DOCUMENTOS DE TRABAJO

BILTOKI

D.T. 2010.02

A note on the implementation of the BFC-MSMIP algorithm in C++
by using COIN-OR as an optimization engine.

L. F. Escudero, M. A. Garín, M. Merino y G. Pérez

eman ta zabal zazu

Universidad del País Vasco
Euskal Herriko Unibertsitatea

Facultad de Ciencias Económicas.
Avda. Lehendakari Aguirre, 83
48015 BILBAO.

Documento de Trabajo BILTOKI DT2010.02
Editado por el Departamento de Economía Aplicada III (Econometría y Estadística)
de la Universidad del País Vasco.

Depósito Legal No.: BI-590/2010
ISSN: 1134-8984
URN: RePEc:ehu:biltok:201002

A note on the implementation of the BFC-MSMIP algorithm in C++ by using *COIN-OR* as an optimization engine

L.F. Escudero[1], **M.A. Garín**[2], **M. Merino**[3] **and G. Pérez**[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: maria.merino@ehu.es

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

Abstract

The aim of this technical report is to present some detailed explanations in order to help to understand and use the algorithm Branch and Fix Coordination for solving MultiStage Mixed Integer Problems (*BFC-MSMIP*). We have developed an algorithmic approach implemented in a C++ experimental code that uses the optimization engine COmputational INfrastructure for Operations Research (*COIN-OR*) for solving the auxiliary linear and mixed 0-1 submodels. Now, we give the computational and implementational description in order to use this open optimization software not only in the implementation of our procedure but also in similar schemes to be developed by the users.

Keywords Multistage stochastic mixed 0-1 programming, Branch-and-Fix Coordination, nonanticipativity constraints, scenario cluster partitioning, COIN-OR library.

[*]This research has been partially supported by the project ECO2008-00777 ECON from the Spanish Ministry of Education and Science, and Grupo de Investigación IT-347-10 from the Basque Government.

1 Introduction

In this paper we present some technical notes for easily using of the algorithm Branch and Fix Coordination for solving MultiStage Mixed Integer Problems (*BFC-MSMIP*), see Escudero et al (2009) and (2010). We have developed an algorithmic approach implemented in a C++ experimental code. It uses the optimization engine COmputational INfrastructure for Operations Research (*COIN-OR*) (see `http://www.coin-or.org` and Laugee-Heimer, R. (2003)) for solving the auxiliary linear and mixed 0-1 submodels. In this technical report we give the computational and implementational description in order to use this open source optimization software not only in the implementation of our own procedure but also in similar schemes to be developed by the users.

The remainder of the papes is as follows. Section 2 presents the optimization problem to be solved and a general scheme of its decomposition in cluster submodels. Some of the main decisions to structure the information in the implementation of the algorithm are given in Section 3. In Section 4 appears the description of the algorithm such as it has been published in Escudero et al (2010). Section 5 describes the main steps of the implementation. An alternative way of storing and branching on the 0-1 variables is presented in Section 6. Section 7 gives some aditional information about the instances named $P1$ to $P16$ taken from the same paper, and Section 8 gives some details about how to compile and link the code with *COIN-OR* library.

2 Mixed integer stochastic model

We will consider, the following multistage mixed 0-1 model

$$\begin{aligned} & \min \sum_{t\in\mathscr{T}} a_t x_t + c_t y_t \\ & \text{s.t. } A'_t x_{t-1} + A_t x_t + B'_t y_{t-1} + B_t y_t = b_t \quad \forall t \in \mathscr{T} \\ & \quad x_t \in \{0,1\}^{n_x}, \;\; y_t \in \mathbb{R}^{+n_y} \qquad \forall t \in \mathscr{T}, \end{aligned} \tag{1}$$

where a_t and c_t are the vectors of the objective function, A'_t, B'_t, A_t and B_t are the constraint matrices for the 0-1 and continuous variables related to stage $t-1$ and t, respectively, and $\mathscr{T}$ the set of stages. b_t is the right-hand-side (*rhs*) and x_t, y_t, are the n_x and n_y dimensional vectors of the 0-1 and continuous variables for stage t, respectively.

We will denote with $T = |\mathscr{T}|$, the number of stages, and $\mathscr{T}^- = \mathscr{T} - \{T\}$ will denote the set of stages except the last one.

This model can be extended to consider uncertainty in some of the main parameters, in our case, the objective function, the *rhs* and the constraint matrix coefficients. To introduce uncertainty in the parameters, we will use a scenario analysis approach. In this sense, Ω will denote the set of scenarios, and $\omega \in \Omega$ will represent one specific scenario.

The *splitting variable* representation of the mixed 0-1 Deterministic Equivalent Model *(DEM)* of the stochastic version with complete recourse of the deterministic multistage problem (1) can be expressed as

$$(MIP) \quad z_{MIP} \;=\; \min \sum_{\omega\in\Omega}\sum_{t\in\mathscr{T}} w^{\omega}\left(a_t^{\omega} x_t^{\omega} + c_t^{\omega} y_t^{\omega}\right)$$

$$\begin{aligned} & \text{s.t. } A_t'^{\omega} x_{t-1}^{\omega} + A_t^{\omega} x_t^{\omega} + B_t'^{\omega} y_{t-1}^{\omega} + B_t^{\omega} y_t^{\omega} = b_t^{\omega}, \qquad \forall \omega \in \Omega,\; t \in \mathscr{T} \\ & \quad x_t^{\omega} - x_t^{\omega'} = 0,\; \forall \omega, \omega' \in \Omega_g : \omega \neq \omega',\; g \in \mathscr{G}_t, \quad t \in \mathscr{T}^- \\ & \quad y_t^{\omega} - y_t^{\omega'} = 0,\; \forall \omega, \omega' \in \Omega_g : \omega \neq \omega',\; g \in \mathscr{G}_t, \quad t \in \mathscr{T}^- \\ & \quad x_t^{\omega} \in \{0,1\},\;\; y_t^{\omega} \in \mathbb{R}^+,\; \forall \omega \in \Omega,\; t \in \mathscr{T}, \end{aligned} \tag{2}$$

where w^ω is the likelihood or probability asigned by the modeler to scenario ω, such that $\sum_{\omega\in\Omega} w^\omega = 1$. The index ω in the model given above denote the copy of the coefficient or variable related to scenario ω.

Let also $\mathscr{G}$ denote the set of scenario groups, and $\mathscr{G}_t$, the subset of scenario groups that belong to stage $t \in \mathscr{T}$, such that $\mathscr{G} = \cup_{t\in\mathscr{T}} \mathscr{G}_t$. Let us suppose that we have selected a number of scenario clusters, say q. This value q can be selected as a divisor of $|\Omega|$. Then, $1 \leq |\Omega^p| = \frac{|\Omega|}{q} \leq |\Omega|$, where Ω^p gives the set of scenarios in cluster p, for $p = 1,...,q$. The idea is to decompose the *DEM* model into scenario cluster models. These scenario cluster models are linked by the nonanticipativity constraints, see below.

As an additional notation, let $\mathscr{G}^p \subset \mathscr{G}$ denote the set of scenario groups for cluster p, such that $\Omega_g \cap \Omega^p \neq \emptyset$ means that $g \in \mathscr{G}^p$ and Ω_g is the set of the scenarios related to group g.

