Risk-Averse Stochastic Optimal Control: an efficiently computable statistical upper bound

Vincent Guigues
School of Applied Mathematics, FGV
Praia de Botafogo, Rio de Janeiro, Brazil
vincent.guigues@fgv.br

Alexander Shapiro
Georgia Institute of Technology
Atlanta, Georgia 30332-0205, USA,
ashapiro@isye.gatech.edu

Yi Cheng
Georgia Institute of Technology
Atlanta, Georgia 30332-0205, USA,
cheng.yi@gatech.edu

Abstract. In this paper, we discuss an application of the SDDP type algorithm to nested risk-averse formulations of Stochastic Optimal Control (SOC) problems. We propose a construction of a statistical upper bound for the optimal value of risk-averse SOC problems. This outlines an approach to a solution of a long standing problem in that area of research. The bound holds for a large class of convex and monotone conditional risk mappings. Finally, we show the validity of the statistical upper bound to solve a real-life stochastic hydro-thermal planning problem.

Key Words: stochastic programming, stochastic optimal control, SDDP, dynamic programming, risk measures, statistical upper bounds.

AMS subject classifications: 90C15, 90C90, 90C30.

1 Introduction

Multistage stochastic optimization problems are challenging to solve and have applications in many areas, for instance in finance and engineering, see for instance [29]. Popular methods to solve these problems often use decomposition techniques such as SDDP, proposed in [21], which is a sampling variant of the decomposition method given in [7]. Initially described for risk-neutral linear problems, the SDDP method has generated a rich literature and many variants in the past three decades, see, e.g., [12, 16, 19, 25, 22, 13, 3, 23, 17, 15].

For risk-neutral problems and a discrete finite sample space, a standard stopping criterion for SDDP is based on a deterministic lower bound and a statistical upper bound on the optimal value of the problem, computed at each iteration of the method. For nested risk-averse problems, a deterministic lower bound can be computed as in the risk-neutral case, but to the best of our knowledge, no working *statistical* upper bound has been proposed so far for SDDP.

Of course, in theory the value of the constructed approximate policy can be computed by evaluating the risk at each node of the scenario tree. However, this computation rapidly becomes

1

prohibitive with increase of the number of stages and the resulting exponential growth of the number of possible realizations of the stochastic data process. A deterministic upper bound on the value of the approximate risk-averse policy was proposed in [24] on the basis of inner approximations of the value functions. which is a natural extension of similar constructions for two stage programs (e.g., [8, section 9.5]). Recently, two variants of Dual SDDP were introduced that also compute a deterministic upper bound, in [17] using conjugate duality and in [15] using Lagrangian duality. The bounds in [17, 15] were developed for risk-neutral problems, and recently extended to risk-averse problems in [9]. However, the computational bulk required to compute the deterministic bounds from [24] and [9] for risk-averse problems increases rapidly with increase of the number of stages, the number of realizations of the stochastic data per stage, and the dimension of the state vectors. The goal of this paper is to fill this gap proposing an efficiently computable statistical upper bound for SDDP applied to nested-risk averse stochastic problems. This will be possible for a large class of monotone convex risk measures that will be studied.

Our developments will be derived for Stochastic Optimal Control (SOC) modeling, instead of the Multistage Stochastic Programming approachoften used in the SDDP and related methods. The SOC is classical with applications documented in a large number of publications (e.g., [6]). We would like to emphasize that many problems discussed in the Stochastic Programming (SP) literature, can be formulated in the SOC framework. One such example is the classical inventory model (it is presented from both points of view in sections 1.2.3 and 7.6.3 in [29]). Another such example is the hydro-thermal planning problem discussed in section 5. One modification in applying an SDDP type algorithm to SOC problems is the fact that it is not necessary anymore to solve the dual problems to compute the required subgradients of the cost-to-go functions. More importantly, from the point of view of the SDDP type algorithms, applied to risk-averse problems, there is an important difference between the SOC modeling, as compared with the SP approach. A straightforward attempt for computation of statistical upper bounds in the SP framework resulted in an exponential growth of the involved bias with increase of the number of stages, which made it practically useless (cf., [30]). On the other hand, we are going to demonstrate that in the SOC framework it is possible to construct such statistical upper bound in a computationally feasible way.

The outline of the paper is the following. In Section 2, we present the class of risk-neutral SOC problems and describe the SDDP type approach for solving this class of problems. In Section 3, we present and study the risk measures which will be used for the risk-averse SOC problem. In Section 4, we present the risk-averse SOC problem and describe the SDDP algorithm for this problem. In Section 4.2, we derive our statistical upper bound. Finally, in Section 5 we present numerical results where our upper bound is computed along iterations of SDDP type algorithm to solve a risk-averse real-life hydro-thermal planning problem.

We use the following notation. By $\xi_{[t]} := (\xi_1, ..., \xi_t)$ we denote the history of a process (ξ_t) up to time t. By $\mathbb{I}_A(x)$ we denote the indicator function of a set A, i.e., $\mathbb{I}_A(x) = 0$ if $x \in A$, and $\mathbb{I}_A(x) = +\infty$ otherwise. For $a \in \mathbb{R}$, $[a]_+ := \max\{a, 0\}$.

