OR Spectrum (2015) 37:559–582
DOI 10.1007/s00291-014-0379-2

REGULAR ARTICLE

Structure of risk-averse multistage stochastic programs

Jitka Dupačová · Václav Kozmík

Published online: 14 October 2014
© Springer-Verlag Berlin Heidelberg 2014

Abstract We deal with risk-averse multistage stochastic programs with coherent risk measures such as multiperiod extensions of conditional value at risk or polyhedral risk measures. Their basic properties are discussed and applied to scenario-based input data. Using the contamination technique we quantify the influence of changes in the structure of the scenario-based approximation to the optimal value of the problem. Stochastic dual dynamic programming algorithm is used to provide illustrative numerical comparisons for different choices of risk measures and changes of input data for a simple multistage risk-averse stock allocation problem with scenario trees based on log-normal distribution of the asset returns.

Keywords Multistage stochastic programming · Multiperiod risk measures · Stochastic dual dynamic programming · Contamination

Mathematics Subject Classification 90C15 · 90C31

1 Multistage stochastic program

In a common T stage stochastic program, one thinks of a stochastic data process $\xi = (\xi_1, \xi_2, \ldots, \xi_T)$ and a decision process $x = (x_1, \ldots, x_T)$. The components $\xi_2, \ldots, \xi_T$ of ξ and the decisions $x_2, \ldots, x_T$ are assumed to be random vectors, not necessarily of the same dimension, defined on some probability space $(\Omega, \mathscr{F}, \mathbf{P})$,

J. Dupačová (✉) · V. Kozmík
Department of Probability and Mathematical Statistics, Charles University in Prague,
Faculty of Mathematics and Physics, Prague, Czech Republic
e-mail: dupacova@karlin.mff.cuni.cz

V. Kozmík
e-mail: vaclav@kozmik.cz

while ξ_1 is deterministic and x_1 is a nonrandom vector-valued variable. The sequence of decisions and observations is

$$x_1, \xi_2, x_2(x_1, \xi_2), \ldots, x_T(x_{T-1}, \xi_2, \ldots, \xi_T). \tag{1}$$

The decision process is *nonanticipative* which means that decisions taken at any stage of the process neither depend on future *realizations* of stochastic data nor on future decisions, whereas the past information as well as the knowledge of the probability distribution of the data process are exploited. This can be expressed as follows: Let $\mathscr{F}_t \subseteq \mathscr{F}$ be the σ-field generated by the projection $\Pi_t \xi = \xi_{[t]} := (\xi_1, \ldots, \xi_t)$ of the stochastic data process ξ that includes data up to stage t, and $\mathscr{F}_1 = \{\emptyset, \Omega\}$ the trivial σ-field. The dependence of the tth stage decision x_t only on the available information means that x_t is $\mathscr{F}_t$-measurable. Similarly, we let $\Pi_t x = x_{[t]} := (x_1, \ldots, x_t)$ denote the sequence of decisions at stages $1, \ldots, t$, $\mathbf{P}$ the probability distribution of ξ, $\mathbf{P}_t$ the marginal probability distribution of ξ_t, and $\mathbf{P}_t\left[\cdot|\xi_{[t-1]}\right]$, $t = 2, , \ldots, T$, its conditional probability distribution.

The first-stage decisions consist of all decisions that have to be selected before further information is revealed, whereas the second-stage decisions are allowed to adapt to this information, etc. In each of the stages, the decisions are limited by constraints that may depend only on the previous decisions and observations. *Stages do not necessarily refer to time periods; rather, they correspond to steps in the decision process.* The outcome of the decision process (1) is $f(x, \xi)$ and one wants to find a nonanticipative feasible decision $x(\xi)$ or policy that minimizes the expectation $\mathbf{E_P}\left[f(x, \xi)\right]$ subject to prescribed constraints; it will be denoted by x^*. Nonanticipativity means that decision x_t at stage t depends only on the realization of $\xi_{[t]}$ and also on previous decisions.

An example is the *nested form* of the multistage stochastic linear program (MSLP) which resembles the backward recursion of stochastic dynamic programming with an additive overall cost function:

$$\min_{x_1 \in \mathsf{X}_1} c_1^\top x_1 + \mathbf{E_P}\left[\mathsf{Q}_2(x_1, \xi_2)\right] \text{ with } \mathsf{X}_1 := \{x_1 |\, A_1 x_1 = b_1,\ x_1 \geq 0\}, \tag{2}$$

and $\mathsf{Q}_t(x_{t-1}, \xi_{[t]})$, $t = 2, \ldots, T$, defined recursively as

$$\mathsf{Q}_t(x_{t-1}, \xi_{[t]}) = \min_{x_t} c_t(\xi_{[t-1]})^\top x_t + \mathbf{E}_{\mathbf{P}_{t+1}[\cdot|\xi_{[t]}]}\left[\mathsf{Q}_{t+1}(x_t, \xi_{[t+1]})\right] \tag{3}$$

subject to constraints $x_t \in \mathsf{X}_t(x_{t-1}, \xi_{[t]})$, e.g.,

$$A_t(\xi_{[t-1]})x_t = b_t(\xi_{[t-1]}) - B_t(\xi_{[t-1]})x_{t-1},\ x_t \geq 0 \text{ a.s.},$$

$\mathsf{Q}_{T+1}(\cdot)$ is explicitly given, e.g., $\mathsf{Q}_{T+1}(\cdot) \equiv 0$.

Matrices A_t are of a fixed (m_t, n_t) type and the remaining vectors and matrices are of consistent dimensions. For the first stage, known values of all elements of c_1, A_1, b_1 are assumed and the main decision variable is x_1 that corresponds to the first stage.

The first-stage problem (2) has the form of the expectation-type stochastic program with the set of feasible decisions *independent* of $\mathbf{P}$.

One can rewrite (2)–(3) briefly as

$$\min_{x_1} c_1^\top x_1 + \mathbf{E}\left[\min_{x_2} c_2(\xi_{[1]})^\top x_2 + \mathbf{E}\left[\cdots + \mathbf{E}\left[\min_{x_T} c_T(\xi_{[T-1]})^\top x_T\right]\right]\right] \tag{4}$$

with corresponding conditional expectations as in (3) and subject to linear constraints $x_t \in \mathsf{X}_t(x_{t-1}, \xi_{[t]})$, $t = 1, \ldots, T$, on decision variables. Constraints involving random elements hold almost surely and for simplicity we assume that *all infima are attained*, which is related to the *relatively complete recourse*, and that *all conditional expectations exist.* In the case of *stage-wise independence*, the conditional probability distributions boil down to marginal distributions $\mathbf{P}_t$ of ξ_t. Notice that the basic nonanticipativity condition is clearly spelled out.

In the *risk-neutral convex case*, one assumes e.g., that $f(x, \xi)$ is an inf-compact convex normal integrand whose finite expectation exists and that the set of feasible decisions $\mathsf{X}(\xi)$ is closed, convex-valued, nonanticipative and uniformly bounded mapping, i.e., the assumption of relatively complete recourse. The optimal decisions can be obtained by application of dynamic programming technique, for example, by telescoping the T stage problem into t stage ones as done in Rockafellar and Wets (1976). The following proposition follows from Theorem 1 of Rockafellar and Wets (1976) where it was formulated for extended real integrand $f(x, \xi)$. (Here, we assume explicitly formulated nonanticipative constraint mappings as well as existence of expectations and of optimal decisions.)

Theorem 1 *Consider the T stage stochastic program*

$$\textit{minimize } \mathbf{E}_{\mathbf{P}}\left[f(x, \xi)\right] \tag{5}$$

subject to constraints $x_t \in \mathsf{X}_t(x_{[t-1]}, \xi_{[t]})$, $t = 1, \ldots, T$.

Take $f_T(x_T, x_{[T-1]}, \xi_{[T]}) := f(x, \xi)$; for $t = 1, \ldots, T-1$ define the t-th stage integrands

$$f_t(x_t, x_{[t-1]}, \xi_{[t]}) = \mathbf{E}_{\mathbf{P}}\left[\min_{x_{[\tau]}}\{f_\tau(x_\tau, x_{[\tau-1]}, \xi_{[\tau]}) : \Pi_t \xi_{[\tau]} = \xi_{[t]}, \Pi_t x_{[\tau]} = x_{[t]}\}\right]$$

for $t \le \tau \le T$ and consider the t stage problems

$$\textit{minimize } \mathbf{E}_{\mathbf{P}}\left[f_t(x_t, x_{[t-1]}, \xi_{[t]})\right] \tag{6}$$

subject to constraints $x_t \in \mathsf{X}_t(x_{[t-1]}, \xi_{[t]})$. Then all programs (6) are solvable and the following property holds true:

If x^* is an optimal solution of (5), then its projection $\Pi_t x^*$ solves (6) and if $x^*_{[t]}$ solves (6), it can be extended to an optimal solution x^* of (5) such that $\Pi_t x^* = x^*_{[t]}$.

This theorem forms a bridge between multistage stochastic programs solved as a sequence of rolling horizon stochastic programs with a reduced number of

stages and the stochastic dynamic programming methodology. For convex multistage expectation-based stochastic programs, it can be evidently linked with the concept of *dynamic* or *time-consistency property* introduced later on, e.g.,

TC1 Carpentier et al. (2012) The sequence of dynamic optimization problems (6) is dynamically consistent if the optimal strategies obtained when solving the original problem remain optimal for all subsequent problems.

