Computers & Operations Research 40 (2013) 2950–2960

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

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

# An algorithmic framework for solving large-scale multistage stochastic mixed 0–1 problems with nonsymmetric scenario trees. Part II: Parallelization

Unai Aldasoro [a], Laureano F. Escudero [b], María Merino [c], Gloria Pérez [c,*]

[a] Matemática Aplicada, Universidad del País Vasco, UPV/EHU, Spain
[b] Estadística e Investigación Operativa, Universidad Rey Juan Carlos, URJC, Spain
[c] Matemática Aplicada, Estadística e Investigación Operativa, Universidad del País Vasco, UPV/EHU, Spain

ARTICLE INFO

Available online 10 July 2013

*Keywords:*
Multistage stochastic mixed 0–1 optimization
Outer and inner parallelization
MPI threads
Break stage scenario clustering
Branch-and-Fix Coordination

ABSTRACT

This note is a sequel of paper (Escudero et al. (2012) [1]), in which the sequential Branch-and-Fix Coordination referred to as the BFC-MS algorithm was introduced for solving large-scale multistage mixed 0–1 optimization problems up to optimality under uncertainty. The aim of the note is to present the parallelization version of the BFC algorithm, referred to as PC-BFCMS, so that the elapsed time reduction on problem solving is analyzed. The parallelization is performed at two levels. The inner level parallelizes the optimization of the MIP submodels attached to the set of scenario clusters that have been created by the modeler-defined break stage to decompose the original problem, where the nonanticipativity constraints are partially relaxed. Several strategies are presented for analyzing the performance of the inner parallel computation based on MPI (Message Passing Interface) threads to solve scenario cluster submodels versus the sequential version of the BFC-MS methodology. The outer level of parallelization defines a set of 0–1 variables, the combinations of whose 0–1 values, referred to as paths (one for each combination), allow independent models to be optimized in parallel, such that each one can itself be internally optimized with the inner parallelization approach. The results of using the outer–inner parallelization are remarkable: the larger the number of paths and MPI threads (in addition to the number of threads that the MIP solver allows to be used), the smaller the elapsed time to obtain the optimal solution. This new approach allows problems to be solved faster, and, can thus solve very large scale problems that could not otherwise be solved by plain use of a state-of the-art MIP solver, or could not be solved by the sequential version of the decomposition algorithm in acceptable elapsed time.


## 1. Introduction

Stochastic Optimization (SO) is broadly studied and applied in today's real-world applications. Uncertainty is the key ingredient in many decision problems. There are several ways in which uncertainty can be formalized and over the past thirty years different approaches to optimization under uncertainty have been developed. The field of SO appeared as a response to the need to incorporate uncertainty into mathematical optimization models. Basically, it deals with situations in which some parameters are random variables (i.e., coefficients in the objective function, the right hand side vector (rhs) and the constraint matrix). It allows the risk inherent into the random variables of the problem to be managed, or at least partially managed, mainly in a time horizon environment. The problem is formulated by the so-called Deterministic Equivalent Model (DEM), a term coined by Wets [2]. See e.g., [1] and references therein for some overviews on the state-of-the-art of two-stage and multistage stochastic continuous and mixed integer optimization algorithms.

In the general formulation of a multistage stochastic integer optimization problem, decisions on each stage have to be made stage-wise. At each stage, there are variables which correspond to decisions that have to be made without anticipating future events, i.e., the so-called nonanticipativity constraints must be satisfied, see [3]. Moreover, there have been few attempts to solve large scale general multistage stochastic mixed 0–1 models up to optimality due to their complexity. In any case this type of problems requires an intensive computing effort, even using decomposition algorithms for problem solving. Parallel Computing (PC) offers an alternative for solving very large scale problems by parallelizing the solution of MIP submodels that appear in the decomposition algorithms.

At hardware level, PC is currently mainly based on clusters and multicore processors. For broad information, see [4,5], among others. The nature of the cooperation between processors can

* Corresponding author. Tel.: +34 946012645; fax: +34 946012516.
*E-mail addresses:* unai.aldasoro@ehu.es (U. Aldasoro), laureano.escudero@urjc.es (L.F. Escudero), maria.merino@ehu.es (M. Merino), gloria.perez@ehu.es (G. Pérez).

0305-0548/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2013.06.015

differ depending on the way in which processors exchange information. One of the parallel architectures is the *message passing interface* (MPI) or distributed memory.

Over the last two decades some papers on SO have appeared in the relevant literature that uses PC for two-stage and multistage stochastic continuous and mixed 0–1 optimization, see e.g., [6–8] and references therein.