An equivalent and alternative representation of the *DEM* (2) can be given by the mixture of the compact (into the clusters) and the splitting variable representation (between them). It can be given in terms of the scenario-cluster models as follows,

$$
\begin{aligned}
(MIP) \quad Z_{MIP} \quad = \quad & \min \sum_{p=1}^{q} \sum_{g\in\mathscr{G}^p} w_g (a^g x^g + c^g y^g) \\
\text{s.t.} \quad & A'_g x^{\pi(g)} + A_g x^g + B'_g y^{\pi(g)} + B_g y^g = b^g, \quad \forall g \in \mathscr{G}^p, p = 1,...,q \\
& x_p^g - x_{p'}^g = 0, \quad \forall g \in \mathscr{G}^p \cap \mathscr{G}^{p'}, p \neq p' \\
& y_p^g - y_{p'}^g = 0, \quad \forall g \in \mathscr{G}^p \cap \mathscr{G}^{p'}, p \neq p' \\
& x^g \in \{0,1\}, \ y^g \in \mathbb{R}^+, \ \forall g \in \mathscr{G}^p, \ p = 1,...,q.
\end{aligned}
\tag{3}
$$

where w_g is the likelihood of scenario group g, with $g \in \mathscr{G}^p$, such that $w_g = \sum_{\omega\in\Omega_g} w^\omega$. x^g and y^g are the copy of the x, y vectors of variables for scenario group g. Moreover, x_p^g, y_p^g, $x_{p'}^g$, and $y_{p'}^g$, for $g \in \mathscr{G}^p \cap \mathscr{G}^{p'}$, denote the set of common variables, i.e, the set of variables related to the scenario group g, and common to scenario clusters p and p'. x_p^g, $x_{p'}^g$, are copies of the variables x^g and y_p^g, $y_{p'}^g$, are copies of the variables y^g. $\pi(g)$ denotes the scenario group related to the immediate predecessor of node g in the scenario tree, such that $\pi(g) \in \mathscr{G}_{t(g)-1}$, for $g \in \mathscr{G} - \mathscr{G}_1$, where $t(g)$ is the stage to which scenario group g belongs to, such that $g \in \mathscr{G}_{t(g)}$.

The model to consider for each scenario *cluster* $p = 1,\ldots,q$ can be expressed by the *compact* representation,

$$
\begin{aligned}
(MIP^p) \quad z^p = & \min \sum_{g\in\mathscr{G}^p} w_g (a^g x^g + c^g y^g) \\
\text{s.t.} \ & A'_g x^{\pi(g)} + A_g x^g + B'_g y^{\pi(g)} + B_g y^g = b^g \quad \forall g \in \mathscr{G}^p \\
& x^g \in \{0,1\}, \ y^g \in \mathbb{R}^+ \qquad \forall g \in \mathscr{G}^p,
\end{aligned}
\tag{4}
$$

The q problems (4) are linked by the *nonanticipativity* constraints:

$$x_p^g - x_{p'}^g = 0 \tag{5}$$

$$y_p^g - y_{p'}^g = 0, \tag{6}$$

$\forall g \in \mathscr{G}^p \cap \mathscr{G}^{p'}$, such that $p \neq p'$. From here to the end of the paper, we will use the notation, x_p^g, only when we want to distinguish between two diferent clusters p and p', as in the nonanticipativity constraints (5) and (6). In other cases, we will use x^g or y^g, with $g \in \mathscr{G}^p$, to denote the vectors of variables for each scenario cluster p, and x^{tp} notation in order to denote the stage t and scenario cluster p.

3 Main decisions for information structuring

The main program for the algorithm BFC-MSMIP is named **BFC_MS.cpp**. It is written in C++ with several references to functions from the library *COIN-OR* to store and solve linear and mixed 0-1 auxilary submodels. The following external references to own functions are as follows:

- **vectores.cpp**. It builds the scenario tree and establishes the relationship between scenario-cluster problems and the *DEM*. It sets the number of contingencies, the last scenario group and the weight for each stage; it also sets the ancestor group, the stage and the last binary variable for each scenario group; it defines the relations between the scenario cluster problems and the *DEM* and assigns the order for the binary variables by indexes and scenario groups.
- **modelos.cpp**. It generates the coefficients of the optimization problem, that is, the vectors a, c, b and the matrices A, A', B and B'. In our case, this function either read data file or generates it seudorandomly.
- **param3.cp**. It structures the coefficients for the *DEM* by setting all the indixes of the objective function, nonzero elements of the matrix of constraints, bounds of variables and bounds of constraints as required by *COIN-OR*.
- **param4.cpp**. It structures the coefficients in similar way as param3.cpp, but for the scenario-cluster models.
- **optimo.cpp**. It checks the integrality of the x variables in the relaxed problem.
- **ooptimo.cpp**. It checks the integrality and nonanticipativity constraints for the x variables between scenario-cluster problems.
- **ooptimo3.cpp**. It checks the nonanticipativity constraints for the continuos y variables between scenario-cluster problems.

To solve any Multistage Stochastic Mixed Integer Problem (MSMIP) with T stages and r contingencies or outlooks at each stage (r^{T-1} scenarios) we have to determine the problem dimensions and tolerances. Also we have used a file named **const_MS.h** with the integer constants, as well as the dimensions of the arrays of the problem. We introduce in the auxiliary file **pm.header** the neccessary includes to the files .hpp of *COIN-OR* and to the files .h of C++.

1. In order to use the *COIN-OR* library, we have introduced the coefficients of each optimization model by using indices representation. Arrays such as, `dobj[]`, `dels[]`, `nrowindx[]`, `mcolindx[]`, `drowlo[]`,`drowup[]`, `dcollo[]` or `dcolup[]`, must be dimensioned with integer constants such as: `ncols` (number of variables), `nrows` (number of constraints) and `nelements` (number of nonzero elements). All of these integer constants are defined in file **const_MS.h**.
2. One of the essential decisions to structure the implementation of the algorithm is the way of building the q clusters.

   We have built the clusters in our computational experience by using the following idea. If $t = 1$ we generate $q = r^1 = r$ clusters ($t = 2$ for $q = r^2$ or $t = 3$ for $q = r^3$); and then, the clusters are explicitly linked by nonanticipativity constraints for the stage 1 (if $t = 1$) (stages 1,2 if $t = 2$ or stages 1,2,3 if $t = 3$).

   That is, for the $q = r$ (or $q = r^2$ or $q = r^3$) cluster problems, we are considering the *splitting* variable representation of the variables of the stages $t = 1$ (or $t = 1,2$ or $t = 1,2,3$). Consequently they are explicitly linked by nonanticipativity constraints in the $q = r$ (or $q = r^2$ or $q = r^3$) cluster models.

Moreover, we are considering the *compact* variable representation of the variables of the stages $t = 2, 3$ (or $t = 3$ or none of them, because in the last stage, T, there is not nonanticipativity constraints).

For example, in case of $q = r$ we are considering the splitting variable representation of the model for the variables of stage 1 in the $q = r$ cluster models.

To select each of these cases: $q = r$, $q = r^2$ or $q = r^3$ we use the following integer constants.

We set `nc1`$=(T-1)-t$ for t=1,2,3 and `nc2`=0 in the file **const_MS.h** as follows,

```
#define nc1 1
#define nc2 0
```

In our computational experience (we have T=4 periods) the posibilities are,

(`nc1`,`nc2`)=(2,0) for $q = r$,

(`nc1`,`nc2`)=(1,0) for $q = r^2$, and

(`nc1`,`nc2`)=(0,0) for $q = r^3$.

3. Other important decision in the implementation of the algorithm is to choose between the strategies to branch with the 0-1 variables. Consequently the type of problems to solve at each cluster $p = 1, \ldots, q$ are MIP^p mixed integer problems with more or less 0-1 variables.

   To do this we define the variable `ivartipo` in file **const_MS.h** as follows: if `ivartipo` is equal to 1, we will use BFC1 (INTEGER), if `ivartipo` is equal to 2, we will use BFC2 (MIXED), and if `ivartipo` is equal to 3, we will use BFC3 (MIXED-INTEGER). In the file **const_MS.h** the sentence is for example

   ```
   #define ivartipo 1
   ```

   That is, by using `ivartipo` we select the strategy to solve the problem.

