaxation of the original problem, Z_{LP}, or alternatively as $\underline{Z}_0 = \sum_{\omega=1}^{|\Omega|} z_0^{\omega}$, although the solution value has not to be the same.

In the case where the optimal solution that has been obtained in each node has 0–1 values and it satisfies the *nonanticipativity* constraints for all the δ-variables, any of two following situations have occurred:

1. The *nonanticipativity* constraints (4) have been satisfied and, then a new solution has been found for the original stochastic mixed 0–1 program. The related incumbent solution can be updated and in any case, the *TNF* is pruned. The optimality of the incumbent solution has been proved, if the sets of active nodes are empty.
2. The *nonanticipativity* constraints (4) have not been satisfied. In this case, we must optimize the *LP* submodel resulting from fixing the δ-variables in the model to the branched on/been fixed at values in the integer *TNF* whose associated models have been optimized. If the *LP* model is feasible the incumbent solution is updated and if the *TNF* cannot be pruned, continue with the tree examination.

See similar decomposition approaches in Carøe and Schultz [14], Hemmecke and Schultz [15], Klein Haneveld and van der Vlerk Kang [18], Nowak et al. [19] and Romisch and Schultz [20], among others. However, those approaches focus more on using a Lagrangian relaxation of the constraints (3) to obtain good lower bounds, and less on branching and variable fixing. In any case, Lagrangian relaxation and BD schemes can be added to the *BFC* procedure. See also Schultz [21].

For the specialization of the *BFC* approach to solving problem *MIP* (5), let $\mathscr{R}^{\omega}$ denote the *BF* tree associated with scenario ω, and $\mathscr{G}^{\omega}$ the set of active nodes in $\mathscr{R}^{\omega}$, $\omega\in\Omega$. Any two active nodes, say $g\in\mathscr{G}^{\omega}$ and $g'\in\mathscr{G}^{\omega'}$ are called *twin* nodes if either they are the *root* nodes or the paths from the *root* nodes to each of them in their own *BF* trees $\mathscr{R}^{\omega}$ and $\mathscr{R}^{\omega'}$, respectively, have branched on/been fixed at the same 0–1 values for the variables in δ^{ω} and $\delta^{\omega'}$, for $\omega,\omega'\in\Omega$. A *TNF*, say, $\mathscr{H}_f$ is a set of nodes, such that any one is a *twin* node to all the other members of the family, for $f\in\mathscr{F}$, where $\mathscr{F}$ is the set of *TNFs*. Notice that $g,g'\in\mathscr{H}_f$ for any family $f\in\mathscr{F}$ implies that $\omega\neq\omega'$ for $g\in\mathscr{G}^{\omega}$ and $g'\in\mathscr{G}^{\omega'}$, $\omega,\omega'\in\Omega$. A *TNF integer set* is a set of integer *BF* nodes, one per *BF* tree, where the *nonanticipativity* constraints (3) of the 0–1 variables are satisfied.

Let us consider the scenario tree and the *BF*trees shown in Fig. 1, which correspond to the illustrative example given in Section 6. Some of the TNF are $\mathscr{H}_0=\{0^1,0^2\}$, $\mathscr{H}_1=\{1^1,1^2\}$, $\mathscr{H}_2=\{2^1,2^2\}$, $\mathscr{H}_3=\{3^1,3^2\}$, $\mathscr{H}_4=\{4^1,4^2\}$, $\mathscr{H}_9=\{9^1,9^2\}$ and $\mathscr{H}_{10}=\{10^1,10^2\}$. At least, the last four families are branched *TNF*integer sets, since all of the 0–1 variables δ have been branched on 0 or 1. There can be more *TNF*integer sets; for example $\mathscr{H}_2=\{2^1,2^2\}$, if the δ_3^{ω} solutions from LP^{ω} at node 2 are binary and such that $\delta_3^1=\delta_3^2$. Notice that the first *TNF*to be used is $\mathscr{H}_0=\{0^1,0^2\}$. Based on the *LP*optimal solution of the scenario related models attached to the nodes in $\mathscr{H}_0$, let us assume that the selected branching variable is δ_1 and, so, the new TNF $\mathscr{H}_1=\{1^1,1^2\}$ and $\mathscr{H}_6=\{6^1,6^2\}$ are created and so forth. The *BFC* algorithm proceeds by branching on the δ-variables, along the *TNF* in the order in which appear in the figure; they can give rise to new *TNF* integer sets, if the corresponding not branching on δ-variables, take the values 0 or 1.

It is clear that the relaxation of the *nonanticipativity* constraints (3) and (4) is not required for all pairs of scenarios in order to gain computational efficiency. For a brief study of the relevance of scenario cluster number choice, see Escudero et al. [22]. So, the number of scenarios to consider in a given model basically depends on

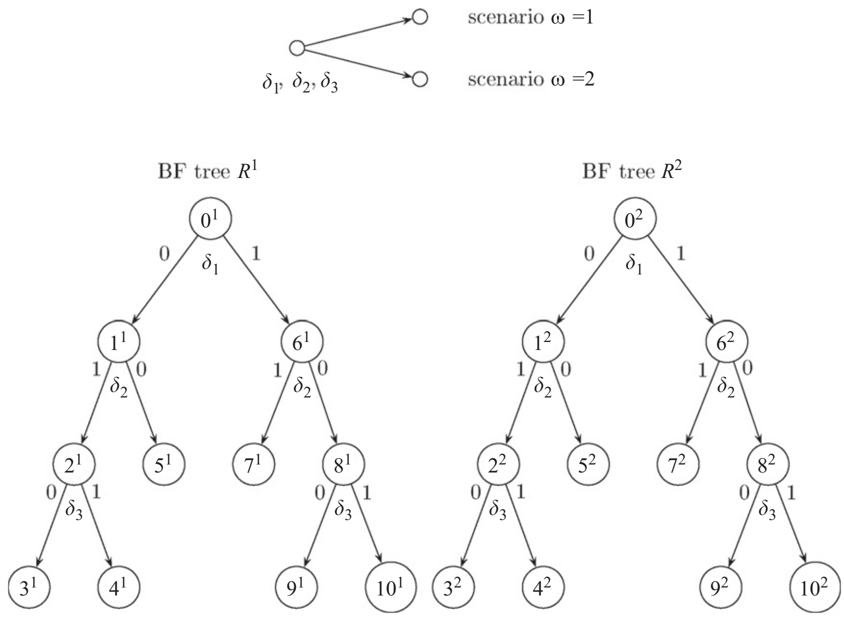


Fig. 1. Twin Node Families (TNF).

the dimensions of the scenario related model MIP^{ω} (6). The criterion for scenario clustering in the sets, say, $\Omega^1,\dots,\Omega^{\hat{p}}$, where $\hat{p}$ is the number of *clusters* to consider, could alternatively be based on the smallest internal deviation of the uncertain parameter, the greatest deviation, etc. The determination of the most efficient criterion is instance dependent. In any case, notice that $\Omega^p\cap\Omega^{p'}=\emptyset, p,p'=1,\dots,\hat{p}$: $p\neq p'$ and $\Omega=\cup_{p=1}^{\hat{p}}\Omega^p$. By abusing the notation slightly, the problem to consider for the scenario *cluster* $p=1,\dots,\hat{p}$ can be expressed as follows:

$$(MIP^p): z^p_{MIP}=\min\sum_{\omega\in\Omega^p} w^{\omega}(c_1^{\mathrm{T}}\delta^p+c_2^{\mathrm{T}}x^p+q^{\omega\mathrm{T}}y^{\omega})$$

$$\begin{aligned}\text{s.t.}\quad & b_1\leqslant A\begin{pmatrix}\delta^p\\ x^p\end{pmatrix}\leqslant b_2,\\ & h^{\omega}_{01}\leqslant T^{\omega}_0\begin{pmatrix}\delta^p\\ x^p\end{pmatrix}\leqslant h^{\omega}_{02},\quad \forall\omega\in\Omega^p,\\ & h^{\omega}_{1}\leqslant T^{\omega}\begin{pmatrix}\delta^p\\ x^p\end{pmatrix}+W^{\omega}y^{\omega}\leqslant h^{\omega}_{2},\quad \forall\omega\in\Omega^p,\\ & x^p,y^{\omega}\geqslant 0,\quad \forall\omega\in\Omega^p,\\ & \delta^p\in\{0,1\}^n. \end{aligned}\tag{7}$$

The $\hat{p}$ problems MIP^p (7) are linked by the *nonanticipativity* constraints

$$\delta^p_i-\delta^{p'}_i=0\quad\forall i\in\mathscr{I},\tag{8}$$

$$x^p_j-x^{p'}_j=0\quad\forall j\in\mathscr{J},\tag{9}$$

where $\mathscr{I}$ and $\mathscr{J}$ denote the sets of variables in the δ-and x-vectors, respectively, $n=|\mathscr{I}|$, and $p,p'=1,\dots,\hat{p}$: $p\neq p'$. We will denote LP^p to the linear relaxation of MIP^p (7), i.e. with $\delta^p\in[0,1]^n$.