2 Risk-neutral Stochastic Optimal Control

Consider the classical Stochastic Optimal Control (SOC) (discrete time, finite horizon) model (e.g., [6]):

$$\min_{u_t \in \mathcal{U}_t} \quad \mathbb{E}^\pi \left[\sum_{t=1}^T c_t(x_t, u_t, \xi_t) + c_{T+1}(x_{T+1}) \right], \tag{2.1}$$

$$\text{s.t.} \quad x_{t+1} = F_t(x_t, u_t, \xi_t), \ t = 1, ..., T. \tag{2.2}$$

Here variables $x_t \in \mathbb{R}^{n_t}$, $t = 1, ..., T + 1$, represent the state of the system, $u_t \in \mathbb{R}^{m_t}$, $t = 1, ..., T$, are controls, $\xi_t \in \mathbb{R}^{d_t}$, $t = 1, ..., T$, are random vectors, $c_t : \mathbb{R}^{n_t} \times \mathbb{R}^{m_t} \times \mathbb{R}^{d_t} \to \mathbb{R}$, $t = 1, ..., T$, are cost functions, $c_{T+1}(x_{T+1})$ is a final cost function, $F_t : \mathbb{R}^{n_t} \times \mathbb{R}^{m_t} \times \mathbb{R}^{d_t} \to \mathbb{R}^{n_{t+1}}$ are (measurable) mappings and $\mathcal{U}_t$ is a (nonempty) subset of $\mathbb{R}^{m_t}$. Values x_1 and ξ_0 are deterministic (initial conditions); it is also possible to view x_1 as random with a given distribution, this is not essential for the following discussion. The optimization in (2.1) - (2.2) is performed over policies π determined by decisions u_t and state variables x_t considered as functions of $\xi_{[t-1]} = (\xi_1, ..., \xi_{t-1})$, $t = 1, ..., T$, this is emphasized in the notation $\mathbb{E}^\pi$. For the sake of simplicity, in order not to distract from the main message of the paper, we assume that the control sets $\mathcal{U}_t$ do not depend on x_t. It is possible to extend the analysis to the general case, where the control sets are functions of the state variables, we give a short discussion of that in section 7.1 of the Appendix.

It is said that the process ξ_t is *stagewise independent* if the probability distribution of ξ_t does not depend on $\xi_{[t-1]}$ for $t = 1, ..., T$.

We make the following basic assumption.

(A) The random data process $\xi_1, ..., \xi_T$ is stagewise independent and its probability distribution does not depend on our decisions.

Since it is assumed that the data process is stagewise independent, it suffices to consider policies of the form $u_t = u_t(x_t)$, $t = 1, ..., T$.

We can consider problem (2.1)-(2.2) in the framework of Stochastic Programming (SP) if we view $y_t = (x_t, u_t)$ as decision variables. Note, however, that here decision u_t should be made before realization of ξ_t becomes known, and x_t and u_t are functions of $\xi_{[t-1]}$ rather than $\xi_{[t]}$. That is, there is a time shift in the SOC formulation as compared with the SP framework. We emphasize that in both the SOC and SP approaches the decisions are made based on information available at time of the decision and both approaches try to solve the *same* problem. Because of the time shift, the dynamic programming equations for the SOC problem are slightly different from the respective equations in the SP framework. This has important implications for the SDDP algorithm, especially in the risk averse setting.

The dynamic programming equations can be written as follows. At the last stage, the value function $V_{T+1}(x_{T+1}) = c_{T+1}(x_{T+1})$ and, going backward in time for $t = T, ..., 1$, the value functions

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} \mathbb{E} \left[c_t(x_t, u_t, \xi_t) + V_{t+1}\big(F_t(x_t, u_t, \xi_t)\big) \right], \tag{2.3}$$

where the expectation is taken with respect to the (marginal) distribution of ξ_t, The optimal policy is defined by the optimal controls $\bar{u}_t(x_t) \in \mathcal{U}_t^*(x_t)$, where

$$\mathcal{U}_t^*(x_t) := \arg\min_{u_t \in \mathcal{U}_t} \mathbb{E} \left[c_t(x_t, u_t, \xi_t) + V_{t+1}\big(F_t(x_t, u_t, \xi_t)\big) \right]. \tag{2.4}$$

The optimal value of the SOC problem (2.1)-(2.2) is given by the first stage value function $V_1(x_1)$, and can be viewed as a function of the initial conditions x_1.

We assume that the sets $\mathcal{U}_t^*(x_t)$, $t = 1, ..., T$, are *nonempty* for every possible realization of state variables. This holds under standard regularity conditions, e.g., if the sets $\mathcal{U}_t$ are compact and the objective function in the right hand side of (2.4) is continuous in $u_t \in \mathcal{U}_t$.

We consider the convex case, by making the following assumption.

(B) For $t = 1, ..., T$: (i) the sets $\mathcal{U}_t$ are closed, convex, (ii) the cost functions $c_t(x_t, u_t, \xi_t)$ are convex in (x_t, u_t), and

$$F_t(x_t, u_t, \xi_t) := A_t x_t + B_t u_t + b_t, \tag{2.5}$$

with matrices $A_t = A_t(\xi_t)$, $B_t = B_t(\xi_t)$ and vectors $b_t = b_t(\xi_t)$ being functions of ξ_t.

It follows that the value functions $V_t(\cdot)$ are convex.