   (a) Strategy BFC1 or INTEGER: (`ivartipo`=1)

   - Branch on/fix the 0-1 variable until stage $T-1$ (except the last stage, since there are not nonanticipativity constraints in the last stage).
   - Solve the mixed integer problems MIP^p for each cluster p, in these problems all 0-1 variables are considered as integer except the fixed variables.
   - The relaxed *DEM* submodels named MIP^{TNF} and MIP^f are mixed integer problems with 0-1 variables in the last stage.

   (b) Strategy BFC2 or MIXED: (`ivartipo`=2)

   - Branch on/fix the 0-1 variable until stage t=1 for $q = r$ (t=2 for $q = r^2$ or t=3 for $q = r^3$). The clusters are explicitly linked by nonanticipativity constraints for the stage 1, (stages 1,2 or stages 1,2,3).
   - Solve the mixed integer problems MIP^p for each cluster p. In these problems we consider as integer variables the 0-1 variables related the stages without explicit nonanticipativity. That is,
     - If $q = r$ we consider as continuous variables, the 0-1 variables of stages 1 and we consider as integer variables, the 0-1 variables of the stages 2 and 3 and of the period $T = 4$,
     - If $q = r^2$ we consider as continuous variables, the 0-1 variables of the stages 1,2 and we consider as integer variables, the 0-1 variables of the stage 3 and of the period $T = 4$
     - If $q = r^3$ we consider as continuous variables, the 0-1 variables of the stages 1,2,3 and we consider as integer variables, the 0-1 variables of the period $T = 4$.

- The problems MIP^{TNF} and MIP^f are mixed integer problems and they have 0-1 variables in the stages without explicit nonanticipativity constraints. In these models the integrality constraints for the 0-1 variables in the stages with implicit nonanticipativity are relaxed.

(c) Strategy BFC3 or MIXED-INTEGER: (`ivartipo`=3)

- Branch on/fix the 0-1 variable until stage t=1 for $q = r$ (t=2 for $q = r^2$ or t=3 for $q = r^3$). The clusters are explicitly linked by nonanticipativity constraints for the stage 1, (stages 1,2 or stages 1,2,3).
- Solve the mixed integer problems MIP^p for each cluster p, in these problems all 0-1 variables are considered as integer except the fixed variables.
- The problems MIP^{TNF} and MIP^f are mixed integer problems with 0-1 variables in the stages without explicit nonanticipativity.

Note that for $q = r^3$ we branch on/fix in the same way for BFC1 and BFC3 .

4. At the top of the procedure, other relevant decision is the lower bound to calculate at the root node. That is, a lower bound of the objective function value (remind that we are in a minimization problem).

   The `inicio` constant is used in **const_MS.h** to make this choice. If `inicio` is equal to 0, the lower bound of the objective function is Z_{LP}, if `inicio` is equal to 1 the lower bound of the objective function is $\max\{Z_{LP}, Sum_p Z_t^p\}$ and if `inicio` is equal to 2 we have to solve $p = 1, \dots, q$ mixed integer problems with optimal function values $Z_{0,0}^p$ and the lower bound of the objective function is $Sum_p Z_{0,0}^p$. In our computational experience this constant always takes the value 2.

4 General description of the algorithm *BFC-MSMIP*

We present the algorithm *BFC-MSMIP* for using the three strategies presented above, *BFC1*, *BFC2* and *BFC3*. We have chosen the *depth first* strategy for the TNF branching selection. When there is a guarantee that the incumbent solution could not be produced by the successor $TNFs$ in both branches, then a *bactracking* to the immediate ancestor TNF is performed.

In the Steps 4, 7, 9 and 10 of the procedure, the subindex i denotes the index in $\mathscr{I}$, i.e., the corresponding set over which the algorithm proceeds by branching on. The cardinality of this last set, $|\mathscr{I}|$ depends upon the branching strategy to select, see Table 2 below.

The procedure is as follows:

Procedure

Step 0: Initialize $\overline{Z}_{MIP} := +\infty$.

Step 1: Solve the scenario-cluster related mixed 0-1 problems, MIP^p, $\forall p = 1, \dots, q$, and compute the lower bound $\underline{Z}_0 = Z_{0,0}^t$, for the choice of $q(t) = r^t$, to use for the root node.

If there is any (0-1) x variable that takes different values in the scenario-related clusters, then go to Step 2.

If there is any (continuous) $y-$variables that takes different values in the scenario-related clusters, then go to Step 6.

Otherwise, the optimal solution to the original problem has been found and, so, $\overline{Z}_{MIP} := \underline{Z}_0$ and stop.

Step 2: Initialize $i := 1$ and go to Step 4.

Step 3: Reset $i := i+1$.

If $i = |\mathscr{I}|+1$ where $\mathscr{I}$ is the subset of variables until stage $T-1$ in $BFC1$ or until stage t with $t \leq T-1$ in $BFC2$ or $BFC3$ then go to Step 8.

Step 4: Branch $x_i^g := 0, \forall g \in \mathscr{G}^p,\ p = 1,\ldots,q$.

Step 5: Solve the mixed 0-1 scenario-cluster related submodels, $MIP^p\ \forall p = 1,\ldots,q$, for the choice of $q(t) = r^t$, $t = 1,...,T-1$ and compute the bound $\underline{Z}_i = Z^t_{i,0}$, for $BFC1$ and $BFC3$, and $\underline{Z}_i = Z^t_{i,t}$, for $BFC2$.

If $\underline{Z}_i \geq \overline{Z}_{MIP}$ then go to Step 7.

If there is any x variable that either takes fractional values or takes different values for some of the q scenario *clusters* then go to Step 3.

If all the y variables take the same value for all scenario *clusters* $p = 1,\ldots,q$ then update $\overline{Z}_{MIP} := \underline{Z}_i$ and go to Step 7.

Step 6: Solve the mixed 0-1 submodel $MIP^{TNF}_{i,T-1}$ for *BFC1* or $MIP^{TNF}_{i,t}$ for *BFC2* and *BFC3* to satisfy the nonanticipativity constraints for the y variables in the given *TNF integer set*. Notice that the solution value is denoted by z_i^{TNF}.

Update $\overline{Z}_{MIP} := \min\{z_i^{TNF}, \overline{Z}_{MIP}\}$.

If $i = |\mathscr{I}|$, then go to Step 7.

Solve the mixed 0-1 submodel $MIP^f_{i,T-1}$ for *BFC1* or $MIP^f_{i,t}$ for *BFC2* and *BFC3*, where the fractional x variables are the non-yet *BF* branched on at the current TNF. Notice that the solution value is denoted by z_i^f.

If $z_i^f < z_i^{TNF}$ and $z_i^f < \overline{Z}_{MIP}$ and all the fractional x variables take 0-1 values in the solution of the model MIP_i^f, update $\overline{Z}_{MIP} := z_i^f$ and go to Step 7.

If $z_i^{TNF} = z_i^f$ or $z_i^f \geq \overline{Z}_{MIP}$, then go to Step 7, otherwise go to Step 3.

Step 7: Prune the branch.

If $x_i^g = 0, \forall g \in \mathscr{G}^p,\ p = 1,\ldots,q$, then go to Step 10.

Step 8: Reset $i := i-1$.

If $i = 0$ then stop, since the optimal solution $\overline{Z}_{MIP}$ has been found.

Step 9: If $x_i^g = 1, \forall g \in \mathscr{G}^p,\ p = 1,\ldots,q$, then go to Step 8.

Step 10: Branch $x_i^g := 1, \forall g \in \mathscr{G}^p,\ p = 1,\ldots,q$.

go to Step 5.

5 Description of the implementation

The steps of the main program **BFC_MS.cpp** are as follows:

1. The sentences `#include "pm.header"`, `#include "const_MS.h"` and the `#include` for the external and own functions as **vectores.cpp, modelos.cpp, param3.cp, param4.cpp, optimo.cpp, ooptimo.cpp** and **ooptimo3.cpp**. The declaration and dimension of all the own external functions and all the external arrays are performed.

