An extension of an interior-point method to include risk aversion in large-scale multistage stochastic optimization

Jordi Castro[a,*], Laureano F. Escudero[b], Juan F. Monge[c]

[a] *Dept. of Statistics and Operations Research, Universitat Politècnica de Catalunya, Barcelona, Catalonia.*
[b] *Area of Statistics and Operations Research, Universidad Rey Juan Carlos, URJC, Móstoles (Madrid), Spain.*
[c] *Center of Operations Research, Universidad Miguel Hernández, UMH, Elche (Alicante), Spain.*

Abstract

In the earlier paper "On solving large-scale multistage stochastic optimization problems with a new specialized interior-point approach, European Journal of Operational Research, 310 (2023), 268–285" the authors presented a novel approach based on a specialized interior-point method (IPM) for solving (risk neutral) large-scale multistage stochastic optimization problems. The method computed the Newton direction by combining Cholesky factorizations with preconditioned conjugate gradients (PCG).

This work extends that approach to the risk averse setting by incorporating coherent risk measures, namely expected conditional value-at-risk and second-order expected conditional stochastic dominance. Introducing risk aversion makes the optimization problems substantially more challenging and, for very large instances, can even exhaust the available memory resources.

The contributions of this work are twofold. First, we propose a reformulation of risk averse stochastic optimization models based on variable splitting, which proves highly effective for general-purpose IPM solvers. Second, we show that the reformulated problem remains fully compatible with the specialized IPM. In particular, the new risk averse constraints extend the PCG preconditioner with an additional diagonal matrix, thereby preserving the efficiency of the solution process.

Extensive computational experiments are reported for large-scale risk averse multistage stochastic revenue management problems with up to 278 million variables and 99 million constraints. The results show that the proposed splitting formulation significantly outperforms the classical compact formulation when using state-of-the-art solvers, and that the specialized IPM achieves superior performance on most of the largest problem instances.

Keywords: Large-scale optimization, interior-point methods, stochastic programming, multistage stochastic optimization, CVaR and stochastic dominance risk averse

1. Introduction and motivation

As is well known, the realization of the uncertain parameters in dynamic mathematical optimization is usually structured as a finite set of scenarios along stages within a given time horizon (Birge and Louveux 2011, Pflug and Pichler 2015, Shapiro et al. 2009). There are basically two types of approaches to deal with the uncertainty, namely risk neutral and risk averse ones (Shapiro 2021). The aim of the former is to minimize the expected objective function value (say, cost) in the scenarios along the time horizon. However, it ignores the variability of the objective function value across the scenarios, in particular, the undesirable *right tail* corresponding to low-probability, high-cost scenarios, often referred to as black swan scenarios. This is because the optimal solution balances that high cost with the smaller ones in other scenarios. The potential drawback is that, in the case of the occurrence of those black swan scenarios, the solution may have very negative consequences; this can be alleviated, at least partially, by incorporating risk averse measures that prevent such types of solutions, at the expense of an increase in the expected cost.

*Corresponding author
Email addresses: jordi.castro@upc.edu (Jordi Castro), laureano.escudero@urjc.es (Laureano F. Escudero), monge@umh.es (Juan F. Monge)

There is a broad literature on risk averse measures for static, mainly two-stage, and multistage stochastic problems with continuous or integer variables. Let us provide a brief overview of the most popular measures:

1. Minimize the expected cost subject to an upper bound on the expected semi-deviation of the scenario optimal cost with respect to the expected one, see Markowitz (1952), Ogryczak and Ruszczyński (1999), Ahmed (2006).
2. Minimize the expected deviation of the scenario optimal cost with respect to the expected one, so-named scenario immunization, see Dembo (1991).
3. Minimize the expected cost jointly with the weighted value-at-risk (VaR), where the probability of cost surplus over VaR is upper bounded and VaR is the highest cost in the scenarios, see Gaivoronski and Pflug (2005), Guigues and Sagastizabal (2013).
4. Minimize the expected cost jointly with the weighted conditional value-at-risk (CVaR), the latter is composed of the same VaR and the expected cost surplus over VaR in the scenarios, see Rockafellar and Uryasev (2000), Schultz and Tiedemann (2006).
5. Minimize the expected cost jointly with the weighted expected CVaR (ECVaR) in the stages, see Rockafellar and Uryasev (2000), Dentcheva and Ruszczyński (2003), Gaivoronski and Pflug (2005), Ahmed (2006), Schultz and Tiedemann (2006), Charpentier and Oulidi (2009), Shapiro et al. (2009), Rudloff et al. (2014), Kormik and Morton (2015), Pflug and Pichler (2015), Homem-de-Mello and Pagnoncelli (2016), Alonso-Ayuso et al. (2018), among others.
6. Minimize the expected cost in the scenarios, where the probability of cost surplus over a given threshold is upper bounded (so-named chance-constrained and also first-order stochastic dominance), see Charnes and Cooper (1959), Gollmer et al. (2008).
7. Minimize the expected cost jointly with the weighted expected cost surplus over a given threshold in the scenarios, see Eppen et al. (1989), Guigues (2014).
8. Minimize the expected cost jointly with the weighted expected conditional (first- and second-order) stochastic dominance (ECSD), see Dentcheva and Ruszczyński (2003), Fabian et al. (2011), Gollmer et al. (2011), Escudero et al. (2017), Alonso-Ayuso et al. (2018), Escudero and Monge (2018), Escudero et al. (2020), Escudero and Monge (2021), Shapiro (2021), Escudero and Monge (2023), Dentcheva (2023), Dentcheva and Ruszczyński (2024).

In the earlier work Castro et al. (2023) we introduced a new approach based on a specialized interior-point method (IPM) (Wright 1996, Gondzio 2012, 2025) for solving risk neutral large stochastic optimization problems. In some of the largest instances (up to 800M variables and 20M constraints) this approach outperformed by a large extent state-of-the-art IPM solvers such as CPLEX.

The aim of this work is to extend the risk neutral multistage stochastic model of Castro et al. (2023) to the risk averse setting. Two coherent risk averse measures are considered: expected conditional value-at-risk (ECVaR) and second-order expected conditional stochastic dominance (ECSD). Those measures are defined as coherent ones since they satisfy the properties of translation invariance, positive homogeneity, monotonicity and convexity, see Artzner et al. (1999).

Unlike risk neutral stochastic problems, for which several specialized IPMs have been developed (e.g., Lustig et al. (1991), Ruszczyński (1993), Gondzio and Grothey (2007), Gondzio et al. (2016), Kibaek and Zavala (2018), Castro and Lama-Zubirán (2020)), the literature on tailored methods to deal with risk averse stochastic optimization is scarce. Moreover, incorporating risk averse constraints into a stochastic optimization model (already challenging on its own) makes the problem hardly tractable when very large-scale instances are solved with general-purpose solvers. For example, the risk neutral variants of the largest instances tested in Section 5 could be solved with state-of-the-art IPM solvers, whereas the risk averse variants exhausted the 1000 gigabytes of RAM available on the computation server.

This work attempts to fill this gap through two main contributions:

- We propose a new reformulation of the risk averse stochastic optimization problem. Similar to the approaches of Castro et al. (2023), Lustig et al. (1991), Ruszczyński (1993), Castro and Lama-Zubirán (2020) for risk neutral stochastic optimization, it relies on a novel variable splitting

scheme. The new split-variable risk averse formulation proves effective for general-purpose solvers, as demonstrated by the computational results reported in this work.

- The structure of the optimization problem provided by the split-variable risk averse reformulation is fully compatible with the specialized IPM of Castro and Cuesta (2011), Castro (2016), which underlies the method for risk neutral problems developed in Castro et al. (2023). This approach solves the normal equations of the IPM by combining Cholesky factorizations with preconditioned conjugate gradients (PCG). The efficient solution of systems with the preconditioner is instrumental to the global performance of the method. In Castro et al. (2023) it was shown that, for risk neutral stochastic optimization problems, the preconditioner is a block tridiagonal matrix, which can be easily factorized. In this work, we show that the inclusion of risk averse constraints simply extends the preconditioner with an additional diagonal block, without complicating the factorization.

The remainder of this paper is organized as follows. Section 2 introduces the basic concepts of multistage stochastic optimization scenario trees and establishes the notation used throughout the paper. In Section 3, we revisit the risk neutral multistage models of Castro et al. (2023) (both compact and split-variable formulations), extend them to their risk averse counterparts under two different risk measures, and propose corresponding split-variable reformulations. Section 4 describes the specialized interior-point method tailored for risk averse multistage stochastic problems. Section 5 reports computational experiments comparing our implementation of this method against CPLEX v22.1.2. Finally, Section 6 concludes with a summary of the main findings and directions for future work.

2. Stochastic scenario tree

The notation follows Escudero and Monge (2023), Castro et al. (2023). Let a *scenario* be the realization of the uncertain parameters over the time horizon. A *node* for a given stage in a multistage scenario tree has one-to-one correspondence with the group of scenarios that have the same realization of the uncertain parameters up to that stage. This information structure can be visualized as the tree in Fig. 1, where each root-to-leaf path corresponds to a specific scenario, i.e., a realization of the whole set of the uncertain parameters.

Lexicographically ordered sets in the multistage tree

$\mathcal{T}$, stages, such that $\mathcal{T} = \{1, \ldots, |\mathcal{T}|\}$.

$\mathcal{N}$, nodes in the scenario tree, such that $\mathcal{N} = \{1, \ldots, |\mathcal{N}|\}$.

$\mathcal{N}_t$, nodes in stage t, where $\mathcal{N}_t \subset \mathcal{N}$, $t \in \mathcal{T}$. By construction, $|\mathcal{N}_1| = 1$.

Ω, scenarios. Each one comprises the nodes in the Hamiltonian path from root node 1 to a node, say, ω in the last stage, up through the stages in set $\mathcal{T}$; therefore, $\omega \in \mathcal{N}_{|\mathcal{T}|}$. For convenience, a scenario has traditionally been denoted by its last node in the path.

$\Omega^n \subset \Omega$, scenario group containing node n in the path from root node 1 to their last node $\omega \in \mathcal{N}_{|\mathcal{T}|}$. Note that $\Omega^1 = \Omega$.

$\mathcal{A}^n$, node n and its ancestors, $n \in \mathcal{N}$. Note that $\mathcal{A}^1$ only contains node $1 \in \mathcal{N}_1$.

$\mathcal{S}^n$, immediate successors of node n, $n \in \mathcal{N}$, where $\mathcal{S}^n = \emptyset$ for $n \in \mathcal{N}_{|\mathcal{T}|}$.

Other elements in node n, for $n \in \mathcal{N}$

w^n, weight factor representing the likelihood that is associated with node n. Notice that $w^n = \sum_{\omega \in \Omega^n} w^\omega$, where w^ω gives the modeler-driven likelihood associated with scenario ω, such that $\sum_{\omega \in \Omega} w^\omega = 1$.

t^n, stage that node n belongs to, therefore, $n \in \mathcal{N}_{t^n}$.

