An SDP approach for multiperiod mixed 0–1 linear programming models with stochastic dominance constraints for risk management*

Laureano F. Escudero[1], **Juan Francisco Monge**[2] **and Dolores Romero Morales**[3]

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

[2] *Centro de Investigación Operativa*
Universidad Miguel Hernández, Elche (Alicante), Spain
e-mail: monge@umh.es

[3] *Saïd Business School*
University of Oxford, Oxford, United Kingdom
e-mail: dolores.romero-morales@sbs.ox.ac.uk

May 3, 2014

Abstract

In this paper we consider multiperiod mixed 0–1 linear programming models under uncertainty. We propose a risk averse strategy using stochastic dominance constraints (SDC) induced by mixed-integer linear recourse as the risk measure. The SDC strategy extends the existing literature to the multistage case and includes both first-order and second-order constraints. We propose a stochastic dynamic programming (SDP) solution approach, where one has to overcome the negative impact the cross-scenario constraints, due to SDC, have on the decomposability of the model. In our computational experience we compare our SDP against a commercial optimization package, in terms of solution accuracy and elapsed time. We use supply chain planning instances, where procurement, production, inventory, and distribution decisions need to be made under demand uncertainty. We confirm the hardness of the testbed, where the benchmark cannot find a feasible solution for half of the test instances while we always find one, and show the appealing tradeoff of SDP, in terms of solution accuracy and elapsed time, when solving medium-to-large instances.

*This research has been partially supported by the grants MTM2009-14087-C04-01 from the Spanish Ministry of Science and Innovation and Risk Management from Comunidad de Madrid, Spain.

Keywords: multiperiod stochastic mixed 0–1 linear programming; risk averse; stochastic dominance constraints; stochastic dynamic programming; cross-scenario constraints.

1 Introduction

Let $\mathcal{T}$ be a time horizon of length T. Let a_t and c_t be row vectors, b_t a column vector, and $A^t_{t'}$ and $B^t_{t'}$ matrices, all of adequate dimensions. Consider the following multiperiod mixed 0–1 linear programming model:

$$\text{minimize} \sum_{t\in\mathcal{T}}(a_t x_t + c_t y_t)$$

subject to (MILP)

$$\sum_{t'=1}^{t}(A^t_{t'}x_{t'} + B^t_{t'}y_{t'}) = b_t \quad \forall t \in \mathcal{T}$$
$$x_t \in \{0,1\}^{nx(t)}, y_t \in I\!R^{ny(t)} \quad \forall t \in \mathcal{T},$$

where x_t (resp. y_t) is a vector of 0–1 (resp. continuous) variables. In this paper we consider the uncertainty on the parameters in (MILP) and assume it to be represented by a multiperiod scenario tree, see Figure 1.

In uncertain environments, it is common to optimize the expected value of the objective function, yielding a risk neutral solution. Because of Risk Management considerations, one would like to implement a risk averse solution to ensure that the variability in the objective function values is low and, in particular, for those on the right tail (for a minimization formulation). The existing risk averse models in the literature consider the following coherent measures [8]: scenario optimization [17, 22], semi-deviations [1, 35], excess probabilities [43], Value-at-Risk [27], conditional Value-at-Risk [1, 4, 9, 38, 40, 44, 31] and stochastic dominance constraints strategies [29, 30]. We refer the reader to [4] for a state-of-the-art survey on risk averse strategies.

In this paper we use stochastic dominance constraints (SDC) as the risk measure [51]. SDC strategies are very appealing for Risk Management. See, for instance, [4, 15] for applications of SDC to energy and copper extraction planning, respectively, where the resulting Deterministic Equivalent Model (DEM), [54], is solved using the commercial Mixed Integer Programming (MIP) solver CPLEX. For some given scenario function, usually the objective function being optimized, SDC strategies aim to reduce the chance of having scenarios with large objective function values (first-order SDC) or the scenario objective function values themselves (second-order SDC). In addition to the increase in the number of constraints and decision variables, including 0–1, the SDC strategies require cross-scenario constraints. These affect the nice block structure of risk neutral DEMs, and therefore pose a challenge to traditional decomposition approaches, the only viable approach to deal with large stochastic models, [11, 13, 14, 21, 23, 49]. Therefore, works on algorithms for SDC strategies do not abound in the literature. Besides, SDC is a relatively new risk averse strategy, though promising, see [18, 19, 20].

Our contribution is to propose an SDC strategy for the mixed 0–1 linear programming problem (MILP) and a decomposition approach that can successfully deal with cross-scenario constraints. Our SDC strategy is the multistage extension of the mixture of the first-order SDC in [30] and the second-order SDC in [29] induced by mixed-integer linear recourse for two-stage programs. Thus, in addition to the computational burden coming from SDC, we have multiple stages whose scenario groups need to satisfy the non-anticipativy principle. In [24], the multistage SDC model is treated via an exact branch-and-fix coordination methodology that requires an affordable elapsed time but it is only intended for medium-scale instances, although difficult ones for commercial MIP solvers as illustrated in their computational experience. To be able to handle larger instances, in this paper we propose a metaheuristic approach.

We extend the stochastic dynamic programming (SDP) algorithm presented in [16] to deal with cross-scenario constraints. This metaheuristic approach decomposes the stochastic problem into a collection of subproblems. To solve the subproblems efficiently, piecewise linear convex estimations for the impact of the decisions to be made at a given subproblem on the objective function value related to the future are constructed. The SDP in [16] was designed to solve risk neutral models, and thus in the absence of cross-scenario constraints, where the linking between subproblems is across one period, [16, 25]. The extension we propose in this paper deals with these two issues using a mechanism that distributes the SDC bounds (or linking variable levels) among the immediate successors of any given subproblem.

We have chosen Supply Chain Management (SCM) as our area of application, [28, 46, 47, 48, 50, 52, 53]. Risk neutral models for SCM abound in the literature, see e.g. [2, 5, 6, 7, 10, 42, 45, 55] for references in the last decade. There have only very recently been some attempts to develop solution methods for risk averse SCM models, [3, 34]. We consider a supply chain planning model, which involves multiple products, multiple periods and a network of players who form the supply chain (typically, markets, production plants, distribution centers and retailers). Raw materials are available at the top of the chain, customers are at the bottom and face demand on a set of end products. The Bill of Materials describes how the players can produce the end products. The goal of the supply chain planning model is to satisfy customer demand at the lowest total costs, where procurement, production, inventory and distribution decisions need to be made.

In the computational experience we benchmark our solution approach against the commercial MIP solver CPLEX in terms of solution accuracy and computation time. We use twelve instances of the SCM model sketched above. These are instances with both 0–1 and continuous variables, where demand is uncertain, and $S2$ sets of variables model nonlinear procurement cost functions, while the SDC risk modelling requires additional 0–1 variables. We illustrate the hardness of the instances, and therefore their adequacy as testbed, in terms of area of application and data generation. From the numerical results, we conclude that SDP is attractive when solving medium-to-large instances, while CPLEX cannot guarantee a feasible solution in half of the instances tested.

The remainder of the paper is organized as follows. In Section 2 we introduce the risk averse SDC formulation. In Section 3 we present the stochastic dynamic programming approach for SDC. We devote Section 4 to the computational experience. Finally, we conclude the paper and discuss future research directions in Section 5.

2 The risk averse SDC strategy

In this section we propose an SDC risk averse strategy for (MILP). This is the multistage extension of the mixture of the first-order SDC in [30] and the second-order SDC in [29] induced by mixed-integer linear recourse for two-stage programs. To avoid scenarios with high objective function values (hereafter, costs), the SDC strategy makes use of the so-called cost thresholds. For each cost threshold the aim is to have all scenario costs below the threshold. However, this may be too conservative and therefore one allows an excess on the cost threshold. In order to control this excess several mechanisms are used. First, a maximum is set on the size of the excess. Second, an upper bound is imposed on the chance that excess occurs. Third, an upper bound is imposed on the expected excess. Expected excess of scenario costs on the imposed cost thresholds has its roots in Integrated Chance Constraints, which were introduced in [32] and further investigated in [33].