The main objective of this note, as a sequel to paper [1], is to present the parallelization version of the BFC-MS algorithm, referred to as PC-BFCMS, so that the reduction in the elapsed time in problem solving is analyzed. The parallelization is performed at two levels. The inner level parallelizes the optimization of the MIP submodels attached to the set of scenario clusters that have been created by the modeler-defined break stage, say $t^*$. The concept of "break stage" was introduced in [1] as a way of decomposing the original problem, in which the nonanticipativity constraints are partially relaxed from the mixture of the splitting and compact representations of the DEM of the stochastic problem. Several strategies are presented for analyzing the performance of using inner parallel computing based on MPI threads strategies for solving scenario cluster based submodels versus the sequential version of the BFC-MS methodology. The outer level of parallelization defines a set of 0–1 variables, the combinations of whose 0–1 values, referred to as paths (one for each combination), enable independent models to be optimized in parallel, such that each one can itself be internally optimized with the inner parallelization. The main results of a broad computational experience are reported to assess whether the performance of the parallel computing approach compares favorably to the sequential one. The elapsed time required by outer–inner parallelization is very frequently some orders of magnitude smaller than that of the sequential version of the algorithm, depending on the computer resources available. So, the larger the number of paths and MPI threads (in addition to the number of threads that the MIP solver allows to be used), the smaller the elapsed time for problem solving.

The rest of the paper is organized as follows. Section 2 presents the main concepts of the sequential version of the scenario cluster BFC-MS algorithm introduced in [1,9], which are also needed to introduce the parallel version. The section also presents some new concepts. Section 3 introduces the Parallel Computing Branch-and-Fix Coordination MultiStage (PC-BFCMS) algorithm. Section 4 reports the main results of a broad computational experience to assess the validity of the PC version of the BFC-MS algorithm versus its sequential version and the plain use of a state-of-the-art MIP solver. Section 5 concludes and outlines future work.

## 2. Basic models in the sequential version of the scenario cluster BFC-MS algorithm

It is assumed in this section that the main concepts and definitions of the sequential version of the algorithm are known, see [1]; so, they are used directly to present the models required by the parallel version, i.e., the PC-BFCMS algorithm.

PC-BFCMS also uses a nonsymmetric scenario tree based approach to represent the uncertainty in the random variables. So, the *compact* representation of the DEM of the multistage stochastic mixed 0–1 problem can be expressed as follows:

$$z^{DEM} = \min \sum_{g\in\mathcal{G}} w_g(a_g x_g + b_g y_g)$$
$$\text{s.t.} \quad A'_{\sigma(g)}x_{\sigma(g)} + A_g x_g + B'_{\sigma(g)}y_{\sigma(g)} + B_g y_g = h_g \ \forall g\in\mathcal{G}_t, t\in$$
$$x_g\in\{0,1\}^{n_g}, y_g\in\mathbb{R}^{+n_g} \ \forall g\in\mathcal{G}, \tag{1}$$

where $\mathcal{T}=\{1,\ldots,T\}$, $w_g$ is the likelihood assigned by the modeler to scenario group $g$, such that $w_g=\sum_{\omega\in\Omega^g} w^\omega$, for $g\in\mathcal{G}$, being $w^\omega$ the likelihood or probability of scenario $\omega\in\Omega$, where $\Omega$ is the set of scenarios under consideration, $\mathcal{G}$ denotes the set of scenario groups, $\Omega^g$ is the set of scenarios in group $g$, and $\sigma(g)$ is the immediate predecessor group of group $g$, such that $\sigma(g)\in\mathcal{G}_{t(g)-1}$, for $g\in\mathcal{G}-\mathcal{G}_1$, where $t(g)$ is the stage from set $\mathcal{T}$ to which group $g$ belongs to. Notice that $g\in\mathcal{G}_{t(g)}$ and $\mathcal{G}_t$ is the set of scenario groups in stage $t$. Observe also that $w_d=w^\omega$ for $\omega\in\Omega^d$, $d\in\mathcal{G}_T$ such that $\Omega^d$ is a singleton set. Additionally, $x_g$ and $y_g$ represent the replicas of the $x$ and $y$ variables for scenario group $g$, respectively, $a_g$ and $b_g$ are the related objective function vector coefficients for the 0–1 and continuous variables, respectively, $A'_g$, $A_g$, $B'_g$ and $B_g$ are the constraint matrices, and $h_g$ is the right-hand-side vector (rhs). See e.g., [10] for the main concepts on stochastic optimization via scenario tree analysis.

On the other hand, the *splitting variable* representation explicitly considers the non-anticipativity constraints (NAC) $x_t^\omega - x_t^{\omega'}=0$ and $y_t^\omega - y_t^{\omega'}=0$ for stage $t\in\mathcal{T}$ under scenario $\omega\in\Omega$, see [9]. It is clear that the explicit representation is not desirable for all pairs $(\omega,\omega')$ of scenarios in order to reduce the model's dimensions. So, the NAC for some pairs of scenarios can be implicitly represented in order to gain computational efficiency.