2. Initializations. For example, the upper bound of the objetive function, $\overline{Z}_{MIP} = \infty$.

3. Calls to external function **vectores** to build the scenario tree and to establish the relationship between scenario-cluster problems and the *DEM*.

4. Calls to external function **modelos** to generate the coefficients of the minimization problem.

5. Pointers for the solver, for each of the `numberOfModels+1` submodels, by using `OsiClpSolverInterface` as follows,

   ```
   OsiClpSolverInterface *sol1;
   sol1=new OsiClpSolverInterface[numberOfModels+1];
   ```

   With these sentences, we declare a pointer to an array in which we store the cluster models in the first positions (from `0` until `numberOfModels-1`), and the whole model in the last position (`numberOfModels`).

6. Calls to external function **param3** to introduce the parameters generated in function **modelos**, as the coefficients of the objective function and constraints for the definition of the whole problem in compact representation. This problem is saved in the last position, `numberOfModels`, of the array of models.

   The following calls avoid to load the whole problem,

   ```
   CoinPackedMatrix ACOMPLETE(true,nrowindx,mcolindx,dels,noncero);
   sol1[numberOfModels].loadProblem (ACOMPLETE,dcolo,dcolup,dobj,drowlo,drowup);
   ```

   This structure of storing allows to calculate a lower bound of the objective function by solving the last model, in numberOfModel position, i.e., the linear problem (linear relaxation) with the nonanticipativity constraints, Z_{LP}. (Remind that we are at the root node).

7. Calls to external function **param4** to introduce the parameters generated in function **modelos**, as coefficients of the objective function and constraints for the definition of each of the $q = r$ (or $q = r^2$ or $q = r^3$) cluster problems in compact representation.

   There are `numberOfModels` models, one for each cluster, and the external function gets the corresponding parameter information given by **modelos** and **param4** to define each cluster model. This structure avoid to load `numberOfModels` cluster problems with the calls to the following functions, where `imodel` varies from `0` to `numberOfModels-1`:

   ```
   CoinPackedMatrix ACLUSTER(true,nrowindxq,mcolindxq,delsq,nonceroq);
   sol1[imodel].loadProblem (ACLUSTER,dcoloq,dcolupq,dobjq,drowloq,drowupq);
   ```

   We can calculate the lower bound of the objective function by solving each of the `numberOfModels` mixed integer problems, MIP_0^p, $p = 1,...,q$ without the nonanticipativity constraints, and compute the lower bound as, $Sum_p Z_{0,0}^p$. (Remind that we are at the root node).

   To solve each of the mixed integer models we use the calls to the functions:

   ```
   CbcModel pm0(sol1[imodel]);
   pm0.branchAndBound();
   ```

8. Before starting the branching procedure, we check with the external function **ooptimo** if the corresponding 0-1 variables satisfy the nonanticipativity constraints; and we check with the external function **ooptimo3** if the corresponding continuous variables satisfy the nonanticipativity constraints. If the 0-1 variables and the continuous variables satisfy the nonanticipativity constraints, we have obtained the optimal solution. In other case we start branching on the 0-1 variables by groups of scenarios.

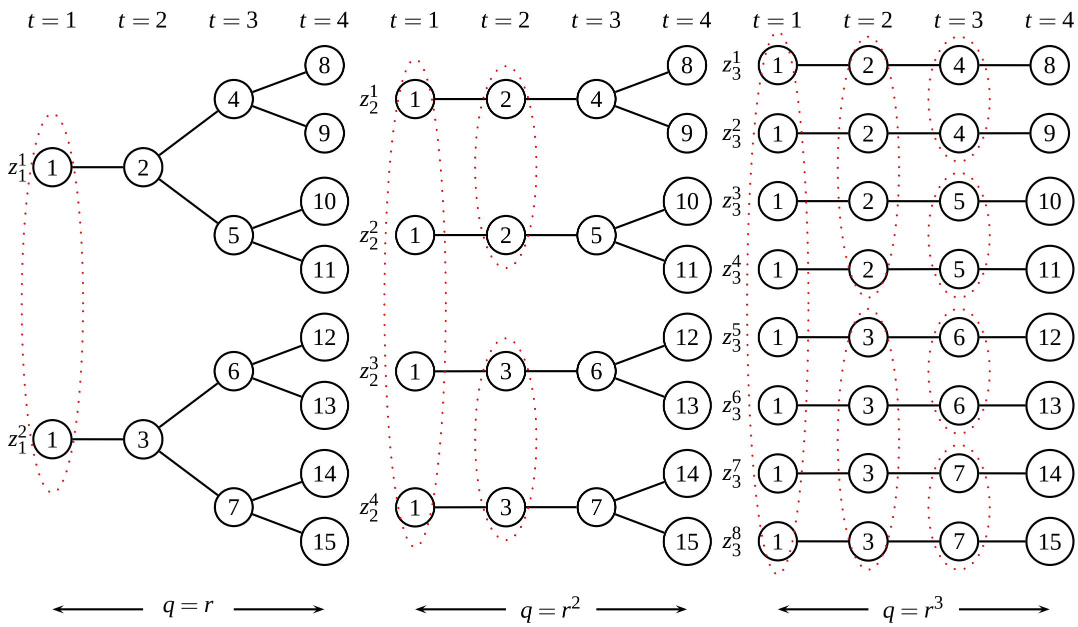

Figure 1: Scenario cluster partitioning, for $q = r = 2$ (left), $q = r^2 = 4$ (central) and $q = r^3 = 8$ (right)

9. Branching procedure. Notice that the number of clusters, $q = r$, $q = r^2$ or $q = r^3$ has been previously selected. We show in Figure 1 an example of the scenario cluster partitioning for these values of q where $T = 4$. Observe that we are in the left situation of the figure if $q = r = 2$, in the central situation if $q = r^2 = 4$ and in the right situation if $q = r^3 = 8$.

   We begin by fixing to zero the first 0-1 variable, i.e $i = 1$, and then, $x_i^g = 0$, $\forall g \in \mathscr{G}^p$, at each cluster, i.e. $p = 1,...,q$.

   Firstly, we explain the branching order for the strategies BFC2 and BFC3 because in both strategies we branch until the stage t, where we consider the splitting representation of the 0-1 variables.

   For the strategy BFC1 we branch until the stage $T-1$, and we have variables in splitting representation until stage t, and variables in compact representation from stagem $t+1$.

   Let x_{pi}^g for $g \in \mathscr{G}^p$, $p = 1,\dots,q$, denote the ith variable of scenario group g and cluster p (see Figure 2).

   For strategies BFC2 and BFC3,

   - If we have $q = r = 2$ clusters, then $\mathscr{G}^1 = \{1,2,4,5,8,9,10,11\}$ and $\mathscr{G}^2 = \{1,3,6,7,12,13,14,15\}$. Then, $x_{1i}^1 = x_{2i}^1 = 0$, $x_{1i+1}^1 = x_{2i+1}^1 = 0,\dots$, until the last 0-1 variable of the first stage ($g = 1$).
   - If we have $q = r^2 = 4$ clusters, then $\mathscr{G}^1 = \{1,2,4,8,9\}$, $\mathscr{G}^2 = \{1,2,5,10,11\}$, $\mathscr{G}^3 = \{1,3,6,12,13\}$ and $\mathscr{G}^4 = \{1,3,7,14,15\}$ and updating $i = i+1$ for each variable:
     $x_{1i}^1 = x_{2i}^1 = x_{3i}^1 = x_{4i}^1 = 0$, $x_{1i+1}^1 = x_{2i+1}^1 = x_{3i+1}^1 = x_{4i+1}^1 = 0$, $\dots$, until the last 0-1 variable of the first stage ($g = 1$ and $p = 1,2,3,4$).
     Then, $x_{1i}^2 = x_{2i}^2 = 0$, $x_{1i+1}^2 = x_{2i+1}^2 = 0$, $\dots$, until the last 0-1 variable of the second stage ($g = 2$ and $p = 1,2$).
     And, $x_{3i}^3 = x_{4i}^3 = 0$, $x_{3i+1}^3 = x_{4i+1}^3 = 0$, $\dots$, until the last 0-1 variable of the second stage ($g = 3$ and $p = 3,4$).
   - If we have $q = r^3 = 8$ clusters, then $\mathscr{G}^1 = \{1,2,4,8\}$, $\mathscr{G}^2 = \{1,2,4,9\}$,$\mathscr{G}^3 = \{1,2,5,10\}$, $\mathscr{G}^4 = \{1,2,5,11\}$, $\mathscr{G}^5 = \{1,3,6,12\}$, $\mathscr{G}^6 = \{1,3,6,13\}$,$\mathscr{G}^7 = \{1,3,7,14\}$ and $\mathscr{G}^8 = \{1,3,7,15\}$. And