When the normal integrand $f(x, \xi)$ is separable with respect to stages, one can design an alternative dynamic programming recursion such as in (2)–(3) or in Pennanen and Perkkiö (2012) for convex multistage stochastic programs where result akin to Theorem 1 can be found.

In applications, one mostly approximates the true probability distribution $\mathbf{P}$ of ξ by a discrete probability distribution carried by a finite number of atoms (scenarios), say, $\xi^1, \ldots, \xi^K$. They are organized in a scenario tree and, in principle, the optimal policy can be obtained by solving a large deterministic program. See, e.g., the recent book Shapiro et al. (2009a) for details and more general cases. Every node of the tree is a root of a scenario subtree which does not contain any branches of other subtrees. Hence, the optimal solutions of a nodal subproblem do not reflect the future information carried by branches of the full tree that, from the point of view of the relevant nodal subproblem, cannot happen in the future. This observation is behind a modified time-consistency concept

TC2 Shapiro (2009b) At each state of the system, optimality of a decision policy should not involve states which cannot happen in the future.

Hence, under modest assumptions, optimal solutions of risk-neutral scenario-based multistage stochastic programs possess both of these time-consistency properties, whereas there are still open questions concerning time-consistency notions for risk-averse multistage stochastic programs.

With reference to the book Pflug and Römisch (2007) and to Chapter 6 of Shapiro et al. (2009a), Sect. 2 opens briefly the risk-averse extensions of multistage stochastic programming problems. In Sect. 3, the selected properties of multiperiod risk measures are presented. Various multistage stochastic linear programs with CVaR-type risk measures are given in Sect. 4. In Sect. 5, an adaptation of stochastic dual dynamic programming algorithm is proposed for their numerical solution. In this problem framework, the contamination technique is elaborated in Sect. 6 to deal with changes of the reference probability distribution $\mathbf{P}$. This is illustrated in Sect. 7 by numerical examples dealing with large-scale multistage portfolio optimization problems.

2 Risk-averse multistage stochastic programs

Maximization of expected gains or minimization of expected losses means to get decisions that are optimal on average and do not reflect possible risks. This need not be an acceptable goal. The present tendency is to spell out explicitly the concern for risk monitoring and control. There are various types of risk and the choice of a suitable risk definition depends on the context, the decision maker's attitude, etc.

To reflect risks in the stochastic programming formulation, it is necessary to quantify it. Both in theoretical considerations and in applications, sensible properties of risk measures are requested. A risk measure is a functional which assigns a real value to the random outcome $f(x, \xi)$. Similarly to the risk-neutral expected value criterion, risk measures ρ should not depend on individual realizations of ξ, but they depend on decisions and on probability distribution $\mathbf{P}$. Moreover, they should also reflect the filtration $\mathscr{F}_1 \subset \cdots \subset \mathscr{F}_t \cdots \subseteq \mathscr{F}$.

Coherence of ρ (monotonicity, translation invariance, positive homogeneity and subadditivity) cf. Artzner et al. (1999) is mostly expected. Risk measures value at risk (VaR), which is not coherent in general, and the coherent conditional value at risk (CVaR) are examples of ρ. Monotonicity with respect to the pointwise partial ordering and subadditivity are evident requirements. Convexity allows to keep a relatively friendly structure of the problem, both for computational and theoretical purposes, polyhedral property, cf. CVaR introduced in Rockafellar and Uryasev (2002) or polyhedral risk measures, cf. Eichhorn and Römisch (2005), to rely on linear programming techniques for scenario-based MSLP.

Whereas there exist many suggestions of risk measures for static stochastic programs, verified by numerical experiments and applications, see, e.g., Krokhmal et al. (2011) and references therein, for multistage problems the situation is much more involved. The first idea is to replace the expectation $\mathbf{E_P}\,[f(x, \xi)]$ by a suitable risk measure ρ and to keep all constraints including nonanticipativity. Assigning a risk measure ρ to the final outcome $f(x, \xi)$ does not take into account the information structure given by the filtration. It corresponds to monitoring risk only at the horizon which need not be sufficient. To include risk monitoring in individual stages, one may relate the risk measure to the partial outcomes, $f_1(x_1)$, $f_t(x_t, x_{[t-1]}, \xi_{[t]})$, $t = 2, \ldots, T$. Different risk measures ρ_t can be applied at individual stages. As a result, we may construct, e.g., objective function

$$\min_{x_1} c_1^\top x_1 + \rho_2 \left(\min_{x_2} c_2(\xi_{[1]})^\top x_2 + \rho_3 \left(\cdots + \rho_T \left(\min_{x_T} c_T(\xi_{[T-1]})^\top x_T \right) \right) \right) \tag{7}$$

and use it in place of (4). It is important to agree on acceptable properties of risk measures, e.g., multiperiod risk measures should be coherent. The full formulation of the risk-averse stochastic program (7) has to include the nonanticipativity constraints. Depending on the risk-averse problem and on the applied solution technique, a form of time consistency of optimal solutions is desirable.

Having in mind tractable numerical techniques such as stochastic dual dynamic programming (SDDP), cf. Pereira and Pinto (1991) applied to sample average approximation (SAA) of the underlying problem, we shall focus on finite discrete probability distributions and study mainly multiperiod extensions of conditional value at risk and multiperiod polyhedral risk measures. The next section comments on basic definitions.

3 Definitions

We will model the risk by representing the loss which could be incurred in stages $1, \ldots, T$ by random functions $\mathbf{Z} = (Z_1, \ldots, Z_T)$ that will be defined on a suitable

linear space $\mathscr{Z}$. The following definition, introduced in Artzner et al. (2007), extends the notion of *coherent* risk measures, introduced in Artzner et al. (1999) and widely accepted in static risk-averse optimization, to the multistage case.

Definition 1 (*Multiperiod risk measures*) A functional ρ on $\mathscr{Z} = \times_{t=1}^{T} \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P})$ with $p \in [1, \infty]$ is called a multiperiod coherent risk measure if it satisfies the following:

1. $Z_t \geq \tilde{Z}_t$ a.s., $t = 1, \ldots, T \Longrightarrow \rho(Z_1, \ldots, Z_T) \geq \rho\left(\tilde{Z}_1, \ldots, \tilde{Z}_T\right)$ (monotonicity);
2. For each $r \in \mathbf{R}$: $\rho(Z_1 + r, \ldots, Z_T + r) = \rho(Z_1, \ldots, Z_T) + r$ (translation invariance);
3. $\rho\left(\mu Z_1 + (1-\mu)\tilde{Z}_1, \ldots, \mu Z_T + (1-\mu)\tilde{Z}_T\right)$
$\leq \mu\rho(Z_1, \ldots, Z_T) + (1-\mu)\rho\left(\tilde{Z}_1, \ldots, \tilde{Z}_T\right)$ for $\mu \in [0, 1]$ (convexity);
4. For $\mu \geq 0$ we have $\rho(\mu Z_1, \ldots, \mu Z_T) = \mu\rho(Z_1, \ldots, Z_T)$ (positive homogeneity).

Two special classes of multiperiod risk measures have received a lot of attention, polyhedral risk measures and conditional risk mappings. Polyhedral risk measures are defined as the optimal value of a multistage stochastic program in the following way; see Eichhorn and Römisch (2005).

Definition 2 (*Multiperiod polyhedral risk measures*) A risk measure ρ on $\times_{t=1}^{T} \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P})$ with $p \in [1, \infty]$ is called multiperiod polyhedral if there are $k_t \in \mathbf{N}$, $c_t \in \mathbf{R}^{k_t}$, $t = 1, \ldots, T$, $w_{t,\tau} \in \mathbf{R}^{k_{t-\tau}}$, $t = 1, \ldots, T$, $\tau = 0, \ldots, t-1$, a polyhedral set $M_1 \subset R^{k_1}$, and polyhedral cones $M_t \subset R^{k_t}$, $t = 2, \ldots, T$, such that

$$\begin{aligned} \rho(\mathbf{Z}) = \inf\ & \mathbf{E}\left[\sum_{t=1}^{T} c_t^\top Y_t\right] \\ \text{s.t. } & Y_t \in \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P}) \quad \forall t \in \{1, \ldots, T\} \\ & Y_t \in M_t \text{ a.s.} \quad \forall t \in \{1, \ldots, T\} \\ & \sum_{\tau=0}^{t-1} w_{t,\tau}^\top Y_{t-\tau} = Z_t \text{ a.s.} \quad \forall t \in \{1, \ldots, T\}. \end{aligned} \tag{8}$$

When replacing the expectation of the total outcome of a multistage risk-neutral SLP by the multiperiod polyhedral risk measure, it is possible to carry out the minimization with respect to the original decision variable x and minimization in (8) simultaneously; see Proposition 4.1 in Eichhorn and Römisch (2005). Moreover, the scenario form of (8) and that of the combined problem is a linear program.

The class of conditional risk mappings resembles the conditional expectations in (4). It is especially convenient for the construction of nested risk measures to obtain the time-consistency property. Let $\mathscr{F} \subset \mathscr{F}'$ be σ-fields of subsets of Ω and $\mathscr{Z}$ and $\mathscr{Z}'$ be linear spaces of real-valued functions $f(\omega)$, $\omega \in \Omega$ measurable with respect to $\mathscr{F}$ and $\mathscr{F}'$, respectively. Following Ruszczyński and Shapiro (2006), we define:

Definition 3 (*Conditional risk mappings*) We say that mapping $\rho : \mathscr{Z}' \to \mathscr{Z}$ is conditional risk mapping if the following properties hold:

(1) Convexity. If $\alpha \in [0, 1]$ and $X, Y \in \mathscr{Z}'$, then

$$\alpha\rho(X) + (1-\alpha)\rho(Y) \succeq \rho(\alpha X + (1-\alpha)Y).$$

(2) Monotonicity. If $Y \succeq X$, then $\rho(Y) \succeq \rho(X)$.
(3) Predictable Translation Equivariance. If $Y \in \mathscr{Z}$ and $X \in \mathscr{Z}'$, then

$$\rho(X + Y) = \rho(X) + Y.$$

The inequalities in (1) and (2) are understood component-wise, i.e., $Y \succeq X$ means that $Y(\omega) \geq X(\omega)$ for every $\omega \in \Omega$.