3.1. LP submodels to be solved at the TNF integer sets

Notice that we have defined a *TNF integer set* as a set of integer *BF* nodes, one per *BF* tree, where the *nonanticipativity* constraints (3) of the 0–1 variables are satisfied. At each *TNF integer set* can be defined two linear submodels.

Let the linear model LP^{TNF} (10) that results after fixing in problem *DEM* (2) all the δ-variables at the 0–1 related values for a given *TNF* integer set. In the new model, $\overline{\delta}$ will denote the 0–1 values of the respective vector δ.

$$(LP^{TNF}): z^{TNF}_{LP}=c_1^{\mathrm{T}}\overline{\delta}+\min c_2^{\mathrm{T}}x+\sum_{\omega\in\Omega} w^{\omega}q^{\omega\mathrm{T}}y^{\omega}$$

$$\begin{aligned}\text{s.t.}\quad & b_1\leqslant A\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix}\leqslant b_2,\\ & h^{\omega}_{01}\leqslant T^{\omega}_0\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix}\leqslant h^{\omega}_{02},\quad \omega\in\Omega,\\ & h^{\omega}_{1}\leqslant T^{\omega}\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix}+W^{\omega}y^{\omega}\leqslant h^{\omega}_{2},\quad \omega\in\Omega,\\ & x,y^{\omega}\geqslant 0,\quad \omega\in\Omega. \end{aligned}\tag{10}$$

The model LP^{TNF} (10) needs to be solved since the coordinated branching in the *BFC* algorithm is only designed for the nonanticipativity constraints (8) on the δ-variables. However, the nonanticipativity constraints (9) on the x-variables must also be satisfied in any feasible solution to the original problem *MIP* (5). The model is solved in Step 6 of the specialization of the *BFC* algorithm that we propose in Section 5.

The second linear submodel to solve at a *TNF* integer set, corresponds to the case in which not all the δ-variables have been branched on in the current *TNF*, but in the corresponding optimal solution to LP^p, all of them have taken the same 0–1 values. Let the linear model LP^f that results when the δ-variables that are not yet branched on/been fixed at in the current *TNF* can take fractional values. Let $\delta=\binom{\overline{\delta}}{\delta^f}$ denote the vector of δ-variables composed by the subset of 0–1 values, $\overline{\delta}$, of the branched on δ-variables and the subset of these variables which have not yet been branched on/fixed at, δ^f, in a given iteration of the *BFC* algorithm. Notice that $0\leqslant\delta^f\leqslant 1$.

The model can be expressed as

$$(LP^f): z^f_{LP} = \min c_1^{\mathrm{T}}\delta + c_2^{\mathrm{T}}x + \sum_{\omega\in\Omega} w^\omega q^{\omega\mathrm{T}} y^\omega$$

$$\text{s.t.}\quad b_1 \leqslant A\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant b_2,$$

$$h^\omega_{01} \leqslant T^\omega_0\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant h^\omega_{02},\quad \omega\in\Omega,$$

$$h^\omega_1 \leqslant T^\omega\begin{pmatrix}\delta\\ x\end{pmatrix} + W^\omega y^\omega \leqslant h^\omega_2,\quad \omega\in\Omega,$$

$$x, y^\omega \geqslant 0,\quad \omega\in\Omega,$$

$$0 \leqslant \delta^f \leqslant 1. \tag{11}$$

The model LP^f (11) needs to be solved for obtaining strong lower bounds of the solution value of the best descendant nodes from a given node by additionally satisfying the nonanticipativity constraints (9) on the x–variables. In this case, the δ^f-variables that have not yet been branched on/been fixed at the current iteration of the *BFC* algorithm are allowed to be fractional. The model is solved in Step 6 of the specialization of the *BFC* algorithm that we propose in Section 5.

3.2. Branching procedure and pruning strategies

The *BFC* algorithm that we propose proceeds by branching on the n δ-variables along the scenario-cluster related *BF* trees, coordinating the satisfaction of integrality and nonanticipativity constraints for all the *TNF*s.

At each *BFC* node $i \in \{0,1,2,\ldots,n\}$, the lower bound $\underline{Z}_i = \sum_{p=1}^{\hat{p}} z^p_i$ is computed, where z^p_i denotes the solution of the LP^p_i model, in which the first i δ-variables have already been fixed to 0 or 1, $\overline{\delta}^p_j$, $j=1,\ldots,i$, in the current iteration of the algorithm, i.e., the solution to the following problem:

$$(LP^p_i): LP^p$$

$$\text{s.t.}\quad \delta^p_j = \overline{\delta}^p_j \quad \forall j\in\{1,\ldots,i\},$$

$$\delta^p_j \in [0,1] \quad \forall j\in\{i+1,\ldots,n\}. \tag{12}$$

If some of the relaxed integrality and nonanticipativity constraints for δ-variables are not satisfied, the branching continues deep until all δ-variables have the same 0–1 values. Otherwise, the satisfaction of the relaxed nonanticipativity constraints for x-variables is tested. If they are satisfied, the upper bound is updated, $\overline{Z} = \underline{Z}_i$. If not, the following LP^{TNF}_i problem is solved:

$$(LP^{TNF}_i):\quad LP^{TNF}$$

$$\text{s.t.}\quad \delta_j = \overline{\delta}_j \quad \forall j\in\{1,\ldots,i\}\ \text{(from } i \text{ th node in the branch)},$$

$$\delta_j = \overline{\delta}_j \quad \forall j\in\{i+1,\ldots,n\}\ \text{(from solution of } LP^p_i) \tag{13}$$

and, the upper bound is updated, if it is necessary, that is, $\overline{Z} = \min\{z^{TNF}_i, \overline{Z}\}$.

Notice that the solution to be obtained by solving the linear model LP^{TNF}_i attached to a *TNF* integer set could be the *incumbent* solution, since all the δ- and x-variables satisfy the *nonanticipativity* constraints and the δ-variables also satisfy integrality. However, it does not necessarily mean that it should be pruned, except if all 0–1 variables have been branched on for the family, i.e., $i=n$. Otherwise, a better solution can still be obtained by branching on the non-yet branched on 0–1 variables. Indeed, notice that for any *BFC* node i it results that, $\underline{Z}_i \leqslant z^{TNF}_i$ and $z^{TNF}_i \geqslant z^f_i$. Recall that z^{TNF}_i is the solution value in (13) that satisfies the *nonanticipativity* constraints (4) by fixing the δ-variables to their 0–1 values, where the constraints (3) are already satisfied. The family can be pruned if $z^{TNF}_i = z^f_i$, where z^f_i is the solution value of model LP^f_i, where both constraint types (3) and (4) are satisfied, but the non-yet branched on δ-variables are allowed to take fractional values, i.e., the solution of the following problem:

$$(LP^f_i):\quad LP^f$$

$$\text{s.t.}\quad \delta_j = \overline{\delta}_j \quad \forall j\in\{1,\ldots,i\},$$

$$\delta_j \in [0,1] \quad \forall j\in\{i+1,\ldots,n\}. \tag{14}$$

In this case, there is no better solution than z^{TNF}_i to be obtained from the descendant *TNF* integer sets. Hence, if $z^{TNF}_i > z^f_i$ and $z^f_i < \overline{Z}$, the branching follows deep to the $i+1$-th node, because it is possible to find a better feasible solution in the tree.