$a(n)$, immediate ancestor node of node n. Note: It is assumed that $a(1) = 0$.

$s^n(i)$, i-th node in set $\mathcal{S}^n : t^n < |\mathcal{T}|$, $i = 1, \ldots, \ell^n$, where $\ell^n = |\mathcal{S}^n|$.

As an illustration, let us consider an instance with $|\mathcal{T}| = 4$ stages (say, years), and the scenario tree depicted in Fig. 1, where the number of immediate successor nodes of node n is $\ell^n = 2$, $n \in \mathcal{N} : t^n < |\mathcal{T}|$. Therefore, the cardinality of the scenario tree is $|\mathcal{N}| = \sum_{t \in \mathcal{T}} |\mathcal{N}_t| = 1 + 2 + 4 + 8 = 2^4 - 1 = 15$ nodes.

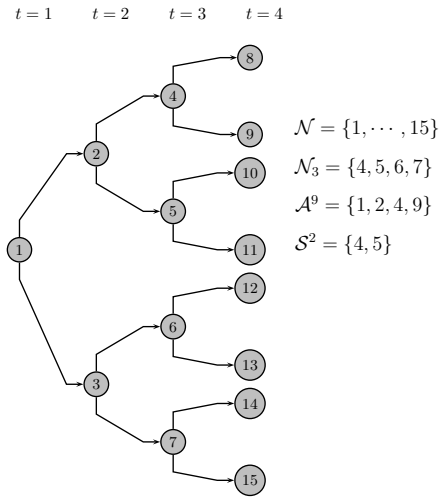


Figure 1: A multistage scenario tree

3. Stochastic metamodels

3.1. Risk neutral metamodel

The risk neutral (RN) metamodel of the multistage stochastic problem follows Castro et al. (2023). Its compact form can be written as

$$\min \quad \sum_{n \in \mathcal{N}} w^n C^n \tag{1a}$$

$$\text{s. to} \quad (T^n x^{a(n)})_{:t^n>1} + W^n x^n + M^n z^n = h^n \quad \forall n \in \mathcal{N} \tag{1b}$$

$$0 \leq x^n \leq u_x^n,\ 0 \leq z^n \leq u_z^n \quad \forall n \in \mathcal{N}, \tag{1c}$$

where the cost function C^n for node n is given by

$$C^n = a^n x^n + b^n z^n \quad \forall n \in \mathcal{N}, \tag{1d}$$

with a^n and b^n denoting the objective coefficient vectors of x^n and z^n, respectively. The remaining elements are defined as follows: T^n and W^n are the constraint matrices of the state variables $x^{a(n)}$ and x^n in the first and second stages, respectively, in the embedded two-stage submodel; M^n is the constraint matrix of the local variables z^n in node n; h^n is the right-hand side (RHS) vector; and u_x^n, u_z^n are the upper bound vectors of x^n and z^n.

Figure 2 illustrates the structure of the constraint matrix of metamodel (1) for the example depicted in Fig. 1. For simplicity, the columns corresponding to the variables z^n (i.e., the terms $M^n z^n$) have been omitted.

To overcome the presence of dense columns in (1) (e.g., x^i, $i = 1, \ldots, 7$, in Fig. 2), split-variable reformulations are required when solving the models with IPMs (see Lustig et al. (1991), Ruszczyński (1993), Castro and Lama-Zubirán (2020), Castro et al. (2023)). The proposed reformulation introduces the following copies of the variables:

x_n^s, copy of x^n in node s, where n is the root of a two-stage subtree and s is a second stage node, for $s \in \mathcal{S}^n$, $n \in \mathcal{N} : t^n < |\mathcal{T}|$.

These copies are forced to take the same value through a set of linking constraints. Therefore, for each node n that is not in the last stage ($n \in \mathcal{N} : t^n < |\mathcal{T}|$), we impose $x^n - x_n^{s(1)} = 0$ and $x_n^{s(i)} - x_n^{s(i+1)} = 0$, $i = 1, \ldots, \ell^n - 1$. Thus, the split-variable scheme covers the entire set of the immediate successors of node n, i.e., the set $\mathcal{S}^n$. The chosen scheme has the advantage that each variable in $\{x^n, x_n^{s(1)}, \ldots, x_n^{s(\ell^n)}\}$ appears in only two linking constraints, except x^n and $x_n^{s(\ell^n)}$, which appear in only one.

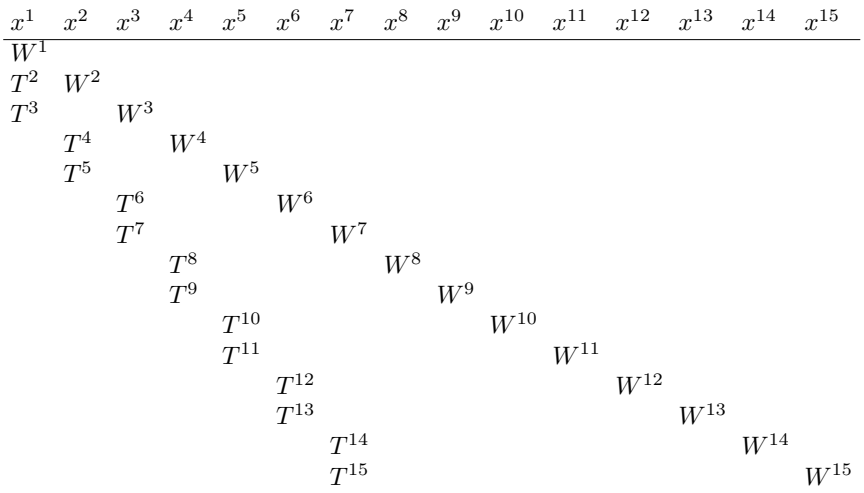


Figure 2: Constraint matrix of metamodel (1) for the multistage tree depicted in Fig. 1, omitting columns z^n, $n \in \mathcal{N}$.

RN split-variable reformulation of metamodel (1)

$$\min \quad \sum_{n\in\mathcal{N}} w^n C^n \tag{2a}$$

$$\text{s. to} \quad (T^n x^n_{a(n)})_{:t^n>1} + W^n x^n + M^n z^n = h^n \qquad \forall n \in \mathcal{N} \tag{2b}$$

$$0 \le x^n \le u^n_x,\ 0 \le z^n \le u^n_z \qquad \forall n \in \mathcal{N} \tag{2c}$$

$$x^n - x^{s(1)}_n = 0,\ x^{s(i)}_n - x^{s(i+1)}_n = 0 \qquad \forall i = 1,\dots,\ell^n - 1,\ n \in \mathcal{N} : t^n < |\mathcal{T}|. \tag{2d}$$

Constraints (2d) are the linking equations for the split variables that are used in the node equations (i.e., (2b)). Figure 3 illustrates the structure of the constraint matrix of the split-variable formulation (2), for the same example depicted in Fig. 2 for metamodel (1), omitting columns z^n to save space.

3.2. Risk averse metamodels for multistage stochastic problems

The compact RN model (1) minimizes the expected objective function value (i.e., the cost) over the scenarios along the time horizon. However, as we mentioned above, it ignores the variability of the cost, in particular, the undesirable *right tail* corresponding to the black swan scenarios to be alleviated by incorporating risk averse measures in the minimization of (1a).

This section focuses on coherent risk averse measures. Subsection 3.2.1 discusses the expected conditional value-at-risk (ECVaR), while Subsection 3.2.2 addresses the expected conditional stochastic dominance (ECSD). Certain variants of those measures are time consistent in the sense presented in Shapiro et al. (2009), see also Rudloff et al. (2014), Homem-de-Mello and Pagnoncelli (2016), Alonso-Ayuso et al. (2018), Shapiro (2021).

3.2.1. Expected conditional value-at-risk (ECVaR)

The classical risk averse measure CVaR in a two-stage setting aims at minimizing the expected cost in the scenarios jointly with the weighted VaR (i.e., the highest cost among those scenarios) and the related expected cost surplus (Rockafellar and Uryasev 2000).

An interesting coherent time-consistent extension, called expected CVaR (ECVaR), is based on a modeler-driven subset of stages $\mathcal{T}' \subseteq \mathcal{T}$. For each $t \in \mathcal{T}'$, the corresponding scenario group $\Omega^{n'}$, for $n' \in \mathcal{N}_t$, is considered for risk contention on high cost. It should be emphasized that the cost is evaluated *from the first stage to the last* along each scenario (see Dentcheva and Ruszczyński (2003), Shapiro et al. (2009), Rudloff et al. (2014), Homem-de-Mello and Pagnoncelli (2016), among others). Notice that the risk averse measure for singleton $\mathcal{T}' = \{1\}$ reduces to the classical CVaR in the multistage context.

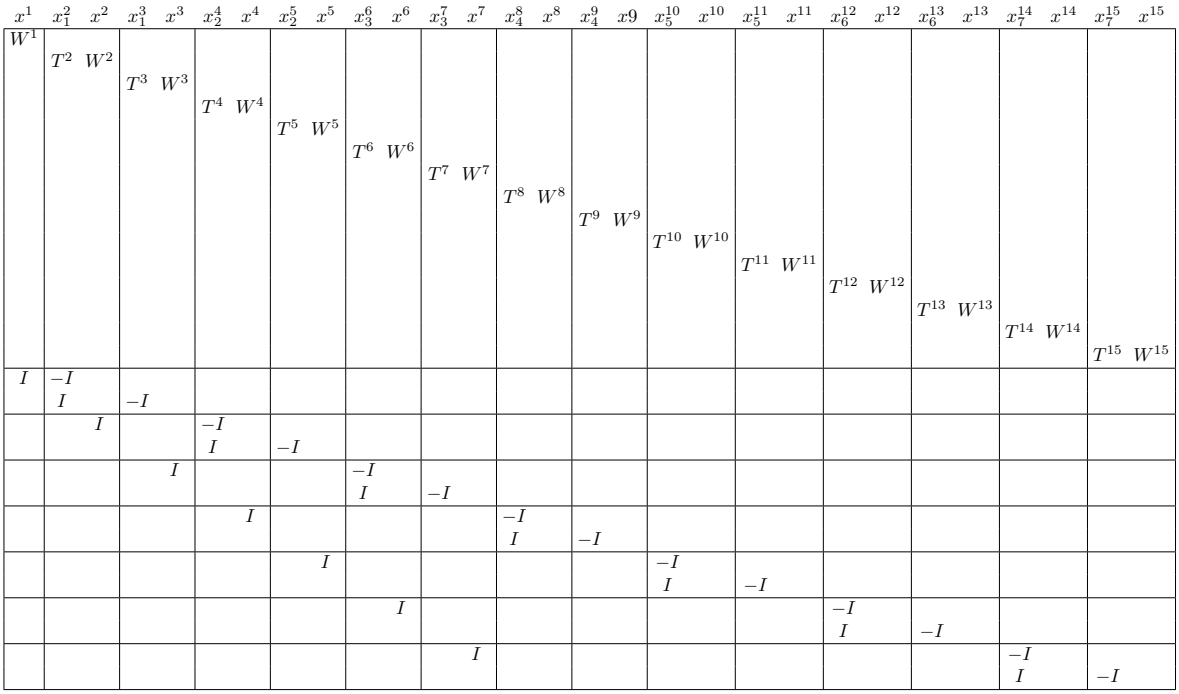


Figure 3: Constraint matrix of split-variable reformulation (2) for the tree depicted in Fig. 1, omitting columns z^n, $n \in \mathcal{N}$.