For conditional risk mappings defined above, we shall use notation $\rho(\cdot|\mathscr{F})$. Notice, that Predictable Translation Equivariance is a natural generalization of translation invariance from Definition 1. Using it, we can construct *composite risk measures* as follows:

Consider conditional risk mappings $\rho_2, \ldots, \rho_T$ and a risk function $\rho : \mathscr{Z}_1 \times \cdots \times \mathscr{Z}_T \to \mathbf{R}$ given by:

$$\rho(Z_1, \ldots, Z_T) = Z_1 + \rho_2(Z_2 + \cdots \rho_{T-1}(Z_{T-1} + \rho_T(Z_T))).$$

Using Predictable Translation Equivariance, we get

$$\rho_{T-1}(Z_{T-1} + \rho_T(Z_T)) = \rho_{T-1} \circ \rho_T(Z_{T-1} + Z_T).$$

By continuing this process, we end up with a composite risk measure $\bar{\rho} := \rho_2 \circ \cdots \circ \rho_T$. It holds that

$$\bar{\rho}(Z_1 + \cdots + Z_T) = \rho(Z_1, \ldots, Z_T).$$

Using notation of Definition 3 we continue by introducing a concept of dynamic or time-consistent conditional risk mappings Kovacevic and Pflug (2009).

Definition 4 (*Time-consistent risk mappings*) A conditional risk mapping $(\rho_t(\cdot|\mathscr{F}_t))_{t=1,\ldots,T}$ is called time consistent if for all $1 \leq t_1 \leq t_2 \leq T$ and $X, Y \in \mathscr{L}_p(\Omega, \mathscr{F}, \mathbf{P})$:

$$\rho_{t_2}\left(X|\mathscr{F}_{t_2}\right) \leq \rho_{t_2}\left(Y|\mathscr{F}_{t_2}\right) \Longrightarrow \rho_{t_1}\left(X|\mathscr{F}_{t_1}\right) \leq \rho_{t_1}\left(Y|\mathscr{F}_{t_1}\right).$$

There exist various other related consistency concepts for risk measures, see e.g., Roorda and Schumacher (2007); when demanded, they may limit substantially the choice of acceptable risk measures up to the risk-neutral case, see e.g., Shapiro (2012b). In comparison to the time-consistency concepts [TC1] and [TC2], which relate to the

decisions, the Definition 4 defines time consistency for the risk measure itself. To evaluate the properties [TC1] and [TC2], we have to specify the subsequent optimization models for *every state of the system*. Without additional assumptions about the model structure, we cannot expect that time consistency of a risk measure automatically guarantees time consistency of the model.

For scenario-based programs, the time-consistency property [TC1] holds true whenever it is possible to reformulate the risk-averse multistage stochastic problem into the form of a classical risk-neutral stochastic program. This is provided by the Theorem 1. It should be noted that such reformulations usually require additional decision variables and are therefore harder to solve than corresponding risk-neutral versions of these models. Further discussion on the structure of risk-averse multistage stochastic programs can be found in articles Guigues (2014) and Guigues and Römisch (2012a, b).

4 Multistage stochastic programs with CVaR-type risk measures

We formulate a multistage stochastic linear program with CVaR risk measure in various versions. With the first version, nested CVaR, we follow the same manner as Shapiro (2011), largely using his notation. Secondly, we formulate a multistage stochastic program with a multiperiod CVaR risk measure and with a sum of CVaR risk measures following the notion of Pflug and Römisch (2007). All models have random parameters in stages $t = 2, \ldots, T$, e.g., $\xi_t = \big(c_t(\xi_{[t-1]}), A_t(\xi_{[t-1]}), B_t(\xi_{[t-1]}), b_t(\xi_{[t-1]})\big)$ in (3), which are governed by a known conditional distribution. All models can be also formulated in a more general convex form (which is solvable using the SDDP algorithm), but we have chosen the linear versions for easier presentation of our results. For simplicity of notation, we will drop the $(\xi_{[t-1]})$ arguments and denote the random parameters only by $\xi_t = (c_t, A_t, B_t, b_t)$. The parameters of the first stage, $\xi_1 = (c_1, A_1, b_1)$, are assumed to be known. Our models allow specification of different risk aversion coefficients λ_t and confidence levels $\alpha_t \in [0, 1]$ at each stage, $t = 1, \ldots, T$.

4.1 Nested CVaR model

The nested CVaR model is based on the following composite risk measure, cf. Pflug and Römisch (2007):

$$\rho^n(\mathbf{Z}) = \mathrm{CVaR}_\alpha\left[\cdot|\mathscr{F}_1\right] \circ \cdots \circ \mathrm{CVaR}_\alpha\left[\cdot|\mathscr{F}_{T-1}\right]\left(\sum_{t=1}^{T} Z_t\right).$$

According to Kovacevic and Pflug (2009), this risk measure is time consistent with respect to the Definition 4. To provide the nested formulation of the model, we introduce the following operator, which forms a weighted sum of conditional expectation and risk associated with random loss Z:

$$\rho_{t,\xi_{[t-1]}}[Z] = (1-\lambda_t)\,\mathbf{E}\left[Z\,\middle|\xi_{[t-1]}\right] + \lambda_t\,\mathrm{CVaR}_{\alpha_t}\left[Z\,\middle|\xi_{[t-1]}\right]. \tag{9}$$

We suppose $\lambda_t \in [0, 1]$, which with $\lambda_t = 0$ covers the risk-neutral problems, whereas $\lambda_t = 1$ puts emphasis on risk control only. The case of $\lambda_t = 0$ for $t < T$ and $\lambda_T \neq 0$ models the importance of risk only at the final stage.

We can write the corresponding risk-averse linear multistage model with T stages in the following form:

$$\min_{\substack{A_1x_1=b_1\\ x_1\geq 0}} c_1^\top x_1 + \rho_{2,\xi_{[1]}}\left[\min_{\substack{A_2x_2=b_2-B_2x_1\\ x_2\geq 0}} c_2^\top x_2 + \cdots + \rho_{T,\xi_{[T-1]}}\left[\min_{\substack{A_Tx_T=b_T-B_Tx_{T-1}\\ x_T\geq 0}} c_T^\top x_T\right]\right]. \tag{10}$$

We assume model (10) is feasible, has relatively complete recourse, and has a finite optimal value.

Our model, with the nested risk measure, allows a dynamic programming formulation to be developed, as described in Shapiro (2011). Using in (9) the definition of conditional value at risk from Rockafellar and Uryasev (2002),

$$\mathrm{CVaR}_\alpha[Z] = \min_u \left(u + \frac{1}{\alpha}\mathbf{E}[Z-u]_+\right), \tag{11}$$

where $[\cdot]_+ \equiv \max\{\cdot, 0\}$, we can rewrite (10) as

$$\begin{aligned} \min_{x_1,u_1}\ & c_1^\top x_1 + \lambda_2 u_1 + Q_2(x_1, u_1, \xi_{[1]}) \\ \text{s.t.}\ & A_1x_1 = b_1 \\ & x_1 \geq 0. \end{aligned} \tag{12}$$

The recourse value $\mathsf{Q}_t(x_{t-1}, \xi_{[t]})$ at stage $t = 2, \ldots, T$ is given by:

$$\begin{aligned} \mathsf{Q}_t(x_{t-1}, \xi_{[t]}) = \min_{x_t,u_t}\ & c_t^\top x_t + \lambda_{t+1}u_t + Q_{t+1}(x_t, u_t, \xi_{[t]}) \\ \text{s.t.}\ & A_tx_t = b_t - B_tx_{t-1} \\ & x_t \geq 0, \end{aligned} \tag{13}$$

where

$$\begin{aligned} & Q_{t+1}(x_t, u_t, \xi_{[t]}) \\ & = \mathbf{E}_{\mathbf{P}_{t+1}[\cdot|\xi_{[t]}]}\left[(1-\lambda_{t+1})\,\mathsf{Q}_{t+1}(x_t, \xi_{[t+1]}) + \frac{\lambda_{t+1}}{\alpha_{t+1}}\left[\mathsf{Q}_{t+1}(x_t, \xi_{[t+1]}) - u_t\right]_+\right]. \end{aligned} \tag{14}$$

We take $Q_{T+1}(\cdot) \equiv 0$ and $\lambda_{T+1} \equiv 0$ so that the objective function of model (13) reduces to $c_T^\top x_T$ when $t = T$; compare with (2)–(3).

The interpretation of the objective function is not straightforward, but it can be viewed as the real cost we would be willing to pay at the first stage instead of incurring the sequence of random costs $Z_1, \ldots, Z_T$; cf. Ruszczyński (2010).

The nested model is formulated in the framework of conditional risk mappings and this formulation is time consistent with respect to both [TC1] and [TC2], cf. Shapiro (2009b). However, due to its nesting structure, it cannot be represented as a polyhedral risk measure.