Suppose further that random vector ξ_t has a finite number of realizations ξ_{ti} with respective probabilities p_{ti}, $i = 1, ..., N$, $t = 1, ..., T$ (for the sake of simplicity assume that the cardinality N is the same for every time t). Denote $c_{ti}(x_t, u_t) := c_t(x_t, u_t, \xi_{ti})$ and $A_{ti} = A_t(\xi_{ti}), B_{ti} = B_t(\xi_{ti}), b_{ti} = b_t(\xi_{ti})$, $i = 1, ..., N$, the respective values of the parameters. In that case, the dynamic programming equations (2.3) can be written as

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} \underbrace{\sum_{i=1}^{N} p_{ti} \left[c_{ti}(x_t, u_t) + V_{t+1}\left(A_{ti} x_t + B_{ti} u_t + b_{ti}\right) \right]}_{\mathbb{E}[c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t)]}. \tag{2.6}$$

The subdifferentials of the value functions are obtained from the dynamic programming equations (2.6). That is, consider function

$$Q_t(x_t, u_t) := \mathbb{E}\left[c_t(x_t, u_t, \xi_t) + V_{t+1}\left(A_t x_t + B_t u_t + b_t\right) \right].$$

Since $c_t(x_t, u_t, \xi_t)$ is convex in (x_t, u_t) and V_{t+1} is convex, $Q_t(x_t, u_t)$ is convex. By (2.6) we have that

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} Q_t(x_t, u_t) = \inf_{u_t \in \mathbb{R}^{m_t}} \{Q_t(x_t, u_t) + \mathbb{I}_{\mathcal{U}_t}(u_t)\}. \tag{2.7}$$

Consequently we have the following formula for the subdifferential of $V_t(\cdot)$ (cf., [26, Theorem 24(a)]):

$$\partial V_t(x_t) = \left\{\gamma_t : (\gamma_t, 0) \in \partial[Q_t(x_t, u_t) + \mathbb{I}_{\mathcal{U}_t}(u_t)]\right\} = \left\{\gamma_t : (\gamma_t, 0) \in \partial Q_t(x_t, \bar{u}_t)\right\}, \tag{2.8}$$

where[1] $\bar{u}_t$ is any point of $\mathcal{U}_t^*(x_t)$. Since the expectation here is a finite sum, we have that (cf., [27, Theorem 23.8])

$$\underbrace{\sum_{i=1}^{N} p_{ti} \left[\partial\left(c_{ti}(x_t, u_t) + V_{t+1}(A_{ti} x_t + B_{ti} u_t + b_{ti})\right)\right]}_{\mathbb{E}[\partial(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t))]} \subset \partial Q_t(x_t, \bar{u}_t). \tag{2.9}$$

---

[1]The indicator function can be removed in the last term of (2.8) since the second component of $(\gamma_t, 0)$ is 0.

Actually under mild regularity conditions the equality in (2.9) holds, in particular the equality holds if the value functions are real valued. Note that the subdifferentials in (2.9) are taken jointly in x_t and u_t. It follows that any subgradient in the left side of (2.9) is also a subgradient in the right side of (2.9). Consequently a subgradient $\nabla V_t(\cdot)$ is given by

$$\nabla V_t(x_t) = \sum_{i=1}^{N} p_{ti} \left[\nabla c_{ti}(x_t, \bar{u}_t) + A_{ti}^{\top} \nabla V_{t+1}\big(A_{ti}x_t + B_{ti}\bar{u}_t + b_{ti}\big) \right], \tag{2.10}$$

for any $\bar{u}_t \in \mathcal{U}_t^*(x_t)$, where $\nabla c_t(x_t, \bar{u}_t, \xi_t)$ is a subgradient of $c_t(\cdot, \bar{u}_t, \xi_t)$ at x_t.

Now suppose that value functions $V_t(\cdot)$ are approximated by (lower bounding) piecewise affine functions

$$\underline{V}_t(x_t) = \max_{j=1,\dots,M} \ell_{tj}(x_t), \tag{2.11}$$

where $\ell_{tj}(x_t) = a_{tj}^{\top} x_t + h_{tj}$, $j = 1, \dots, M$. Then $\nabla \underline{V}_t(x_t) = a_{t\nu}$, where $\nu \in \{1, \dots, M\}$ is such that $\nu \in \arg\max_{j=1,\dots,M} \ell_{tj}(x_t)$. This suggests a way for computing a subgradient of a current approximation of the value functions in a cutting planes type algorithm discussed below. There is no need to solve dual problems as in the classical SDDP method.

A cutting planes (SDDP type) algorithm for the SOC problem can be described as follows. In the forward step at iteration k of the algorithm, for given convex piecewise affine lower bounding approximations $\underline{V}_t^k$ of the value functions and for a generated sample path (scenario) $\hat{\xi}_1, \dots, \hat{\xi}_T$ of realizations of the random data process, starting with the initial value $\hat{x}_1 = x_1$, compute a minimizer in the right hand side of (2.6) for the current approximation of the value function, that is

$$\hat{u}_t \in \arg\min_{u_t \in \mathcal{U}_t} \sum_{i=1}^{N} p_{ti} \left[c_{ti}(x_t, u_t) + \underline{V}_{t+1}^k\big(A_{ti}x_t + B_{ti}u_t + b_{ti}\big) \right], \tag{2.12}$$

for $x_t = \hat{x}_t$, and set $\hat{x}_{t+1} = F_t(\hat{x}_t, \hat{u}_t, \hat{\xi}_t)$. If the set $\mathcal{U}_t$ is polyhedral, the cost functions $c_{ti}(x_t, u_t)$ are piecewise affine functions of u_t, this minimization problem can be written as a linear programming problem. In the next backward step of the algorithm, the cutting planes approximation of the value functions are updated going backwards in time by adding the cuts at the computed trial points $\hat{x}_t$. In computing the cuts use subgradients (at the trial points) of the current approximations of the value functions.

3   Preliminaries on risk measures

Let $(\Omega, \mathcal{F}, P)$ be a probability space and let $\mathcal{Z}$ be a linear space of $\mathcal{F}$-measurable functions (random variables) $Z : \Omega \to \mathbb{R}$. A risk measure is a function $\mathcal{R} : \mathcal{Z} \to \mathbb{R}$ which assigns to a random variable Z a real number representing its risk. Typical example of the linear space $\mathcal{Z}$ is the space of random variables with finite p-th order moments, denoted $L_p(\Omega, \mathcal{F}, P)$, $p \in [1, \infty)$. It is said that risk measure $\mathcal{R}$ is *convex* if it possesses the properties of convexity, monotonicity, and translation equivariance. If moreover it is positively homogeneous, then it is said that risk measure $\mathcal{R}$ is *coherent* (coherent risk measures were introduced in [2]). We can refer to [11] and [29] for a thorough discussion of risk measures.