Finally, the conditions for pruning a branch in the *BFC* procedure that we propose can be enumerated as follows:

(i) After computing $\underline{Z}_i$:
- The linear scenario-*cluster* model LP^p_i attached to a given node member is infeasible, for any $p=1,\ldots,\hat{p}$.
- The lower bound $\underline{Z}_i$ is not better than the *incumbent* solution $\overline{Z}$, i.e., $\underline{Z}_i \geqslant \overline{Z}$.
- The δ-integrality and δ- and *x-nonanticipativity* constraints are satisfied whenever $\underline{Z}_i$ is computed. In this case, $\overline{Z} := \min\{\underline{Z}_i, \overline{Z}\}$.

(ii) After computing z^{TNF}_i:
- z^{TNF}_i is calculated at $i=n$, i.e. we are at the end of the branch (all 0–1 variables have already been branched on for the family).

(iii) After computing z^f_i:
- The linear model LP^f_i does not have a better solution value than model LP^{TNF}_i, i.e., $z^{TNF}_i = z^f_i$.
- The solution value of the linear model LP^f_i is not better than the *incumbent* solution, i.e., $z^f_i \geqslant \overline{Z}$.
- $z^f_i < z^{TNF}_i$ and $z^f_i < \overline{Z}$, and all the fractional variables δ take 0–1 values in the solution to model (14); in this case, the *incumbent* solution is updated.

In this way, the proposed approach always finds the optimal solution or determines that the instance is infeasible. However its computation for large-scale problems is not trivial at all because they combine several classes of difficulties: the number of branches to test can be huge, i.e., the cardinality of the set of feasible solutions can be too big and a high number of linear models z^p, z^{TNF} and z^f can exist to solve, or it can happen that the last two linear models have large dimensions. To gain computational efficiency, we propose the decomposition of these linear submodels for large-scale problems.

4. *LP* submodels to be solved via BD

We present in this section two submodels to be solved via BD for obtaining the LP optimal solution for the *TNF* integer sets, see Section 3 and the specialization that we propose of the *BFC* algorithm, see Section 5.

The computational experience on using BD within the framework of the *BFC* algorithm is reported in Section 7. We use BD, in particular, in Steps 1 and 6 of the algorithm, see Section 5. As a result, *BFC* with BD gives the optimal solution for large-scale problems in reasonable computing time, while using our specialization of the *BFC* algorithm alone does not obtain any solution due to running out of memory. On the contrary, for small and medium sized instances, *BFC* without BD requires smaller computing time than the strategy *BFC–BD* does for obtaining the optimal solution.

4.1. All δ-variables fixed to 0 or 1

The linear model LP^{TNF} (10) results after fixing in model *DEM* (2) the δ-variables at the 0–1 related values for a given *TNF* integer set. In this model, $\overline{\delta}$ will denote the 0–1 values of the respective vector of variables δ, all of them fixed to 0 or 1.

This linear problem can be decomposed and its solution can be iteratively obtained by identifying extreme points and rays based cuts from the optimization of the so-called *auxiliary program* (*AP*), and appending them to the so-called *relaxed master program* (*RMP*) for its optimization, see Benders [3]. The *RMP* can be expressed as

$$\overline{z}_{LP}^{TNF} = c_1^{\mathrm{T}}\overline{\delta} + \min c_2^{\mathrm{T}}x + \theta$$

$$\text{s.t.}\quad b_1 \leqslant A\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix} \leqslant b_2,$$

$$h_{01}^{\omega} \leqslant T_0^{\omega}\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix} \leqslant h_{02}^{\omega},\quad \omega\in\Omega,$$

$$\theta \geqslant \sum_{\omega\in\Omega} w^{\omega}v^{\omega\mathrm{T}}\left[\begin{pmatrix}h_1^{\omega}\\ -h_2^{\omega}\end{pmatrix} + \begin{pmatrix}-T^{\omega}\\ T^{\omega}\end{pmatrix}\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix}\right],\quad v^{\omega}\in\overline{\mathscr{J}}^{e_p},$$

$$0 \geqslant v^{\omega\mathrm{T}}\left[\begin{pmatrix}h_1^{\omega}\\ -h_2^{\omega}\end{pmatrix} + \begin{pmatrix}-T^{\omega}\\ T^{\omega}\end{pmatrix}\begin{pmatrix}\overline{\delta}\\ x\end{pmatrix}\right],\quad v^{\omega}\in\overline{\mathscr{J}}^{e_r},$$

$$x\geqslant 0, \theta\in IR,$$

where $\overline{\mathscr{J}}^{e_p} \subseteq \mathscr{J}^{e_p}$ and $\overline{\mathscr{J}}^{e_r} \subseteq \mathscr{J}^{e_r}$ are the subsets of the extreme points and extreme rays already identified, respectively.

4.2. Fractional δ-variables

Throughout this subsection we will consider the decomposition of the other *LP* model to solve for a given *TNF* integer set, the linear model LP^f (11). In this situation, the δ-variables can take fractional values, δ^f, or even the values 0 or 1, $\overline{\delta}$, if they are not yet branched on/fixed at in the current *TNF*. In terms of the BD, z_{LP}^f can be expressed by the *RMP* as

$$\overline{z}_{LP}^{f} = \min c_1^{\mathrm{T}}\delta + c_2^{\mathrm{T}}x + \theta$$

$$\text{s.t.}\quad b_1 \leqslant A\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant b_2,$$

$$h_{01}^{\omega} \leqslant T_0^{\omega}\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant h_{02}^{\omega},\quad \omega\in\Omega,$$

$$\theta \geqslant \sum_{\omega\in\Omega} w^{\omega}v^{\omega\mathrm{T}}\left[\begin{pmatrix}h_1^{\omega}\\ -h_2^{\omega}\end{pmatrix} + \begin{pmatrix}-T^{\omega}\\ T^{\omega}\end{pmatrix}\begin{pmatrix}\delta\\ x\end{pmatrix}\right],\quad v^{\omega}\in\overline{\mathscr{J}}^{e_p},$$

$$0 \geqslant v^{\omega\mathrm{T}}\left[\begin{pmatrix}h_1^{\omega}\\ -h_2^{\omega}\end{pmatrix} + \begin{pmatrix}-T^{\omega}\\ T^{\omega}\end{pmatrix}\begin{pmatrix}\delta\\ x\end{pmatrix}\right],\quad v^{\omega}\in\overline{\mathscr{J}}^{e_r},$$

$$0\leqslant \delta^f \leqslant 1,\quad x\geqslant 0,\quad \theta\in IR.$$

In order to gain computational efficiency, we present a scenario cluster based procedure for using the BD procedure, known as the L-shaped method, see Birge and Louveaux [23] and Van Slyke and Wets [24]. In our particular approach, Step 2 solves the feasibility problem in each scenario cluster, p. Notice that the objective function depends on a set of slack variables v^+, v^-, whose dimension is the number of constraints. This objective function can be optimized for a given scenario cluster instead of a particular scenario. Moreover, once the feasibility cut has been defined, the procedure goes back to Step 1 in order to solve the new RMP.

L-shaped primal scenario cluster based procedure.