In this work we consider a variant of ECVaR whose goal is to minimize the highest cost *from the first stage 1 to stage t*, jointly with the expected cost surplus in the nodes $n' \in \mathcal{N}_t, t \in \mathcal{T}'$ (Alonso-Ayuso et al. 2018). This measure is also a coherent risk averse functional, but it is time-inconsistent. Nevertheless, it is highly attractive in real-life applications since, in addition to controlling high costs over the entire horizon when $|\mathcal{T}| \in \mathcal{T}'$, it also allows risk control at intermediate stages. Notice that when $|\mathcal{T}| \in \mathcal{T}'$, ECVaR again reduces to the classical CVaR in the multistage setting.

The following additional notation is required:

γ, β, continuous parameters in [0,1].

α_t, variable representing the highest cost (1a) for the sum of nodes $n \in \mathcal{A}^{n'}$, for $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$.

$s^{n'}$, variable representing the cost surplus over α_t for any node $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$.

The compact ECVaR metamodel can be expressed as

$$\min \quad \gamma \sum_{n \in \mathcal{N}} w^n C^n + (1-\gamma) \sum_{t \in \mathcal{T}'} \Big(\alpha_t + \frac{1}{\beta} \sum_{n' \in \mathcal{N}_t} w^{n'} s^{n'} \Big) \tag{3a}$$

$$\text{s. to} \quad (1b) - (1c) \tag{3b}$$

$$\sum_{n \in \mathcal{A}^{n'}} C^n - s^{n'} \leq \alpha_t \quad \forall n' \in \mathcal{N}_t,\, t \in \mathcal{T}'. \tag{3c}$$

The objective (3a) aims to minimize the weighted combination of the expected cost and the target cost $\{\alpha_t\}_{t \in \mathcal{T}'}$, while penalizing the expected cost surplus in the nodes.

ECVaR split-variable reformulation of metamodel (3)

The ECVaR split-variable reformulation requires special handling of the objective function terms for nodes $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$, traced back to the root node 1. The scheme accumulates the costs from node n' to the root of the multistage tree.

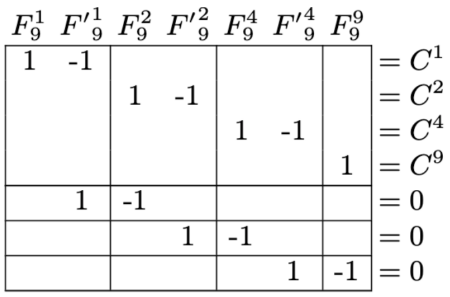


Figure 4: Split-variables constraint (5f), and the related block ones (5c) and (5d) in path (4)

We extend the split-variable formulation of the risk neutral model (2) as an alternative to model (3). This extension is needed to compute the accumulated sum $\sum_{n\in\mathcal{A}^{n'}} C^n$ in (3c) by means of additional split variables and linking constraints.

For this purpose, let us define the following ordered set of nodes along the path from node $n' \in \mathcal{N}_t$ to root node 1, for $t \in \mathcal{T}'$:

$$\mathcal{P}^{n'} = \{1, \ldots, a(n'), n'\}. \tag{4}$$

Let $\overline{p}(n')$ denote the last node in $\mathcal{P}^{n'}$, and $p+1$ denote the immediate successor of node p in $\mathcal{P}^{n'}$, for $p \in \mathcal{P}^{n'} \setminus \{\overline{p}(n')\}$. The following variables are introduced to represent the accumulated cost:

$F^p_{n'}$, accumulated cost from the last node $\overline{p}(n')$ to node p in $P^{n'}$.

$F'^p_{n'}$, copy of $F^{p+1}_{n'}$ used at node p.

Therefore, the split-variable reformulation of the ECVaR multistage problem can be expressed as

$$\min \quad \gamma \sum_{n\in\mathcal{N}} w^n C^n + (1-\gamma) \sum_{t\in\mathcal{T}'} \Big(\alpha_t + \frac{1}{\beta} \sum_{n'\in\mathcal{N}_t} w^{n'} s^{n'}\Big) \tag{5a}$$

$$\text{s. to} \quad (2b) - (2d) \tag{5b}$$

$$F^{\overline{p}(n')}_{n'} = C^{\overline{p}(n')} \qquad \forall n' \in \mathcal{N}_t,\ t \in \mathcal{T}' \tag{5c}$$

$$F^p_{n'} = C^p + F'^p_{n'} \qquad \forall p \in \mathcal{P}^{n'} \setminus \{\overline{p}(n')\},\ n' \in \mathcal{N}_t,\ t \in \mathcal{T}' \tag{5d}$$

$$F^1_{n'} - s^{n'} - \alpha_t + \xi^{n'} = 0 \qquad \forall n' \in \mathcal{N}_t,\ t \in \mathcal{T}' \tag{5e}$$

$$F'^p_{n'} - F^{p+1}_{n'} = 0 \qquad \forall p \in \mathcal{P}^{n'} \setminus \{\overline{p}(n')\},\ n' \in \mathcal{N}_t,\ t \in \mathcal{T}' \tag{5f}$$

$$0 \le s^{n'} \le \overline{s}^{n'},\ 0 \le \alpha_t \le \overline{\alpha}_t,\ 0 \le \xi^{n'} \le \overline{\xi}^{n'} \qquad \forall n' \in \mathcal{N}_t,\ t \in \mathcal{T}', \tag{5g}$$

where $\xi^{n'}$ in (5e) is a slack variable to enforce equality, and $\overline{s}^{n'}$, $\overline{\alpha}_t$ and $\overline{\xi}^{n'}$ are given upper bounds. Formulation (5) matches the standard format needed by the specialized IPM, as shown in Section 4.

Clearly, the function (5a) coincides with (3a). As an illustration, Fig. 4 depicts the sparsity of constraints (5c), (5d), (5f) corresponding to constraint (3c) for the node $n' = 9$ and time stage $t = 4$ in Fig. 1. In this case, the path (4) is $\mathcal{P}^9 = \{1, 2, 4, 9\}$. The system in Fig. 4 guarantees that F^1_9 is equal to $\sum_{n\in\mathcal{A}^9} C^n$ in (3c). This quantity is in fact the cost under scenario n', for $n' \in \Omega$, provided that $n' \in \mathcal{N}_{|\mathcal{T}|}$, $|\mathcal{T}| \in \mathcal{T}'$.

Therefore, (3) and (5) are mathematically equivalent formulations. Moreover, the linking constraints (5f) involve only two adjacent nodes p and $p+1$ in $\mathcal{P}^{n'}$. This particular structure is advantageous for decomposition approaches, which can exploit it effectively.

3.2.2. Expected conditional stochastic dominance (ECSD)

Stochastic dominance (SD) was introduced in stochastic optimization with risk aversion in Dentcheva and Ruszczyński (2003). The risk averse measure ECSD corresponds to second-order SD; see Gollmer et al. (2011) for the two-stage case and Escudero and Monge (2018), Escudero and Monge (2023) for

multi-horizon environments. It ensures that the scenario cost (from the first to the last stage) remains within modeler-defined bounds for the scenarios in each group of the selected stages. This type of risk averse measure is coherent and time-consistent (Shapiro et al. 2009, Escudero et al. 2015, 2016, Escudero and Monge 2018, Escudero et al. 2020). In addition, this work considers a coherent but time-inconsistent second-order ECSD variant, which is also attractive for the same types of applications as ECVaR, since risk control can be imposed not only over the entire horizon but also at intermediate stages. Note: by construction, the second-order version does not include the chance-constrained case; see Gollmer et al. (2008). The latter corresponds to a first-order SD functional.

The following notation is required:

$\mathcal{T}' \subseteq \mathcal{T}$, modeler-driven set of stages from which nodes are selected for risk control on cost, *from the first stage up to stage* t, for $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$.

$\overline{\alpha}_t$, cost threshold used as a target for the sum of nodes $n \in \mathcal{A}^{n'}$, for $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$.

$s^{n'}$, surplus over the cost threshold $\overline{\alpha}_t$ in any node $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$. Let $\overline{s}_t$ be the upper bound on $s^{n'}$.

$\overline{\overline{s}}_t$, upper bound on the expected surplus relative to the cost threshold $\overline{\alpha}_t$ across the nodes $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$.

The compact ECSD metamodel can be expressed as

$$\min \quad \sum_{n \in \mathcal{N}} w^n C^n \tag{6a}$$

$$\text{s. to} \quad (1b) - (1c) \tag{6b}$$

$$\sum_{n \in \mathcal{A}^{n'}} C^n - s^{n'} \leq \overline{\alpha}_t, \quad 0 \leq s^{n'} \leq \overline{s}_t \quad \forall n' \in \mathcal{N}_t,\, t \in \mathcal{T}' \tag{6c}$$

$$\sum_{n' \in \mathcal{N}_t} w^{n'} s^{n'} \leq \overline{\overline{s}}_t \qquad \forall t \in \mathcal{T}'. \tag{6d}$$

Constraints (6c) define the cost surplus for the nodes at the modeler-driven stages. Constraints (6d) impose bounds on the expected cost surplus in the nodes at those stages.

The main difference between the ECVaR model (3) and the ECSD model (6) lies in the treatment of the threshold parameter. In the former, α represents a cost target (see (3c)), which is minimized jointly with the expected cost surplus (see the objective function (3a)). In the latter, $\overline{\alpha}$ is a fixed cost threshold (see (6c)), and both the scenario-wise cost surplus and its expectation are explicitly bounded (see (6d)).