In this paper we consider a class of convex risk measures which can be represented in the following parametric form:
$$\mathcal{R}(Z) := \inf_{\theta \in \Theta} \mathbb{E}_P[\Psi(Z, \theta)], \tag{3.13}$$

where $\Theta \subset \mathbb{R}^k$ and $\Psi : \mathbb{R} \times \Theta \to \mathbb{R}$ is a real valued function. The notation $\mathbb{E}_P$ in (3.13) emphasizes that the expectation is taken with respect to the probability measure (distribution) P of random variable Z. We consider risk measures of the form (3.13) for every time period. That is, for every $t = 1, ..., T$, we consider a probability space $(\Omega_t, \mathcal{F}_t, P_t)$, and risk measure[2]

$$\mathcal{R}_t(Z_t) := \inf_{\theta_t \in \Theta} \mathbb{E}_{P_t}[\Psi(Z_t, \theta_t)], \ Z_t \in \mathcal{Z}_t, \tag{3.14}$$

defined on the respective linear space of random variables, say $\mathcal{Z}_t := L_p(\Omega_t, \mathcal{F}_t, P_t)$.

We make the following assumptions.

(C) (i) The set Θ is a nonempty closed convex. (ii) For every $Z_t \in \mathcal{Z}_t$, $t = 1, ..., T$, the expectation in the right hand side of (3.14) is well defined and the infimum is finite valued. (iii) The function $\Psi(z, \theta)$ is convex in $(z, \theta) \in \mathbb{R} \times \Theta$. (iv) For every $\theta \in \Theta$, the function $\Psi(\cdot, \theta)$ is monotone nondecreasing, i.e., if $z_1 \leq z_2$ then $\Psi(z_1, \theta) \leq \Psi(z_2, \theta)$. (v) For every $z, a \in \mathbb{R}$,
$$\inf_{\theta \in \Theta} \Psi(z + a, \theta) = a + \inf_{\theta \in \Theta} \Psi(z, \theta).$$

Proposition 3.1 *Assumptions* (B) *and* (C) *imply that the functional* $\mathcal{R}$, *defined in* (3.13), *satisfies the axioms of convex risk measures*[3].

Proof. It follows from assumption (C)(iii) that $\mathbb{E}[\Psi(Z, \theta)]$ is convex in $(Z, \theta) \in \mathcal{Z} \times \Theta$, and hence its minimum over convex set Θ is convex. That is, the functional $\mathcal{R} : \mathcal{Z} \to \mathbb{R}$ is convex. By Assumptions (B) and (C)(iv) the functional $\mathcal{R}$ is monotone, i.e., if $Z, Z' \in \mathcal{Z}$ are such that $Z \geq Z'$ almost surely (a.s.), with respect to the measure P, then $\mathcal{R}(Z) \geq \mathcal{R}(Z')$. Assumptions (B) and (C)(v) imply the translation equivariance property, i.e., $\mathcal{R}(Z + a) = \mathcal{R}(Z) + a$ for any $Z \in \mathcal{Z}$ and $a \in \mathbb{R}$. $\square$

Moreover, Assumption (C)(iv) implies that $\mathcal{R}$ is consistent with the stop-loss order meaning that if $Z_1 \leq_{\text{icx}} Z_2$ then $\mathcal{R}(Z_1) \leq \mathcal{R}(Z_2)$ where the relation $\leq_{\text{icx}}$ between random variables (representing losses) is given by

$$Z_1 \leq_{\text{icx}} Z_2 \iff \mathbb{E}[f(Z_1)] \leq \mathbb{E}[f(Z_2)]$$

for any nondecreasing convex function f such that the respective expectations are well defined (see [28, 20]). For random variables representing incomes, the stop-loss order is replaced with the second order stochastic dominance, see for instance [28, 20, 14].

---

[2]It is possible to consider different parametric sets Θ_t and different functions Ψ_t for different time periods. For the sake of simplicity, we consider the same set Θ and function Ψ, this is in line with the examples below. On the other hand, the probability distributions P_t could be different for different time periods.

[3]In fact, in our construction of the statistical upper bound we do not need the translation equivariance property. We assume this property in order for $\mathcal{R}$ to satisfy the standard axioms of convex risk measures.

Recall that $Z, Z' \in \mathcal{Z}$ are said to be distributionally equivalent (with respect to the reference measure P) if $P(Z \leq z) = P(Z' \leq z)$ for all $z \in \mathbb{R}$. It is said that a functional $\mathcal{R} : \mathcal{Z} \to \mathbb{R}$ is *law invariant* if $\mathcal{R}(Z) = \mathcal{R}(Z')$ for any distributionally equivalent $Z, Z' \in \mathcal{Z}$. It follows immediately from the definition (3.14) that $\mathcal{R}_t$, defined in (3.14), is a function of its cdf $F_t(z) = P_t(Z_t \leq z)$, and hence is law invariant. For every t, consider direct product $P_1 \times \cdots \times P_t$ of probability measures and the corresponding space $\mathcal{Z}_1 \times \cdots \times \mathcal{Z}_t$. Conditional mapping $\mathcal{R}_{t|\xi_{[t-1]}} : \mathcal{Z}_t \to \mathcal{Z}_{t-1}$ is defined as a counterpart of the law invariant functional $\mathcal{R}_t$, $t = 1, ..., T$. Since ξ_0 is deterministic, $\mathcal{R}_{1|\xi_0} = \mathcal{R}$. The associated nested functional is defined in the composite form

$$\mathfrak{R}(\cdot) := \mathcal{R}_{1|\xi_0}\Big(\mathcal{R}_{2|\xi_{[1]}}\big(\cdots \mathcal{R}_{T|\xi_{[T-1]}}(\cdot)\big)\Big). \tag{3.15}$$

We refer to [29, section 7.6] for a detailed discussion of constructions of such conditional mappings and nested functionals. Note that in this framework the process $\xi_1, ..., \xi_T$, viewed as a random process with respect to the reference probability distributions, is *stagewise independent* with P_t being the marginal distribution of ξ_t.