Step 0: Set $k:=e_p:=e_r:=0$.
Step 1: Solve the RMP (with $\theta=0$ if $e_p=0$). Set $\omega:=0$, $p:=0$ and $k:=k+1$.

$$\min\quad c_1^{\mathrm{T}}\delta + c_2^{\mathrm{T}}x + \theta$$

$$\text{s.t.}\quad b_1 \leqslant A\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant b_2,$$

$$h_{01}^{\omega} \leqslant T_0^{\omega}\begin{pmatrix}\delta\\ x\end{pmatrix} \leqslant h_{02}^{\omega}\quad \forall\omega\in\Omega,$$

$$\hat{v}_{j_1}^{\omega\mathrm{T}}\begin{pmatrix}T^{\omega}\\ -T^{\omega}\end{pmatrix}\begin{pmatrix}\delta\\ x\end{pmatrix} \geqslant \hat{v}_{j_1}^{\omega\mathrm{T}}\begin{pmatrix}h_1^{\omega}\\ -h_2^{\omega}\end{pmatrix},\quad \omega\in\Omega^p,\ j_1=0,\dots,e_r, \tag{15}$$

$$\sum_{\omega\in\Omega} w^{\omega}\left[\hat{v}_{j_2}^{\omega\mathrm{T}}\begin{pmatrix}T^{\omega}\\ -T^{\omega}\end{pmatrix}\begin{pmatrix}\delta\\ x\end{pmatrix}\right] + \theta \geqslant \sum_{\omega\in\Omega} w^{\omega}\hat{v}_{j_2}^{\omega\mathrm{T}}\begin{pmatrix}h_1^{\omega}\\ -h_2^{\omega}\end{pmatrix},\quad j_2=0,\dots,e_p, \tag{16}$$

$$0\leqslant\delta^f\leqslant 1,\quad x\geqslant 0,\quad \theta\in IR.$$

Save primal variables $\hat{\delta}$, $\hat{x}$ and $\hat{\theta}$.
Step 2: Set $p:=p+1$. Solve the feasibility problem in scenario cluster p, $\omega\in\Omega^p$

$$(FEAS):\ z_{FEAS}^{p} = \min e^{\mathrm{T}}v_1^{+\omega} + e^{\mathrm{T}}v_1^{-\omega} + e^{\mathrm{T}}v_2^{+\omega} + e^{\mathrm{T}}v_2^{-\omega}$$

$$\text{s.t.}\quad W^{\omega}y^{\omega} - Iu^{-\omega} + Iv_1^{+\omega} - Iv_1^{-\omega} = h_1^{\omega} - T^{\omega}\begin{pmatrix}\hat{\delta}\\ \hat{x}\end{pmatrix},\quad \omega\in\Omega^p,$$

$$W^{\omega}y^{\omega} + Iu^{+\omega} - Iv_2^{+\omega} + Iv_2^{-\omega} = h_2^{\omega} - T^{\omega}\begin{pmatrix}\hat{\delta}\\ \hat{x}\end{pmatrix},\quad \omega\in\Omega^p,$$

$$y^{\omega},\ v_1^{+\omega},\ v_1^{-\omega},\ v_2^{+\omega},\ v_2^{-\omega},\ u^{+\omega},\ u^{-\omega}\geqslant 0,\quad \omega\in\Omega^p, \tag{17}$$

where the dimension of $e^{\mathrm{T}}=(1,\dots,1)$ is the number of constraints for scenario cluster p.
If $z_{FEAS}^p\neq 0$ (infeasible): Set $e_r:=e_r+1$, $\phi^{\omega}=+\infty$, $\forall\omega\in\Omega^p$, save the dual variables $\hat{v}^{\omega}$, $\omega\in\Omega^p$ and define the feasibility cut (15). Go to Step 1.
If $z_{FEAS}^p=0$ (feasible) and $p<\hat{p}$, go to Step 2.
Step 3: Solve the auxiliary primal problem in scenario $\omega\in\Omega$

$$\phi^{\omega} = \min\quad q^{\omega\mathrm{T}}y^{\omega}$$

$$\text{s.t.}\quad \begin{pmatrix}W^{\omega}\\ -W^{\omega}\end{pmatrix}y^{\omega} \geqslant \begin{pmatrix} h_1^{\omega} - T^{\omega}\begin{pmatrix}\hat{\delta}\\ \hat{x}\end{pmatrix}\\ -h_2^{\omega} + T^{\omega}\begin{pmatrix}\hat{\delta}\\ \hat{x}\end{pmatrix}\end{pmatrix},$$

$$y^{\omega} > 0. \tag{18}$$

Set $e_p := e_p + 1$, save ϕ^ω and the dual variables $\hat{v}^\omega$, reset $\theta := 0$ and define the optimality cut (16).
Step 4: Set $\phi := \sum_{\omega\in\Omega} w^\omega \phi^\omega$. If $\phi \leqslant \theta$, Stop. Optimal solution has been found in k-th iteration.
Save $\theta := \theta + c_1\hat{\delta} + c_2\hat{x}$ and go to Step 1.

The dimensions of the cluster-based dual vector to be used for obtaining the feasibility cut, problem *FEAS* (17), are greater than the dimensions for a scenario based scheme. In effect, problem (17) has, in this case, $2\cdot|\Omega^p|$ constraints for each scenario cluster p, and 2 constraints for each scenario feasibility problem. However, notice that the solution to this problem for each scenario cluster forces the feasibility in more scenarios than by using the scheme for each individual scenario. Then, the scenario cluster based scheme allows us to define tighter feasibility cuts than when using a scenario based procedure.

Moreover, it is not necessary to choose the same dimension for the scenario cluster, Ω^p, in the L-shaped algorithm as in the general procedure.

5. *BFC* implementation

We present the version that has been implemented for performing the computational experimentation reported in Section 7.

Let us introduce the elements that we use in our *depth first* strategy, see Alonso-Ayuso et al. [6] and Sherali and Zhu [13], among others, for selecting the branching variable and the two descendant *TNF*s from one given, where ρ_i is the selection parameter for the δ-variables to branch.

$$\rho_i = \min\left\{\sum_{p=1}^{\hat{p}} \overline{\delta}_i^p, \hat{p} - \sum_{p=1}^{\hat{p}} \overline{\delta}_i^p\right\}, \quad i \in \mathscr{I},$$

where $\overline{\delta}_i^p$ give the current values of the variables δ_i^p, and $\mathscr{I}$ gives the set of the δ-variables. Additionally, $\langle i\rangle$ will denote the i-th variable in a nonincreasing order of the ρ-parameter.

Let σ_i be the branching parameter, such that

$$\sigma_i = \begin{cases} 0 & \text{if } \sum_{p=1}^{\hat{p}} \overline{\delta}_i^p \leqslant \hat{p} - \sum_{p=1}^{\hat{p}} \overline{\delta}_i^p \\ 1 & \text{otherwise,} \end{cases}, \quad i \in \mathscr{I}.$$

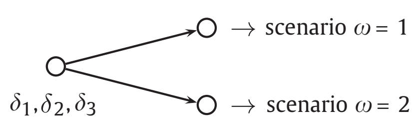
By convention, $z^{TNF} = +\infty$, for the infeasible problem LP^{TNF} (10) related to a given *TNF* integer set, and $z^f = +\infty$, for the infeasible problem LP^f (11).

BFC *procedure*.