updating $i = i+1$, we fix: $x^1_{1i} = x^1_{2i} = x^1_{3i} = x^1_{4i} = x^1_{5i} = x^1_{6i} = x^1_{7i} = x^1_{8i} = 0$, and, $x^1_{1i} = x^1_{2i+1} = x^1_{3i+1} = x^1_{4i+1} = x^1_{5i+1} = x^1_{6i+1} = x^1_{7i+1} = x^1_{8i+1} = 0, \ldots$, until the last 0-1 variable of the first stage ($g = 1$ and $p = 1, \ldots, 8$).

Then $x^2_{1i} = x^2_{2i} = x^2_{3i} = x^2_{4i} = 0$, $x^2_{1i+1} = x^2_{2i+1} = x^2_{3i+1} = x^2_{4i+1} = 0, \ldots$, until the last 0-1 variable of the second stage ($g = 2$ and $p = 1,2,3,4$).
$x^3_{5i} = x^3_{6i} = x^3_{7i} = x^3_{8i} = 0$, $x^3_{5i+1} = x^3_{6i+1} = x^3_{7i+1} = x^3_{8i+1} = 0, \ldots$, until the last 0-1 variable of the second stage ($g = 3$ and $p = 5,6,7,8$).
And finally,
$x^4_{1i} = x^4_{2i} = 0$, $x^4_{1i+1} = x^4_{2i+1} = 0, \ldots$, until the last 0-1 variable of the $T-1$ stage ($g = 4$ and $p = 1,2$).
$x^5_{3i} = x^5_{4i} = 0$, $x^5_{3i+1} = x^5_{4i+1} = 0, \ldots$, until the last 0-1 variable of the $T-1$ stage ($g = 5$ and $p = 3,4$).
$x^6_{5i} = x^6_{6i} = 0$, $x^6_{5i+1} = x^6_{6i+1} = 0, \ldots$, until the last 0-1 variable of the $T-1$ stage ($g = 6$ and $p = 5,6$).
$x^7_{7i} = x^7_{8i} = 0$, $x^7_{7i+1} = x^7_{8i+1} = 0, \ldots$, until the last 0-1 variable of the $T-1$ stage ($g = 7$ and $p = 7,8$).

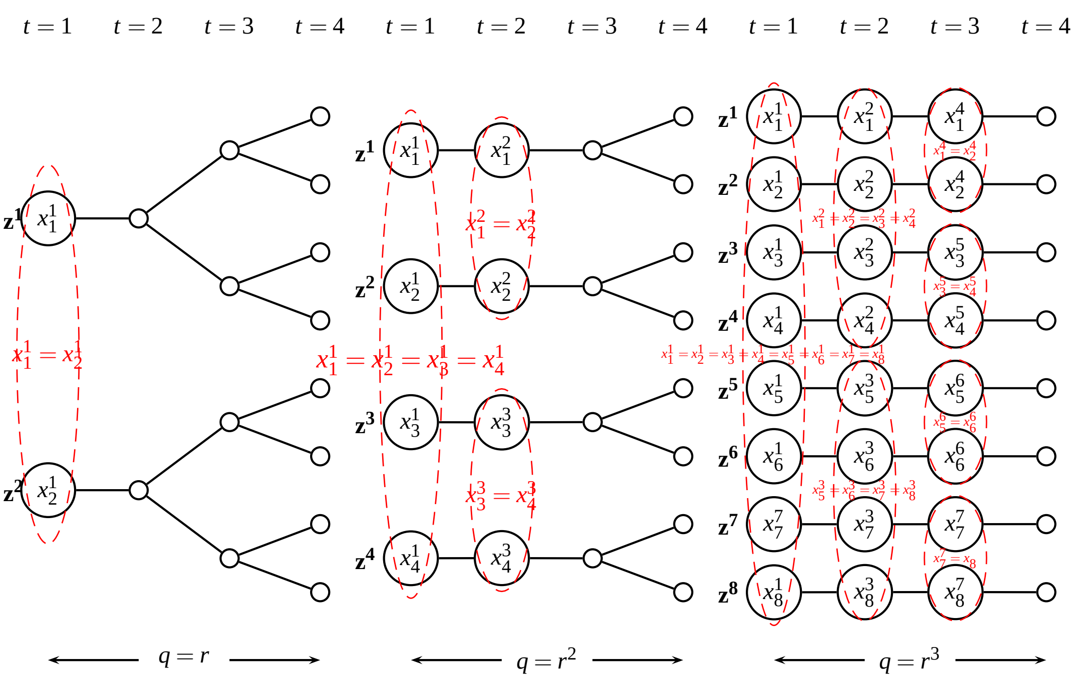

Figure 2: x^g_p variables, for $q = r$, $q = r^2$, and $q = r^3$, where $r = 2$

For strategy BFC1, we show only one case because the others are very similar.

If we have $q = r = 2$ clusters, then $\mathscr{G}^1 = \{1,2,4,5,8,9,10,11\}$ and $\mathscr{G}^2 = \{1,3,6,7,12,13,14,15\}$ and we branch on/fix until stage $T-1$.

Since the variables are stored in compact representation from stage $t+1 = 2$ to stage $T-1 = 3$, we must fix $x^1_{1i} = x^1_{2i} = 0$, $x^1_{1i+1} = x^1_{2i+1} = 0, \ldots$, until the last 0-1 variable of the first stage with the same value in the two clusters $p = 1,2$ ($g = 1$).

Then $x^2_{1i} = 0$, $x^2_{1i+1} = 0$, $\dots$, until the last 0-1 variable of the second stage since the variables are compacted in x^2_{1i}, for the cluster $p = 1$ $(g = 2)$.

Then $x^3_{2i} = 0$, $x^3_{2i+1} = 0$, $\dots$, until the last 0-1 variable of the second stage since now the variables are compacted in x^3_{2i}, for the cluster $p = 2$ $(g = 3)$.

$x^4_{1i} = 0$, $x^4_{1i+1} = 0$, $\dots$, until the last 0-1 variable of the $T-1$ stage since now the variables are compacted in x^4_{1i}, for the cluster $p = 1$ $(g = 4)$.

$x^5_{1i} = 0$, $x^5_{1i+1} = 0$, $\dots$, until the last 0-1 variable of the $T-1$ stage since now the variables are compacted in x^5_{2i}, for the cluster $p = 1$ $(g = 5)$.

$x^6_{2i} = 0$, $x^6_{2i+1} = 0$, $\dots$, until the last 0-1 variable of the $T-1$ stage since now the variables variables are compacted in x^6_{2i}, for the cluster $p = 2$ $(g = 6)$.

$x^7_{2i} = 0$, $x^7_{2i+1} = 0$, $\dots$, until the last 0-1 variable of the $T-1$ stage since now the variables are compacted in x^7_{2i}, for the cluster $p = 2$ $(g = 7)$.