In the following we introduce some notation to model the scenario tree, some of which is illustrated in Figure 1, and the SDC strategy. There is a one-to-one correspondence between nodes in the tree and the so-called scenario groups (denoted by g), and a one-to-one correspondence between the leaf nodes and the scenarios (denoted by w). We use the term immediate ancestor of a node g to refer to the only node in the previous period (level of the tree) connected to g, and use the term ancestor of g to refer to another node in an earlier period that is connected to g by a chain of ancestors. We also use the term successor, which is defined similarly.

Ω, set of scenarios.

w_ω, likelihood given by the modeler to scenario ω, with $\sum_{\omega\in\Omega} w_\omega = 1$.

$\mathcal{G}$, set of scenario groups.

Ω_g, set of scenarios in group g, for $g \in \mathcal{G}$, with $\Omega_g \subseteq \Omega$.

$\mathcal{G}_t$, set of scenario groups in period t, for $t \in \mathcal{T}$, with $\mathcal{G}_t \subseteq \mathcal{G}$.

$t(g)$, period to which scenario group g belongs to, for $g \in \mathcal{G}$, with $g \in \mathcal{G}_{t(g)}$.

$\sigma(g)$, immediate ancestor node of scenario group g, for $g \in \mathcal{G}$.

$\mathcal{A}_g$, set consisting of scenario group g and its ancestor nodes, for $g \in \mathcal{G}$.

$\mathcal{S}_g$, set consisting of immediate successor nodes to scenario group g, for $g \in \mathcal{G}$. Note that $\mathcal{S}_g = \emptyset$ for $g \in \mathcal{G}_T$.

w_g, likelihood associated with scenario group g, for $g \in \mathcal{G}$. Note that $w_g = \sum_{\omega \in \Omega_g} w_\omega$ and therefore $\sum_{g \in \mathcal{G}_t} w_g = 1$, for $t \in \mathcal{T}$.

$\mathcal{P}$, set of SDC profiles, where profiles are defined by the 4-tupla $(\phi^p, \beta^p, e^p, E^p)$, for $p \in \mathcal{P}$.

ϕ^p, cost threshold of the profile.

β^p, upper bound on failure probability on reaching cost threshold ϕ^p.

e^p, upper bound on expected excess on cost threshold ϕ^p.

E^p, maximum excess on cost threshold ϕ^p.

Let a_g, c_g, A_g^q, B_g^q and b_g be the counterparts of a_t, c_t, $A_{t'}^t$, $B_{t'}^t$ and b_t. The compact representation of the SDC strategy for (MILP) has the following DEM:

$$\text{minimize} \sum_{g \in \mathcal{G}} w_g (a_g x_g + c_g y_g) \tag{1}$$

subject to

$$\text{s.t.} \sum_{g \in \mathcal{A}_q} (A_g^q x_g + B_g^q y_g) = b_g \qquad \forall q \in \mathcal{G} \tag{2}$$

$$\sum_{g \in \mathcal{A}_\omega} (a_g x_g + c_g y_g) - v_\omega^p \leq \phi^p \qquad \forall \omega \in \Omega, p \in \mathcal{P} \tag{3}$$

$$v_\omega^p \leq E^p \nu_\omega^p \qquad \forall \omega \in \Omega, p \in \mathcal{P} \tag{4}$$

$$\sum_{\omega \in \Omega} w_\omega v_\omega^p \leq e^p \qquad \forall p \in \mathcal{P} \tag{5}$$

$$\sum_{\omega \in \Omega} w_\omega \nu_\omega^p \leq \beta^p \qquad \forall p \in \mathcal{P} \tag{6}$$

$$x_g \in \{0,1\}^{nx(g)}, y_g \in I\!R^{ny(g)} \qquad \forall g \in \mathcal{G} \tag{7}$$

$$v_\omega^p \in I\!R^+, \, \nu_\omega^p \in \{0,1\} \qquad \forall \omega \in \Omega, p \in \mathcal{P}, \tag{8}$$

where x_g and y_g are the counterparts of x_t and y_t, and variables v_ω^p and ν_ω^p are used to model the first-order and the second-order stochastic dominance constraints associated with the p-th profile. The objective function (1) and constraints (2) are the stochastic versions of those in (MILP). Constraints (3) and (4) define the correct range for the excess variable v_ω^p. In particular, it is easy to see that if $\nu_\omega^p = 0$, then $v_\omega^p = 0$, and thus the cost of scenario ω is below the threshold ϕ^p, while if $v_\omega^p > 0$, then $\nu_\omega^p = 1$. Constraints (4) impose the expected excess to be not greater than bound E^p. In addition, constraints (5) force the excess on threshold ϕ^p to be not greater than bound e^p. Similarly, constraints (6) force the fraction of scenarios with excess to be not greater than bound β^p. Constraints (7) and (8) define the domain of the variables. Throughout the rest of the paper we will refer to (1)-(8) as the SDC model.

3 The solution approach

3.1 Introduction

In this section we propose a decomposition approach to solve the SDC model. We extend the stochastic dynamic programming (SDP) metaheuristic approach, [36, 37, 39, 41], in [16]. In short, this SDP starts by combining consecutive time periods into stages, creating a collection of subtrees/subproblems, see Figure 1. The subproblems are obviously not independent but linked by decision variables in ancestor subproblems, we will refer to those as *linking* variables. The SDP solves the subproblems iteratively, making use of the so-called Expected Future Value (EFV) curves, which estimate the impact of the decisions to be made at a given stage on the objective function value related to the future stages. Each EFV curve is obtained using strong duality theory and Taylor's expansion at the so-called *reference* levels, and the result is a piecewise linear and convex function.

The SDP in [16] was designed to solve risk neutral models in production planning, where the linking is across one period. The extension we propose in this paper to deal with the SDC model is not trivial. Apart from its size and the possibility of linking across more periods, when solving the SDC model, one has another major challenge, namely cross-scenario constraints that link all scenarios, see constraints (5) and (6). To deal with these constraints, we device a mechanism that distributes the SDC bounds, namely e^p and β^p, among the immediate successors of any given subproblem.

The remainder of the section is organized as follows. In Section 3.2 we formulate the subproblems. In Section 3.3 we introduce the EFV curves. In Section 3.4 the SDP approach is outlined. In the following we introduce some notation to formulate the subproblems.

$\mathcal{E}$, set of stages in the time horizon.

$\mathcal{G}^e \subseteq \mathcal{G}$, set of scenario groups in stage e, for $e \in \mathcal{E}$.

$\mathcal{R}^e \subseteq \mathcal{G}^e$, set of scenario groups associated with the root nodes in the subtrees of stage e, for $e \in \mathcal{E}$.

$\mathcal{C}_r \subseteq \mathcal{G}^e$, set of nodes in $\mathcal{G}^e$ that belong to the subtree rooted at r, for $r \in \mathcal{R}^e, e \in \mathcal{E}$.

$\mathcal{L}_r \subseteq \mathcal{C}_r$, set of leaf nodes in $\mathcal{C}_r$, for $r \in \mathcal{R}^e$, $e \in \mathcal{E}$.

$\tilde{\mathcal{A}}_q \subseteq \mathcal{A}_q$, set of ancestor nodes to scenario group q in previous stages e', $\forall e' < e$, such that their variables have nonzero elements in the constraints associated with node q, for $q \in \mathcal{C}_r$, $r \in \mathcal{R}^e$, $e \in \mathcal{E}$. Note that $\tilde{\mathcal{A}}_q = \emptyset$ for $q \in \mathcal{C}_1$.