4.2 Multiperiod CVaR model

The multiperiod CVaR model is based on the following risk measure, see Pflug and Römisch (2007):

$$\rho^m(\mathbf{Z}) = \sum_{t=2}^{T} \mu_t \mathbf{E}\left[\mathrm{CVaR}_{\alpha_t}\left[Z_t|\mathscr{F}_{t-1}\right]\right]. \tag{15}$$

with $\sum_{t=2}^{T} \mu_t = 1,\ \mu_t \geq 0\,\forall t$. The multiperiod CVaR risk measure is time consistent with respect to the Definition 4, according to Theorem 3.3.11 of Kovacevic and Pflug (2009).

Using this risk measure and mean-risk operator (9), we obtain a multiperiod CVaR model:

$$\begin{array}{ll} \min\limits_{x_1,\ldots,x_T} & c_1^\top x_1 + \mu_2 \rho_{2,\xi_{[1]}}\left[c_2^\top x_2\right] + \cdots + \mu_T \mathbf{E}\left[\rho_{T,\xi_{[T-1]}}\left[c_T^\top x_T\right]\right] \\ \text{s.t.} & A_1 x_1 = b_1 \\ & A_2 x_2 = b_2 - B_2 x_1 \\ & \quad\vdots \\ & A_T x_T = b_T - B_T x_{T-1} \\ & x_t \geq 0,\ x_t \in \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P}), \quad t = 1, \ldots, T. \end{array} \tag{16}$$

We assume that model (16) is feasible, has a relatively complete recourse and a finite optimal value. While $\rho_{2,\xi_{[1]}}$ is deterministic, $\rho_{t,\xi_{[t-1]}}$, $t = 3, \ldots, T$ are random variables and expectation is applied to get a sensible model. The difference between this model and the nested CVaR model (10) is that here we apply expectation instead of the risk measure nesting. We also give a reformulation which uses the definition (11) of conditional value at risk:

$$\begin{array}{lll} \min\limits_{x_t,u_t,q_t \forall t} & c_1^\top x_1 + \sum\limits_{t=1}^{T-1} \mu_{t+1} \mathbf{E}\left[\lambda_{t+1} u_t\right] + \sum\limits_{t=2}^{T} \mu_t \mathbf{E}\left[(1-\lambda_t)\, c_t^\top x_t + \frac{1}{\alpha_t}\lambda_t q_t\right] & \\ \text{s.t.} & A_1 x_1 = b_1 & \\ & A_t x_t = b_t - B_t x_{t-1}, & t = 2, \ldots, T \\ & q_t \geq c_t^\top x_t - u_{t-1}, & t = 2, \ldots, T \\ & q_t \geq 0, & t = 2, \ldots, T \\ & u_t \in \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P}), & t = 1, \ldots, T-1 \\ & q_t \in \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P}), & t = 2, \ldots, T \\ & x_t \geq 0,\ x_t \in \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P}), & t = 1, \ldots, T. \end{array} \tag{17}$$

As shown in the book Pflug and Römisch (2007), the multiperiod risk measure (15) is polyhedral. Moreover, with reference to the dynamic programming equations, multi-

period CVaR model is time consistent with respect to [TC1] and [TC2]. Other concepts of time consistency with this risk measure are discussed in an example in Artzner et al. (2007) and also in Kovacevic and Pflug (2009).

Similarly as in the case with nested CVaR model, we develop dynamic programming equations. Using the interchangeability principle, we have:

$$\begin{aligned} \min_{x_1,u_1} \; & c_1^\top x_1 + \mu_2\lambda_2 u_1 + Q_2(x_1, u_1, \xi_{[1]}) \\ \text{s.t.} \; & A_1 x_1 = b_1 \\ & x_1 \geq 0 \end{aligned} \tag{18}$$

with the recourse value $\mathsf{Q}_t(x_{t-1}, u_{t-1}, \xi_{[t]})$ at stage $t = 2, \ldots, T$ given by:

$$\begin{aligned} \mathsf{Q}_t(x_{t-1}, u_{t-1}, \xi_{[t]}) = \min_{x_t,u_t,q_t} \; & \mu_t\,(1-\lambda_t)\,c_t^\top x_t + \mu_{t+1}\lambda_{t+1}u_t \\ & + \mu_t \tfrac{1}{\alpha_t}\lambda_t q_t + Q_{t+1}(x_t, u_t, \xi_{[t]}) \\ \text{s.t.} \; & A_t x_t = b_t - B_t x_{t-1} \\ & q_t \geq c_t^\top x_t - u_{t-1} \\ & q_t \geq 0 \\ & x_t \geq 0, \end{aligned} \tag{19}$$

where:

$$Q_{t+1}(x_t, u_t, \xi_{[t]}) = \mathbf{E}_{\mathbf{P}_{t+1}[\cdot|\xi_{[t]}]}\left[\mathsf{Q}_{t+1}(x_t, u_t, \xi_{[t+1]})\right]. \tag{20}$$

We take $Q_{T+1}(\cdot) \equiv 0$ and $\lambda_{T+1} \equiv 0$.

4.3 Sum of CVaR model

The weighted sum of CVaR model is based on the following risk measure, see Eichhorn and Römisch (2005):

$$\rho^s(\mathbf{Z}) = \sum_{t=2}^{T} \mu_t \,\mathrm{CVaR}_{\alpha_t}[Z_t]$$

with $\sum_{t=2}^{T} \mu_t = 1,\ \mu_t \geq 0\,\forall t$.

It can be shown that the sum of CVaR is not a time-consistent risk measure with respect to Definition 4; see Artzner et al. (2007) and Rudloff et al. (2010). The sum of CVaR model does not include nesting of the recourse values. It can be deduced from the scalarization technique of the multiobjective optimization. Using (9), it reads as

$$\begin{aligned} \min_{x_1,\ldots,x_T} \; & c_1^\top x_1 + \mu_2\rho_{2,\xi_{[1]}}\left[c_2^\top x_2\right] + \cdots + \mu_T\rho_{T,\xi_{[1]}}\left[c_T^\top x_T\right] \\ \text{s.t.} \; & A_1 x_1 = b_1 \\ & A_2 x_2 = b_2 - B_2 x_1 \\ & \vdots \\ & A_T x_T = b_T - B_T x_{T-1} \\ & x_t \geq 0,\ x_t \in \mathscr{L}_p(\Omega, \mathscr{F}_t, \mathbf{P}), \quad t = 1, \ldots, T. \end{aligned} \tag{21}$$

We assume again that model (21) is feasible, has relatively complete recourse and a finite optimal value. Please note that no nesting of the CVaR values is present and that we always condition the operator ρ with the first stage information $\xi_{[1]}$, i.e., $\rho_{t,\xi_{[1]}}$ is deterministic $\forall t = 2, \ldots, T$.

Using the mean-risk operator (9) and auxilliary variables q_t to express the nonlinear term in (11), we can rewrite the model as the following multistage stochastic linear program:

$$\begin{array}{lll} \min\limits_{x_t, u_t, q_t \forall t} & c_1^\top x_1 + \sum\limits_{t=1}^{T-1} \mu_{t+1}\lambda_{t+1} u_t + \sum\limits_{t=2}^{T} \mu_t \mathbf{E}\left[(1-\lambda_t)\, c_t^\top x_t + \frac{1}{\alpha_t}\lambda_t q_t\right] & \\ \text{s.t.} & A_1 x_1 = b_1 & \\ & A_t x_t = b_t - B_t x_{t-1}, & t = 2, \ldots, T \\ & q_t \geq c_t^\top x_t - u_{t-1}, & t = 2, \ldots, T \\ & q_t \geq 0, & t = 2, \ldots, T \\ & u_t \in \mathscr{L}_p\left(\Omega, \mathscr{F}_1, \mathbf{P}\right), & t = 1, \ldots, T-1 \\ & q_t \in \mathscr{L}_p\left(\Omega, \mathscr{F}_t, \mathbf{P}\right), & t = 2, \ldots, T \\ & x_t \geq 0,\ x_t \in \mathscr{L}_p\left(\Omega, \mathscr{F}_t, \mathbf{P}\right), & t = 1, \ldots, T. \end{array} \tag{22}$$

It can be seen that the risk measure $\rho^s(\mathbf{Z})$ used in this linear program satisfies the requirements of Definition 2 and is therefore polyhedral. Moreover, the corresponding optimization model is time consistent under the Definition [TC1]. On the other side, all variables u_t are decided in the first stage and the model is therefore not time consistent according to Definition [TC2].

We again develop dynamic programming equations using the interchangeability principle:

$$\begin{array}{ll} \min\limits_{x_1, u_1, \ldots u_{T-1}} & c_1^\top x_1 + \sum\limits_{t=1}^{T-1} \mu_{t+1}\lambda_{t+1} u_t + Q_2(x_1, u_1, \ldots, u_{T-1}, \xi_{[1]}) \\ \text{s.t.} & A_1 x_1 = b_1 \\ & x_1 \geq 0 \end{array} \tag{23}$$

with recourse value $\mathsf{Q}_t(x_{t-1}, u_{t-1}, \ldots, u_{T-1}, \xi_{[t]})$ at stage $t = 2, \ldots, T$, given by:

$$\begin{array}{l} \mathsf{Q}_t(x_{t-1}, u_{t-1}, \ldots, u_{T-1}, \xi_{[t]}) \\ \quad = \min\limits_{x_t, q_t} \mu_t\,(1-\lambda_t)\, c_t^\top x_t + \mu_t \dfrac{1}{\alpha_t}\lambda_t q_t + Q_{t+1}(x_t, u_t, \ldots, u_{T-1}, \xi_{[t]}) \\ \quad\quad \text{s.t. } A_t x_t = b_t - B_t x_{t-1} \\ \quad\quad\quad q_t \geq c_t^\top x_t - u_{t-1} \\ \quad\quad\quad q_t \geq 0 \\ \quad\quad\quad x_t \geq 0, \end{array} \tag{24}$$

where:

$$Q_{t+1}(x_t, u_t, \ldots, u_{T-1}, \xi_{[t]}) = \mathbf{E}_{\mathbf{P}_{t+1}[\cdot|\xi_{[t]}]}\left[\mathsf{Q}_{t+1}(x_t, u_t, \ldots, u_{T-1}, \xi_{[t+1]})\right]. \quad (25)$$

We take $Q_{T+1}(\cdot) \equiv 0$ and $\lambda_{T+1} \equiv 0$.