ECSD split-variable reformulation of metamodel (6)

Similarly to ECVaR, the split-variable reformulation of the ECSD model can be written as

$$\min \quad \sum_{n \in \mathcal{N}} w^n C^n \tag{7a}$$

$$\text{s. to} \quad (2b) - (2d),\ (5c) - (5f) \tag{7b}$$

$$F^1_{n'} - s^{n'} + \xi^{n'} = \overline{\alpha}_t, \quad 0 \leq s^{n'} \leq \overline{s}_t \quad \forall n' \in \mathcal{N}_t,\, t \in \mathcal{T}' \tag{7c}$$

$$\sum_{n' \in \mathcal{N}_t} w^{n'} s^{n'} + \nu_t = \overline{\overline{s}}_t \qquad \forall t \in \mathcal{T}' \tag{7d}$$

$$0 \leq \xi^{n'} \leq \overline{\xi}^{n'},\ 0 \leq \nu_t \leq \overline{\nu}_t \qquad \forall n' \in \mathcal{N}_t,\, t \in \mathcal{T}', \tag{7e}$$

where $\xi^{n'}$ in (7c) and ν_t in (7d) are slack variables, and $\overline{\xi}^{n'}$ and $\overline{\nu}_t$ in (7e) denote their upper bounds. Notice that the surplus relative to the cost threshold $\overline{\alpha}_t$ is expressed as $s^{n'} - \xi^{n'}$, for $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$.

4. The specialized IPM for multistage stochastic optimization with a risk averse functional

As it will be shown below, the split-variable formulations (5) and (7) for, respectively, the ECVaR and ECSD models, match the following general formulation of a primal block-angular optimization problem:

$$\begin{aligned} \min_{x^1,\ldots,x^k,x^0} \quad & \sum_{i=1}^{k} (c^i)^\top x^i \\ \text{s. to} \quad & \begin{bmatrix} N_1 & & & & \\ & N_2 & & & \\ & & \ddots & & \\ & & & N_k & \\ R_1 & R_2 & \ldots & R_k & I \end{bmatrix} \begin{bmatrix} x^1 \\ x^2 \\ \vdots \\ x^k \\ x^0 \end{bmatrix} = \begin{bmatrix} b^1 \\ b^2 \\ \vdots \\ b^k \\ b^0 \end{bmatrix} \\ & 0 \le x^i \le u^i \quad i = 0, \ldots, k, \end{aligned} \tag{8}$$

where x in (8), for notation saving purposes, denotes all the optimization variables in (5) and (7) (namely: $x, z, s, F, F', \alpha, \xi$ for ECVaR; and x, z, s, F, F', ν for ECSD). The matrices $N_i \in \mathbb{R}^{\overline{m}_i \times \overline{n}_i}$ and $R_i \in \mathbb{R}^{l \times \overline{n}_i}$, $i = 1, \ldots, k$, define, respectively, the block and linking constraints, where k is the number of blocks, l is the number of linking constraints, and $\overline{m}_i$ and $\overline{n}_i$ denote the number of constraints and variables of block i. Vectors $x^i \in \mathbb{R}^{\overline{n}_i}$ and $u^i \in \mathbb{R}^{\overline{n}_i}$ are the variables and their upper bounds, respectively, for block $i = 1, \ldots, k$. The components of $x^0 \in \mathbb{R}^l$ are the slacks of the linking constraints; if they are equalities their upper bounds $u^0 \in \mathbb{R}^l$ can be set to 0 or to a very small feasibility tolerance. The linear objective function is defined by vectors $c^i \in \mathbb{R}^{\overline{n}_i}$, $i = 1, \ldots, k$. The vectors $b^i \in \mathbb{R}^{\overline{m}_i}$, $i = 1, \ldots, k$ and $b^0 \in \mathbb{R}^l$ are the RHS of the block and the linking constraints.

In Castro et al. (2023) it was proven the following for risk neutral stochastic optimization problems (2):

- The number of blocks k is the number of nodes in the multistage scenario tree.
- Any risk neutral multistage stochastic optimization problem based on the split-variable formulation (2) can be recast in the form (8), where block matrices N_i are related to constraints (2b), and the linking matrices R_i correspond to constraints (2d) equating copies of variables (Proposition 1 of Castro et al. (2023)). For example, Fig. 3 illustrates the primal block-angular structure of the risk neutral model from the scenario tree in Fig. 1.

Proposition 1 of Castro et al. (2023) can be easily extended to include the split-variable formulations (5) and (7) of the risk averse ECVaR and ECSD models. Let us consider first the split-variable formulation (5) of the ECVaR model. Without loss of generality, let us assume that $\ell^n = \ell$, that is, the number of successors is the same for all nodes in the scenario tree, but the leaf ones. Then, the number of new constraints added by the ECVaR model to the diagonal blocks $N_1, \ldots, N_k$ for each $n' \in \mathcal{N}_t, t \in \mathcal{T}'$ is:

- One constraint of type (5c);
- $t - 1$ constraints of type (5d);
- and one constraint of type (5e).

And the number of new linking constraints (5f) added to matrices $R_1, \ldots, R_k$ for each $n' \in \mathcal{N}_t, t \in \mathcal{T}'$ is $t - 1$. Since the number of nodes in $\mathcal{N}_t$ is $|\mathcal{N}_t| = \ell^{t-1}$, the total number of new constraints (5c), (5d), (5e) is

$$\sum_{t \in \mathcal{T}'} (t+1)\ell^{t-1},$$

and the total number of new linking constraints (5f) is

$$\sum_{t \in \mathcal{T}'} (t-1)\ell^{t-1}. \tag{9}$$

Similarly, new variables are added by the ECVaR model to problem (8):

- For each $n' \in \mathcal{N}_t, t \in \mathcal{T}'$:
  - $|\mathcal{P}^{n'}|$ variables $F^p_{n'}$ (one for each $p \in \mathcal{P}^{n'}$, as defined in (4));
  - $|\mathcal{P}^{n'}| - 1$ variables $F'^p_{n'}$ (one for each $p \in \mathcal{P}^{n'} \setminus \{\overline{p}_{n'}\}$);
  - one variable $s^{n'}$;
  - one slack $\xi^{n'}$.
- For each $t \in \mathcal{T}'$:
  - one variable α_t.

Since $|\mathcal{N}_t| = \ell^{t-1}$, the total number of new variables in problem (8) is

$$|\mathcal{T}'| + \sum_{t \in \mathcal{T}'} (2|\mathcal{P}^{n'}| + 1)\ell^{t-1}.$$

Given that the k blocks of (8) correspond to the nodes of the scenario tree, the new variables $F^p_{n'}$ and $F'^p_{n'}$ appear in the blocks associated to nodes n'. On the other hand, the new variables $s^{n'}, \xi^{n'}, \alpha_t$ only intervene in constraints (5e) with $F^1_{n'}$, therefore these variables and constraints can be included in the first block of (8). Figure 5 shows the structure of the constraints for a subset of the the scenario tree of Fig. 1 considering only stages 1, 2 and 3, for $\mathcal{T}' = \{3\}$. We notice that there are two groups of linking constraints, which correspond, respectively, to constraints (2d) of the risk neutral model and (5f) of the risk averse ECVaR model.

A similar analysis can be performed for ECSD. The distribution of variables across the different blocks is the same as in ECVaR, with the only difference being that the first block does not include the variables α_t (in ECSD, these are replaced by parameters $\bar{\alpha}_t$), and that this first block now includes the new constraints (7d) and new variables ν_t. The linking constraints are the same for both ECVaR and ECSD. Therefore, the constraint structure shown in Fig. 5 also applies to the ECSD formulation (7), with the only difference being that the specific form of matrix N_1 differs from that in the ECVaR model (5).

The previous discussion can be summarized in the following result:

Proposition 1. *Any risk averse multistage stochastic optimization problem that is based on the split-variable formulations (5) (for risk averse ECVaR models) or (7) (for risk averse ECSD measures), can be recast as a primal block-angular problem (8).*

Proof. The risk neutral part of models (5) and (7) was shown to match (8) in Proposition 1 of Castro et al. (2023).

As for the risk averse part, and as previously discussed, all variables in formulations (5) and (7) can be distributed across the different k blocks as follows:

For the ECVaR model (5), constraints (5c)–(5d) are distributed across the blocks according to node n', while constraints (5e) appear in the first block, and constraints (5f) act as the linking constraints.

The ECSD model (7) shares constraints (5c)–(5f) with the ECVaR model (5), which follow the same distribution across blocks. The additional constraints of the ECSD model, (7c)–(7d), are included in block N_1. □

Block-angular problems in the form (8) can be solved using the specialized IPM described in Castro (2000, 2007), Castro and Cuesta (2011), which is implemented in the BlockIP package (Castro 2016). This approach has recently been adapted to risk neutral two-stage (Castro and Lama-Zubirán 2020) and multi-stage (Castro et al. 2023) stochastic optimization problems. The specialized IPM is based on an infeasible long-step primal-dual path-following method (a comprehensive description of path-following methods can be found in the monograph Wright (1996)). An outline of the specialized IPM is provided in the earlier paper (Castro et al. 2023), which focused specifically on risk neutral multi-stage stochastic optimization problems. The most time-consuming operation of the method is solving the *normal equations* at each IPM iteration. For completeness, we review below how normal equations are solved by this specialized IPM.