There is a large class of risk measures which can be represented in the parametric form (3.13).

Example 3.1 *The Average Value-at-Risk measure*

$$\mathsf{AV@R}_\alpha(Z) = \inf_{\theta \in \mathbb{R}} \mathbb{E}\left[\theta + \alpha^{-1}[Z - \theta]_+\right], \ \alpha \in (0, 1), \tag{3.16}$$

is of form (3.13) *with* $\Psi(z, \theta) = \theta + \alpha^{-1}[z - \theta]_+$, *and* $\Theta = \mathbb{R}$, $\mathcal{Z} = L_1(\Omega, \mathcal{F}, P)$.

Example 3.2 *A convex combination of the expectation and of Average Value-at-Risk measures given by*

$$\mathcal{R}(Z) := \lambda_0 \mathbb{E}[Z] + \sum_{i=1}^{k} \lambda_i \mathsf{AV@R}_{\alpha_i}(Z),$$

where λ_i *are positive numbers with* $\sum_{i=0}^{k} \lambda_i = 1$, *and* $\alpha_i \in (0, 1)$. *Here* $\mathcal{R}$ *is of form* (3.13) *with* $\Theta = \mathbb{R}^k$, $\mathcal{Z} = L_1(\Omega, \mathcal{F}, P)$, *and* $\Psi(z, \theta) = \lambda_0 z + \sum_{i=1}^{k} \lambda_i \left(\theta_i + \alpha_i^{-1}[z - \theta_i]_+\right)$.

Example 3.3 (ϕ-divergence) *Another example is risk measures constructed from ϕ-divergence ambiguity sets (cf., [4],[5],[29, section 7.2.2]). Let $\phi : \mathbb{R} \to \mathbb{R}_+ \cup \{+\infty\}$ be a convex lower semicontinuous function such that $\phi(1) = 0$ and $\phi(x) = +\infty$ for $x < 0$. By duality arguments the ditributionally robust functional associated with the ambiguity set determined by the respective ϕ-divergence constraint with level $\epsilon > 0$ can be written in the form (3.13) with*

$$\mathcal{R}_\epsilon(Z) = \inf_{\mu, \lambda > 0} \{\lambda \epsilon + \mu + \lambda \mathbb{E}_P[\phi^*((Z - \mu)/\lambda)]\}, \tag{3.17}$$

$\theta = (\mu, \lambda)$, $\lambda > 0$, *and* $\Psi(z, \theta) = \lambda \epsilon + \mu + \lambda \phi^*((Z - \mu)/\lambda)$, *where* ϕ^* *is the conjugate of* ϕ. *In particular for the Kullback-Leibler (KL)-divergence,* $\phi(x) = x \ln x - x + 1$, $x \geq 0$, *and*

$$\mathcal{R}_\epsilon(Z) = \inf_{\mu, \lambda > 0} \{\lambda \epsilon + \mu + \lambda e^{-\mu/\lambda} \mathbb{E}_P[e^{Z/\lambda}] - \lambda\}. \tag{3.18}$$

Given $\lambda > 0$ the minimizer over μ in (3.18) is given by $\mu = \lambda \ln \mathbb{E}_P[e^{Z/\lambda}]$ and hence

$$\mathcal{R}_\epsilon(Z) = \inf_{\lambda > 0} \{\lambda \epsilon + \lambda \ln \mathbb{E}_P[e^{Z/\lambda}]\}. \tag{3.19}$$

Risk measures in the above examples are positively homogeneous, and hence are coherent.

Example 3.4 *Let $u : \mathbb{R} \to [-\infty, +\infty)$ be a proper closed concave and nondecreasing utility function with nonempty domain. The functional*

$$\mathcal{R}(Z) := \inf_{\theta \in \mathbb{R}} \left\{ \theta - \mathbb{E}[u(Z + \theta)] \right\},$$

is of form (3.13) with $\Theta = \mathbb{R}$ and $\Psi(z, \theta) = \theta - u(z + \theta)$. This risk measure is convex, but is not necessarily positively homogeneous. It can be viewed as the opposite of the OCE (Optimized Certainty Equivalent (see [1]).

Extended polyhedral risk measures, introduced in [14], are also of form (3.13).