Step 0: Initialize $\overline{Z} := +\infty$.
Step 1: Solve the *LP* relaxation of the original problem *MIP* (5) and compute $\underline{Z}$. If there is any 0–1 variable that takes a fractional value then go to Step 2. Otherwise, the optimal solution to the original problem has been found and, so, $\overline{Z} := \underline{Z}$ and stop.
Step 2: Initialize $i := 1$ and go to Step 4.
Step 3: Reset $i := i + 1$. If $i = |\mathscr{I}| + 1$ then go to Step 8.
Step 4: Set $\langle i\rangle := \mathrm{argmax}\{\rho_j, j \in \mathscr{I}\}$, such that j has not been previously branched on or not been fixed at in the current branching path.
Branch $\delta_{\langle i\rangle}^p := \sigma_{\langle i\rangle}$, $\forall p = 1,\ldots,\hat{p}$.
Step 5: Solve the linear relaxations $LP_{\langle i\rangle}^p$ (12), $\forall p = 1,\ldots,\hat{p}$ and compute $\underline{Z}$.
If $\underline{Z} \geqslant \overline{Z}$ then go to Step 7.
If there is any δ-variable that either takes fractional values or takes different values for some of the $\hat{p}$ scenario *clusters* then go to Step 3.
If all the x–variables take the same value for all scenario *clusters* $p = 1,\ldots,\hat{p}$ then update $\overline{Z} := \underline{Z}$ and go to Step 7.
Step 6: Solve the submodel $LP_{\langle i\rangle}^{TNF}$ (13) to satisfy the nonanticipativity constraints (9) for the x-variables in the given *TNF* integer set. Notice that the solution value is denoted by z^{TNF}.
Update $\overline{Z} := \min\{z^{TNF}, \overline{Z}\}$. If $i = |\mathscr{I}|$ then go to Step 7.
Solve the submodel $LP_{\langle i\rangle}^f$ (14), where the fractional δ-variables are the non-yet branched on at the current *TNF*. Notice that the solution value is denoted by z^f. If all the fractional δ- variables take 0–1 values in the solution to submodel (14), update $\overline{Z} := \min\{z^f, \overline{Z}\}$ and go to Step 7.
If $z^{TNF} = z^f$ or $z^f \geqslant \overline{Z}$ then go to Step 7, otherwise go to Step 3.
Step 7: Prune the branch.
If $\delta_{\langle i\rangle}^p = \sigma_{\langle i\rangle}$, $\forall p = 1,\ldots,\hat{p}$ then go to Step 10.
Step 8: Reset $i := i - 1$.
If $i = 0$ then stop, since the optimal solution $\overline{Z}$ has been found.
Step 9: If $\delta_{\langle i\rangle}^p = 1 - \sigma_{\langle i\rangle}$, $\forall p = 1,\ldots,\hat{p}$, then go to Step 8.
Step 10: Branch $\delta_{\langle i\rangle}^p = 1 - \sigma_{\langle i\rangle}$, $\forall p = 1,\ldots,\hat{p}$. go to Step 5.

6. Illustrative instance

Consider the following instance: where $|\Omega| = 2$ scenarios; $n_\delta = 3$, δ—variables; $n_x = 3$, x—variables and $n_y = 3$, y—variables.



Let *DEM* mixed 0–1 problem have the following form:

$$z_{MIP} = \min 9\delta_1 - 18\delta_2 + 27\delta_3 + 65x_1 - 65x_2 + 65x_3 - 0.15y_1^1 - 355y_2^1 - 355y_3^1 + 205y_1^2 + 21y_2^2 + 21y_3^2$$

$$\begin{aligned}
\text{s.t.}\quad & 1.5 \leqslant \delta_1 + \delta_2 + \delta_3 \leqslant 2,\\
& 1 \leqslant 0.3\delta_1 + 0.6\delta_2 + 0.9\delta_3 + 0.51x_1 + 0.77x_2 + 1.03x_3 \leqslant 100,\\
& 2 \leqslant 0.24x_1 + 0.23x_2 + 0.22x_3 \leqslant 120,\\
& 9.26\delta_1 + 17.52\delta_2 + 25.78\delta_3 + 2y_1^1 + 2y_2^1 + 2y_3^1 \leqslant 510,\\
& 8\delta_1 + 8\delta_2 + 8\delta_3 + 12x_1 + 12x_2 + 12x_3 + 12y_1^1 + 12y_2^1 + 12y_3^1 \leqslant 200,\\
& 0.4x_1 + 0.7x_2 + x_3 + 2y_1^1 + 2.1y_2^1 + 2.2y_3^1 \leqslant 240,\\
& 10.26\delta_1 + 18.52\delta_2 + 26.78\delta_3 + y_1^2 + y_2^2 + 3.27y_3^2 \leqslant 501,\\
& 6\delta_1 + 6\delta_2 + 6\delta_3 + 14x_1 + 14x_2 + 14x_3 + 10y_1^2 + 10y_2^2 + 10y_3^2 \leqslant 355,\\
& 0.5x_1 + 0.8x_2 + 1.1x_3 + 1.9y_1^2 + 2y_2^2 + 2.1y_3^2 \leqslant 240,\\
& x_1, x_2, x_3, y_1^\omega, y_2^\omega, y_3^\omega \geqslant 0, \quad \omega \in \{1,2\},\\
& \delta_1, \delta_2, \delta_3 \in \{0,1\}.
\end{aligned}$$

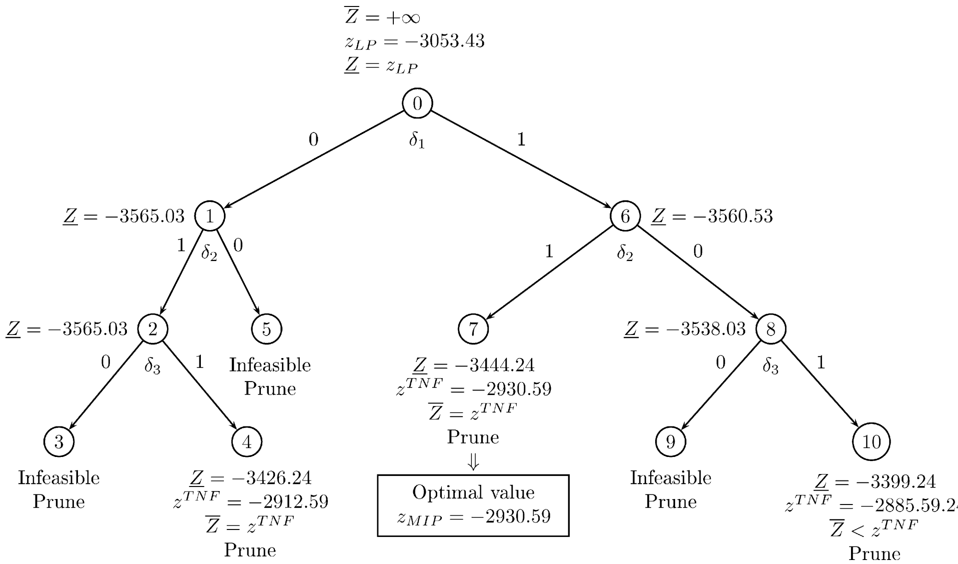


Fig. 2. Algorithm *BFC* for the illustrative instance.

We have considered $\hat{p} = 2$ scenario clusters, both of them with the same probability, i.e., $w^\omega = \frac{1}{2}$, $\omega \in \Omega$.

$$
\begin{aligned}
z^1_{MIP} = \min \quad & 4.5\delta_1 - 9\delta_2 + 13.5\delta_3 + 32.5x_1 - 32.5x_2 + 32.5x_3 \\
& \quad - 1.15y^1_1 - 355y^1_2 - 355y^1_3 \\
\text{s.t.} \quad & 1.5 \leqslant \delta_1 + \delta_2 + \delta_3 \leqslant 2, \\
& 1 \leqslant 0.3\delta_1 + 0.6\delta_2 + 0.9\delta_3 + 0.51x_1 + 0.77x_2 \\
& \quad + 1.03x_3 \leqslant 100, \\
& 2 \leqslant 0.24x_1 + 0.23x_2 + 0.22x_3 \leqslant 120, \\
& 9.26\delta_1 + 17.52\delta_2 + 25.78\delta_3 + 2y^1_1 + 2y^1_2 + 2y^1_3 \leqslant 510, \\
& 8\delta_1 + 8\delta_2 + 8\delta_3 + 12x_1 + 12x_2 + 12x_3 + 12y^1_1 \\
& \quad + 12y^1_2 + 12y^1_3 \leqslant 200, \\
& 0.4x_1 + 0.7x_2 + x_3 + 2y^1_1 + 2.1y^1_2 + 2.2y^1_3 \leqslant 240, \\
& x_1, x_2, x_3, y^1_1, y^1_2, y^1_3 \geqslant 0, \\
& \delta_1, \delta_2, \delta_3 \in \{0, 1\}. \qquad (19)
\end{aligned}
$$