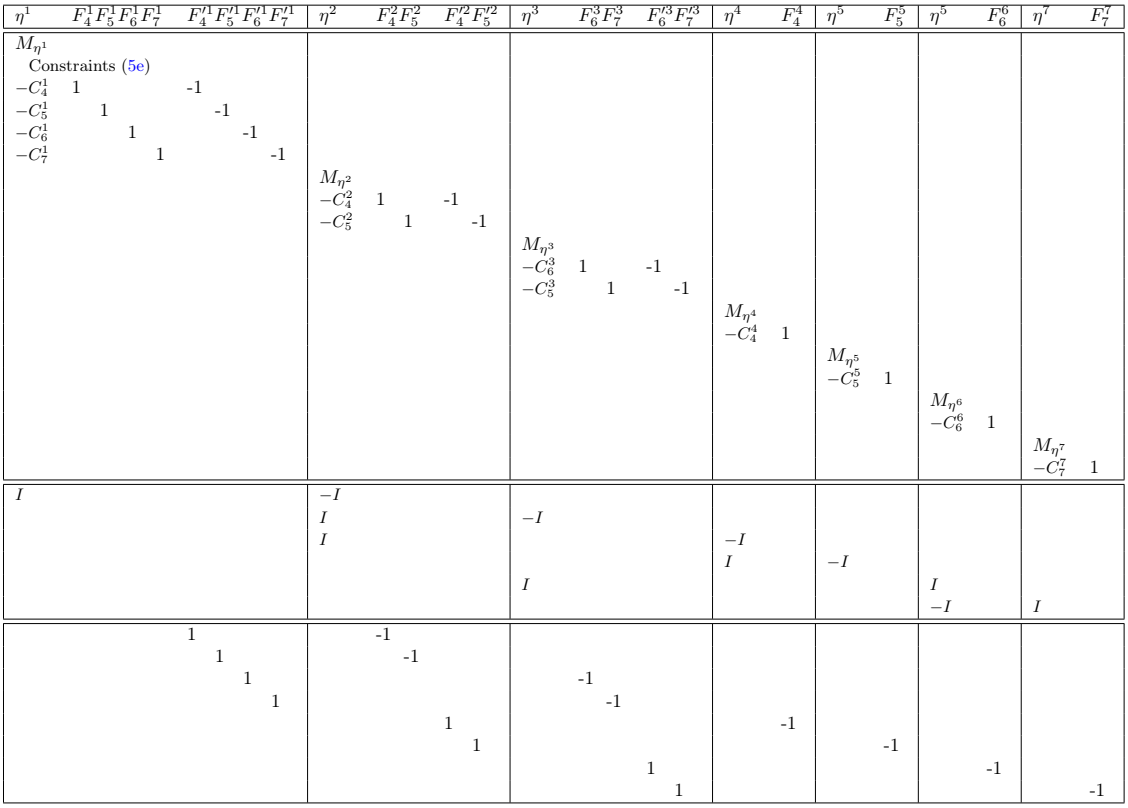


Figure 5: Constraints structure for a subset of the scenario tree of Fig. 1 considering only stages 1, 2 and 3 and ECVaR constraints for $\mathcal{T}' = \{3\}$. We focus on the structure of variables $F_{n'}^p$, denoting by η the rest of variables $(x_a, x, z, s, \xi, \alpha)$ and by M_η the submatrices $[T, W, M]$ of the constraints associated to variables η. Notice that the variables (x, ξ, α) and constraints (5e) only appear in η^1 (first block). For a detailed structure of M_η see Fig. 3. The first and second group of linking constraints correspond, respectively, to the constraints of the risk neutral and risk averse models.

Let us denote by $\overline{n} = \sum_{i=1}^{k} \overline{n_i} + l$ and $\overline{m} = \sum_{i=1}^{k} \overline{m_i} + l$ the numbers of equality constraints and variables, respectively, in problem (8). The normal equations can be written as the linear system

$$(A\Theta A^\top)\Delta\lambda = g, \tag{10}$$

where $g \in \mathbb{R}^{\overline{m}}$ is a given RHS, $\Delta\lambda \in \mathbb{R}^{\overline{m}}$ is the search direction for the Lagrange multipliers associated with the equality constraints of (8), $A \in \mathbb{R}^{\overline{m}\times\overline{n}}$ is the constraints matrix of (8), and $\Theta \in \mathbb{R}^{\overline{n}\times\overline{n}}$ is a diagonal matrix that can be decomposed into smaller submatrices $\Theta_i, i = 0, \ldots, k$, corresponding to each block of (8). The specialized IPM exploits the structure of Θ and A, such that the matrix $A\Theta A^\top$ of the normal equations can be recast as

$$\begin{aligned} A\Theta A^\top &= \left[\begin{array}{ccc|c} N_1\Theta_1 N_1^\top & & & N_1\Theta_1 R_1^\top \\ & \ddots & & \vdots \\ & & N_k\Theta_k N_k^\top & N_k\Theta_k R_k^\top \\ \hline R_1\Theta_1 N_1^\top & \ldots & R_k\Theta_k N_k^\top & \Theta_0 + \sum_{i=1}^{k} R_i\Theta_i R_i^\top \end{array}\right] \\ &= \left[\begin{array}{cc} B & C \\ C^\top & E \end{array}\right], \end{aligned} \tag{11}$$

where $B \in \mathbb{R}^{\widetilde{m}\times\widetilde{m}}$ ($\widetilde{m} = \sum_{i=1}^{k} \overline{m}_i$), $C \in \mathbb{R}^{\widetilde{m}\times l}$, and $E \in \mathbb{R}^{l\times l}$ are the blocks of $A\Theta A^\top$. Partitioning the RHS of (10) as

$$g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}, \quad g_1 \in \mathbb{R}^{\widetilde{m}}, g_2 \in \mathbb{R}^{l},$$

and the direction of Lagrange multipliers λ as

$$\Delta\lambda = \begin{bmatrix} \Delta\lambda_1 \\ \Delta\lambda_2 \end{bmatrix}, \quad \Delta\lambda_1 \in \mathbb{R}^{\widetilde{m}}, \Delta\lambda_2 \in \mathbb{R}^{l},$$

the normal equations can be written as

$$\left[\begin{array}{cc} B & C \\ C^T & E \end{array}\right] \left[\begin{array}{c} \Delta\lambda_1 \\ \Delta\lambda_2 \end{array}\right] = \left[\begin{array}{c} g_1 \\ g_2 \end{array}\right]. \tag{12}$$

Eliminating $\Delta\lambda_1$ from the first block of equations in (12) by block Gaussian elimination yields

$$(E - C^\top B^{-1} C)\Delta\lambda_2 = (g_2 - C^\top B^{-1} g_1) \tag{13}$$

$$B\Delta\lambda_1 = (g_1 - C\Delta\lambda_2). \tag{14}$$

As suggested in Castro (2000, 2007), Castro and Cuesta (2011), the specialized IPM solves system (14) by performing a Cholesky factorization for each diagonal block $N_i\Theta_i N_i^\top, i = 1, \ldots, k$ of matrix B. In contrast, solving (13) by Cholesky factorization would be impractical, since forming the system matrix $E - C^\top B^{-1} C \in \mathbb{R}^{l\times l}$ requires solving l linear systems with B. Moreover, even if formed explicitly, this matrix could be large and dense. For this reason, system (13) is solved using an iterative method, namely the preconditioned conjugate gradient (PCG) . The dimension of this system equals the number of linking constraints l, which can be very large in some applications (in particular in risk averse multistage stochastic optimization). Therefore, a good preconditioner is essential for the efficient solution of (13) with PCG.

The preconditioner considered is based on the following Neumann series expansion of the inverse of the system matrix in (13):

$$(E - C^\top B^{-1} C)^{-1} = \left(\sum_{i=1}^{\infty} (E^{-1}(C^\top B^{-1} C))^{i-1}\right) E^{-1}. \tag{15}$$

In Castro (2000) it was shown that the eigenvalues of $E^{-1}(C^\top B^{-1} C)$ lie in $(0, 1)$, which guarantees the convergence of the infinite sum in (15). The preconditioner is then obtained by truncating this series

after a certain number of terms, denoted by ϕ. Originally developed for multicommodity flow problems in Castro (2000), this preconditioner was later proven to be valid for any block-angular problem of the form (8) (Castro 2007).

The efficiency of the preconditioner depends mainly on two factors:

- *The spectral radius ρ (i.e., the largest eigenvalue) of the matrix* $(E^{-1}(C^\top B^{-1}C))$. When ρ is small, higher order terms in the series decay rapidly, so retaining only a few initial terms (small ϕ) is enough for an effective preconditioner. Unfortunately, the value of ρ is problem dependent and not know a priori. Only a few results are available: (1) ρ is smaller for quadratic than for linear problems (Castro and Cuesta 2011, Theorem 1, Proposition 2)); (2) ρ can be approximated at each interior-point iteration using the Ritz values of PCG (Bocanegra et al. 2013). Although, in theory, larger values of ϕ yield better preconditioners, increasing ϕ by one means solving an additional system with matrices E and B at every PCG iteration. For most large-scale optimization problems (including risk neutral multistage stochastic optimization (Castro et al. 2023)), the best performance of the specialized IPM is typically obtained with $\phi = 1$ (i.e., the preconditioner is simply E^{-1}). All computational results in this work are therefore reported with $\phi = 1$.
- *The efficient solution of systems involving E.* At each PCG iteration, systems with matrix E must be solved, so their efficient solution is crucial. Unlike the spectral radius ρ, the structure of E can be determined a priori from the application. For risk neutral multistage stochastic optimization problems, Castro et al. (2023) showed that E has a block tridiagonal structure, which admits a fast factorization. In the next section, we show that the addition of risk averse constraints does not complicate the structure of E: it merely introduces a new diagonal block into the block tridiagonal matrix.

By abuse of notation, matrix E will be referred to as the preconditioner throughout the remainder of this work.

4.1. The structure of preconditioner E

From (11), the preconditioner E is a $l \times l$ matrix defined as

$$E = \Theta_0 + \sum_{i=1}^{k} R_i \Theta_i R_i^\top .$$

The rows of each linking constraints matrix R_i, $i = 1, \ldots, k$ can be partitioned into two groups, depending on whether they correspond to risk neutral or risk averse constraints. Similarly, the columns of each R_i, $i = 1, \ldots, k$ can also be partitioned into two groups, corresponding to the split variables of either the risk neutral or risk averse part. Therefore, each R_i can be decomposed as

$$R_i = \begin{bmatrix} R_i^{RN} & \\ & R_i^{RA} \end{bmatrix}, \tag{16}$$

where RN and RA denote *risk neutral* and *risk averse*, respectively. Figure 5 illustrates this decomposition of the R_i matrices for the particular scenario tree of Fig. 1. By appropriately partitioning each Θ_i, $i = 0, \ldots, k$ into risk neutral and risk averse components, we can write

$$\Theta_i = \begin{bmatrix} \Theta_i^{RN} & \\ & \Theta_i^{RA} \end{bmatrix},$$

and the preconditioner E can be recast as

$$E = \begin{bmatrix} \Theta_0^{RN} + \sum_{i=1}^{k} R_i^{RN} \Theta_i^{RN} (R_i^{RN})^\top & \\ & \Theta_0^{RA} + \sum_{i=1}^{k} R_i^{RA} \Theta_i^{RA} (R_i^{RA})^\top \end{bmatrix} = \begin{bmatrix} E^{RN} & \\ & E^{RA} \end{bmatrix}. \tag{17}$$

The matrix E^{RN} is the preconditioner for risk neutral multistage stochastic optimization problems introduced in Castro et al. (2023). It was shown to be block diagonal, with each block being tridiagonal, and with the number of blocks equal to the number of nested two-stage trees (denoted as $\bar{k} = \frac{\ell^{|\mathcal{T}|-1}-1}{\ell-1}$) in the multistage scenario tree (Castro et al. 2023, Proposition 2).

The risk averse constraints extend the preconditioner with a new block $E^{RA} \in \mathbb{R}^{l^{RA} \times l^{RA}}$, where l^{RA} denotes the number of linking risk averse constraints (5f) defined in (9). The following proposition shows that E^{RA} is diagonal.

Proposition 2. *The submatrix*

$$E^{RA} = \Theta_0^{RA} + \sum_{i=1}^{k} R_i^{RA} \Theta_i^{RA} (R_i^{RA})^\top$$

of the preconditioner E is diagonal.