4 Risk-averse Stochastic Optimal Control

4.1 Risk-averse Setting

Consider the risk averse setting in the nested form. That is, the expectation operator in the risk neutral formulation (2.1) - (2.2) is replaced by the nested risk measure $\mathfrak{R}$, defined in (3.15) under the assumption that the data process is stagewise independent with respect to the reference distributions. Suppose further that the state equations are affine of the form (2.5). This leads to the following risk averse problem[4] (in the nested form)

$$\min_{u_t \in \mathcal{U}_t} \quad \mathcal{R}_{1|\xi_0}\Big(\mathbf{c}_1 + \mathcal{R}_{2|\xi_{[1]}}\big(\mathbf{c}_2 + \cdots + \mathcal{R}_{T|\xi_{[T-1]}}(\mathbf{c}_T)\big) + \mathbf{c}_{T+1}\Big), \tag{4.20}$$

$$\text{s.t.} \quad x_{t+1} = A_t x_t + B_t u_t + b_t, \ t = 1, ..., T, \tag{4.21}$$

where we use notation $\mathbf{c}_t := c_t(x_t, u_t, \xi_t)$, $t = 1, ..., T$, and $\mathbf{c}_{T+1} := c_{T+1}(x_{T+1})$. Recall that the optimization (minimization) in (4.20) is over policies which are functions of the data process, with (x_t, u_t) being a function of $\xi_{[t-1]}$, this is suppressed in the notation. The constraints in the above problem should be satisfied with probability one with respect to the reference measures.

The risk averse counterpart of dynamic equations (2.6) can be written as $V_{T+1}(x_{T+1}) = c_{T+1}(x_{T+1})$ and for $t = T, ..., 1$,

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t} \mathcal{R}_t\big(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t)\big) \tag{4.22}$$

$$= \inf_{u_t \in \mathcal{U}_t, \theta_t \in \Theta} \mathbb{E}_{P_t}\left[\Psi\big(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t), \theta_t\big)\right], \tag{4.23}$$

where formulation (4.23) is obtained by applying definition (3.14) of $\mathcal{R}_t$. Note that it is possible to write dynamic equations (4.22) in terms of the (static) risk measures $\mathcal{R}_t$ because of the basic assumption of stagewise independence of the process ξ_t (with respect to the reference measures) (e.g., [29, section 6.5.4, Remark 39]). The respective optimal policy is defined by the optimal controls

$$\bar{u}_t(x_t) \in \arg\min_{u_t \in \mathcal{U}_t} \mathcal{R}_t\big(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t)\big). \tag{4.24}$$

---

[4]Recall that $\mathcal{R}_{1|\xi_0} = \mathcal{R}$.

As in the risk neutral setting, we assume that the set of minimizers in the right hand side of (4.24) is *nonempty* for all possible realizations of state variables.

The developments of Section 2 can be adapted to this risk-averse framework. Under the convexity assumption (B), the value functions $V_t(\cdot)$ are convex in the risk averse setting as well. There are explicit formulas how to compute a subgradient of the functional $\mathcal{R} : \mathcal{Z} \to \mathbb{R}$ for various examples of risk measures (cf., [29, section 6.3.2]).

Recall definition (3.14) of risk measure $\mathcal{R}_t$. For x_t and the optimal control $\bar{u}_t = \bar{u}_t(x_t)$, determined by (4.24), consider a minimizer

$$\bar{\theta}_t \in \arg\min_{\theta_t \in \Theta} \mathbb{E}_{P_t} \left[\Psi\big(c_t(x_t, \bar{u}_t, \xi_t) + V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t), \theta_t\big) \right]. \tag{4.25}$$

Then, similar to (2.10) and using the Chain rule, a subgradient $\nabla V_t(x_t)$ of the value function V_t at x_t can be computed as

$$\nabla V_t(x_t) = \mathbb{E}_{P_t} \left[\Psi'(y_t, \bar{\theta}_t)\big(\nabla c_t(x_t, \bar{u}_t, \xi_t) + A_t^\top \nabla V_{t+1}\big(A_t x_t + B_t \bar{u}_t + b_t\big)\big) \right], \tag{4.26}$$

where $\Psi'(y_t, \bar{\theta}_t)$ is a subgradient[5] of $\Psi(\cdot, \bar{\theta}_t)$ at y_t, $\nabla c_t(x_t, \bar{u}_t, \xi_t)$ is a subgradient of $c_t(\cdot, \bar{u}_t, \xi_t)$ at x_t, $\nabla V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t)$ is a subgradient of V_{t+1} at $A_t x_t + B_t \bar{u}_t + b_t$, and $y_t := c_t(x_t, \bar{u}_t, \xi_t) + V_{t+1}(A_t x_t + B_t \bar{u}_t + b_t)$.

As a special case, consider Example 3.1 of the Average Value-at-Risk measure. In that case the minimizer $\bar{\theta}$ in the right hand side of (3.16) is given by the $(1 - \alpha)$-quantile of the considered distribution. That is, suppose that the reference distribution P_t has a finite number of N realizations with equal probabilities $1/N$. Then $\bar{\theta}_t$ can be computed by arranging values $c_{ti}(x_t, \bar{u}_t) + V_{t+1}(A_{ti} x_t + B_{ti} \bar{u}_t + b_{ti})$, $i = 1, \ldots, N$, in the increasing order and taking the respective empirical $(1 - \alpha)$-quantile. Consequently, the required subgradient of the current approximation of the value function can be computed in a straightforward way (cf., [32]).

4.2 Statistical upper bounds on the value of the policy

In this section, we discuss the construction of a statistical upper bound on the optimal value of the risk averse problem. As before, all probabilistic statements and expectations are taken with respect to the *reference distributions*. Let $\underline{V}_t(x_t)$, $t = 1, ..., T$, be current approximations of the value functions. This defines the corresponding (approximate) policy $(\hat{x}_t, \hat{u}_t)$ with

$$\hat{u}_t \in \arg\min_{u_t \in \mathcal{U}_t} \mathcal{R}_t\big(c_t(\hat{x}_t, u_t, \xi_t) + \underline{V}_{t+1}(A_t \hat{x}_t + B_t u_t + b_t)\big), \tag{4.27}$$

with value given by $\underline{V}_1(x_1)$.