$\tilde{\tilde{\mathcal{A}}}_\ell$, set consisting of leaf node $\ell \in \mathcal{L}_r$ and its ancestor nodes in $\mathcal{A}_\ell$, such that their variables have nonzero elements in constraints associated with the nodes in the immediate successor subproblems to node ℓ, defined by $\bigcup_{r' \in \mathcal{S}_\ell} C_{r'}$, for $\ell \in \mathcal{L}_r, r \in \mathcal{R}^e$, $e \in \mathcal{E} \backslash \{|E|\}$.

3.2 The subproblems

In this section we formulate the SDC subproblem defined by node r, $r \in \mathcal{R}^e$, $e \in \mathcal{E}$. Let $\overline{x}_g$ and $\overline{y}_g$ be given values of vectors x_g and y_g, $g \in \tilde{\mathcal{A}}_q$, $q \in \mathcal{C}_r$. Similarly, let $\overline{e}_r^p$ and $\overline{\beta}_r^p$, $p \in \mathcal{P}$. Let $\lambda'_\ell(\cdot)$ denote the expected future objective function value in the set of scenarios Ω_ℓ, $\ell \in \mathcal{L}_r$. The SDC subproblem defined by node set $\mathcal{C}_r$ can be written as follows:

$$
\begin{aligned}
&F_r^{'}(\overline{x}_g, \overline{y}_g \forall g \in \tilde{\tilde{\mathcal{A}}}_{\sigma(r)};\, \overline{e}_r^p, \overline{\beta}_r^p \forall p \in \mathcal{P}) = w_r \sum_{g \in \mathcal{A}_{\sigma(r)}} (a_g x_g + c_g y_g) + \\
&\min \sum_{\ell \in \mathcal{L}_r} w_\ell [\sum_{g \in \mathcal{A}_\ell \setminus \mathcal{A}_{\sigma(r)}} (a_g x_g + c_g y_g) + \\
&\qquad \lambda'_\ell(x_q, y_q \forall q \in \tilde{\tilde{\mathcal{A}}}_\ell;\, e_{r'}^p, \beta_{r'}^p \forall p \in \mathcal{P}, r' \in \mathcal{S}_\ell)] + \\
&\sum_{p \in \mathcal{P}} (M_r^{pE} \epsilon_r^{pE} + M_r^{pe} \epsilon_r^{pe} + M_r^{p\beta} \epsilon_r^{p\beta})
\end{aligned} \tag{9}
$$

subject to:

$$\sum_{g \in \mathcal{A}_q} (A_g^q x_g + B_g^q y_g) = b_q \qquad \forall q \in \mathcal{C}_r \tag{10}$$

$$\sum_{g \in \mathcal{A}_\ell} (a_g x_g + c_g y_g) + \lambda'_\ell(\cdot) - v_\ell^p \le \phi^p \qquad \forall \ell \in \mathcal{L}_r,\, p \in \mathcal{P} \tag{11}$$

$$v_\ell^p \le E^p \nu_\ell^p + \epsilon_r^{pE} \qquad \forall \ell \in \mathcal{L}_r,\, p \in \mathcal{P} \tag{12}$$

$$\sum_{\ell \in \mathcal{L}_r} w_\ell v_\ell^p \le e_r^p + \epsilon_r^{pe} \qquad \forall p \in \mathcal{P} \tag{13}$$

$$\sum_{\ell \in \mathcal{L}_r} w_\ell \nu_\ell^p \le \beta_r^p + \epsilon_r^{p\beta} \qquad \forall p \in \mathcal{P} \tag{14}$$

$$\sum_{\ell \in \mathcal{L}_r} \sum_{r' \in \mathcal{S}_\ell} e_{r'}^p = e_r^p \qquad \forall p \in \mathcal{P} \tag{15}$$

$$\sum_{\ell \in \mathcal{L}_r} \sum_{r' \in \mathcal{S}_\ell} \beta_{r'}^p = \beta_r^p \qquad \forall p \in \mathcal{P} \tag{16}$$

$$x_g = \overline{x}_g,\, y_g = \overline{y}_g \qquad \forall g \in \tilde{\mathcal{A}}_q,\, q \in \mathcal{C}_r \tag{17}$$

$$e_r^p = \overline{e}_r^p,\, \beta_r^p = \overline{\beta}_r^p \qquad \forall p \in \mathcal{P} \tag{18}$$

$$x_g \in \{0,1\}^{nx(g)}, y_g \in I\!R^{+ny(g)} \qquad \forall g \in \mathcal{A}_\ell,\, \ell \in \mathcal{L}_r \tag{19}$$

$$e_r^p \in I\!R^+,\, \beta_r^p \in \{0,1\} \qquad \forall p \in \mathcal{P} \tag{20}$$

$$\epsilon_r^{pE} \in I\!R^+,\, \epsilon_r^{pe} \in I\!R^+,\, \epsilon_r^{p\beta} \in I\!R^+ \qquad \forall p \in \mathcal{P} \tag{21}$$

$$e_{r'}^p \in I\!R^+,\, \beta_{r'}^p \in \{0,1\} \qquad \forall r' \in \mathcal{S}_\ell,\, \ell \in \mathcal{L}_r,\, p \in \mathcal{P} \tag{22}$$

$$v_\ell^p \in I\!R^+,\, \nu_\ell^p \in \{0,1\} \qquad \forall \ell \in \mathcal{L}_r,\, p \in \mathcal{P}. \tag{23}$$

Before we explain the ingredients of this formulation, we highlight the main differences between this subtree formulation and the SDC model on the whole tree, i.e., model (1)-(8). First, model (9)-(23) has fixed the x, y-variables of ancestor nodes to the subtree, see constraints (17), as well as the SDC variables of the immediate ancestor node, see constraints (18). The right-hand-side values in these two constraints are obtained while solving the subproblems at previous stages. Second, new variables are introduced, the slack variables ϵ_r^{pE}, ϵ_r^{pe}, $\epsilon_r^{p\beta}$, $p \in \mathcal{P}$, to ensure that constraints (12)-(14) are always satisfied. The slack variables are penalized in the objective function using big M type parameters, M_r^{pE}, M_r^{pe} and $M_r^{p\beta}$, with the aim of having them equal to zero at the end of the procedure. Third, in order for the subproblem (9)-(23) to be an appropriate approximation, the expected objective function value of the successor subproblems is represented by the function $\lambda'_\ell(\cdot)$, $\ell \in \mathcal{L}_r$, and added to the objective function (9), and therefore to constraints (11). Note that no decisions are taken prior to the root node $r = 1$, while no estimation is required in the subproblems in the last stage $|\mathcal{E}|$ since the time horizon ends there. Therefore, any reference to $\mathcal{A}_1$ or $\lambda'_\ell(\cdot)$, $\ell \in \Omega$, will be dropped out from model (9)-(23).

The objective function (9) gives the expected cost along the time horizon for the set of scenarios in $\mathcal{L}_r$. It can be split into four terms, namely the constant term related to the ancestor subproblems in the earlier stages e' to stage e, $e' < e$, the second term is related to stage e, the third one is an approximation of the cost of all successor subproblems in future stages e', $e' > e$, and the last one is the penalization term discussed above. The big M parameters are monotonically increased up to the subproblems in the last stage.

Constraints (10) are the ones related to the scenario groups in the subproblem.

Constraints (11) define the average excess variable v_ℓ^p of the cost on threshold ϕ^p for the set of scenarios Ω_ℓ that belong to each leaf node $\ell \in \mathcal{L}_r$. So, it is an approximation, since the cost excess is an average for the set of scenarios in group ℓ, unless the subproblem belongs to the last stage $|\mathcal{E}|$ where Ω_ℓ is a singleton set for $\ell \in \mathcal{L}_r$. In this case, $\Omega = \bigcup_{\ell \in \mathcal{L}_r} \Omega_\ell$.