Other multiperiod polyhedral risk measures and their comparison can be found in Eichhorn and Römisch (2005). The final decision regarding which of the multiperiod risk measures to choose depends on the solved problem.

5 Stochastic dual dynamic programming

We use stochastic dual dynamic programming to solve, or rather approximately solve, models presented in the previous section. SDDP does not operate directly on these models. Instead, we first form a sample average approximation (SAA) of the model, and SDDP approximately solves the SAA. Thus in our context, SDDP forms estimators by sampling within an empirical scenario tree. In the remainder of this article, we restrict attention to solving SAA via SDDP. See Shapiro (2003) for a discussion of asymptotics of SAA for multistage problems, Philpott and Guan (2008) for convergence properties of SDDP, and Chiralaksanakul and Morton (2004) for procedures to assess the quality of an SDDP-based policy.

To apply the SDDP algorithm, we have to assume $\xi_t, t = 2, \ldots, T$, to be stage-wise independent. We further assume that for each stage $t = 2, \ldots, T$, there is a known (possibly continuous) distribution P_t of ξ_t and that we have a procedure to sample i.i.d. observations from this distribution. Using this procedure, we obtain empirical distributions $\hat{P}_t, t = 2, \ldots, T$. The scenarios generated by this procedure all have the same probabilities, but this is not required by the SDDP algorithm, which also applies to the case where the scenario probabilities differ.

We let $\hat{\Omega}_t$ denote the stage t sample space, where $|\hat{\Omega}_t| = N_t$. We use $j_t \in \hat{\Omega}_t$ to denote a stage t sample point, which we call a stage t scenario. We define the mapping $a(j_t) : \hat{\Omega}_t \to \hat{\Omega}_{t-1}$, which specifies the unique stage $t-1$ ancestor for the stage t scenario j_t. Similarly, we use $\Delta(j_t) : \hat{\Omega}_t \to 2^{\hat{\Omega}_{t+1}}$ to denote the set of descendant nodes for j_t, where $|\Delta(j_t)| = D_{t+1}$. The empirical scenario tree therefore has stage t realizations denoted by $\xi_t^{j_t}$, $j_t \in \hat{\Omega}_t$. At the last stage, we have $\xi_T^{j_T}$, $j_T \in \hat{\Omega}_T$, and each stage T scenario corresponds to a full path of observations through each stage of the scenario tree. That is, given j_T, we recursively have $j_{t-1} = a(j_t)$ for $t = T, T-1, \ldots, 2$. For this reason and for notational simplicity, when possible, we suppress the stage T subscript and denote $j_T \in \hat{\Omega}_T$ by $j \in \hat{\Omega}$.

We emphasize using the same set of D_t observations at stage t to form the descendant nodes of all N_{t-1} scenarios at stage $t-1$. This ensures the resulting empirical scenario tree is stage-wise independent. The SDDP algorithm does not apply, for example, to a scenario tree in which we instead use a separate, independent set of i.i.d. observations $\xi_t^1, \ldots, \xi_t^{D_t}$ for each of the stage $t-1$ scenarios, because the resulting empirical scenario tree would not be stage-wise independent. Note that fully general forms of dependency lead to inherent computational intractability as even the memory requirements to store a general sampled scenario tree grow exponentially in the number

of stages. Tractable dependency structures are typically rooted in some form of the independent increments between stages; e.g., autoregressive models, moving-average models, and dynamic linear models.

Under the stage independence assumption, the expected future cost function simplifies in the following way. In the nested CVaR model, function $Q_{t+1}(x_t, u_t, \xi_{[t]})$ from Eq. (13) now takes the form $Q_{t+1}(x_t, u_t)$. The dynamic programming equations are given by:

$$\begin{aligned} \min_{x_1, u_1} \; & c_1^\top x_1 + \lambda_2 u_1 + Q_2(x_1, u_1) \\ \text{s.t.} \quad & A_1 x_1 = b_1 \\ & x_1 \geq 0 \end{aligned} \tag{26}$$

with the recourse value $\mathsf{Q}_t(x_{t-1}, \xi_t)$ at stage $t = 2, \ldots T$ given by:

$$\begin{aligned} \mathsf{Q}_t(x_{t-1}, \xi_t) = \min_{x_t, u_t} \; & c_t^\top x_t + \lambda_{t+1} u_t + Q_{t+1}(x_t, u_t) \\ \text{s.t.} \quad & A_t x_t = b_t - B_t x_{t-1} \\ & x_t \geq 0, \end{aligned} \tag{27}$$

where

$$Q_{t+1}(x_t, u_t) = \mathbf{E}\left[(1 - \lambda_{t+1})\, \mathsf{Q}_{t+1}(x_t, \xi_{t+1}) + \frac{\lambda_{t+1}}{\alpha_{t+1}} \left[\mathsf{Q}_{t+1}(x_t, \xi_{t+1}) - u_t\right]_+\right]. \tag{28}$$

Similar development applies to the function $Q_{t+1}(x_t, u_t, \xi_{[t]})$ from Eq. (19) and the function $Q_{t+1}(x_t, u_t, \ldots, u_{T-1}, \xi_{[t]})$ from Eq. (24) which will be written as $Q_{t+1}(x_t, u_t, \ldots, u_{T-1})$ in the stage-independent case.

We give a brief description of the SDDP algorithm to give sufficient context for presenting our results. For further related details on SDDP, see Kozmík and Morton (2014), Pereira and Pinto (1991) and Shapiro (2011). SDDP applies to the dynamic programming equations developed in Sect. 4. During a typical iteration of the SDDP algorithm, cuts have been accumulated at each stage. These represent a piecewise linear outer approximation of the expected future cost function, $Q_{t+1}(x_t, u_t)$ or $Q_{t+1}(x_t, u_t, \ldots, u_{T-1})$. On a forward pass we sample a number of linear paths through the tree. As we solve a sequence of master programs (which we specify below) along these forward paths, the cuts that have been accumulated so far are used to form decisions at each stage. Solutions found along a forward path in this way form a policy, which does not anticipate the future. In fact, the solutions can be found at a node on a sample path via the stage t master program, even before we sample the random parameters at stage $t + 1$. The sample mean of the costs incurred along all the forward sampled paths through the tree forms an estimator of the expected cost of the current policy, which is determined by the master programs.

In the backward pass of the algorithm, we add cuts to the collection defining the current approximation of the expected future cost function at each stage. We do this by solving subproblems at the descendant nodes of each node in the linear paths from

the forward pass, except in the final stage, T. The cuts collected at any node in stage t apply to all the nodes in that stage, and hence we maintain a *single set of cuts* for each stage. We let C_t denote the number of cuts accumulated so far in stage t. This reduction is possible because of our stage-wise independence assumption.

The following model (29) is based on a nested formulation (13) and acts as a master program for its stage $t+1$ descendant scenarios and as a subproblem for its stage $t-1$ ancestor. We will refer to it as sub(j_t) for scenario j_t.

$$\hat{\mathsf{Q}}_t = \min_{x_t, u_t, \theta_t} c_t^\top x_t + \lambda_{t+1} u_t + \theta_t \tag{29a}$$

$$\text{s.t. } A_t x_t = b_t - B_t x_{t-1} \quad : \boldsymbol{\pi}_t \tag{29b}$$

$$\theta_t \geq \hat{Q}_{t+1}^j + \left(\mathbf{g}_{t+1}^j\right)^\top \left[(x_t,\, u_t) - \left(x_t^j,\, u_t^j\right)\right], \quad j = 1, \ldots, C_t \tag{29c}$$

$$x_t \geq 0. \tag{29d}$$

Decision variable θ_t in the objective function (29a), coupled with cut constraints in (29c), forms the outer linearization of the recourse function $Q_{t+1}(x_t, u_t)$ from model (27) and equation (28). This outer linearization is represented by the average values of the recourse function, $\hat{Q}_{t+1}^j$, and its subgradients $\mathbf{g}_{t+1}^j$. The structural and nonnegativity constraints in (29b) and (29d) simply repeat the same constraints from model (27). In the final stage T, we omit the cut constraints and the θ_T term.

As we indicate in constraint (29b), we use $\boldsymbol{\pi}_t$ to denote the dual vector associated with the structural constraints. As detailed in the articles Kozmík and Morton (2014) and Shapiro (2011), this dual vector is used to develop the cuts in the backward pass of the SDDP algorithm. For simplicity in stating the SDDP algorithm below, we assume we have known lower bounds L_t on the recourse functions. The presentation applies to the nested CVaR model (27), and the remaining models are solved in a similar fashion.