For a given realization (scenario) $\xi_1, ..., \xi_T$ of the data process, $\hat{x}_t$ and $\hat{u}_t$ are computed in the forward step of the SDDP algorithm, and can be viewed as functions $\hat{x}_t = \hat{x}_t(\xi_{[t-1]})$ and $\hat{u}_t = \hat{u}_t(\xi_{[t-1]})$. When each reference probability distribution has a finite support (of N points), i.e., for the discretized version of the problem, these values are computable.

Now let $\hat{\theta}_t \in \Theta$ be a specified function of the data process, $\hat{\theta}_t = \hat{\theta}_t(\xi_{[t-1]})$, $t = 1, ..., T$. Note that $\hat{\theta}_t$ is non-anticipative in the sense that it does not depend on unobserved values $\xi_t, ..., \xi_T$ at

---

[5]If $\Psi(\cdot, \bar{\theta}_t)$ is differentiable at y_t, then $\Psi'(y_t, \bar{\theta}_t)$ is given by the derivative of $\Psi(\cdot, \bar{\theta}_t)$ at y_t.

time t. Denote $\hat{c}_t := c_t(\hat{x}_t, \hat{u}_t, \xi_t)$, $t = 1, ..., T$, and $\hat{c}_{T+1} := c_{T+1}(\hat{x}_{T+1})$. Consider the following sequence of random variables (functions of the data process) defined iteratively going backward in time: $\mathfrak{v}_{T+1} := \hat{c}_{T+1}$ and

$$\mathfrak{v}_t := \Psi(\hat{c}_t + \mathfrak{v}_{t+1}, \hat{\theta}_t), \ t = T, \dots, 1. \tag{4.28}$$

Of course, values $\mathfrak{v}_t$ depend on a choice of parameters $\hat{\theta}_t$. We will discuss an appropriate choice of $\hat{\theta}_t$ later. Our statistical upper bound on the value of a risk-averse approximate policy is given in the following proposition.

Proposition 4.1 *Consider the risk-averse problem* (4.20) *-* (4.21). *Let $\mathfrak{v}_t$ be the sequence of random variables (defined iteratively by* (4.28)*) associated with current approximations of the value functions. Then for $t = 1, ..., T$,*

$$\mathcal{R}_{t|\xi_{[t-1]}}\big(\hat{c}_t + \ldots + \mathcal{R}_{T|\xi_{[T-1]}}(\hat{c}_T + \hat{c}_{T+1})\big) \le \mathbb{E}_{|\xi_{[t-1]}}[\mathfrak{v}_t], \quad w.p.1. \tag{4.29}$$

In particular, $\mathbb{E}[\mathfrak{v}_1]$ is greater than or equal to the value of the policy defined by the considered approximate value functions, and is an upper bound on the optimal value of the risk averse problem.

Proof. For $t = T$, using the definition of $\hat{u}_T$ and since $\hat{\theta}_T \in \Theta$, we get

$$
\begin{aligned}
\mathcal{R}_{T|\xi_{[T-1]}}(\hat{c}_T + \hat{c}_{T+1}) &= \inf_{u_T \in \mathcal{U}_T} \mathcal{R}_T\Big(c_T(\hat{x}_T, u_T, \xi_T) + \hat{V}_{T+1}(A_T\hat{x}_T + B_T u_T + b_T)\Big) \\
&\le \mathbb{E}_{|\xi_{[T-1]}}\Big[\Psi\Big(c_T(\hat{x}_T, \hat{u}_T, \xi_T) + c_{T+1}(A_T\hat{x}_T + B_T\hat{u}_T + b_T), \hat{\theta}_T\Big)\Big] \\
&= \mathbb{E}_{|\xi_{[T-1]}}[\mathfrak{v}_T].
\end{aligned}
$$

We now use induction in t going backward in time. For $t - 1$ we have

$$
\begin{aligned}
&\mathcal{R}_{t-1|\xi_{[t-2]}}\Big(\hat{c}_{t-1} + \mathcal{R}_{t|\xi_{[t-1]}}\big(\hat{c}_t + \ldots + \mathcal{R}_{T|\xi_{[T-1]}}(\hat{c}_T + c_{T+1}(\hat{x}_{T+1}))\big)\Big) \\
&\le \mathcal{R}_{t-1|\xi_{[t-2]}}\Big(\hat{c}_{t-1} + \mathbb{E}_{|\xi_{[t-1]}}[\mathfrak{v}_t]\Big) \quad \text{(monotonicity and induction step)} \\
&\le \mathbb{E}_{|\xi_{[t-2]}}\Big[\Psi\Big(\hat{c}_{t-1} + \mathbb{E}_{|\xi_{[t-1]}}[\mathfrak{v}_t], \hat{\theta}_{t-1}\Big)\Big] \quad \text{(because } \hat{\theta}_{t-1} \in \Theta) \\
&= \mathbb{E}_{|\xi_{[t-2]}}\Big[\Psi\Big(\mathbb{E}_{|\xi_{[t-1]}}[\hat{c}_{t-1} + \mathfrak{v}_t], \hat{\theta}_{t-1}\Big)\Big] \quad \text{(since } \hat{c}_{t-1} \text{ is a function of } \xi_{[t-1]}) \\
&\le \mathbb{E}_{|\xi_{[t-2]}}\mathbb{E}_{|\xi_{[t-1]}}\Big[\Psi\big(\hat{c}_{t-1} + \mathfrak{v}_t, \hat{\theta}_{t-1}\big)\Big] \quad \text{(by Jensen's inequality)} \\
&= \mathbb{E}_{|\xi_{[t-2]}}\Big[\Psi\big(\hat{c}_{t-1} + \mathfrak{v}_t, \hat{\theta}_{t-1}\big)\Big] \\
&= \mathbb{E}_{|\xi_{[t-2]}}[\mathfrak{v}_{t-1}].
\end{aligned}
\tag{4.30}
$$

This completes the induction step. $\qquad\square$

Therefore, for a sample path (scenario) of the data process, an unbiased point estimate of an upper bound on the corresponding policy value can be computed recursively starting with $\mathfrak{v}_{T+1} = c_{T+1}(\hat{x}_{T+1})$ and going backward in time using the iteration procedure (4.28). Finally $\mathfrak{v}_1$ gives a point estimate of an upper bound on the corresponding value of the policy. Therefore by generating a sample of scenarios, of the random data process, and averaging the corresponding

point estimates it is possible to construct the respective statistical upper bound for the optimal value of the risk averse problem.