Constraints (12)-(14) are the counterparts of constraints (4)-(6), where as mentioned above we have added slack variables to make them always feasible.

Constraints (15) (resp. (16)) distribute the SDC variables e_r^p (resp. β_r^p) among the subproblems whose root nodes r' are the immediate successors of any leaf node ℓ, $\ell \in \mathcal{L}_r$. The distribution is at random for the first SDP iteration until obtaining a solution to the original problem in a front-to-back scheme. However, when the SDP iterations go on the function $\lambda'(\cdot)$ will correctly assign the e- and β-variables to the root nodes in the scenario subtrees.

Constraints (17)-(18) are the fixing ones.

Constraints (19)-(23) define the domain of the variables. Note that $\overline{e}_1^p$ (resp. $\overline{\beta}_1^p$), i.e., at the root node in the only subproblem for stage $e = 1$, is precisely the SDC bound e^p (resp. β^p) in the p-th profile.

Note that $\lambda'_\ell(\cdot)$ and $F'_{r'}(\cdot)$ are closely related. Indeed, for leaf node ℓ, $\ell \in \mathcal{L}_r$, $r \in$

$\mathcal{R}^e$, $e \in \mathcal{E}\backslash\{|\mathcal{E}|\}$, we can express $\lambda'_\ell(\cdot)$ as:

$$\lambda'_\ell(\cdot) = \sum_{r' \in \mathcal{S}_\ell} F'_{r'}(\cdot). \tag{24}$$

3.3 The EFV curves

In general, $\lambda'_\ell(\cdot)$ is difficult to compute and thus is $F'_r(\cdot)$. The SDP approach approximates $\lambda'_\ell(\cdot)$ by a piecewise linear convex function $\lambda_\ell(\cdot)$, which we will refer to as the EFV curve for leaf node ℓ. See Figure 2 for an example. Let $\mathcal{Z}_\ell$ denote the set of reference levels, where the z-th reference level is included by vector

$$(\overline{X}^z_\ell,\, \pi^z_g, \gamma^z_g\, \forall g \in \tilde{\tilde{\mathcal{A}}}_\ell,\, \delta^{p^z}_\ell, \tau^{p^z}_\ell \forall p \in \mathcal{P},\, \mu^z_\ell), \tag{25}$$

with

$$\overline{X}^z_\ell \equiv (\overline{x}^z_g, \overline{y}^z_g\, \forall g \in \tilde{\tilde{\mathcal{A}}}_\ell,\, \overline{e}^{p^z}_{r'}, \overline{\beta}^{p^z}_{r'}\, \forall r' \in \mathcal{S}_\ell,\, p \in \mathcal{P}). \tag{26}$$

The SDP approach approximates $\lambda'_\ell(\cdot)$ by

$$\lambda_\ell(\cdot) = \max_{z \in \mathcal{Z}_\ell} \Big\{ \mu^z_\ell + \sum_{g \in \tilde{\tilde{\mathcal{A}}}_\ell} (\pi^z_g x_g + \gamma^z_g y_g) + \sum_{p \in \mathcal{P}} \sum_{r' \in \mathcal{S}_\ell} (\delta^{p^z}_\ell e^p_{r'} + \tau^{p^z}_\ell \beta^p_{r'}) \Big\}. \tag{27}$$

Consider again $r \in \mathcal{R}^e$, $e \in \mathcal{E}\backslash\{|E|\}$, and suppose that we have EFV curves (27) for stage e. Subproblem (9)-(23), defining $F'_r(\cdot)$, can be approximated by

$$\begin{aligned} F_r(\overline{X}^z_{\sigma(r)}\,(26)\, \forall z \in \mathcal{Z}_{\sigma(r)}) = w_r \sum_{g \in \mathcal{A}_{\sigma(r)}} (a_g x_g + c_g y_g) + \\ \min \Big\{ \sum_{\ell \in \mathcal{L}_r} w_\ell \big[\sum_{g \in \mathcal{A}_\ell \backslash \mathcal{A}_{\sigma(r)}} (a_g x_g + c_g y_g) + \lambda_\ell \big] + \\ \sum_{p \in \mathcal{P}} (M^{pE}_r \epsilon^{pE}_r + M^{pe}_r \epsilon^{pe}_r + M^{p\beta}_r \epsilon^{p\beta}_r) \Big\} \end{aligned} \tag{28}$$

subject to

$$\begin{aligned} &\text{Constraints (10) and (12)-(23)} \\ &\sum_{g \in \mathcal{A}_\ell} (a_g x_g + c_g y_g) + \lambda_\ell - v^p_\ell \le \phi^p && \forall \ell \in \mathcal{L}_r,\, p \in \mathcal{P} \\ &\lambda_\ell \ge \mu^z_\ell + \sum_{g \in \tilde{\tilde{\mathcal{A}}}_\ell} (\pi^z_g x_g + \gamma^z_g y_g) + \sum_{p \in \mathcal{P}} \sum_{r' \in \mathcal{S}_\ell} (\delta^{p^z}_\ell e^p_{r'} + \tau^{p^z}_\ell \beta^p_{r'}) && \forall z \in \mathcal{Z}_\ell,\, \ell \in \mathcal{L}_r \quad (29) \\ &\lambda_\ell \in I\!R && \forall \ell \in \mathcal{L}_r. \quad (30) \end{aligned}$$

In the next section we outline the SDP approach and explain how the parameters in (29) are obtained. Two observations need to be made at this point. First, a refinement of the EFV curve in leaf node ℓ, and thus a better approximation of $\lambda'_\ell(\cdot)$, is obtained when the set $\mathcal{Z}_\ell$ is enlarged. Second, as in (24), the EFV curve can be written in terms of $F_{r'}(\cdot)$.

3.4 The SDP approach

Each iteration of the SDP algorithm consists of a *front-to-back* scheme, followed by a *back-to-front* scheme. A flowchart of the entire SDP decomposition algorithm can be found in Figure 3, including the initialization step. Below we briefly explain these two schemes, as well as the procedure to refine the EFV curves in each iteration.

The front-to-back scheme is aimed at building a solution for the problem, say, $\overline{X} = (\overline{X}_g, \forall g \in \mathcal{G})$ and, if it is feasible for the original SDC model (i.e., if $\epsilon_r^{pE} = \epsilon_r^{pe} = \epsilon_r^{p\beta} = 0$, $r \in \mathcal{R}^{|\mathcal{E}|}$), then checking whether $\overline{X}$ improves the incumbent solution, say X^*, in which case $\overline{X}$ becomes the incumbent one. Subproblems from stage 1 to stage $|\mathcal{E}|$ are solved by passing the obtained values of linking variables onto the subproblems in the next stage. (Note that in the first iteration of the SDP algorithm the λ-values are zero for the front-to-back scheme.)

The back-to-front scheme is aimed at refining the EFV curves around the solution $\overline{X}$ built in that iteration. Subproblems from stage $|\mathcal{E}|$ to stage 1 are solved, passing the refinement of the EFV curves onto the subproblems in the previous stage. The algorithm will stop if the relative change in the value $\sum_{g\in\mathcal{G}} w_g(a_g\overline{x}_g + c_g\overline{y}_g)$ between two consecutive front-to-back iterations is below a tolerance parameter, $\varepsilon > 0$, or an upper bound on the number of iterations, say `miter`, is reached.

The EFV curve for $\ell \in \mathcal{L}_{r'}$, $r' \in \mathcal{R}^e$, $e \in \mathcal{E}\backslash\{|\mathcal{E}|\}$, is refined in a back-to-front scheme by adding a new reference level, z', and therefore a new linear function to the collection in (29). We briefly work out the details on how $(\pi_g^{z'}, \gamma_g^{z'}\, \forall g \in \tilde{\tilde{\mathcal{A}}}_\ell, \delta_\ell^{p^{z'}}, \tau_\ell^{p^{z'}}\, \forall p \in \mathcal{P}, \mu_\ell^{z'})$ is obtained.