Algorithm 1 Stochastic dual dynamic programming algorithm

1. Let iteration $k = 1$ and append lower bounding cuts $\theta_t \geq L_t$, $t = 1, \ldots, T-1$.
2. Solve the stage 1 master program ($t = 1$) and obtain x_1^k, u_1^k, θ_1^k.
   Let $\underline{z}_k = c_1^\top x_1^k + \lambda_2 u_1^k + \theta_1^k$.
3. Forward pass: sample i.i.d. paths from $\hat{\Omega}$ and index them by S^k.

   For all $j \in S^k$ {
   for $t = 2, \ldots, T$ {
   form and solve sub(j_t) to obtain $\left(x_t^{j_t}\right)^k$ and $\left(u_t^{j_t}\right)^k$;
   }
   }

   Form the upper bound estimator $\overline{z}_k$ based on equation (31) in Kozmík and Morton (2014).
4. If a stopping criterion, given $\overline{z}_k$ and $\underline{z}_k$, is satisfied. then stop and output first stage solution $x_1 = x_1^k$ and lower bound $\underline{z} = \underline{z}_k$, otherwise continue to step 5.

5. Backward pass:
   For $t = T-1, \ldots, 1$ {
   for all $j \in S^k$ {
   for all descendant nodes $j_{t+1} \in \Delta(j_t)$ {
   form and solve sub(j_{t+1}) to obtain $\hat{\mathsf{Q}}_{t+1}^{j_{t+1}}$ and $\boldsymbol{\pi}_{t+1}^{j_{t+1}}$;
   calculate subgradient
   (see Kozmík and Morton (2014), Shapiro (2011));
   }
   obtain $\hat{Q}_{t+1}$ and $\mathbf{g}_{t+1}$ by averaging the optimal values and subgradients;
   append the resulting cut to the collection (29c) for stage t
   (see Kozmík and Morton (2014), Shapiro (2011)) ;
   }
   }
6. Let $k = k+1$ and go to step 2 with extended sets of cuts.

See Bayraksan and Morton (2011) for stopping rules that can be employed in step 4 and Philpott et al. (2013) for an alternative upper bound evaluation procedure.

6 Contamination for multistage risk-averse problems

The contamination technique for stochastic programs was developed in a series of papers as one of the tools for analysis of robustness of the optimal value with respect to deviations from the assumed probability distribution $\mathbf{P}$ and/or its parameters. The results were applied mainly to scenario-based two-stage stochastic linear programs; see, e.g., Dupačová (1990, 1996) for static and two- stage stochastic programs with risk-averse objective functions, Dupačová and Polívka (2007) for CVaR and VaR risk criteria and Dupačová (2008) for polyhedral risk measures.

The first ideas dealing with contamination for multistage stochastic linear programs were presented in Dupačová (1995) and their application to study the influence of changes in the structure of multistage problems with polyhedral risk measures can be found in Dupačová et al. (2009).

For construction of global contamination bounds, it is important that the stochastic program gets reformulated as

$$\min_{x \in \mathsf{X}} F(x, \mathbf{P}) := \min_{x \in \mathsf{X}} \int_{\Omega} f(x, \xi) \mathbf{P}(\mathrm{d}\xi) \tag{30}$$

with X independent of $\mathbf{P}$, such as in (12). Notice that the reformulations of the three considered models with CVaR-type risk measures comply with this requirement.

Possible changes in probability distribution $\mathbf{P}$ are modeled using contaminated distributions $\mathbf{P}_k$,

$$\mathbf{P}_k := (1-k)\mathbf{P} + k\mathbf{Q}, \; k \in [0, 1],$$

with $\mathbf{Q}$ as another *fixed* probability distribution. Via contamination, robustness analysis with respect to changes in $\mathbf{P}$ gets reduced to a much simpler analysis with respect to a scalar parameter k.

Assume that (30) was solved for a probability distribution $\mathbf{P}$ and denote $\varphi(\mathbf{P})$ the optimal value and $\mathsf{X}^*(\mathbf{P})$ the set of optimal (first stage) solutions.

The objective function in (30) is linear in $\mathbf{P}$; hence,

$$F(x,k) := \int_\Omega f(x,\xi)\mathbf{P}_k(\mathrm{d}\xi) = (1-k)F(x,\mathbf{P}) + kF(x,\mathbf{Q})$$

is linear in k. Suppose that the stochastic program (30) has an optimal solution for all considered distributions $\mathbf{P}_k,\ 0 \le k \le 1$. Then the optimal value function

$$\varphi(k) := \min_{x\in\mathsf{X}} F(x,k)$$

is concave on $[0,1]$ which implies its continuity and existence of directional derivatives in $(0,1)$. Continuity at the point $k=0$ is a property related with the stability results for the stochastic program in question. In general, one needs a nonempty, bounded set of optimal solutions $\mathsf{X}^*(\mathbf{P})$ of the initial stochastic program (30). This assumption together with stationarity of derivatives $\frac{\partial F(x,k)}{\partial k} = F(x,\mathbf{Q}) - F(x,\mathbf{P})$ is used to derive the form of the directional derivative:

$$\varphi'(0^+) = \min_{x\in\mathsf{X}^*(\mathbf{P})} F(x,\mathbf{Q}) - \varphi(0), \tag{31}$$

which enters the upper bound for the concave optimal value function $\varphi(k)$

$$\varphi(0) + k\varphi'(0^+) \ge \varphi(k) \ge (1-k)\varphi(0) + k\varphi(1),\ k\in[0,1]; \tag{32}$$

see Dupačová (1990, 1996) and Dupačová and Polívka (2007) and references therein. Contamination bounds (32) can be relaxed to

$$(1-k)\varphi(\mathbf{P}) + kF(x,\mathbf{Q}) \ge \varphi(\mathbf{P}_k) \ge (1-k)\varphi(\mathbf{P}) + k\varphi(\mathbf{Q}) \tag{33}$$

valid for an arbitrary $x \in \mathsf{X}^*(\mathbf{P})$ and $k\in[0,1]$.

The development so far applies to general distributions $\mathbf{P}$ and $\mathbf{Q}$. To be able to solve the problem in practice, we usually form a sample average approximation version of the problem and obtain a finite discrete distribution $\hat{\mathbf{P}}$. To make the contaminated problem solvable, the same property is required for the distribution $\hat{\mathbf{Q}}$. To apply the SDDP algorithm, we have to assume that stage independence holds true for both the original distribution $\hat{\mathbf{P}}$ and the contaminating distribution $\hat{\mathbf{Q}}$. That way, the contaminated distribution $\hat{\mathbf{P}}_k$ shares the same property.

When the problem is too large to be solved precisely, we obtain only approximate suboptimal solutions. In that case, an SDDP algorithm provides a lower bound $\underline{\varphi}$ on the optimal objective function value; see Step 2 of the Algorithm 1. Since we always have $\underline{\varphi} < \varphi$, the contamination lower bound follows easily:

$$\varphi(\hat{\mathbf{P}}_k) \geq (1-k)\underline{\varphi}(\hat{\mathbf{P}}) + k\underline{\varphi}(\hat{\mathbf{Q}}). \tag{34}$$

With the approximate (suboptimal) solution $\tilde{x}^*$ of the problem with original distribution $\hat{\mathbf{P}}$, we proceed in the following fashion. Since $\tilde{x}^*$ is feasible, but in general suboptimal for the contaminated problem with $\hat{\mathbf{P}}_k$, we have:

$$\begin{aligned}\varphi(\hat{\mathbf{P}}_k) \leq F(\tilde{x}^*, k) &= F(\tilde{x}^*, \hat{\mathbf{P}}) + k\frac{\partial F(\tilde{x}^*, \hat{\mathbf{P}}_k)}{\partial k} \\ &= F(\tilde{x}^*, \hat{\mathbf{P}}) + k\left(F(\tilde{x}^*, \hat{\mathbf{Q}}) - F(\tilde{x}^*, \hat{\mathbf{P}})\right).\end{aligned} \tag{35}$$

Therefore, the following is a valid upper bound for the contaminated problem:

$$\varphi(\hat{\mathbf{P}}_k) \leq (1-k)F(\tilde{x}^*, \hat{\mathbf{P}}) + kF(\tilde{x}^*, \hat{\mathbf{Q}}). \tag{36}$$

Since the upper bound (36) is provided by the approximate solution $\tilde{x}^*$, we replace the deterministic quantity $F(\tilde{x}^*, \hat{\mathbf{P}})$ by its statistical estimator $\overline{\varphi}(\hat{\mathbf{P}})$. This estimator is provided by the SDDP algorithm; see Step 3 of the Algorithm 1. By Proposition 3 from Kozmík and Morton (2014), we have that $\overline{\varphi}(\hat{\mathbf{P}}) \to \overline{\varphi}^*(\hat{\mathbf{P}})$, $w.p.1$, as the number of scenarios used to compute $\overline{\varphi}$ grows to infinity and that $\overline{\varphi}^*(\hat{\mathbf{P}}) \geq F(\tilde{x}^*, \hat{\mathbf{P}})$.