10. We solve the p clusters and compute the lower bound $\underline{Z}_i$, for example with the expression $\underline{Z}_i = Z^t_{i,0} = \sum_{p=1}^{q} z^p_{i,0}$ for $BFC3$.

    If $\underline{Z}_i \geq \overline{Z}_{MIP}$, then prune the branch.

    If $\underline{Z}_i < \overline{Z}_{MIP}$ and all variables satisfy the nonanticipativity constraints, then update the lower bound $\overline{Z}_{MIP} = \underline{Z}_i$ and prune the branch.

    If $\underline{Z}_i < \overline{Z}_{MIP}$ and there is any x variable that either takes fractional values or takes different values for some of the q scenario *clusters*, then $i = i+1$ and $x^g_i = 0\ \forall g \in \mathcal{G}^p$, $p = 1, ..., q$.

    If all 0-1 variables satisfy the nonanticipativity constraints but the continuous variables do not satisfy the nonanticipativity constraints, then solve the MIP^{TNF}_i submodel and compute z^{TNF}_i. Update $\overline{Z}_{MIP} = min\{z^{TNF}_i, \overline{Z}_{MIP}\}$. Solve the MIP^f_i submodel and compute z^f_i.

    If $z^f_i < z^{TNF}_i$ and $z^f_i < \overline{Z}_{MIP}$ and all 0-1 variables satisfy the nonanticipativity constraints. Update $\overline{Z}_{MIP} = z^f_i$.

    If $z^f_i = z^{TNF}_i$ or $z^f_i \geq \overline{Z}_{MIP}$, then prune the branch.

11. If we have fixed to 0 all 0-1 variables until the stage that we have decided, we go up fixing to 1 the last variable.

    $x^g_i = 1\ \forall g \in \mathcal{G}^p$, $p = 1, ..., q$

    We solve the p cluster models and compute $\underline{Z}_i = Z^t_{i,0} = \sum_{p=1}^{q} z^p_{i,0}$, for example with this expression for each lower bound at each node by using the strategy $BFC3$.

    Repeat the step 9.

    Update $i = i-1$.

6 Another procedure for branching/storing the variables

Alternatively to the criteria given in item 9 of Section 5, there is another equivalent procedure for storing the variables and, consequently, branching. We will use the following notation for this purpose:

(In Figure 3 we introduce the new notation for the indexes in the variables). Let x^{tp}_i denote the corresponding variable for $t = 1, 2, T-1 = 3$, $p = 1, ..., q$, and where i varies in the subset of variables of the t-stage. Now, we have to fix $x^{tp}_i = x^{tp'}_i = 0$, $p, p' = 1, \dots, q$ until stage t.

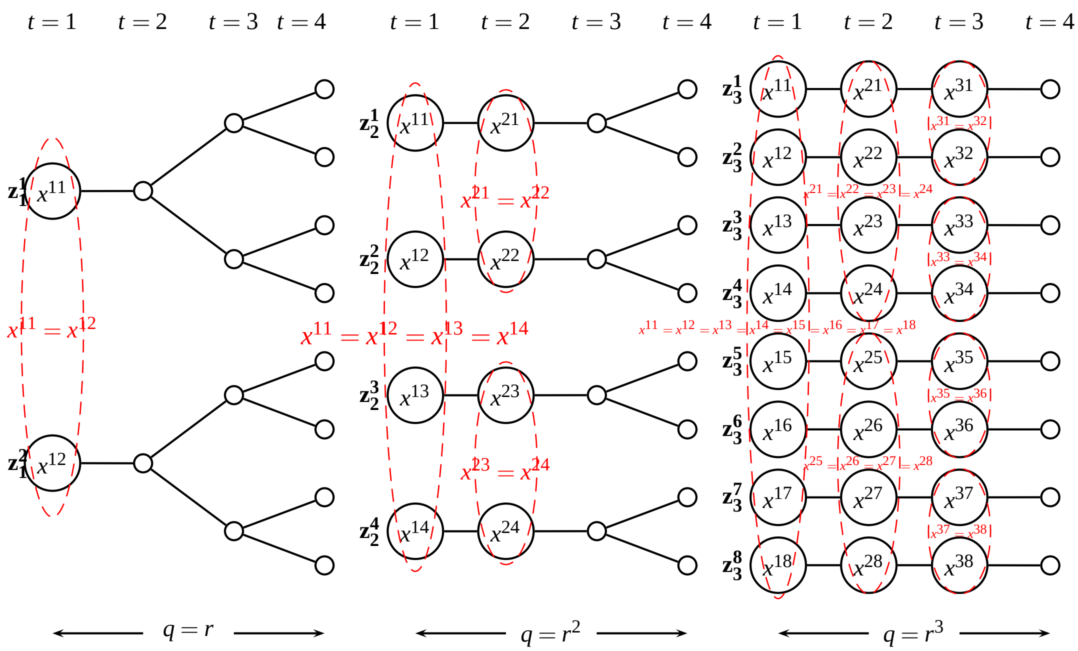

Figure 3: x^{tp} variables, for $q=r$, $q=r^2$, and $q=r^3$, where $r=2$

For strategies BFC2 and BFC3,

- If we have $q=r=2$ clusters, updating $i=i+1$:

  $x_i^{11}=x_i^{12}=0$, $x_{i+1}^{11}=x_{i+1}^{12}=0$, …, until the last 0-1 variable of the first stage, for clusters $p=1,2$.

- If we have $q=r^2=4$ clusters, updating $i=i+1$:

  $x_i^{11}=x_i^{12}=x_i^{13}=x_i^{14}=0$, $x_{i+1}^{11}=x_{i+1}^{12}=x_{i+1}^{13}=x_{i+1}^{14}=0,\ldots$, until the last 0-1 variable of the first stage, for clusters $p=1,2,3,4$.

  Then, $x_i^{21}=x_i^{22}=0$, $x_{i+1}^{21}=x_{i+1}^{22}=0$, …, until the last 0-1 variable of the second stage, for clusters $p=1,2$.

  And, $x_i^{23}=x_i^{24}=0$, $x_{i+1}^{23}=x_{i+1}^{24}=0$, …, until the last 0-1 variable of the second stage, for clusters $p=3,4$.

- If we have $q=r^3=8$ clusters, we have to branch on/fix variables until stage $T-1=3$, updating $i=i+1$:

  $x_i^{11}=x_i^{12}=x_i^{13}=x_i^{14}=x_i^{15}=x_i^{16}=x_i^{17}=x_i^{18}=0$,

  $x_{i+1}^{11}=x_{i+1}^{12}=x_{i+1}^{13}=x_{i+1}^{14}=x_{i+1}^{15}=x_{i+1}^{16}=x_{i+1}^{17}=x_{i+1}^{18}=0$, …, until the last 0-1 variable of the first stage, for clusters $p=1,2,3,4,5,6,7,8$.

  $x_i^{21}=x_i^{22}=x_i^{23}=x_i^{24}=0$, $x_{i+1}^{21}=x_{i+1}^{22}=x_{i+1}^{23}=x_{i+1}^{24}=0,\ldots$, until the last 0-1 variable of the second stage, for clusters $p=1,2,3,4$.

  $x_i^{25}=x_i^{26}=x_i^{27}=x_i^{28}=0$, $x_{i+1}^{25}=x_{i+1}^{26}=x_{i+1}^{27}=x_{i+1}^{28}=0,\ldots$, until the last 0-1 variable of the second stage, for clusters $p=5,6,7,8$.

  $x_i^{31}=x_i^{32}=0$, $x_{i+1}^{31}=x_{i+1}^{32}=0,\ldots$, until the last 0-1 variable of the $T-1$ stage, for clusters $p=1,2$.

  $x_i^{33}=x_i^{34}=0$, $x_{i+1}^{33}=x_{i+1}^{34}=0,\ldots$, until the last 0-1 variable of the $T-1$ stage, for clusters $p=3,4$.