We first apply an ad-hoc sensitivity analysis of $F_r(\cdot)\, \forall r \in \mathcal{S}_\ell$ on a small perturbation of solution $\overline{X}_\ell^{z'}$ to model (28)-(30). Let $(\pi_{g,r}^{z'}, \gamma_{g,r}^{z'}\, \forall g \in \tilde{\tilde{\mathcal{A}}}_\ell, \delta_{\ell,r}^{p^{z'}}, \tau_{\ell,r}^{p^{z'}}\, \forall p \in \mathcal{P})$ be the dual vector of constraints (17)-(18).

We now use a similar property as (24) but for their corresponding approximations, i.e., the EFV curve in node ℓ, $\lambda_\ell(\cdot)$, and $F_r(\cdot)$. By strong duality and Taylor's expansion, we have

$$\mu_{\ell,r}^{z'} = F_r(\overline{X}_\ell^{z'}) - \sum_{g\in\tilde{\tilde{\mathcal{A}}}_\ell} (\pi_{g,r}^{z'}\overline{x}_g + \gamma_{g,r}^{z'}\overline{y}_g) - \sum_{p\in\mathcal{P}} (\delta_{\ell,r}^{p^{z'}}\overline{e}_r^p + \tau_{\ell,r}^{p^{z'}}\overline{\beta}_r^p) \tag{31}$$

$$\mu_\ell^{z'} = \sum_{r\in\mathcal{S}_\ell} w_r \mu_{\ell,r}^{z} \tag{32}$$

$$\pi_g^{z'} = \sum_{r\in\mathcal{S}_\ell} w_r \pi_{g,r}^{z'}, \quad \gamma_g^{z'} = \sum_{r\in\mathcal{S}_\ell} w_r \gamma_{g,r}^{z'} \quad \forall g \in \tilde{\tilde{\mathcal{A}}}_\ell \tag{33}$$

$$\delta_\ell^{p^{z'}} = \sum_{r\in\mathcal{S}_\ell} w_r \delta_{\ell,r}^{p^{z'}}, \quad \tau_\ell^{p^{z'}} = \sum_{r\in\mathcal{S}_\ell} w_r \tau_{\ell,r}^{p^{z'}} \quad \forall p \in \mathcal{P}. \tag{34}$$

4 Computational experience

4.1 Introduction

In this section we illustrate the performance of the SDP metaheuristic when solving the SDC model. We compare the solution accuracy and elapsed time of SDP to the commercial MIP solver CPLEX [12]. In order to illustrate the increase computational complexity of the SDC model, we show performance results for the risk neutral (RN) strategy using both solution approaches. Note that the RN model can be seen as the SDC model (1)-(8) with one single SDC profile with cost thresholds equal to ∞. In Section 4.2 we describe the test problem instances and we report the results in Section 4.3.

4.2 The test instances

In our tests we consider supply chain planning instances. These involve multiple products, multiple periods and piecewise linear concave functions. Raw materials are available at the top of the chain and are procured at the beginning of the planning horizon. Customers are at the bottom and face demand on a set of end products, though uncertain. The Bill of Materials describes the required components/sub-assemblies to produce the end products. Procurement, production, inventory and distribution decisions need to be made in order to satisfy customer demand, subject to balance and capacity constraints. For more details we refer the reader to [26]. These are problem instances with both 0–1 and continuous variables, as well as $S2$ sets of variables to model the piecewise linear concave shape of the procurement cost functions. Since procured raw materials can be used in different periods of the planning horizon, $S2$ variables have an impact on multiple subproblems. Our SDP uses a distribution mechanism to handle these variables, as we do for the cross-scenario constraints.

We have considered two pilot cases and six values of T, yielding twelve instances in total. For each instance, we use two SDC profiles $(\phi^p, \beta^p, e^p, E^p)$, $p = 1, 2$, where ϕ^1 (resp. ϕ^2) is defined by cutting off 5% from the mean (resp. maximum) scenario cost on the RN solution vector. For each value of the threshold, we find β^p by cutting off 5% from the fraction of scenarios, and similarly for the other SDC bound, namely e^p. Although these are plausible choices, the modeling of the SDC profiles is a very interesting topic, but beyond the scope of this paper.

Table 1 describes the scenario tree and gives the dimensions of both stochastic SCM formulations. The first column is the identifier for the instance, where the first six instances are generated using the first pilot case, and the rest using the second one. The following four columns focus on the scenario tree. We first report the predefined structure $A_1^{B_1} A_2^{B_2}$ of the scenario tree, where A_i denotes the number of children each node in stage i has and B_i denotes the number of periods in stage i. The next three columns give the number of periods T, the number of nodes $|\mathcal{G}|$ and the number of scenarios $|\Omega|$. The remaining columns report the dimensions of the models, where nc is the number of continuous variables, $n01$ is the number of 0–1 variables, m is the number of constraints and nel is the number of nonzero elements in the constraint matrix. Note that for a given value of T, the pilot 2 instance is larger than the corresponding pilot 1 one.

Except for the first two, these are medium-to-large instances. As far as the authors know, the risk averse SDC instances 3-6 and 9-12 are larger than the ones that have been tested in the literature. Even the risk neutral RN instances are, in general, of larger dimensions than the ones in the state-of-the-art.

4.3 The numerical results

In this section we present the performance results of the SDP metaheuristic and CPLEX. We use CPLEX v12.5 for solving the DEM models but also all MIP formulations arising in the SDP metaheuristic. Our experiments were conducted on a PC with a 2.5 GHz dual-core Intel Core i5 processor, 8 Gb of RAM and the operating system was OS X 10.9. In terms of stopping criteria, we set $\varepsilon = 0.01$, $\texttt{miter} = 45$ for pilot 1 instances, and $\texttt{miter} = 75$ for pilot 2 ones. For CPLEX, we impose a limit on the elapsed time of 28800 seconds.

The results for the RN model and the SDC model can be found in Tables 2 and 3, respectively. The first column refers again to the identifier of the instance. The following four columns reports the results for CPLEX, where $objval_{LP}$ is the solution value of the Linear Programming (LP) relaxation of the model, t_{LP} is the elapsed time in seconds to solve it, $objval_{IP}$ is the solution value returned by CPLEX, and t_{IP} is the elapsed time to obtain it. When CPLEX is not able to find a feasible solution, we denote this with '-'. The remaining columns report the results for the SDP approach. Column $objval_{SDP}$ gives the solution value of the solution returned by SDP and t_{SDP} is the total elapsed time to obtain it. The following three columns report other SDP metrics, namely $niter$ is the number of iterations performed, nz is the total number of reference levels generated, and $nprob$ the total number of MIP subproblems solved. Finally, the last column reports the deviation of the solution value obtained by SDP from the CPLEX value, $dev = (objval_{SDP} - objval_{IP})/objval_{IP}$ in (%), if CPLEX returns a feasible solution.

From Table 3, we can see that solving the SDC model (1)-(8) by CPLEX requires a high computational effort for half of instances, namely, instances 4-6 and 10-12. Note that these are instances with larger values of the time horizon T. There, CPLEX solves the LP relaxation in less than a minute (60 seconds), but it is not able to find even a feasible solution when the integrality constraints of the SDC model (1)-(8) are brought back, with a time limit of eight hours (28800 seconds). If we now turn to the CPLEX results in Table 2, we can confirm what is common knowledge, namely SDC strategies are computationally much harder than RN ones. In all instances, CPLEX is able to find a feasible solution to the RN model (though the time limit is reached in five of them). The four instances with the shortest time horizon are solved to optimality in less than 5 seconds. As we show below, our SDP approach is also challenged by the SDC instances, confirming the adequacy of the instances when benchmarking the SDP approach for the

SDC model, but remains attractive against CPLEX.