Proof. Let

$$R^{RA} = \begin{bmatrix} \mathbf{0}_1 & R_1^{RA} & \mathbf{0}_2 & R_2^{RA} & \cdots & \mathbf{0}_k & R_k^{RA} \end{bmatrix} \in \mathbb{R}^{l^{RA} \times \widetilde{n}}$$

be the matrix corresponding to the risk averse rows of the linking constraints, where $\widetilde{n} = \sum_{i=1}^{k} \overline{n}_i$ is the number of variables of (8) excluding the slacks x^0, and $\mathbf{0}_i$ denotes the zero matrix that precedes R_i^{RA} in the last l^{RA} rows of R_i, as shown in (16). Let $\widetilde{\Theta} \in \mathbb{R}^{\widetilde{n} \times \widetilde{n}}$ be a diagonal matrix composed of the diagonal block Θ_i, $i = 1, \ldots, k$ (that is, $\widetilde{\Theta}$ coincides with Θ excluding the block Θ_0). Then E^{RA} can be written as

$$E^{RA} = \Theta_0^{RA} + R^{RA} \widetilde{\Theta} (R^{RA})^\top.$$

Each row of R^{RA} corresponds to one constraint (5f), which contains exactly two nonzero coefficients (a $+1$, and a -1) for the variables $F'^{p}_{n'}$ and $F^{p+1}_{n'}$, with $p \in \mathcal{P}^{n'} \setminus \{\overline{p}(n')\}$, $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$. Since $F'^{p}_{n'}$ and $F^{p+1}_{n'}$ are auxiliary variables used to compute the accumulated cost from the last node $\overline{p}(n')$ up to node p of $\mathcal{P}^{n'}$, for $n' \in \mathcal{N}_t$ and $t \in \mathcal{T}'$, they appear in exactly one risk averse linking constraint (5f). Therefore, each row of R^{RA} has only two nonzero coefficients ($+1$ and -1), and each column contains at most one nonzero coefficient (either $+1$ or -1). This is illustrated in Figs. 4 and 5. Consequently, $R^{RA}\widetilde{\Theta}(R^{RA})^\top$ is diagonal, and E^{RA} is the sum of two diagonal matrices. □

Proposition 2 can be used for an efficient computation of E^{RA}. Let $j_{F'^{p}_{n'}}$ and $j_{F^{p+1}_{n'}}$ denote the two columns corresponding to the $+1$ and -1 nonzeros entries in row $p \in \mathcal{P}^{n'} \setminus \{\overline{p}(n')\}$, $n' \in \mathcal{N}_t$, $t \in \mathcal{T}'$ of E^{RA}. Then E^{RA} can be computed as:

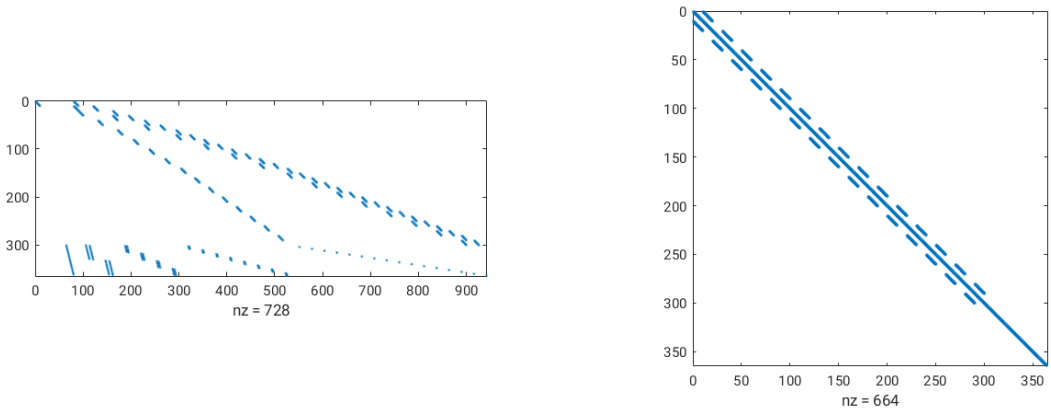
$$E^{RA} = \Theta_0^{RA} + \operatorname{diag}\left(\left(\widetilde{\Theta}_{j_{F'^{p}_{n'}}} + \widetilde{\Theta}_{j_{F^{p+1}_{n'}}} \right)_{p \in \mathcal{P}^{n'} \setminus \{\overline{p}(n')\},\, n' \in \mathcal{N}_t,\, t \in \mathcal{T}'} \right), \tag{18}$$

where diag denotes the operator that constructs a diagonal matrix from a vector.

Figure 6 illustrates the structure of $\begin{bmatrix} R_1 & \ldots & R_k \end{bmatrix}$ and the preconditioner E for a problem with $|\mathcal{T}| = 5$ stages, $\ell^n = 2$ for $n \in \mathcal{N} : t^n < |\mathcal{T}|$, and $\mathcal{T}' = \{5\}$. Note that the number of tridiagonal blocks of E^{RN} (corresponding to the two-stage subtrees of the multistage scenario tree) is $\bar{k} = \frac{2^4-1}{2-1} = 15$, and that the dimension of E^{RA} is, according to (9), $(5-1)2^{(5-1)} = 64$.

5. Computational results

The code MSSO-BlockIP, originally developed in Castro et al. (2023) for risk neutral multistage stochastic optimization problems, has been extended in this work to handle the split-variable formulations (5) and (7) of the ECVaR and ECSD risk averse models. MSSO-BlockIP runs on top of BlockIP, an efficient C++ implementation of the specialized IPM (Castro 2007, 2016). The PCG preconditioner has been augmented with the new block E^{RA}, associated with the risk averse linking constraints, which is computed efficiently using (18).



(a) (b)

Figure 6: Structure of (a) linking constraints and (b) preconditioner E, for a problem with $|\mathcal{T}| = 5$ stages, $\ell^n = 2$ for $n \in \mathcal{N} : t^n < |\mathcal{T}|$, and $\mathcal{T}' = \{5\}$.

Table 1: Sizes of revenue management instances. ECVaR and ECSD instances have the same sizes, except that ECSD instances include one additional constraint.

Instance	$\lvert\mathcal{T}\rvert$	ℓ	k	# var.	# cons.	l
T5-L5	5	5	781	1,943,757	691,831	626,500
T8-L3	8	3	3,280	8,130.360	2,865,663	2,638,509
T3-L100	3	100	10,101	25,321,702	9,130,901	8,100,000
T4-L25	4	25	16,276	40,781,002	14,677,701	13,066,875
T10-L3	10	3	29,524	73,267,968	25,833,537	23,795,547
T8-L5	8	5	97,656	243,611,882	86,738,081	78,670,875
T6-L10	6	10	111,111	278,076,712	99,699,911	89,388,000

We considered a set of seven large multistage stochastic optimization instances from the realistic revenue management problem in Escudero et al. (2013). Risk neutral versions of this model were used and are fully detailed in Castro et al. (2023). For each of these seven instances, we constructed two risk averse models using the ECVaR and ECSD risk measures. Table 1 reports the dimensions of these instances, including the number of stages $|\mathcal{T}|$; the number of immediate successor nodes ℓ for each node (excluding last-stage nodes); the number of blocks in the optimization problem k (which coincides with the number of nodes in the multistage tree); and the total number of variables ("# var.", excluding slacks), block constraints ("# cons."), and linking constraints l of the optimization problem. The ECVaR and ECSD instances have the same size, with the only minor difference being that each ECSD instance includes one additional block constraint. Both the MSSO-BlockIP code and the instances are available from the authors by request.

Due to the large size of the test instances, we considered an optimality tolerance of 10^{-3} in MSSO-BlockIP (i.e., a primal-dual feasible solution with a relative duality gap below 10^{-3} was required). We compared MSSO-BlockIP with the barrier algorithm in CPLEX v22.1.2, a state-of-the-art implementation of an IPM. Indeed, such an IPM is the only reliable, non-heuristic option for solving instances of this scale. Default CPLEX parameter values were used, except for the following:

- The optimality tolerance was set to 10^{-3}, as in MSSO-BlockIP, to ensure a fair comparison.
- We selected the standard primal–dual algorithm of the CPLEX barrier, instead of the homogeneous self-dual algorithm (used by default in most cases). Although the homogeneous self-dual algorithm is generally more robust and less prone to infeasibility issues, it does not benefit from a relaxed optimality tolerance, making it slower and leading to an unfair comparison with MSSO-BlockIP.

By contrast, the standard primal–dual algorithm can be terminated early with feasible primal and dual solutions that satisfy the relaxed optimality tolerance.

- The CPLEX crossover was deactivated (so that the solver provides an interior-point solution rather than a basic one), again to ensure a fair comparison.
- The default aggregator in CPLEX preprocessing was disabled in runs with the split-variable formulations (5) and (7) of the ECVaR and ECSD models, since it was observed to destroy the structure of the constraint matrix and slow down the solution process. To confirm this effect, for each instance we performed two runs of models (5) and (7), with and without the default aggregator.

It is worth noting that BlockIP carries out the matrix factorizations required by the linear systems through the academic block sparse Cholesky package described in Ng and Peyton (1993). In contrast, modern commercial IPM solvers such as CPLEX v22.1.2 incorporate highly optimized numerical linear algebra routines designed to take advantage of hardware features (Mészáros 2016). Consequently, the superior performance of MSSO-BlockIP over CPLEX should be attributed to the specialized algorithm proposed in this work, rather than to low-level implementation details.

The computational experiments were performed on a DELL PowerEdge R7625 with two 3.6GHz AMD EPYC 9474F CPUs (192 total cores) and 1024 GB of RAM, running GNU/Linux (openSuse 15.5). All runs were executed sequentially, without exploiting multithreading capabilities.

Tables 2 and 3 report the results obtained for, respectively, the ECVaR and ECSD instances, with MSSO-BlockIP and CPLEX. For CPLEX, we considered three different configurations: CPLEX$^{(1)}$ solves the split-variable formulations with the aggregator in preprocessing; CPLEX$^{(2)}$ solves the split-variable formulations without the aggregator in preprocessing; and CPLEX$^{(3)}$ solves the compact formulations with the default preprocessing (which includes the aggregator), since this was the best option for the compact formulation. For MSSO-BlockIP and CPLEX, the tables provide the number of IPM iterations (columns "it") and execution time (columns "CPU"). For MSSO-BlockIP, the total number of PCG iterations (columns "PCG") is also provided. The last rows of Tables 2 and 3 show the total CPU time needed to solve all the instances.

Table 2: Results for ECVaR revenue management instances. CPU times are in seconds, or in days if exceeding 86,400 seconds (1 day).