$x_i^{35} = x_i^{36} = 0$, $x_{i+1}^{35} = x_{i+1}^{36} = 0$, $\ldots$, until the last 0-1 variable of the $T-1$ stage, for clusters $p = 5,6$.

And finally, $x_i^{37} = x_i^{38} = 0$, $x_{i+1}^{37} = x_{i+1}^{38} = 0$, $\ldots$, until the last 0-1 variable of the $T-1$ stage, for clusters $p = 7,8$.

For strategy BFC1 we show only one case because the others are similar.

If we have $q = r = 2$ clusters and we branch on/fix until stage $T-1$, updating $i = i+1$:

$x_i^{11} = x_i^{12} = 0$, $x_{i+1}^{11} = x_{i+1}^{12} = 0$, $\ldots$, until the last 0-1 variable of the first stage, for clusters $p = 1,2$.

The variables of the second and the third stage are compacted in each cluster.

$x_i^{21} = 0$, $x_{i+1}^{21} = 0$, $\ldots$, until the last 0-1 variable of the second stage ($t = 2$) for the cluster $p = 1$.

$x_i^{22} = 0$, $x_{i+1}^{22} = 0$, $\ldots$, until the last 0-1 variable of the second stage ($t = 2$) for the cluster $p = 2$.

$x_i^{31} = 0$, $x_{i+1}^{31} = 0$, $\ldots$, until the last 0-1 variable of the stage ($T-1$) for the cluster $p = 1$.

$x_i^{32} = 0$, $x_{i+1}^{32} = 0$, $\ldots$, until the last 0-1 variable of the stage ($T-1$) for the cluster $p = 2$.

7 Computational experience. Instances P1 to P16

The instances named P1 to P16 have been generated as a perturbation of a pilot case. As a result, small, medium and large scale sized instances have been generated.

In this section we present some of the main results obtained in the computational experience while optimizing a general multistage mixed 0-1 problem. Table 1 gives the structure of the scenario tree, and the linear relaxation and stochastic solutions for the *DEM*. The headings are as follows: Tree, scenario tree in terms of the outcomes and the number of stages; $|\Omega|$, number of scenarios; $|\mathscr{G}|$, number of scenario groups; Z_{LP} solution value of the *LP* relaxation of the original problem; z_{MIP}, solution value of the original problem; *GAP*, optimality gap defined as $\frac{(Z_{MIP}-Z_{LP})}{Z_{LP}}$ (in %); and T_{LP}, elapsed time (in seconds) to obtain the Z_{LP} solution.

Table 1: Stochastic solution

Instance	Tree	$\lvert\Omega\rvert$	$\lvert\mathscr{G}\rvert$	Z_{LP}	z_{MIP}	*GAP*	T_{LP}
P1	2^3	8	15	4116.3	408827	9831.9	0.0
P2	2^3	8	15	11753.2	624636	5214.6	0.0
P3	2^3	8	15	11994.4	664294	5438.4	0.0
P4	2^3	8	15	5262.5	356204	6668.7	0.1
P5	3^3	27	40	4952.5	436517	8714.0	0.1
P6	3^3	27	40	4577.7	335477	7228.5	0.1
P7	2^3	8	15	24622.6	1398840	5581.1	0.1
P8	3^3	27	40	10811.9	617938	5615.3	0.1
P9	3^3	27	40	11781.1	612964	5102.9	0.1
P10	4^3	64	85	13089.5	810074	6088.7	0.3
P11	6^3	216	259	6126.2	604693	9770.6	0.7
P12	6^3	216	259	8758.5	763052	8612.1	1.3
P13	7^3	343	400	6245.6	544014	8610.4	1.6
P14	9^3	729	820	5263.66	473741	8900.2	3.0
P15	10^3	1000	1111	4973.9	468610	9321.3	8.6
P16	10^3	1000	1111	6117.4	654229	10594.5	10.5

Table 2 shows, in the three last columns, the number of 0-1 variables over which the algorithm proceeds by branching on. The cardinality of this set, $|\mathscr{I}|$, depends upon the branching strategy to select. The new headings are as follows: $|\mathscr{I}^1|$, number of the 0-1 variables to branch on/fix until stage 1; $|\mathscr{I}^2|$, number of the 0-1 variables to branch on/fix until stage 2; and $|\mathscr{I}^3|$, number of the 0-1 variables to branch on/fix until stage 3.

Table 2: Number of 0-1 variables to branch on/fix

Instance	Tree, r^3	$\lvert\Omega\rvert$	$\lvert\mathscr{G}\rvert$	$\lvert\mathscr{I}^1\rvert$	$\lvert\mathscr{I}^2\rvert$	$\lvert\mathscr{I}^3\rvert$
P1	2^3	8	15	5	15	35
P2	2^3	8	15	10	30	70
P3	2^3	8	15	10	30	70
P4	2^3	8	15	10	30	70
P5	3^3	27	40	5	20	65
P6	3^3	27	40	5	20	65
P7	2^3	8	15	20	60	140
P8	3^3	27	40	10	40	130
P9	3^3	27	40	10	40	130
P10	4^3	64	85	10	40	210
P11	6^3	216	259	6	42	258
P12	6^3	216	259	8	56	344
P13	7^3	343	400	6	48	342
P14	9^3	729	820	5	50	455
P15	10^3	1000	1111	5	55	555
P16	10^3	1000	1111	6	66	666

Notice that for BFC1, this number is always equal to $|\mathscr{I}^3|$.

For instance P4, we have $n_x = 10$.

- If $q = r^1 = 2$, we have 2 clusters, and $|\mathscr{I}^1| = n_x = 10$.

  In the first stage we branch on/fix $x^1_{1i} = x^1_{2i} = 0$ or 1, in cluster $p = 1$ we use x^1_{1i} for $i = 1,\dots 10$ and in cluster $p = 2$ we use x^1_{2i} for $i = 1,\dots 10$, there are 10 variables in each cluster.

- If $q = r^2 = 4$, we have 4 clusters, and $|\mathscr{I}^2| = n_x(1+r) = 10+20 = 30$.

  In the first stage we branch on/fix $x^1_{1i} = x^1_{2i} = x^1_{3i} = x^1_{4i} = 0$ or 1 $i = 1,\dots 10$, there are 10 variables in each cluster.

  In the second stage we branch on/fix $x^2_{1i} = x^2_{2i} = 0$ or 1 $i = 1,\dots 10$ then we branch on/fix $x^3_{3i} = x^3_{4i} = 0$ or 1 $i = 1,\dots 10$, there are 20 in each cluster.

- If $q = r^3 = 8$, we have 8 clusters, and $|\mathscr{I}^3| = n_x(1+r+r^2) = 10+20+40 = 70$.

  In the first stage we branch on/fix $x^1_{1i} = x^1_{2i} = x^1_{3i} = x^1_{4i} = x^1_{5i} = x^1_{6i} = x^1_{7i} = x^1_{8i} = 0$ or 1 $i = 1,\dots 10$, there are 10 variables in each cluster.

  In the second stage we branch on/fix $x^2_{1i} = x^2_{2i} = x^2_{3i} = x^2_{4i} = 0$ or 1 $i = 1,\dots 10$ then we branch on/fix $x^3_{5i} = x^3_{6i} = x^3_{7i} = x^3_{8i} = 0$ or 1, $i = 1,\dots 10$, there are 20 variables in each cluster.

  In the third stage we branch on/fix $x^4_{1i} = x^4_{2i} = 0$ or 1, $i = 1,\dots 10$, then $x^5_{3i} = x^5_{4i} = 0$ or 1, and we branch on/fix $x^6_{5i} = x^6_{6i} = 0$ or 1, $i = 1,\dots 10$, then $x^7_{7i} = x^7_{8i} = 0$ or 1, $i = 1,\dots 10$, there are 40 variables in each cluster.

When we solve the q cluster models stage by stage we only solve the clusters p that have changed.