In all instances, SDP is able to find a feasible solution for the SDC model. This means that we also find a feasible solution for the six instances in which CPLEX is the most challenged, namely, instances 4-6 and 10-12. The time spent by SDP on those instances ranges between 1000 and 8000 seconds, which can be considered to be high, but recall that for those instances CPLEX stops after 28800 seconds with no feasible solution found. For the three smallest ones, instances 1-2 and 7, CPLEX gives a better solution in less time, which confirms that SDP is a metaheuristic decomposition approach that pays off for larger instances. For medium size instances, instances 3 and 8-9, the tradeoff between elapsed time and solution accuracy of SDP is preferred.

5 Conclusions

In this work we have presented a stochastic dynamic programming (SDP) metaheuristic based on a recursive refinement of the so-called EFV curves. This is an extension of the SDP decomposition approach in [16]. Our solution approach allows to solve risk averse models with first-order and second-order stochastic dominance constraints. Those strategies have been recently proposed in [29, 30] for dealing with the two-stage problem and their implementation requires cross-scenario constraints. In addition to the multistage environment, our stochastic dominance constraint strategy jointly considers bounds on the probability of failure and the expected shortfall over the scenarios on reaching the modeler-driven thresholds. The computational experience shows that SDP is attractive when solving medium-to-large instances, where plain use of MIP solvers cannot guarantee a feasible solution.

The main contribution of this paper is twofold. First, the new SDP algorithm allows the treatment of any type of cross-scenario constraint, not only those required by SDC strategies, which are a challenge to traditional decomposition algorithms. Our approach also allows us to consider variables that do not belong to any scenario model in particular, but have an impact on all subproblems in the tree, which are also a disturbance to decomposition algorithms. $S2$ variables used to define piecewise linear functions, such as procurement costs functions as the ones used in our SCM test instances, fit this framework. Second, the new SDP algorithm paves the way for considering other risk averse strategies, see [4], although the SDC strategy is much more computationally demanding, and it is one of the few attempting to develop solution methods for risk averse SCM models, see also [3, 34].

Notice that the decomposition of the original multiperiod problem into linked multistage subproblems allows to solve the subproblems in each stage independently. These tasks can be performed in parallel, saving a substantial amount of elapsed time. However, this is a non trivial task. In the future, we plan to apply inner and outer parallelization to those subproblems. This will allow us to solve even larger instances. We are also interested in parallel computation based on MPI (Message Passing Interface) cores to solve the subproblems. Notice that subproblems with different sets of successor nodes in successor stages to their own stage do not need to wait for their solving until the optimization of all the subproblems of successor stages but only their own successor subproblems. The additional elapsed time reduction can be used to further refine the Expected Future Value (EFV) curves, an essential ingredient in the SDP algorithm. The refinement can be performed by adding new (hopefully active) reference levels to the EFV curve of each leaf node of a given subproblem. Each new reference level can be obtained by solving the successor subproblems for each related value of the linking variables in the ancestor nodes that have been obtained at each previous iteration of the algorithm. All of those new subproblems can be themselves solved in parallel.

References

[1] S. Ahmed. Convexity and decomposition of mean-risk stochastic programs. *Mathematical Programming (Series B)*, 106(3):433–446, 2006.

[2] S. Ahmed, A.J. King and G. Parija. A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization*, 26:3–24, 2003.

[3] D. Alem and R. Morabito. Risk averse two-stage stochastic programs in furniture plants. *OR Spectrum*, 35(4):773–806, 2013.

[4] A. Alonso-Ayuso, F. Carvallo, L.F. Escudero, M. Guignard, J. Pi, R. Puranmalka, and A. Weintraub. Medium range optimization of copper extraction planning under uncertainty in future copper prices. *European Journal of Operational Research*, 233(3):711–726, 2014.

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

[6] A. Alonso-Ayuso, L.F. Escudero, A. Garín, M.T. Ortuño, and G. Pérez. On the product selection and plant dimensioning problem under uncertainty. *Omega*, 33(4):307–318, 2005.

[7] A. Alonso-Ayuso, L.F. Escudero, C. Pizarro, H.E. Romeijn, and D. Romero Morales. On solving the multi-period single-sourcing problem under uncertainty. *Computational Management Science*, 3(1):29–53, 2006.

[8] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent Measures of Risk. *Mathematical Finance*, 9(3):203–228, 1999.

[9] P. Beraldi, A. Violi, F. De Simone, M. Costabile, I. Massabò, and E. Russo. A multistage stochastic programming approach for capital budgeting problems under uncertainty. *IMA Journal of Management Mathematics*, 24(1):89–110, 2013.

[10] H.M. Bidhandi and R.M. Yusuff. Integrated supply chain planning under uncertainty using an improved stochastic approach. *Applied Mathematical Modelling*, 35(6):2618–2630, 2011.

[11] J.R. Birge. Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, 33:989-1007, 1985.

[12] IBM ILOG CPLEX. CPLEX v12.5. `http://www.ilog.com/products/cplex`, 2013.

[13] C.C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24(1):37–45, 1999.

[14] C.C. Carøe and J. Tind. L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming*, 83(1–3):451–464, 1998.

[15] M. Carrión, U. Gotzes, and R. Schultz. Risk aversion for an electricity retailer with second-order stochastic dominance constraints. *Computational Management Science*, 6(2):233–250, 2009.

[16] M.P. Cristóbal, L.F. Escudero, and J.F. Monge. On stochastic dynamic programming for solving large-scale planning problems under uncertainty. *Computers and Operations Research*, 36(8):2418–2428, 2009.

[17] R.S. Dembo. Scenario optimization. *Annals of Operations Research*, 30(1):63–80, 1991.

[18] D. Dentcheva and G. Martinez. Two-stage stochastic optimization problems with stochastic ordering constraints on the recourse. *European Journal of Operational Research*, 219:1–8, 2012.

[19] D. Dentcheva and A. Ruszczyński. Optimization with stochastic dominance constraints. *SIAM Journal on Optimization*, 14:548–566, 2003.

[20] D. Dentcheva and A. Ruszczyński. Robust stochastic dominance and its application to risk-averse optimization. *Mathematical Programming (Series B)*, 123:85–100, 2010.

[21] C.J. Donohue and J.R. Birge. The abridged nested decomposition method for multistage stochastic linear programs with relatively complete recourse. *Algorithmic Operations Research*, 1:20-30, 2006.

[22] L.F. Escudero. Robust portfolios for mortgage-backed securities. In S.A. Zenios, editor. *Quantitative Methods. AI and Supercomputers in Finance*. Unicom, London, 201–228, 1995.

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

[24] L.F. Escudero, A. Garín, M. Merino, and G. Pérez. On solving multistage stochastic mixed 0–1 problems with time consistent stochastic dominance constraints recourse-integer risk averse strategy. Technical Report. Department of Statistics and Operations Research, Universidad Rey Juan Carlos, Móstoles (Madrid), Spain, 2014.

[25] L.F. Escudero, J.F. Monge, D. Romero Morales, and J. Wang. Expected future value decomposition based bid price generation for large-scale network revenue management. *Transportation Science*, 47(2):181–197, 2013.

[26] L.F. Escudero, J.F. Monge, and D. Romero Morales. On supply chain planning under uncertainty. Working paper. Department of Statistics and Operations Research, Universidad Rey Juan Carlos, Móstoles (Madrid), Spain, 2014.

[27] A.A. Gaivoronski and G. Pflug. Value-at-risk in portfolio optimization: properties and computational approach. *Journal of Risk*, 7(2):1–31, 2005.

[28] J. Geunes, P.M. Pardalos, and H.E. Romeijn, editors. *Supply Chain Management: Models, Applications, and Research Directions*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[29] R. Gollmer, U. Gotzes, and R. Schultz. A note on second-order stochastic dominance constraints induced by mixed-integer linear recourse. *Mathematical Programming*, 126(1):179–190, 2011.