The value $F(\tilde{x}^*, \hat{\mathbf{Q}})$ cannot be computed directly as well. Instead, we form an upper bound estimator under the distribution $\hat{\mathbf{Q}}$ in a similar fashionas the SDDP algorithm does (see again Step 3 in Algorithm 1), but this time we use decisions $\tilde{x}^*$ as given. In a practical large-scale application, the decisions $\tilde{x}^*$ cannot be stored in a memory. We solve the original problem and store the cuts that are collected when the algorithm ends. Then, we proceed with the upper bound estimator under the distribution $\hat{\mathbf{Q}}$ and sample the scenarios. For these sampled scenarios, we use the stored cuts to obtain decisions $\tilde{x}^*$ and apply these under the new distribution $\hat{\mathbf{Q}}$. With this approach, we again get a statistical upper bound, say $\bar{F}(\tilde{x}^*, \hat{\mathbf{Q}})$, for the value of $F(\tilde{x}^*, \hat{\mathbf{Q}})$. Plugging this value into our formula (36), we reach the final upper contamination bound:

$$(1-k)\overline{\varphi}(\hat{\mathbf{P}}) + k\bar{F}(\tilde{x}^*, \hat{\mathbf{Q}}). \tag{37}$$

This bound is based on the set of scenarios used to compute the upper bounds and therefore it is not deterministic. We have an asymptotic validity of this bound provided by the results of Proposition 3 of Kozmík and Morton (2014):

$$(1-k)\overline{\varphi}(\hat{\mathbf{P}}) + k\bar{F}(\tilde{x}^*, \hat{\mathbf{Q}}) \to \overline{\varphi}_F^*, w.p.1.; \quad \varphi(\hat{\mathbf{P}}_k) \leq \overline{\varphi}_F^*, \tag{38}$$

as the sizes of samples to compute both upper bounds grow to infinity.

The contamination bounds we have just developed depend on the specification or choice of $\mathbf{Q}$ and $\hat{\mathbf{Q}}$, respectively. When the contaminating distribution $\mathbf{Q}$ is not fully specified, a natural idea is to use the worst distribution of the considered alternatives. For simple uncertainty sets and under the assumption of stage independence, such worst-case or robust contamination bounds can be applied. See, e.g., Shapiro (2012a)

for an example of a manageable choice of uncertainty set of one-dimensional probability distributions specified by a known support and expectation; or for two-stage problems see also Bertsimas et al. (2010).

7 Numerical illustrations

We begin with a simple asset allocation model without transaction costs and emphasize that our primary purpose is to illustrate the performance of contamination bounds as opposed to building a high-fidelity model for practical use. At stage t, the decisions x_t denote the allocations, and $\boldsymbol{p}_t$ denotes gross return per stage; i.e., the ratio of the price at stage t to that in stage $t-1$. These represent the only random parameters in the model. Without transaction costs, the nested CVaR model (13) specializes to:

$$\mathsf{Q}_t(x_{t-1}, \xi_t) = \min_{x_t, u_t} -\mathbf{1}^\top x_t + \lambda_{t+1} u_t + Q_{t+1}(x_t, u_t) \tag{39a}$$

$$\text{s.t. } \mathbf{1}^\top x_t = \boldsymbol{p}_t^\top x_{t-1} \tag{39b}$$

$$x_t \geq 0, \tag{39c}$$

except that in the first stage: (i) the right-hand side of (39b) is instead 1 and (ii) because $-\mathbf{1}^\top x_1$ is then identically -1, we drop this constant from the objective function.

We also consider the case with transaction costs, which are proportional to the value of the assets sold or bought. We must modify the rebalancing equation between stage $t-1$ and stage t to include the transaction costs of $f_t \mathbf{1}^\top |x_t - x_{t-1}|$, where the $|\cdot|$ function applies component-wise. Linearizing we obtain the following special case of model (13):

$$\begin{aligned} \mathsf{Q}_t(x_{t-1}, \xi_t) = \min_{x_t, z_t, u_t} \quad & -\mathbf{1}^\top x_t + \lambda_{t+1} u_t + Q_{t+1}(x_t, u_t) \\ \text{s.t.} \quad & \mathbf{1}^\top x_t + f_t \mathbf{1}^\top z_t = \boldsymbol{p}_t^\top x_{t-1} \\ & z_t - x_t \geq -x_{t-1} \\ & z_t + x_t \geq x_{t-1} \\ & x_t \geq 0. \end{aligned}$$

We used monthly price data of the most important assets traded on the Prague Stock Exchange, January 2009 to February 2012. The summary of the month-to-month price ratios can be found in Table 1. We have fitted a multidimensional correlated log-normal distribution to the price ratios to obtain the original distribution $\mathbf{P}$. The contaminating distribution $\mathbf{Q}$ was then constructed from $\mathbf{P}$ by increasing the variance by 20 %. Scenario trees were constructed by sampling $\hat{\mathbf{P}}$ and $\hat{\mathbf{Q}}$ from these distributions, using the polar method, cf. Knopp (1966) for normal distribution sampling. The L'Ecuyer random generator, cf. L'Ecuyer (2004) was used to generate the required uniform random variables. The computation was implemented in our own C++ software, using IBM ILOG CPLEX (2014) to solve the required linear programs and Armadillo (2014) library for matrix computations. The CVaR levels α_t were always set to 5 %.

Table 1 Data summary

Asset	Mean	SD
AAA	1.0290	0.1235
CETV	0.9984	0.2469
CEZ	0.9990	0.0647
ERSTE GROUP BANK	1.0172	0.1673
KOMERCNI BANKA	1.0110	0.1157
ORCO	1.0085	0.2200
PEGAS NONWOVENS	1.0221	0.0863
PHILIP MORRIS CR	1.0213	0.0719
TELEFONICA C.R.	0.9993	0.0595
UNIPETROL	1.0079	0.0843
VIENNA INSURANCE GROUP	1.0074	0.1100

Table 2 Testing problems setup

Stages	Descendants per node	Total scenarios
3	1, 000	1, 000, 000
5	1, 000	10^{12}

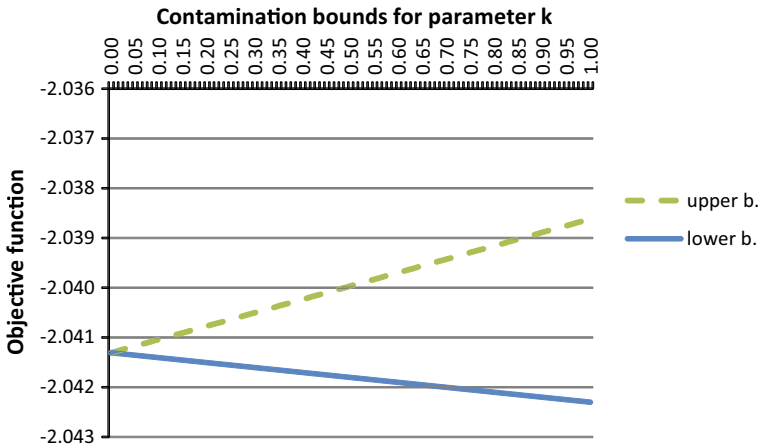

Fig. 1 Three-stage problem contamination bounds with no transaction costs

We evaluated the model with risk coefficients $\lambda_t = 0.1$. Both cases, with transaction costs of 0.3 % and without transaction costs, were considered. We have computed the contamination bounds for problems with 3 and 5 stages. In Table 2, we show the setup for the scenario trees used in our algorithm.

The three-stage problems can be solved to optimality using our SDDP algorithm, meaning that there is no gap between the lower bound and the upper bound, which is formed by computing the population mean rather than sampling. Figures 1 and 2

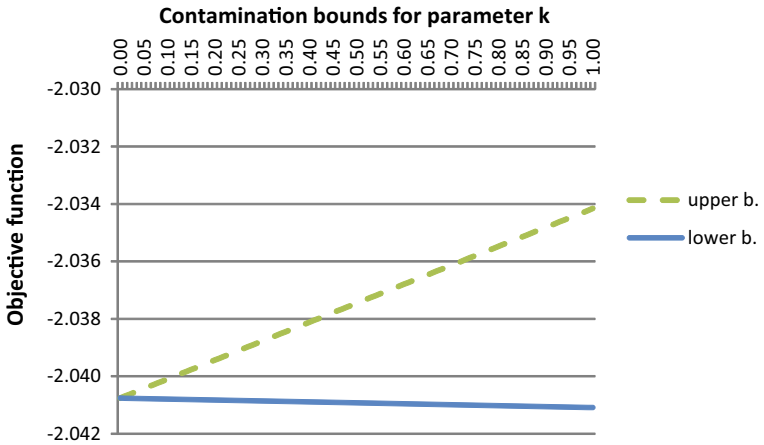

Fig. 2 Three-stage problem contamination bounds with transaction costs 0.3 %

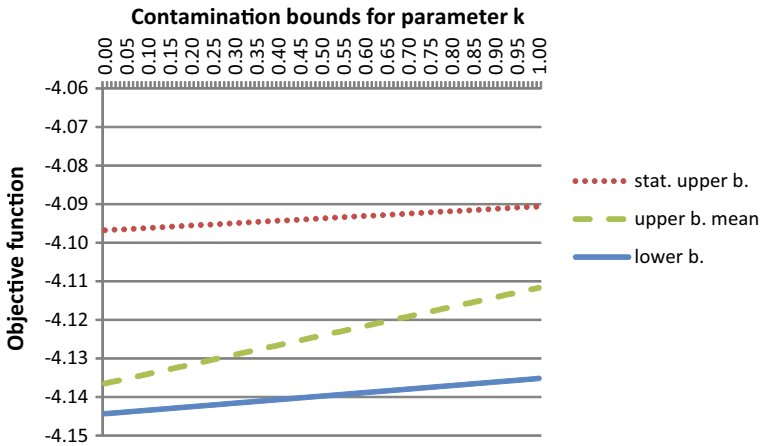

Fig. 3 Five-stage problem contamination bounds with no transaction costs

present the obtained lower and upper contamination bounds based on inequalities (34) and (36).