$$
\begin{aligned}
z^2_{MIP} = \min \quad & 4.5\delta_1 - 9\delta_2 + 13.5\delta_3 + 32.5x_1 - 32.5x_2 + 32.5x_3 \\
& \quad + 205y^2_1 + 21y^2_2 + 21y^2_3 \\
\text{s.t.} \quad & 1.5 \leqslant \delta_1 + \delta_2 + \delta_3 \leqslant 2, \\
& 1 \leqslant 0.3\delta_1 + 0.6\delta_2 + 0.9\delta_3 + 0.51x_1 + 0.77x_2 \qquad (20) \\
& \quad + 1.03x_3 \leqslant 100, \\
& 2 \leqslant 0.24x_1 + 0.23x_2 + 0.22x_3 \leqslant 120, \\
& 10.26\delta_1 + 18.52\delta_2 + 26.78\delta_3 + y^2_1 + y^2_2 + 3.27y^2_3 \leqslant 501, \\
& 6\delta_1 + 6\delta_2 + 6\delta_3 + 14x_1 + 14x_2 + 14x_3 + 10y^2_1 \\
& \quad + 10y^2_2 + 10y^2_3 \leqslant 355, \\
& 0.5x_1 + 0.8x_2 + 1.1x_3 + 1.9y^2_1 + 2y^2_2 + 2.1y^2_3 \leqslant 240, \\
& x_1, x_2, x_3, y^2_1, y^2_2, y^2_3 \geqslant 0, \\
& \delta_1, \delta_2, \delta_3 \in \{0, 1\}. \qquad (21)
\end{aligned}
$$

We will show the performance of our *BFC* procedure in this small instance, which is illustrated in Fig. 2.

At the root node of the *BFC* tree, we initialize $\overline{Z}=+\infty$. We solve the linear relaxation of the mixed 0–1 *DEM*, and obtain $\underline{Z}=z_{LP}=-3053.43$ and $(\delta_1, \delta_2, \delta_3) = (0.5, 1, 0)$. We solve the two scenario-cluster linear problems, LP^ω, for $\omega = 1, 2$. The solution values of the δ variables are $(\delta_1, \delta_2, \delta_3)^1 = (0.5, 1, 0)$ and $(\delta_1, \delta_2, \delta_3)^2 = (0.5, 1, 0)$.

What is the order of branching in δ-variables? Let us compute ρ_i:

$\rho_1 = \min\{1, 1\} = 1$, $\rho_2 = \min\{2, 0\} = 0$ and $\rho_3 = \min\{0, 2\} = 0$.
As $\rho_1 \geqslant \rho_2 \geqslant \rho_3$, the order of branching for the 0–1 first-stage variables will be the natural one, i.e., $\langle 1 \rangle = 1$, $\langle 2 \rangle = 2$ and $\langle 3 \rangle = 3$.
And, how will we start branching on each δ-variable? Let us compute σ_i:
$1 \leqslant 1 \Rightarrow \sigma_1 = 0$, $2 \nleqslant 0 \Rightarrow \sigma_2 = 1$ and $0 \leqslant 2 \Rightarrow \sigma_3 = 0$.
So, we will start fixing δ_1 to 0, then δ_2 to 1 and finally, δ_3 to 0.

The steps of the procedure are presented in Table 1, where the first two columns give the node of the *TNF* under consideration (*f* denotes a fractional δ-variable), the next three columns represent the solution of the scenario models (Step 5), the sixth column is full when Step 6 occurs, and the last column represents the yes–no decision of pruning (P), it appears as *y* if the decision is to prune, and it appears as *n* if the decision is to continue branching.

7. Computational experience

We report the results of the computational experience obtained while optimizing model *MIP* (5) by using the *BFC* approach presented in Section 5. The scenarios have been randomly generated for a broad variety of instances.

Our algorithmic approach has been implemented in a C++ experimental code, which uses the simplex method from the library COIN-OR to solve the *LP* models. The computations were carried out on a Workstation Dell precision 690 (Intel Xeon Quad Core) under the LINUX operating system, having a cpu speed of 3.73 GHz.

Table 2 presents the dimensions of *DEM* (2), compact representation. The headings are as follows: m, number of constraints; n_δ, number of δ-variables; n_x, number of x-variables; n_y, number of y-variables; *nel*, number of nonzero elements in the constraint matrix and *dens*, constraint matrix density %. The table has been split into three categories. The first one includes cases with $|\Omega| = 1000$

Table 1
Steps of the illustrative instance.

Node	Fix			$\underline{z}$	Compute					z^{TNF}	$\overline{z}$	P
	δ_1	δ_2	δ_3		z_{LP}^{ω}	$\overline{\delta}_1$	$\overline{\delta}_2$	$\overline{\delta}_3$				
0	*f*	*f*	*f*	−3574.03	−2764.07	0.5	1	0	$\overline{\delta}_1$ *f*		$+\infty$	
					−809.96	0.5	1	0				*n*
1	**0**	*f*	*f*	−3565.03	−2759.57	0	1	0.5	$\overline{\delta}_3$ *f*		$+\infty$	
					−805.46	0	1	0.5				*n*
2	0	**1**	*f*	−3565.03	−2759.57	0	1	0.5	$\overline{\delta}_3 f$		$+\infty$	
					−805.46	0	1	0.5				*n*
3	0	1	**0**	$-\infty$	Infeasible						$+\infty$	*y*
4	0	1	**1**	−3426.24	−2634.49	0	1	1			$+\infty$	
					−791.75	0	1	1	$\overline{x}_2^1 \neq \overline{x}_2^2$	−2912.59	−2912.59	y
5	0	**0**	*f*	$-\infty$	Infeasible						−2912.59	*y*
6	**1**	*f*	*f*	−3560.53	−2757.32	1	0.5	0	$\overline{\delta}_2$ *f*		−2912.59	
					−803.21	1	0.5	0				*n*
7	1	**1**	*f*	−3444.24	−2643.49	1	1	0			−2912.59	
					−800.75	1	1	0	$\overline{x}_2^1 \neq \overline{x}_2^2$	−2930.59	−2930.59	*y*
8	1	**0**	*f*	−3538.03	−2746.07	1	0	0.5	$\overline{\delta}_3$ *f*		−2930.59	
					−791.64	1	0	0.5				*n*
9	1	0	**0**	$-\infty$	Infeasible						−2930.59	*y*
10	1	0	**1**	−3399.24	−2620.99	1	0	1			−2930.59	
					−778.25	1	0	1	$\overline{x}_2^1 \neq \overline{x}_2^2$	−2885.59	−2930.59	*y*

Table 2
MIP model dimensions.

Case	Deterministic equivalent model						
	$\lvert\Omega\rvert$	*m*	n_δ	n_x	n_y	*nel*	*dens*
P1	1000	8008	100	100	100 000	2 401 600	0.299
P2	1000	8008	120	120	120 000	2 881 920	0.299
P3	1000	8008	150	150	150 000	3 602 400	0.299
P4	1000	8038	100	300	100 000	4 015 200	0.497
P5	1000	8008	200	200	150 000	4 403 200	0.365
P6	1000	8053	120	600	120 000	6 758 160	0.695
P7	1000	7075	200	450	450 000	7 395 000	0.232
P8	1000	6070	150	750	450 000	8 163 000	0.298
P9	1000	8008	500	500	150 000	9 208 000	0.761
P10	1000	8008	1000	1000	200 000	17 616 000	1.089
P11	2000	16008	100	100	200 000	4 801 600	0.149
P12	2000	18009	150	150	300 000	8 102 700	0.149
P13	2000	12007	600	600	500 000	17 408 400	0.289
P14	2000	12070	600	600	600 000	18 084 000	0.249
P15	2000	14070	500	500	690 000	18 900 000	0.194
P16	2000	16007	500	500	500 000	20 007 000	0.249
P17	2000	12070	600	1000	700 000	23 512 000	0.277
P18	2000	12110	300	1400	700 000	24 787 000	0.292
P19	2000	14009	750	750	1 000 000	28 013 500	0.199
P20	2000	14009	1000	1000	1 000 000	35 018 000	0.249
P21	3000	27009	100	100	300 000	8 101 800	0.099
P22	3000	27009	500	500	1 500 000	40 509 000	0.099
P23	3000	24007	1000	1000	1 500 000	60 014 000	0.166

Compact representation.