[30] R. Gollmer, F. Neise, and R. Schultz. Stochastic programs with first-order dominance constraints induced by mixed-integer linear recourse. *SIAM Journal on Optimization*, 19(2):552–571, 2008.

[31] V. Guigues and C. Sagastizábal. Risk-averse feasible policies for large-scale multistage stochastic linear programs. *Mathematical Programming*, 138(1–2):167–198, 2013.

[32] W.K. Klein Haneveld. *Duality in Stochastic Linear and Dynamic Programming*. Lecture Notes in Economics and Mathematical Systems, volume 274. Springer-Verlag, 1986.

[33] W.K. Klein Haneveld and M.H. van der Vlerk. Integrated chance constraints: reduced forms and an algorithm. *Computational Management Science*, 3(4):245–269, 2006.

[34] S. Nickel, F. Saldanha da Gama, and H.-P. Ziegler. A multi-stage stochastic supply network design problem with financial decisions and risk management. *Omega*, 40(5):511–524, 2012.

[35] W. Ogryczak and A. Ruszczyński. From stochastic dominance to mean-risk models: Semideviations as risk measures. *European Journal of Operational Research*, 116(1):33–50, 1999.

[36] M.V.F. Pereira and L.M.V.G. Pinto. *Stochastic optimization of a multireservoir hydroelectric system, a decomposition approach*. Water Resources Research, 21:779–792, 1985.

[37] M.V.F. Pereira and L.M.V.G. Pinto. *Multistage stochastic optimization applied to energy planning.* Mathematical Programming, 52:359–375, 1991.

[38] G.Ch. Pflug and W. Römisch. *Modeling, Measuring and Managing Risk.* World Scientific, 2007.

[39] W.B. Powell. Approximate Dynamic Programming: Solving the Curses of Dimensionality. Wiley, 2nd edition, 2011.

[40] R.T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2:21–42, 2000.

[41] S.M. Ross. Introduction to Stochastic Dynamic Programming, Academic Press, 1995.

[42] T. Santoso, S. Ahmed, M. Goetschalckx, and A. Shapiro. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167(1):96–115, 2005.

[43] R. Schultz and S. Tiedemann. Risk aversion via excess probabilities in stochastic programs with mixed-integer recourse. *SIAM Journal on Optimization*, 14(1):115–138, 2003.

[44] R. Schultz and S. Tiedemann. Conditional value-at-risk in stochastic programs with mixed-integer recourse. *Mathematical Programming (Series B)*, 105(2–3):365–386, 2006.

[45] P. Schütz, A. Tomasgard, and S. Ahmed. Supply chain design under uncertainty using sample average approximation and dual decomposition. *European Journal of Operational Research*, 199(2):409–419, 2009.

[46] J.F. Shapiro. *Modeling the Supply Chain.* Duxbury, 2001.

[47] D. Simchi-Levi, P. Kaminsky, and E. Simchi-Levi. *Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies.* McGraw-Hill/Irwin, Boston, Massachusetts, 3rd edition, 2008.

[48] D. Simchi-Levi, S.D. Wu, and Z.-J. Shen, editors. *Handbook of Quantitative Supply Chain Analysis: Modeling in the E-Business Era.* Kluwer Academic Publishers, Boston, 2004.

[49] R.M. van Slyke and R. Wets. L–shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17:638-663, 1969.

[50] L.V. Snyder and Z.J.M. Shen. *Fundamentals of Supply Chain Theory.* John Wiley & Sons, 2011.

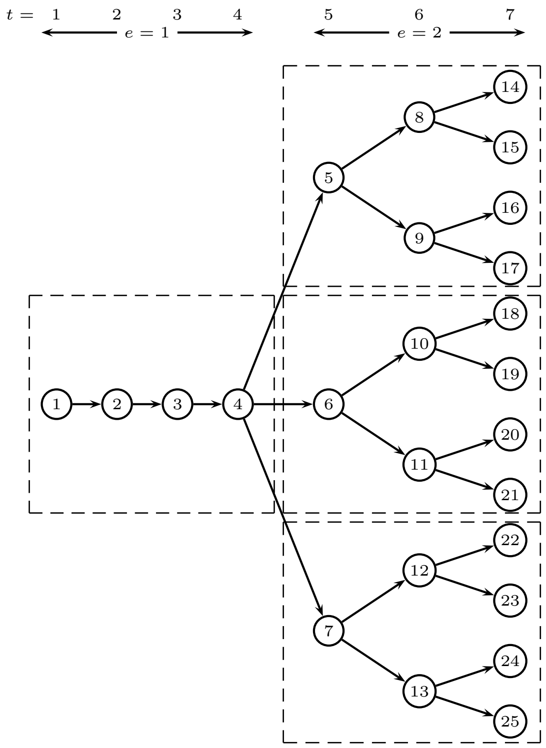
[51] S. Sriboonchita, W.-K. Wong, S. Dhompongsa, and H.T. Nguyen. *Stochastic Dominance and Applications to Finance, Risk and Economics.* CRC Press, 2010.

[52] H. Stadtler and C. Kilger, editors. *Supply Chain Management and Advanced Planning. Concepts, Models, Software and Case Studies.* Springer-Verlag, Berlin, 4th edition, 2008.

[53] S. Tayur, R. Ganeshan, and M. Magazine, editors. *Quantitative Models for Supply Chain Management.* Kluwer Academic Publishers, Boston, 2nd edition, 1999.

[54] R.J.-B. Wets. Stochastic programs with fixed recourse: The equivalent deterministic program. *SIAM Review*, 16:309-339, 1974.

[55] F. You and I.E. Grossmann. Multicut benders decomposition algorithm for process supply chain planning under uncertainty. *Annals of Operations Research*, 210(1):191–211, 2013.



$\mathcal{G}^2 = \{5, \ldots, 25\}$

$\mathcal{R}^2 = \{5, 6, 7\}$

$\mathcal{C}_5 = \{5, 8, 9, 14, \ldots, 17\}$

$\mathcal{L}_1 = \{4\}$

Figure 1: Breaking the time horizon into stages

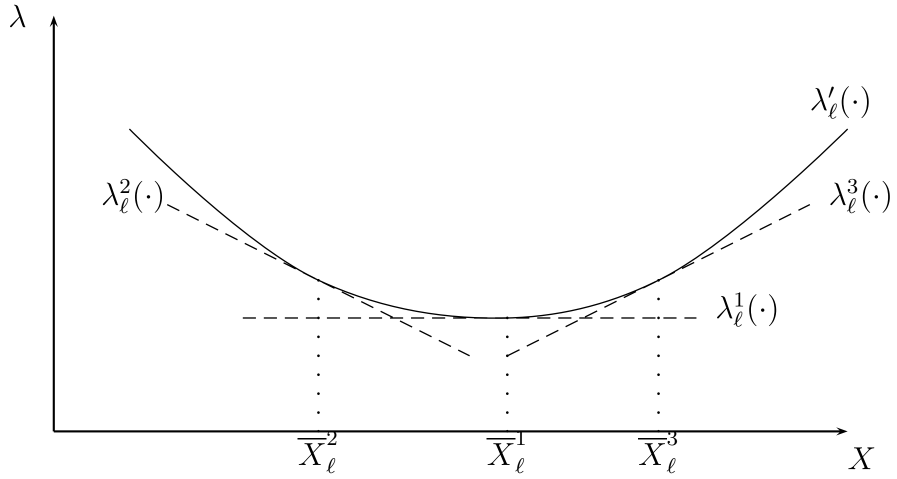


Figure 2: An example of EFV curve with three reference levels, $\lambda_\ell(\cdot) = \max_{z \in \mathcal{Z}_\ell} \lambda^z_\ell(\cdot)$, to approximate $\lambda'_\ell(\cdot)$