If we branch on/fix until $t = 2$ (r^2=4) when we branch on/fix $x^2_{1i} = x^2_{2i} = 0$ or 1 we use the values of the variables x^2_{1i} and x^2_{2i} to calculate z^1 and z^2 hence we do not solve the submodels for $p = 3$ and $p = 4$ to obtain

z^3 and z^4 since these values will not change. Then, we branch on/fix $x^2_{3i} = x^2_{4i} = 0$ or 1, and we use the values of the variables x^2_{3i} and x^2_{4i} to calculate z^3 and z^4. For the same reason, we will not solve the submodels for $p = 1$ and $p = 2$, z^1 and z^2. See Figure 2.

When we are branching on/fixing the i variable to 0 we solve q submodels, then previously to fix the $i+1$-variable to 0 we check if this variable is fixed to 0 in all submodels p to solve if we are in this situation we check the same to the $i+2$ variable, and so on. In other case we continue branching on/fixing the $i+1$ variable.

We do the same if we branching on/fixing the i variable to 1.

Tables 3, 4 and 5 show the main parameters in the cluster analysis for each of the strategies $BFC1$, $BFC2$ and $BFC3$, respectively.

In each table, and for each of the cluster partitioning selection, $q = r$, $q = r^2$ or $q = r^3$, the number of branching nodes is given, i.e., the number of nodes examined in the corresponding branching tree, n^n; the number of MIP^{TNF} problems that are solved, n^{TNF}; and the elapsed time (in seconds) for obtaining the optimal solution with the corresponding strategy, T_{BFC}. In the last column, the heading T_{COIN} corresponds to the total time (in seconds) to obtain the optimal solution, by the plain use of the optimization engine COIN-OR over the whole model without any decomposition.

Table 3: Cluster analysis for *BFC1*

Instance	$q=r$			$q=r^2$			$q=r^3$			
	n^n	n^{TNF}	T_{BFC}	n^n	n^{TNF}	T_{BFC}	n^n	n^{TNF}	T_{BFC}	T_{COIN}
P1	1	0	0.2	1	0	0.2	38	3	1.7	0.3
P2	11	0	32.8	1431	594	416.2	23402	151	1310.7	268.3
P3	9	0	15.9	333	134	106.7	2621	234	315.9	279.3
P4	132	62	53.6	153	41	32.8	745	15	31.9	4.8
P5	1	0	0.7	10	0	1.3	851	13	63.7	5.9
P6	1	0	1.1	114	49	63.7	1269	36	117.5	7.6
P7	782	371	3722.0	5237	2078	5314.5	100938	182	13646.3	-
P8	27	0	194.8	5962	2606	10459.7	69531	902	10222.5	7034.1
P9	423	210	1204.5	-	-	-	-	-	-	-
P10	12	0	482.9	-	-	-	-	-	-	-
P11	-	-	-	-	-	-	-	-	-	-
P12	15	0	1043.2	-	-	-	-	-	-	-
P13	12	0	271.4	-	-	-	-	-	-	13910.1
P14	-	-	-	-	-	-	3337	9	12401.3	3790.0
P15	8	0	1942.5	-	-	-	-	-	-	-
P16	10	0	1327.6	133	0	443.6	-	-	-	-

Table 4: Cluster analysis for *BFC2*

Instance	$q = r$			$q = r^2$			$q = r^3$			
	n^n	n^{TNF}	T_{BFC}	n^n	n^{TNF}	T_{BFC}	n^n	n^{TNF}	T_{BFC}	T_{COIN}
P1	1	0	0.2	1	0	0.2	84	2	3.1	0.3
P2	69	0	67.2	1185	21	183.7	186295	151	8935.2	268.3
P3	29	0	36.1	609	8	136.5	54505	71	2642.0	279.3
P4	33	2	19.9	155	1	16.7	1607	15	79.6	4.8
P5	1	0	0.7	75	0	7.0	2023	13	113.7	5.9
P6	1	0	1.1	71	2	12.6	2277	30	138.5	7.6
P7	257	8	7413.6	18947	31	5744.1	-	-	-	-
P8	107	0	372.1	1937	26	6360.1	-	-	-	7034.1
P9	-	-	-	5051	52	18022.8	-	-	-	-
P10	35	0	569.5	-	-	-	-	-	-	-
P11	23	1	13230.5	201	2	19849.0				-
P12	21	0	926.8	-	-	-	-	-	-	-
P13	15	0	313.2	1235	26	12585.5	-	-	-	13910.1
P14	17	1	1680.8	335	4	4515.7	5174	9	8351.2	3790.0
P15	11	0	2119.8	-	-	-	-	-	-	-
P16	13	0	1581.6	211	0	581.7	-	-	-	-

Table 5: Cluster analysis for *BFC3*

Instance	$q = r$			$q = r^2$			$q = r^3$			
	n^n	n^{TNF}	T_{BFC}	n^n	n^{TNF}	T_{BFC}	n^n	n^{TNF}	T_{BFC}	T_{COIN}
P1	1	0	0.2	1	0	0.2	38	3	1.7	0.3
P2	11	0	32.8	347	50	330.0	23402	151	1310.7	268.3
P3	9	0	15.9	93	14	255.1	2621	234	315.9	279.3
P4	12	2	14.1	73	1	6.5	745	15	31.9	4.8
P5	1	0	0.7	10	0	1.3	851	13	63.7	5.9
P6	1	0	1.1	24	4	14.8	1269	36	117.5	7.6
P7	68	14	10945.5	1279	99	5412.9	100938	182	13646.3	-
P8	27	0	194.8	822	26	12051.4	69531	902	10222.5	7034.1
P9	-	-	-	2223	52	17339.0	-	-	-	-
P10	12	0	482.9	-	-	-	-	-	-	-
P11	-	-	-	-	-	-	-	-	-	-
P12	15	0	1043.2	-	-	-	-	-	-	-
P13	12	0	271.4	776	26	12433.3	-	-	-	13910.1
P14	13	1	1582.9	225	4	8541.0	3337	9	12401.3	3790.0
P15	8	0	1942.5	-	-	-	-	-	-	-
P16	10	0	1327.6	133	0	443.6	-	-	-	-

8 Compilation and linking with COIN-OR library

Our algorithmic approach has been implemented in a C++ experimental code. This code is going to be submitted for publication through a new project, to the *COIN-OR* web page.

It uses the optimization engine *COIN-OR* for solving the linear and mixed-integer submodels and the complete linear model. The computational experiments were conducted in a Workstation Sun FIRE v245, under Solaris System 1.0, with 2 CPU of 1.5 Ghz and 4 Gb of RAM.

We have downloaded and installed the source code for the CoinAll package for UNIX-like systems. After doing this, you can find the executables, libraries and header files in the "bin", "lib" and "include" subdirectory, respectively.

Then, you can link your own code with the installed libraries. In particular, our computational scheme uses the libraries: `Clp`, `Cbc` and `CoinUtils`.

For compiling and linking, a Makefile must be created with the dependences between the main program, BFC_MS, the own external functions and the COIN library.

You can find examples for a Makefile in the examples subdirectory, see also the information at

`https://projects.coin-or.org/BuildTools/wiki/user-examples`

References

Escudero LF, Garín A, Merino M, Pérez G (2009) BFC-MSMIP: an exact Branch-and-Fix Coordination approach for solving multistage stochastic mixed 0-1 problems. *TOP*, 17(1):96-122.

Escudero LF, Garín A, Merino M, Pérez G (2010) On BFC-MSMIP strategies for scenario cluster partitioning, and twin node family branching selection and bounding for multistage stochastic mixed 0-1 problems. *Computers and Operations Research* 37: 738-753.

Laugee-Heimer, R (2003) The Common INterface for Operations Research. *IBM, Journal of Research and Development*, 47(1):57-66. (`http://www.coin-or.org`)