Table 3
MIP^p model dimensions.

Case	Scenario cluster model						
	$\lvert\Omega^p\rvert$	*m*	n_δ	n_x	n_y	*nel*	*dens*
P1	40	208	100	100	2500	61 600	10.969
P2	40	208	120	120	3000	73 920	10.969
P3	40	208	150	150	3750	92 400	10.969
P4	40	238	100	300	2500	115 200	16.690
P5	40	208	200	200	3750	113 200	13.114
P6	40	253	120	600	3000	73 920	10.969
P7	40	250	200	450	11 200	228 750	7.721
P8	40	220	150	750	11 250	265 500	9.943
P9	40	208	500	500	3750	238 000	24.089
P10	40	208	1000	1000	5000	456 000	31.319
P11	50	328	100	100	4000	97 600	7.085
P12	50	369	150	150	6000	164 700	7.085
P13	50	247	600	600	10 000	356 400	12.883
P14	50	310	600	600	12 000	444 000	10.850
P15	50	350	500	500	13 800	44 660	8.622
P16	50	327	500	500	10 000	407 000	11.315
P17	50	310	600	1000	14 000	580 000	11.993
P18	50	350	300	1400	14 000	679 000	12.357
P19	50	289	750	750	20 000	573 500	9.230
P20	50	289	1000	1000	20 000	718 000	11.293
P21	60	459	100	100	5000	136 800	5.731
P22	60	459	500	500	25 000	684 000	5.731
P23	60	407	1000	1000	25 000	1 014 000	9.227

Compact representation.

scenarios, and the second and third categories include cases with 2000 and 3000 scenarios, respectively. Notice the large-scale dimensions of the testbeds.

Table 3 presents the dimensions of the scenario cluster model MIP^p (7). The headings are similar to Table 1 but, now, instead of $|\Omega|$, $|\Omega^p|$ gives the dimension of each scenario cluster considered. For each instance, the number of scenario-cluster related submodels that the procedure solves is $|\Omega|/|\Omega^p| = \hat{p}$. Each cluster contains $|\Omega^p|$ scenarios consecutive, starting from the first one and following in natural order.

Table 4 presents the main results of our computational experimentation for given values of the number of scenario *clusters*. The headings are as follows: Z_{LP}, solution value of the *LP* relaxation of the original problem (2); Z_{MIP}, solution value of the original problem; *GAP*, optimality gap defined as $(Z_{MIP} - Z_{LP})/Z_{LP}\%$; *nn*, number of *TNF* branches for the set of *BF* trees; T_{LP} and T_{LP}^{B}, the elapsed time (s) for obtaining the *LP* solution without using the BD and using it, respectively; T, T^B and T^{COIN}, the total elapsed time (s) to obtain the optimal solution to the original problem by using the *BFC* procedure without *BD*, by using *BFC* jointly with *BD* and by plain use of the optimization engine in COIN-OR for solving *DEM* (2), respectively. Notice that when using the strategy *BFC–BD*, the *LP* relaxation of the original problem in Step 1, and the linear programs LP_i^{TNF} (13) and LP_i^{f} (14) in Step 6 are optimized using *BD*, but this is not the case for the *LP* relaxation of the scenario *cluster* model LP_i^p (12) in Step 5, which is not decomposed.

Table 4
Stochastic solution.

Case	$\hat{p}$	Z_{LP}	Z_{MIP}	GAP	nn	T_{LP}	T	T^B_{LP}	T^B	T^{COIN}
P1	25	−1 026 540	−996 961	2.88	336	9.24	121.539	8.81	264.897	266.529
P2	25	−1 004 790	−975 767	2.89	336	12.02	136.297	10.32	313.124	340.430
P3	25	−2 243 630	−2 194 520	2.19	330	17.20	257.128	11.93	297.089	676.814
P4	25	−3 563 630	−3 523 100	1.137	132	9.03	137.340	6.54	134.816	257.719
P5	25	−14 964 200	−14 939 200	0.17	380	16.23	191.001	8.57	558.871	698.793
P6	25	−41 739 600	−41 726 800	0.03	128	21.52	142.245	18.72	127.076	498.763
P7	25	−4 207 640	−4 207 630	0.00	251	22.92	231.919	16.357	226.73	641.928
P8	25	−1 140 720	−1 107 570	2.90	335	12.64	166.662	11.92	357.534	591.750
P9	25	−9 536 070	−9 518 680	0.183	383	36.83	324.448	9.64	777.157	2291.730
P10	25	−42 031 500	−42 011 100	0.048	394	62.18	812.241	15.07	2270.66	7257.616
P11	40	−315 991 000	−315 933 000	0.018	390	19.39	171.412	14.82	340.857	684.131
P12	40	−442 956 000	−442 897 000	0.013	390	35.55	286.542	20.96	507.880	1344.797
P13	40	−218 408 000	−218 392 000	0.007	67	57.41	142.813	37.23	113.855	1525.194
P14	40	−148 389 000	−148 376 000	0.009	64	59.03	160.578	36.58	131.956	–
P15	40	−160 293 000	−160 277 000	0.009	69	69.42	248.003	46.59	214.132	928.182
P16	40	−258 877 000	−258 855 000	0.008	390	68.11	668.312	44.69	1136.630	6412.737
P17	40	−193 010 000	−192 993 000	0.009	68	70.86	205.130	44.80	180.935	2462.026
P18	40	−190 622 000	−190 596 000	0.013	128	43.94	388.032	39.35	352.678	–
P19	40	−95 545 900	−95 532 500	0.014	65	–	–	48.34	1047.970	–
P20	40	−95 808 800	−95 795 500	0.014	86	–	–	58.67	1762.630	–
P21	50	−490 077 000	−489 987 000	0.018	662	36.71	496.39	24.12	792.306	1809.765
P22	50	−421 943 000	−421 900 000	0.010	834	–	–	132.85	7161.880	–
P23	50	−288 501 000	−288 473 000	0.009	1498	–	–	147.117	10 120.900	–

The number *nn* of *TNFs* that have been branched on is relatively small. The *COIN-OR* strategy requires more elapsed time than the other two strategies. The *BFC* strategy alone requires much less elapsed time than the *BFC–BD* strategy, while there is enough memory to store and solve the LP submodels. Obviously, the simplex scheme is enough to solve linear models of these small and medium scale dimensions. For bigger instances (in particular, P19, P20, P22 and P23), BD is needed to solve the LP submodels. For these large-scale models, *BFC–BD* is the only able strategy for solving them.

We will use the $\log_2$ scaled performance profile described in Dolan and Moré [25], for comparing the three strategies mentioned, namely, *COIN*, *BFC* and *BFC–BD*.

For each problem $p \in \{P1, P2, \ldots, P23\}$, and strategy $s \in \{COIN, BFC, BFC\text{–}BD\}$, we define:

$T_{p,s}$, computing time required to solve problem p by strategy s.
$r_{p,s}$, performance ratio, $r_{p,s} = T_{p,s}/\min_s\{T_{p,s}\}$ and $r_{p,s} = r_M$ iff s does not solve problem p, where r_M is a big number, say 12.
$\rho_s(\tau)$, performance profile, $\rho_s(\tau) = P(r_{p,s} \leqslant \tau) = Card\{p : r_{p,s} \leqslant \tau\}/Card\{p\}$. $\rho_s : IR \rightarrow [0, 1]$ for a strategy s is a nondecreasing, piecewise constant function, continuous from the right at each breakpoint. The value of $\rho_s(1)$ is the probability that the strategy will win over the rest of the strategies. In particular, $1 - \rho_s(\tau)$ is the fraction of problems that the strategy s cannot solve within a factor τ of the best strategy, including problems for which the strategy in question fails.
$\pi_s(\tau)$, $\log_2$ scaled performance profile, $\pi_s(\tau) = P(\log_2(r_{p,s}) \leqslant \tau) = Card\{p : \log_2(r_{p,s}) \leqslant \tau\}/Card\{p\}$.