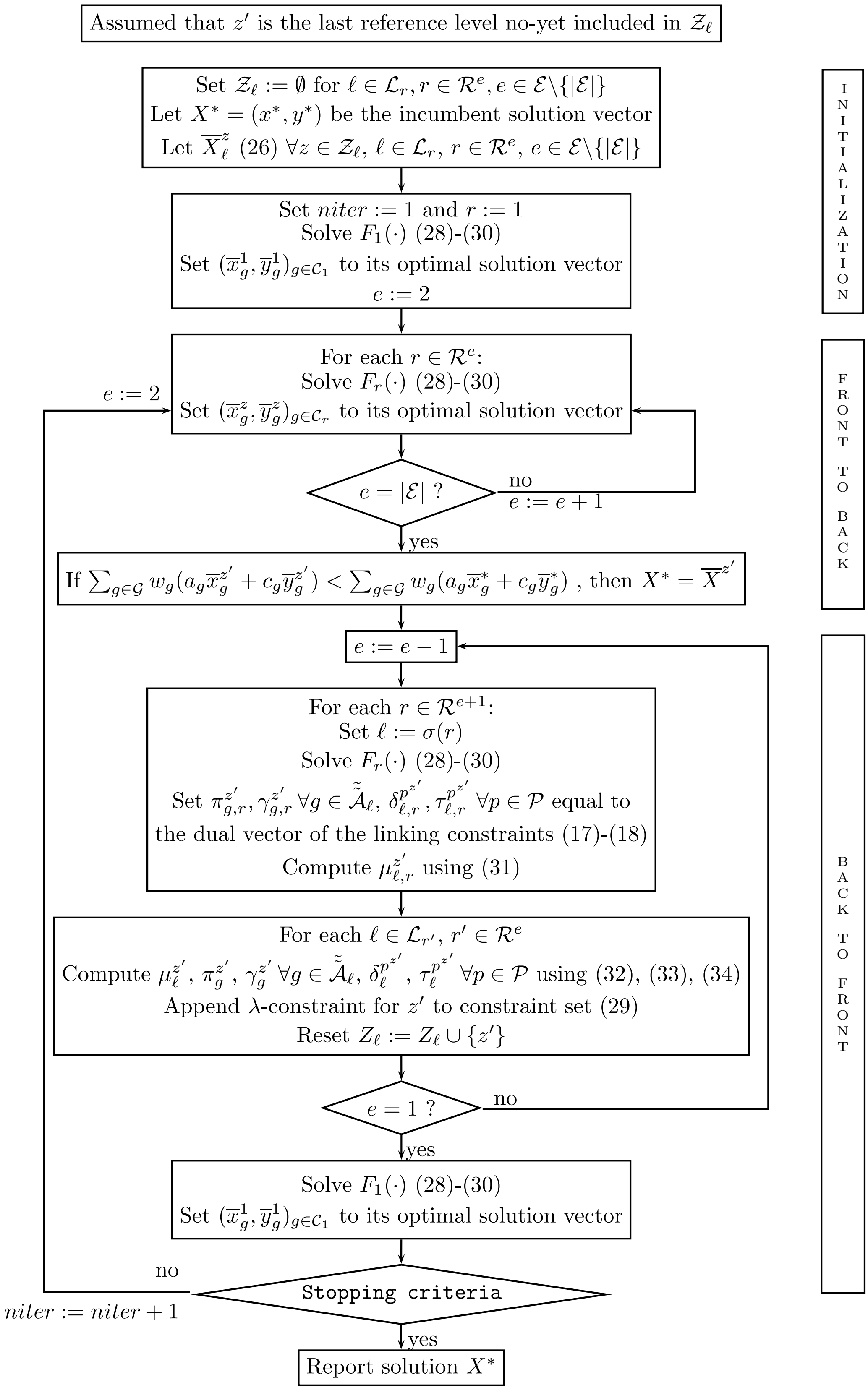


Figure 3: Flowchart of the SDP approach

Table 1: Dimensions of the problem instances

	Scenario tree				RN model				SDC model			
ID	$structure$	T	$\|\mathcal{G}\|$	$\|\Omega\|$	nc	$n01$	m	nel	nc	$n01$	m	nel
1	$2^3 2^2$	5	32	16	1165	320	1903	8526	1197	352	1971	15374
2	$2^3 2^3$	6	64	32	2285	640	3791	18222	2349	704	3923	34094
3	$2^4 2^3$	7	128	64	4525	1280	7567	38862	4653	1408	7827	74958
4	$2^4 2^4$	8	256	128	9005	2560	15119	82638	9261	2816	15635	163534
5	$2^4 2^5$	9	512	256	17965	5120	30223	175182	18477	5632	31251	354382
6	$2^4 2^6$	10	1024	512	35885	10240	60431	370254	36909	11264	62483	763470
7	$2^3 2^2$	5	32	16	4036	1120	6636	34261	4068	1152	6704	57141
8	$2^3 2^3$	6	64	32	7892	2240	13212	72517	7956	2304	13344	125637
9	$2^4 2^3$	7	128	64	15604	4480	26364	153349	15732	4608	26624	274309
10	$2^4 2^4$	8	256	128	31028	8960	52668	323653	31284	9216	53184	595013
11	$2^4 2^5$	9	512	256	61876	17920	105276	681541	62388	18432	106304	1283141
12	$2^4 2^6$	10	1024	512	123572	35840	210492	1431877	124596	36864	212544	2752837

Table 2: RN model solved with CPLEX and SDP metaheuristic

	CPLEX				SDP					
ID	$objval_{LP}$	t_{LP}	$objval_{IP}$	t_{IP}	$objval_{SDP}$	t_{SDP}	$niter$	nz	$nprob$	dev
1	431385	0.03	454185	1.24	456610	4.71	12	96	108	0.53
2	415313	0.03	438526	2.28	444247	16.36	14	112	126	1.30
3	433022	0.09	458531	61.89	460591	21.65	15	240	255	0.45
4	532109	0.73	562648	13798	570579	41.44	14	224	238	1.41
5	560343	1.04	592671	28800	615148	19.61	15	240	255	3.79
6	574435	3.20	608957	28800	637992	38.15	16	256	272	4.77
7	218432	0.08	220815	2.23	224460	18.42	23	184	207	1.65
8	227124	0.12	230346	4.24	231776	46.35	27	216	243	0.62
9	259253	0.24	262864	1130	268819	63.43	29	464	493	2.27
10	253955	0.56	258245	28800	262856	91.24	32	512	544	1.79
11	270881	1.10	275849	28800	283066	168.95	34	544	578	2.62
12	281047	1.57	286707	28800	301281	245.69	36	576	612	5.08

Table 3: SDC model solved with CPLEX and SDP metaheuristic

ID	CPLEX				SDP					
	$objval_{LP}$	t_{LP}	$objval_{IP}$	t_{IP}	$objval_{SDP}$	t_{SDP}	$niter$	nz	$nprob$	dev
1	622791	0.04	671825	3.16	695198	35.15	30	240	270	3.48
2	608621	0.13	662788	85.00	698155	125.65	39	312	351	5.34
3	587297	0.45	686769	17480	699255	610.52	32	512	544	1.82
4	742368	4.54	–	28800	879581	2558.73	45*	720	765	–
5	732797	10.61	–	28800	859626	4484.16	45*	720	765	–
6	832611	31.84	–	28800	840182	7923.85	45*	720	765	–
7	278973	0.14	300266	29.42	309891	229.00	62	496	558	3.21
8	271869	0.18	303235	28800	317273	375.91	75**	600	675	4.63
9	315585	0.81	344419	28800	356748	1113.43	75**	1200	1275	3.58
10	306016	15.08	–	28800	348162	1187.78	75**	1200	1275	–
11	324661	17.93	–	28800	369718	4848.33	74	1184	1258	–
12	335667	31.12	–	28800	391658	6540.56	74	1184	1258	–

*`miter` = 45

**`miter` = 75