The scenario tree is decomposed into a set of subtrees, each for a scenario cluster in the set denoted as $\mathcal{C}=\{1,\ldots,C\}$ with $C=|\mathcal{C}|$. Let $\Omega_c$ denote the set of scenarios that belong to cluster $c$, such that $\Omega_c\bigcap\Omega_{c'}=\emptyset$, $c,c'\in\mathcal{C}: c\neq c'$ and $\Omega=\cup_{c\in\mathcal{C}}\Omega_c$. Additionally, let a break stage, say, $t^*$ be a stage such that the number of scenario clusters is $C=|\mathcal{G}_{t^*+1}|$, where $t^*+1\in\mathcal{T}$. In this case, any cluster $c\in\mathcal{C}$ is induced by a group $g\in\mathcal{G}_{t^*+1}$ and contains all scenarios belonging to that group, i.e., $\Omega_c=\Omega^g$. So, the models of the scenario clusters in set $\mathcal{C}$ result from the relaxation of the NAC until break stage $t^*$ in the original DEM (1). See [1] for details.

The original DEM can now be formulated via a mixture of the splitting variable and compact representations, so that the submodels are linked by the explicit NAC up to break stage $t^*$. To that end we slightly abuse the notation such that $\mathbf{x}_t^c$ and $\mathbf{y}_t^c$ denote the vectors of the 0–1 and continuous variables, respectively, for scenario cluster $c\in\mathcal{C}$ and stage $t\in\mathcal{T}$, $\mathbf{a}_t^c$ and $\mathbf{b}_t^c$ are the vectors of the objective function coefficients of the variables vectors $\mathbf{x}_t^c$ and $\mathbf{y}_t^c$, respectively, and $\mathbf{nx}_t^c$ and $\mathbf{ny}_t^c$ denote the number of 0–1 and continuous variables, respectively, for the pair $(c,t)$. Similarly, let $\mathbf{h}_t^c$ denote the new rhs. Additionally, let $\mathcal{G}^c\subset\mathcal{G}$ denote the set of scenario groups for cluster $c$, such that $\Omega^g\cap\Omega_c\neq\emptyset$ means that $g\in\mathcal{G}^c$, and let $\mathcal{G}_t^c=\mathcal{G}_t\cap\mathcal{G}^c$ denote the set of scenario groups for cluster $c\in\mathcal{C}$ in stage $t\in\mathcal{T}$.

The MIP submodel for cluster $c\in\mathcal{C}$ can be formulated as follows:

$$z^c = \min \sum_{t\in\mathcal{T}} \mathbf{w}_t^c(\mathbf{a}_t^c\mathbf{x}_t^c + \mathbf{b}_t^c\mathbf{y}_t^c)$$
$$\text{s.t.} \quad \text{cluster } c \text{ constraint system (3)}$$
$$\mathbf{x}_t^c\in[0,1]^{\mathbf{nx}_t^c}, \mathbf{y}_t^c\in\mathbb{R}^{+\mathbf{ny}_t^c} \quad \forall t\in\mathcal{T}, \tag{2}$$

where the cluster $c$ constraint system is expressed as

$$\mathbf{A}_t^{'c}\mathbf{x}_{t-1}^c + \mathbf{A}_t^c\mathbf{x}_t^c + \mathbf{B}_t^{'c}\mathbf{y}_{t-1}^c + \mathbf{B}_t^c\mathbf{y}_t^c = \mathbf{h}_t^c \quad \forall t\le t^*+1$$
$$[\mathbf{A}_{t'}]^c\mathbf{x}_{t-1}^c + [\mathbf{A}_t]^c\mathbf{x}_t^c + [\mathbf{B}_{t'}]^c\mathbf{y}_{t-1}^c + [\mathbf{B}_t]^c\mathbf{y}_t^c = \mathbf{h}_t^c \quad \forall t^*+1<t\le T. \tag{3}$$

The set of constraints in model (2) is split into two blocks. The first represents the constraints related to the vectors of variables from stage $t=1$ up to stage $t^*+1$ (i.e., stages with explicit NAC), since those variables must be linked with their own replicas in the appropriate clusters in set $\mathcal{C}$. The second block represents the constraints related to the vectors of variables from stage $t^*+2$ up to the last stage $T$ (i.e., stages with implicit NAC). See [1,9] for details.

Let us split the set of stages $\mathcal{T}$ into two subsets, such that $\mathcal{T}=\mathcal{T}_1\bigcup\mathcal{T}_2$, where $\mathcal{T}_1=\{1,\ldots,t^*\}$, and $\mathcal{T}_2=\{t^*+1,\ldots,T\}$. For modeling the (explicit) NAC of the scenario clusters for the stages in set $\mathcal{T}_1$, consider the following definition.