In Fig. 3, we can observe and compare the efficiency of the strategies (at $\tau = 0$), and also the stability of them (at $\tau = r_M$). Both items can be shown in the same figure due to the $\log_2$ scaled performance profile.

If we are interested in the strategy that can solve most of the problems with the greatest efficiency, then *BFC–BD* and *BFC* stand out. The strategy *COIN* alone fails on 100% of the problems. These results indicate that for any $\tau \geqslant 0$, *BFC–BD* and *BFC* solve more problems within a factor of τ than *COIN* strategy does. In other words, by examining $\tau = 0$ we can say that *BFC–BD* is the fastest strategy on approximately in 52.17% of the problems and *BFC* is the fastest one on the 47.83% of the problems. By examining $\tau = 12$, we can say that *BFC–BD* solves all the problems to optimality, and *BFC* solves most of the problems (about 82.61%).

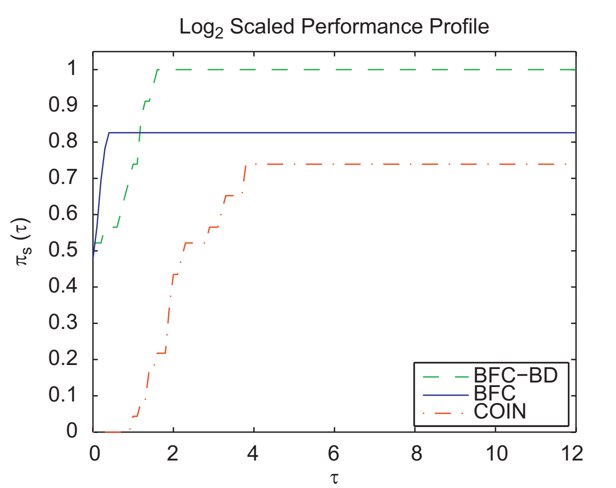


Fig. 3. Comparison, in terms of CPU time, on 23 two-stage stochastic MIP problems

8. Conclusions and future work

We have presented an algorithmic framework for solving two-stage stochastic mixed 0–1 problems where the uncertainty appears anywhere in the problem, and the 0–1 variables and the continuous variables appear both in the first stage. The framework is based on a specialization of the *BFC* method. The BD is used to solve the linear submodels in the first step and other places in the algorithm. The nonanticipativity constraints for the first-stage continuous variables are satisfied by solving linear submodels at the *TNF* integer sets. The computational experience shows that the *BFC* and *BFC–BD* strategies require much less elapsed time for obtaining the optimal solution than the time required by the plain use of *COIN-OR*, a state-of-the-art optimization engine. For the first class and some examples of the second class, with small and middle dimensions, the use of BD gives worse results than using *BFC* alone. Otherwise, the *BFC–BD* strategy gives the optimal solution for large-scale problems, in a reasonable elapsed time while the others cannot.

As a future work we are planning to perform an extensive computational experimentation to assess the possibility of introducing in our approach the feature that allows second stage variables to be also 0–1 variables. Another future work under consideration is the extension of the proposed methodology to the multi-stage case.

Acknowledgments

This research has been partially supported by the Projects MEC2005/168 and MTM2004-01095 from the Spanish Ministry of Education and Science, Research Group IT-321-07 from the Basque Government, and Project URJC-CM-2007-CET-1622 from Comunidad de Madrid, Spain.

Moreover we wish to thank the anonymous referees for valuable comments and suggestions that helped to improve the presentation of the paper.

References

[1] Wets RJ-B. Programming under uncertainty: the equivalent convex program. SIAM Journal on Applied Mathematics 1966;14:89–105.
[2] Laporte G, Louveaux F. An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. Operations Research 2002;50: 415–23.
[3] Benders J. Partitioning procedures for solving mixed variables programming problems. Numerische Mathematik 1962;4:238–52.
[4] Alonso-Ayuso A, Escudero LF, Ortuño MT. 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 2003;151:503–19.
[5] Alonso-Ayuso A, Escudero LF, Garín MA, Ortuño MT, Pérez G. An approach for strategic supply chain planning based on stochastic 0–1 programming. Journal of Global Optimization 2003;26:97–124.
[6] Alonso-Ayuso A, Escudero LF, Garín MA, Ortuño MT, Pérez G. On the product selection and plant dimensioning problem under uncertainty. Omega—International Journal of Management Science 2005;33:307–18.
[7] Ahmed S, Tawarmalani M, Sahinidis NV. A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. Mathematical Programming 2004;100:355–77.
[8] Carøe C, Tind J. L-shaped decomposition of two-stage stochastic programs with integer recourse. Mathematical Programming 1998;83:451–64.
[9] Sherali HD, Fraticelli BMP. A modification of Benders' decomposition algorithm for discrete subproblems: an approach for stochastic programs with integer recourse. Journal of Global Optimization 2002;22:319–42.
[10] Sen SS, Higle JL. The c^3 theorem and the d^2 algorithm for large scale stochastic mixed-integer programming: set convexification. Mathematical Programming 2005;104:1–20.
[11] Ntaimo L, Sen S. The million variable 'march' for stochastic combinatorial optimization. Journal of Global Optimization 2005;32:385–400.
[12] Sen S, Sherali H. Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. Mathematical Programming Series A 2006;105:203–23.
[13] Sherali H, Zhu X. On solving discrete two-stage stochastic programs having mixed-integer first- and second-stage variables. Mathematical Programming Series B 2006;108:597–611.
[14] Carøe C, Schultz R. Dual decomposition in stochastic integer programming. Operations Research Letters 1999;24:37–45.
[15] Hemmecke R, Schultz R. Decomposition methods for two-stage stochastic integer programs. In: Grotschel M, Krumke SO, Rambau J, editors. Online optimization of large scale systems. Berlin: Springer; 2001. p. 601–22.
[16] Takriti S, Birge J. Lagrangean solution techniques and bounds for loosely coupled mixed-integer stochastic programs. Operations Research 2000;48:91–8.
[17] Rockafellar RT, Wets R J-B. Scenario and policy aggregation in optimisation under uncertainty. Mathematics of Operations Research 1991;16:119–47.
[18] Klein Haneveld W, van der Vlerk Kang M. Stochastic integer programming: general models and algorithms. Annals of Operations Research 1999;85:39–57.
[19] Nowak M, Schultz R, Westphalen W. Optimization of simultaneous power production and trading by stochastic integer programming. Technical report, Stochastic programming e-print series ⟨http://dochost.rz.hu-berlin.de/speps⟩, 2002.
[20] Romisch W, Schultz R. Multi-stage stochastic integer programs: an introduction. In: Grotschel M, Krumke SO, Rambau J, editors. Online optimization of large scale systems. Berlin: Springer; 2001. p. 581–600.
[21] Schultz R. Stochastic programming with integer variables. Mathematical Programming Series B 2003;97:285–309.
[22] Escudero LF, Garín MA, Merino M, Pérez G. A multistage stochastic integer programming model and algorithm for incorporating logical constraints in assets and liabilities under uncertainty. Computational Management Science, 2007, in press, doi:10.2007/s10287-006-0035-7.
[23] Birge J, Louveaux F. Introduction to stochastic programming. Berlin: Springer; 1997.
[24] Van Slyke R, Wets RJ-B. L-shaped linear programs with application to optimal control and stochastic programming. SIAM Journal on Applied Mathematics 1969;17:638–63.
[25] Dolan ED, Moré JJ. Benchmarking optimization software with performance profiles. Mathematical Programming Series A 2002;91:201–13.