For the case of five-stage problems, we are unable to compute the solutions exactly and provide the contamination bounds based on the lower and upper bounds from the SDDP algorithm. The lower bound based on the inequality (34) remains deterministic, but the terms present in inequality (37) are estimated ten times and we provide their mean as well as the empirical statistical upper bounds with confidence level of 95 %. The results are presented in Figs. 3 and 4.

The first-stage decisions found for all considered setups are listed in Table 3.

The presented results show that for smaller problems, we are able to obtain tight contamination bounds; in our testing setup with three stages we have a spread of 0.09 % in the case without transaction costs and 0.17 % spread in the case with transaction costs, both cases considering $k = 50$ % contamination. For large-scale problems,

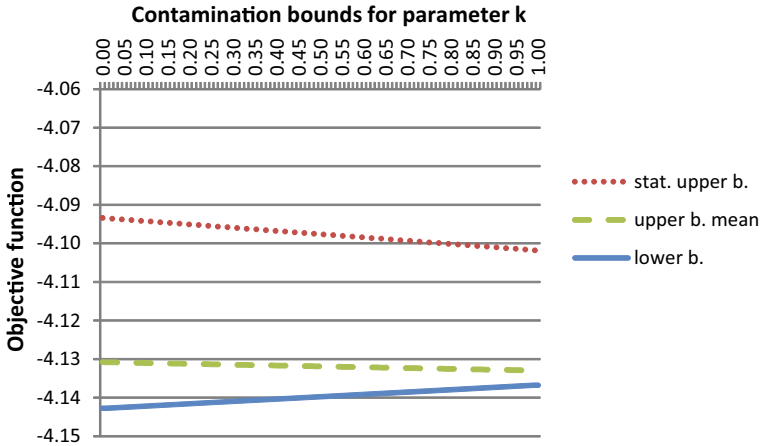

Fig. 4 Five-stage problem contamination bounds with transaction costs 0.3%

Table 3 First-stage decisions for three and five stages with the original and contaminating distributions

Stages	Trans. costs (%)	Distr.	AAA	PEGAS	PHILIP M.
3	0	$\hat{\mathbf{P}}$	0.1978	0.6214	0.1808
3	0	$\hat{\mathbf{Q}}$	0.3394	0.2834	0.3773
3	0.3	$\hat{\mathbf{P}}$	0.2065	0.5294	0.2642
3	0.3	$\hat{\mathbf{Q}}$	0.4204	0.1391	0.4405
5	0	$\hat{\mathbf{P}}$	0.4096	0.1868	0.4036
5	0	$\hat{\mathbf{Q}}$	0.3399	0.4143	0.2458
5	0.3	$\hat{\mathbf{P}}$	0.3652	0.2113	0.4235
5	0.3	$\hat{\mathbf{Q}}$	0.2547	0.4011	0.3442

we can rely on the statistically valid bound or on the mean of sampled estimates. For our five-stage setup, we obtained spreads of 1.13 and 1.03 % in the analogous cases when using empirical statistical upper bounds. Even though we consider these bounds pretty tight, we can also rely on the mean estimators, which are usually used in the SDDP algorithms. That gives us spreads of 0.38 and 0.19 %, respectively. The straightforward interpretation of our results would then state that the results of our model can be considered stable with respect to growing variance of the underlying random distribution which drives the asset price evolution.

8 Conclusion

We have shown three different models based on the CVaR risk measure for modeling risk in the multistage stochastic programs and discussed their basic properties and differences. Under the assumption of stage-wise independence, we present and apply

stochastic dual dynamic programming algorithm to the model with nested CVaR risk measure. For the purposes of output analysis, the contamination technique is extended to cover the large-scale cases where we are not able to solve the problem precisely, but we can obtain approximate solutions through SDDP. Numerical results with the asset allocation problem provide sufficiently tight bounds that can be used in practical applications to test the stability.

Further research should include extended numerical experiments, including all three presented models and problems with more stages. Other risk measures and probability distributions than those presented in the article could be also considered. More general structures without the stage independence assumption would provide another topic for further application of our ideas. In such a case, the SDDP algorithm does not apply and some other way to compute the contamination bounds for large-scale problems has to be developed.

Acknowledgments We thank the two anonymous referees for their valuable comments. The research was partly supported by the project of the Czech Science Foundation P/402/12/G097 'DYME/Dynamic Models in Economics. We gratefully acknowledge the helpful comments of Werner Römisch.

References

Armadillo (2014) C++ linear algebra library. http://arma.sourceforge.net/

Artzner P, Delbaen F, Eber J-M, Heath D (1999) Coherent measures of risk. Math Financ 9:203–228

Artzner P, Delbaen F, Eber J-M, Heath D, Ku H (2007) Coherent multiperiod risk adjusted values and Bellman's principle. Ann Oper Res 152:2–22

Bayraksan G, Morton DP (2011) A sequential sampling procedure for stochastic programming. Oper Res 59:898–913

Bertsimas D, Duan XV, Natarajan K, Teo C-P (2010) Model for minimax stochastic linear optimization problems with risk aversion. Math Oper Res 35:580–602

Carpentier P, Chancelier JP, Cohen G, De Lara M, Girardeau P (2012) Dynamic consistency for stochastic optimal control problems. Ann Oper Res 200:247–263

Chiralaksanakul A, Morton DP (2004) Assessing policy quality in multi-stage stochastic programming. Stochastic Programming E-Print Series

IBM ILOG CPLEX (2014) http://www.ibm.com/software/integration/optimization/cplex-optimization-studio/

Dupačová J (1990) Stability and sensitivity analysis for stochastic programming. Ann Oper Res 27:115–142

Dupačová J (1995) Postoptimality for multistage stochastic linear programs. Ann Oper Res 56:65–78

Dupačová J (1996) Scenario based stochastic programs: resistance with respect to sample. Ann Oper Res 64:21–38

Dupačová J, Polívka J (2007) Stress testing for VaR and CVaR. Quant Financ 7:411–421

Dupačová J (2008) Risk objectives in two-stage stochastic programming problems. Kybernetika 44:227–242

Dupačová J, Bertocchi M, Moriggia V (2009) Testing the structure of multistage stochastic programs. Comput Manag Sci 6:161–185

Eichhorn A, Römisch W (2005) Polyhedral risk measures in stochastic programming. SIAM J Optim 16:69–95

Guigues V (2014) SDDP for some interstage dependent risk-averse problems and application to hydro-thermal planning. Comput Optim Appl 57:167–203

Guigues V, Römisch W (2012a) SDDP for multistage stochastic linear programs based on spectral risk measures. Oper Res Lett 40:313–318

Guigues V, Römisch W (2012b) Sampling-based decomposition methods for multistage stochastic programs based on extended polyhedral risk measures. SIAM J Optim 22:286–312

Knopp R (1966) Remark on algorithm 334 [G5]: normal random deviates. Commun ACM 12:281

Kovacevic R, Pflug GCh (2009) Time consistency and information monotonicity of multiperiod acceptability functionals. Radon Ser Comput Appl Math 8:1–23

Kozmík V, Morton D (2014) Evaluating policies in risk-averse multi-stage stochastic programming. Math Program doi:10.1007/s10107-014-0787-8

Krokhmal P, Zabarankin M, Uryasev S (2011) Modeling and optimization of risk. Surv Oper Res Manag Sci 16:49–66

L'Ecuyer random streams generator (2004) http://www.iro.umontreal.ca/lecuyer/myftp/streams00/

Pennanen T, Perkkiö AP (2012) Stochastic programs without duality gaps. Math Program Ser B 136:91–110

Pereira MVF, Pinto LMVG (1991) Multi-stage stochastic optimization applied to energy planning. Math Program 52:359–375

Pflug GCh, Römisch W (2007) Modeling, measuring and managing risk. World Scientific Publishing, Singapore

Philpott AB, de Matos VL, Finardi EC (2013) On solving multistage stochastic programs with coherent risk measures. Oper Res 61:957–970

Philpott AB, Guan Z (2008) On the convergence of sampling-based methods for multi-stage stochastic linear programs. Oper Res Lett 36:450–455

Rockafellar RT, Uryasev S (2002) Conditional value at risk for general loss distributions. J Bank Finance 26:1443–1471

Rockafellar RT, Wets RJ-B (1976) Nonanticipativity and $\mathscr{L}^1$ martingales in stochastic optimization problems. Math Program Study 6:170–187

Roorda B, Schumacher JM (2007) Time consistency conditions for acceptability measures, with an application to tail value at risk. Insur Math Econ 40:209–230

Rudloff B, Street A, Valladao D (2010) Time consistency and risk averse dynamic decision models: definition, interpretation and practical consequences. Optim Online

Ruszczyński A, Shapiro A (2006) Conditional risk mappings. Math Oper Res 31:544–561

Ruszczyński A (2010) Risk-averse dynamic programming for Markov decision processes. Math Program 125:235–261

Shapiro A (2003) Inference of statistical bounds for multistage stochastic programming problems. Math Methods Oper Res 58:57–68

Shapiro A, Dentcheva D, Ruszczyński A (2009a) Lectures on stochastic programming: modeling and theory. SIAM

Shapiro A (2009b) On a time consistency concept in risk averse multistage stochastic programming. Oper Res Lett 37:143–147

Shapiro A (2011) Analysis of stochastic dual dynamic programming method. Eur J Oper Res 209:63–72

Shapiro A (2012a) Minimax and risk averse multistage stochastic programming. Eur J Oper Res 219:719–726

Shapiro A (2012b) Time consistency of dynamic risk measures. Oper Res Lett 40:436–439