Instance	MSSO-BlockIP			CPLEX$^{(1)}$		CPLEX$^{(2)}$		CPLEX$^{(3)}$	
	it.	PCG	CPU	it.	CPU	it.	CPU	it.	CPU
ECVaR-T5-L5	314	2,579	125	17	460	31	143	26	254
ECVaR-T8-L3	201	8,253	1,089	26	6,730	38	892	45	44,324
ECVaR-T3-L100	524	8,974	4,532	31	3,457	41	5,794	27	83,366
ECVaR-T4-L25	467	39,674	25,541	26	32,818	61	8,858	32	19,648
ECVaR-T10-L3	541	40,136	42,587	34	2.70d	48	21,297		‡
ECVaR-T8-L5	768	41,352	1.75d	—	>32d†	66	5.50d		‡
ECVaR-T6-L10	827	55,808	2.69d	—	>36d†	57	2.68d	—	>26d†
CPU time for all instances:			*5.3d*		*>71d*		*8.6d*		‡

(1) CPLEX v22.1.2 solved the split-variable formulation (5), with default aggregator.
(2) CPLEX v22.1.2 solved the split-variable formulation (5), deactivating default aggregator.
(3) CPLEX v22.1.2 solved the compact formulation (3), with default aggregator.
— Execution was stopped early by excessive expected CPU time.
† Estimated number of days of CPU from the total number of arithmetic operations for factorizations reported by CPLEX.
‡ Out of memory: the solver required more than 1000 GB of RAM.

Table 3: Results for ECSD revenue management instances. CPU times are in seconds, or in days if exceeding 86,400 seconds (1 day).

Instance	MSSO-BlockIP			CPLEX$^{(1)}$		CPLEX$^{(2)}$		CPLEX$^{(3)}$	
	it.	PCG	CPU	it.	CPU	it.	CPU	it.	CPU
ECSD-T5-L5	321	2,067	112	17	443	31	131	26	255
ECSD-T8-L3	334	8,491	1,208	26	7,037	37	981	43	23,380
ECSD-T3-L100	779	7,942	4,463	32	3,777	42	5,691	20	28,389
ECSD-T4-L25	460	30,320	18,781	31	28,713	50	7,331	33	4,141
ECSD-T10-L3	526	34,654	36,595	44	3.11d	43	25,667		‡
ECSD-T8-L5	767	40,179	1.66d	—	>42d†	56	5.60d		‡
ECSD-T6-L10	674	45,864	2.17d	—	>59d†	80	3.72d	65	4.44d
CPU time for all instances:			*4.5d*		*>101d*		*9.8d*		‡

(1) CPLEX v22.1.2 solved the split-variable formulation (7), with default aggregator.
(2) CPLEX v22.1.2 solved the split-variable formulation (7), deactivating default aggregator.
(3) CPLEX v22.1.2 solved the compact formulation (6), with default aggregator.
— Execution was stopped early by excessive expected CPU time.
† Estimated number of days of CPU from the total number of arithmetic operations for factorizations reported by CPLEX.
‡ Out of memory: the solver required more than 1000 GB of RAM.

From Tables 2 and 3, it can be clearly seen that, for both ECVaR and ECSD risk averse measures, the new split-variable models (5) and (7) provide a much better formulation than the standard compact models (3) and (6) for state-of-the-art IPM solvers, with the only exception of instance ECSD-T4-L25. For the largest instances, the compact models proved intractable for CPLEX, as they required either more than 1000 GB of RAM or several days of execution time. It is also evident that CPLEX$^{(2)}$ outperforms CPLEX$^{(1)}$ (and by a large margin in the larger instances). This indicates that preserving the original structure of formulations (5) and (7), by deactivating the aggregator, is preferable when solving the split-variable formulation with a general-purpose solver. Only in two cases (namely, ECVaR-T3-L100 and ECSD-T3-L100, corresponding to the same problem T3-L100) did the aggregator prove to be a good option for the split-variable formulation. Overall, however, CPLEX$^{(2)}$ can be considered the fastest of the three CPLEX configurations.

When comparing CPLEX$^{(2)}$, the best variant of the state-of-the-art solver, with the specialized IPM MSSO-BlockIP, we find that, unlike in risk neutral problems, where MSSO-BlockIP consistently outperformed CPLEX (Castro et al. 2023), this does not always hold for risk averse problems. For the five smaller instances in Tables 2 and 3, CPLEX$^{(2)}$ is generally faster than MSSO-BlockIP, with the exception of ECVaR-T5-L5, ECVaR-T3-L100, ECSD-T5-L5, and ECSD-T3-L100. However, for the two larger instances in those tables, MSSO-BlockIP significantly outperformed CPLEX$^{(2)}$. For example, in ECVaR-T8-L5, ECSD-T8-L5 and ECSD-T6-L10, MSSO-BlockIP required 1.75, 1.66 and 2.17 days of CPU time, respectively, while CPLEX$^{(2)}$ ran for 5.5, 5.6 and 3.72 days. For ECVaR-T6-L10, both solvers reported similar execution times. As shown in the last rows of Tables 2 and 3, MSSO-BlockIP required substantially less total CPU time than the other approaches to solve all instances of the test set.

It is also observed in Tables 2 and 3 that MSSO-BlockIP required many more IPM iterations than CPLEX. This is due to the use of PCG in MSSO-BlockIP, which provides inexact Newton search directions. However, it is known that inexact directions can still be used in IPMs while preserving convergence guarantees (Gondzio 2013).

Memory usage shows an even clearer advantage for MSSO-BlockIP over a general IPM solver. Tables 4 and 5 report the RAM consumption (in gigabytes) of MSSO-BlockIP and the three CPLEX configurations (CPLEX$^{(1)}$, CPLEX$^{(2)}$, and CPLEX$^{(3)}$) for all ECVaR and ECSD instances. The results demonstrate that MSSO-BlockIP consistently requires only a fraction of the memory needed by CPLEX. This efficiency allows MSSO-BlockIP to be executed on substantially less powerful hardware, leading to lower energy demand and, in turn, reduced CO_2 emissions. Moreover, this lower memory consumption makes it possible to run several problem instances with MSSO-BlockIP on different CPUs simultaneously on the same hardware, thereby reducing overall wall-clock execution times, whereas only one run of the general-

Table 4: Memory requirements (in GB of RAM) for ECVaR revenue management instances.

Instance	MSSO-BlockIP	CPLEX$^{(1)}$	CPLEX$^{(2)}$	CPLEX$^{(3)}$
ECVaR-T5-L5	0.6	3.3	2.2	2.7
ECVaR-T8-L3	2.6	13	9.1	54
ECVaR-T3-L100	11	29	31	52
ECVaR-T4-L25	16	56	49	37
ECVaR-T10-L3	41	102	89	> 1000
ECVaR-T8-L5	137	482	373	> 1000
ECVaR-T6-L10	134	553	391	656

(1) CPLEX solved the split-variable formulation (5), with default aggregator.
(2) CPLEX solved the split-variable formulation (5), deactivating default aggregator.
(3) CPLEX solved the compact formulation (3), with default aggregator.

Table 5: Memory requirements (in GB of RAM) for ECSD revenue management instances.

Instance	MSSO-BlockIP	CPLEX$^{(1)}$	CPLEX$^{(2)}$	CPLEX$^{(3)}$
ECSD-T5-L5	0.6	3.3	2.5	2.7
ECSD-T8-L3	2.6	13	9.0	32
ECSD-T3-L100	13	29	36	30
ECSD-T4-L25	23	55	49	26
ECSD-T10-L3	41	99	104	> 1000
ECSD-T8-L5	137	502	378	> 1000
ECSD-T6-L10	157	557	396	338

(1) CPLEX solved the split-variable formulation (7), with default aggregator.
(2) CPLEX solved the split-variable formulation (7), deactivating default aggregator.
(3) CPLEX solved the compact formulation (6), with default aggregator.

purpose solver can be performed for large instances.

6. Conclusions

The standard (compact) formulations of risk averse multistage optimization problems have proven to be intractable for state-of-the-art solvers when applied to large-scale instances. The split-variable reformulation introduced in this work made it possible to solve all such large-scale instances with general-purpose IPM solvers. In addition, the specialized IPM implemented in MSSO-BlockIP was extended to include risk averse measures, and it outperformed the state-of-the-art IPM solver on the largest instances. Thus, this work contributes two novel approaches for addressing challenging risk averse multistage stochastic optimization problems.

Future work in this line of research includes extending the approach to multi-horizon problems, that is, models that simultaneously account for strategic (long-term) and operational (short-term) uncertainties and decisions in risk averse multistage stochastic optimization.

Acknowledgements

J. Castro was supported by the MCIN/AEI/10.13039/501100011033/FEDER,EU grant PID2022-139219OB-I00. L.F. Escudero was supported by the MCIN/AEI/10.13039/501100011033 grant PID2021-122640OB-I00. J.F. Monge was supported by the MCIN/AEI/10.13039/501100011033 grants PID2021-122344NB-I00 and PID2022-136383NB-I00.

References

Ahmed, S. (2006). Convexity and decomposition of mean-risk stochastic programs. *Mathematical Programming, Ser. B*, 106, 433–446. https://doi.org/10.1007/s10107-005-0638-8. 2

Alonso-Ayuso, A., Escudero, L.F. Guignard, M. & Weintraub, A. (2018). Risk management for forestry planning under uncertainty in demands and prices. *European Journal of Operational Research*, 267, 1051–1074. https://doi.org/10.1016/j.ejor.2017.12.022. 2, 5, 6

Artzner, P., Delbaen, F., Eber, J.M. & Heath, D. (1999). Coherent measures of risk. *Mathematical Finance*, 9, 203-–228. https://doi.org/10.1111/1467-9965.00068. 2

Birge, J.R., & Louveaux, F.V. (2011). *Introduction to Stochastic Programming (2nd ed.)*. New York: Springer, https://doi.org/10.1007/978-1-4614-0237-4. 1

Bocanegra, S., Castro, J., & Oliveira, A.R.L. (2013). Improving an interior-point approach for large block-angular problems by hybrid preconditioners. *European Journal of Operational Research*, 231, 263–273. https://doi.org/10.1016/j.ejor.2013.04.007. 13

Castro, J. (2000). A specialized interior-point algorithm for multicommodity network flows. *SIAM Journal on Optimization*, 10, 852–877. https://doi.org/10.1137/S1052623498341879. 10, 12, 13

Castro, J. (2007). An interior-point approach for primal block-angular problems. *Computational Optimization and Applications*, 36, 195–219. https://doi.org/10.1007/s10589-006-9000-1. 10, 12, 13, 14

Castro, J. (2016). Interior-point solver for convex separable block-angular problems. *Optimization Methods and Software*, 31, 88–109. https://doi.org/10.1080/10556788.2015.1050014. 3, 10, 14

Castro, J., & Cuesta, J. (2011). Quadratic regularizations in an interior-point method for primal block-angular problems. *Mathematical Programming, Ser. A*, 130, 415–445. https://doi.org/10.1007/s10107-010-0341-2. 3, 10, 12, 13

Castro, J., Escudero, L.F. & Monge, J.F. (2023). On solving large-scale multistage stochastic problems with a new specialized interior-point approach. *European Journal of Operational Research*, 310, 268–285. https://doi.org/10.1016/j.ejor.2023.03.042. 2, 3, 4, 9, 10, 13, 14, 15, 17

Castro, J. & de la Lama-Zubirán, P. (2020). A new interior-point approach for large two-stage stochastic problems. *Optimization Methods and Software*, 37, 801–829. https://doi.org/10.1080/10556788.2020.1841190. 2, 4, 10

Charnes, A. & Cooper, W.W. (1959). Chance-constrained programming. *Management Sciences*, 6, 73–79. https://doi.org/10.1287/mnsc.6.1.73. 2

Charpentier, A. & Oulidi, A. (2009). Estimating allocations for Value-at-Risk portfolio optimization. *Mathematical Methods of Operations Research*, 69, 395–410. https://doi.org/10.1007/s00186-008-0244-7. 2

Dembo, R. (1991). Scenario optimization. *Annals of Operations Research*, 30, 63–80. https://doi.org/10.1007/BF02204809. 2

Dentcheva, D. (2023). Relations between risk averse model in extended two-stage stochastic optimization. *Serdica Mathematical Journal*, 49, 49–76. https://doi.org/10.55630/serdica.2023.49.49-76. 2

Dentcheva, D. & Ruszczyński, A. (2003). Optimization with stochastic dominance constraints. *SIAM Journal on Optimization*, 14, 548–566. https://doi.org/10.1137/S1052623402420528. 2, 5, 7

Dentcheva, D. & Ruszczyński, A. (2024). Numerical methods for problems with stochastic dominance constraints. In: Risk-Averse Optimization and Control, 327–368. Springer. https://doi.org/10.1007/978-3-031-57988-2_7. 2

Eppen, G.D., Martin, R.K. & Schrage, L. (1989). OR Practice—A Scenario approach to capacity planning. *Operations Research*, 34, 517–527. https://doi.org/10.1287/opre.37.4.517 2

Escudero, L.F., Garín, A. & Unzueta, A. (2017). Cluster Lagrangean Decomposition for risk averse in multistage stochastic optimization. *Computers and Operations Research*, 85, 154–171. https://doi.org/10.1016/j.cor.2015.09.005. 2

Escudero, L.F., Garín, A., Monge; J.F. & Unzueta, A. (2020). Some matheuristic algorithms for multistage stochastic optimization models with endogenous uncertainty and risk management. *European Journal of Operational Research*, 285, 988–1001. https://doi.org/10.1016/j.ejor.2020.02.046. 2

Escudero, L.F. & Monge, J.F. (2018). On capacity expansion planning under strategic and operational uncertainties based on stochastic dominance risk averse management. *Computational Management Science*, 15, 479–500. https://doi.org/10.1007/s10287-018-0318-9. 2, 7, 8

Escudero, L.F. & Monge, J.F. (2021). On multistage multiscale stochastic capacitated multiple allocation hub network expansion planning. *Mathematics*, 9, 3177. https://doi.org/10.3390/math9243177. 2

Escudero, L.F. & Monge, J.F. (2023). On risk management of multistage multiscale FLP under uncertainty. Chapter in Eiselt, H.A.& Marianov, V. (eds.), Uncertainty in facility location models. International series in Operations Reseaarch & Management Sciences, 437, 355–390, Springer. https://doi.org/10.1007/978-3-031-32338-6_14. 2, 3, 7

Escudero, L.F., Garín, A., Merino, M. & Pérez, G. (2016). On time stochastic dominance induced by mixed integer-linear recourse in multistage stochastic programs. *European Journal of Operational Research*, 249, 164–176. https://doi.org/10.1016/j.ejor.2015.03.050. 8

Escudero, L.F., Garín, M.A., Monge, J.F. & Unzueta, A. (2020). Some matheuristic algorithms for multistage stochastic optimization models with endogenous uncertainty and risk management. *European Journal of Operational Research*, 285, 988–1001. https://doi.org/10.1016/j.ejor.2020.02.046. 8

Escudero, L.F., Monge, J.F. & Romero-Morales, D. (2015). An SDP approach for multiperiod mixed 0–1 linear programming models with stochastic dominance constraints for risk management. *Computers & Operations Research*, 58, 32–40. https://doi.org/10.1016/J.COR.2014.12.007. 8

Escudero, L.F., Monge, J.F., Romero-Morales, D., & Wang, J. (2013). Expected future value decomposition based bid price generation for large-scale network revenue management. *Transportation Science*, 47, 181–197. https://doi.org/10.1287/trsc.1120.0422. 15

Fabian, C, Mitra, G. Roman, D. &. Zverovich, V. (2011). An enhanced model for portfolio choice with SSD criteria: a constructive approach. *Quantitative Finance*, 11, 1525–1534. https://doi.org/10.1080/14697680903493607. 2

Gaivoronski, A.A & Pflug, G. (2005). Value-at-risk in portfolio optimization: properties and computational approach. *Journal of Risk*, 7, 11–31. https://doi.org/10.21314/JOR.2005.106. 2

Gollmer, R., Gotzes, Y. & Schultz, R. (2011). A note on second-order stochastic dominance constraints induced by mixed-integer linear recourse. *Mathematical Programming, Ser. A*, 126, 179–190. https://doi.org/10.1007/s10107-009-0270-0. 2, 7

Gollmer, R., Neise, F. & Schultz, R. (2008). Stochastic programs with first-order stochastic dominance constraints induced by mixed-integer linear recourse. *SIAM Journal on Optimization*, 19, 552–571. https://doi.org/10.1137/060678051. 2, 8

Gondzio, J. (2012). Interior point methods 25 years later. *European Journal of Operational Research*, 218, 587–601. https://doi.org/10.1016/j.ejor.2011.09.017. 2

Gondzio, J. (2013). Convergence analysis of an inexact feasible interior point method for convex quadratic programming. *SIAM Journal on Optimization*, 23, 1510–1527. https://doi.org/10.1137/120886017. 17

Gondzio, J. (2025). Interior point methods in the year 2025. *EURO Journal on Computational Optimization*, 13, 100105. https://doi.org/10.1016/j.ejco.2025.100105. 2

Gondzio, J., González-Brevis, P., & Munari, P. (2016). Large-scale optimization with the primal-dual column generation method. *Mathematical Programming Computation*, 8, 47–82. https://doi.org/10.1007/s12532-015-0090-6. 2

Gondzio, J., & Grothey, A. (2007) Solving nonlinear portfolio optimization problems with the primal-dual interior point method. *European Journal of Operational Research*, 181, 1019–1029. https://doi.org/10.1016/j.ejor.2006.03.006. 2

Guigues, V. (2014). SDDP for some interstage dependent risk-averse problems and application to hydro-thermal planning. *Computational Optimization and Applications*, 57, 167–203. https://doi.org/10.1007/s10589-013-9584-1. 2

Guigues, V. &. Sagastizabal, C. (2013). Risk averse policies for large-scale multistage stochastic linear programs. *Mathematical Programming, Ser. A*, 138, 167–198. https://doi.org/10.1007/s10107-012-0592-1. 2

Homem-de-Mello, T. & Pagnoncelli, B.K. (2016). Risk aversion in multistage stochastic programming: A modeling and algorithmic perspective. *European Journal of Operational Research*, 249, 188–1996. https://doi.org/10.1016/j.ejor.2015.05.048. 2, 5

Kibaek, K., & Zavala, V.M. (2018). Algorithmic innovations and software for the dual decomposition method applied to stochastic mixed-integer programs. *Mathematical Programming Computation*, 10, 225–266. https://doi.org/10.1007/s12532-017-0128-z. 2

Kormik, V. & Morton, D.P. (2015). Evaluating policies in risk averse multistage stochastic programming. *Mathematical Programming, Ser. A*, 152, 275–300. https://doi.org/10.1007/s10107-014-0787-8. 2

Lustig, I.J., Mulvey, J.M., & Carpenter, T.J. (1991). Formulating stochastic programs for interior point methods. *Operations Research*, 39, 757–770. https://doi.org/10.1287/opre.39.5.757. 2, 4

Markowitz, H.M. (1952). Portfolio selection. *Journal of Finance*, 7, 77–91. https://doi.org/10.2307/2975974 2

Mészáros, C. (2016). Exploiting hardware capabilities in interior point methods. *Optimization Methods and Software*, 31, 435–443. https://doi.org/10.1080/10556788.2015.1104677. 16

Ng, E., & Peyton, B.W. (1993). Block sparse Cholesky algorithms on advanced uniprocessor computers. *SIAM Journal on Scientific Computing*, 14, 1034–1056. https://doi.org/10.1137/0914063. 16

Ogryczak, W. & Ruszczyński, A. (1999). From stochastic dominance to mean-risk models: semideviations as risk measures.. *European Journal of Operational Research*, 116, 33–50. https://doi.org/10.1016/S0377-2217(98)00167-2. 2

Pflug, G.CH. &. Pichler, A. (2015). Time consistent decisions and temporal decomposition of coherent risk functional. *Mathematics of Operations Research*, 41, 682–699. https://doi.org/10.1287/moor.2015.0747. 1, 2

Rockafellar, R.T. & Uryasev, S. (2000). Optimization on conditional value-at-risk. *Journal of Risk*, 2, 21–41. https://doi.org/10.21314/JOR.2000.038. 2, 5

Rudloff, B., Street, A. & Valladao, D. (2014). Time consistency and risk averse dynamic decision models: Definition, interpretation and practical consequences. *European Journal of Operational Research*, 234, 743–750. https://doi.org/10.1016/j.ejor.2013.11.037. 2, 5

Ruszczyński, A. (1993). Interior point methods in stochastic programming. International Institute for Applied Systems Analysis, WP-93-8, Laxenburg, Austria. http://pure.iiasa.ac.at/id/eprint/3804/1/WP-93-008.pdf. Accessed online 23 September 2025. 2, 4

Shapiro, A., Dentcheva, D. & Ruszczyński, A. (2009). Lectures on Stochastic Programming. *MPS-SIAM Series on Optimization*, 253–332. https://doi.org/10.1137/1.9780898718751. 1, 2, 5, 8

Shapiro, A. (2021). Tutorial on risk neutral, distributionally robust and risk averse multistage stochastic programming. *European Journal of Operational Research*, 288, 1–13. https://doi.org/10.1016/j.ejor.2020.03.065. 1, 2, 5

Schultz, R. & Tiedemann, S. (2006). Conditional value-at-risk in stochastic programs with mixed integer recourse *Mathematical Programming, Ser. B*, 105, 365-386. https://doi.org/10.18452/8324. 2

Wright, S.J. (1997). *Primal-Dual Interior-Point Methods*. Philadelphia, PA: SIAM. https://doi.org/10.1137/1.9781611971453. 2, 10