The quality of such statistical bound depends on the choice of the parameter value function $\hat{\theta}_t$. It is natural to use the corresponding minimizer of the form (4.25). That is, to take

$$\hat{\theta}_t \in \arg\min_{\theta_t \in \Theta} \mathbb{E}\left[\Psi\big(c_t(\hat{x}_t, \hat{u}_t, \xi_t) + \underline{V}_{t+1}(A_t\hat{x}_t + B_t\hat{u}_t + b_t), \theta_t\big)\right]. \tag{4.31}$$

The so defined $\hat{\theta}_t$ is a function of $\hat{x}_t$ and $\hat{u}_t$, which in turn are functions of $\xi_{[t-1]}$. For example, as it was pointed at the end of Section 4.1, in case of the Average Value-at-Risk measure such $\hat{\theta}_t$ can be easily computed by using the respective quantile. Note that even for $\hat{\theta}_t$ of the form (4.31) the inequality (4.29) can be strict. This is because Jensen's inequality was used in derivations (4.30). Nevertheless, this approach performed well in the numerical experiments discussed in the next section.

Remark 4.1 We would like to point to the important difference between the corresponding SOC and SP approaches to construction of the statistical upper bound for the risk averse problems. Computation of the parameter $\hat{\theta}_t$ in (4.31) is based on the distribution of random vector ξ_t. When ξ_t has a finite number of realizations ξ_{ti}, $i = 1, ..., N$, the parameter $\hat{\theta}_t$ is a function of all corresponding costs $\hat{c}_{ti}$ and A_{ti}, B_{ti}, b_{ti}, $i = 1, ..., N$, and therefore in a sense is a consistent estimate of $\bar{\theta}_t$ defined in (4.25). On the other hand, in the SP setting it was not possible to construct a computationally feasible consistent estimate of the respective parameter of the risk measure. As a result a straightforward attempt for computation of such statistical upper bound in the SP framework resulted in an exponential growth of the involved bias with increase of the number of stages, which made it practically useless (cf., [30]).

We close this section by presenting Algorithm 1 for computing the statistical upper bound for a T-stage SOC problem.

5   Numerical Experiments

In this section numerical experiments are performed on the Brazilian Inter-connected Power System problem (we refer to [31] for more details on the problem description). All experiments were run using Python 3.8.5 under Ubuntu 20.04.1 LTS operating system with a 4.20 GHz Intel Core i7 processor and 32Gb RAM. We extended the `MSPPy` solver[6][10] for the SDDP algorithm solving for the SOC problem. We report numerical results of the convergence guided by the deterministic lower bound and the statistical upper bound of the risk averse stochastic optimal control problem.

The hydro-thermal planning problem is a large-scale problem with $T = 120$ planning horizon stages and four state variables related to the energy reservoirs in four interconnected regions. The monthly energy inflows define the stochastic data process in the model. For the sake of simplicity, it is assumed in the experiments below that the random inflow process is stagewise independent. The (discretization) samples are generated from log-normal distributions (with 100 realizations

---

[6]https://github.com/lingquant/msppy

Algorithm 1 SDDP-type Algorithm for SOC Problem

1: Inputs: stage-wise independent samples $\xi_t := \{\xi_{tj}\}_{1 \leq j \leq N_t}, t = 1, \cdots, T$, initializations of $V_t(\cdot) : \underline{V}_t^0(\cdot), t = 1, \cdots, T$, initial point $\hat{x}_1$

2: **for** $k = 1, 2, \ldots, K$ **do**

3:     **for** $t = 1, \cdots, T$ **do**   ▷ Forward Step

4:         Draw a sample $(\hat{A}_t, \hat{B}_t, \hat{b}_t)$ from $\{\xi_t\}$

5:         $\hat{u}_t = \underset{u_t \in \mathcal{U}_t}{\arg\min} \, \mathcal{R}_t \left(c_t(\hat{x}_t, u_t, \xi_t) + \underline{V}_{t+1}^{k-1}(A_t \hat{x}_t + B_t u_t + b_t) \right)$

6:         $\hat{x}_{t+1} = \hat{A}_t \hat{x}_t + \hat{B}_t \hat{u}_t + \hat{b}_t$

7:     **end for**

8:     **for** $t = T, \cdots, 1$ **do**   ▷ Backward Step

9:         $\hat{\theta}_t = \underset{\theta_t \in \Theta}{\arg\min} \, \frac{1}{N_t} \sum_{j=1}^{N_t} \Psi \left(c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}^{k-1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj}), \theta_t \right)$

10:         $v_t = \frac{1}{N_t} \sum_{j=1}^{N_t} \Psi \left(c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}^{k-1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj}), \hat{\theta}_t \right)$

11:         $g_t = \frac{1}{N_t} \sum_{j=1}^{N_t} \Psi'(y_t, \hat{\theta}_t) \left(\nabla c_t(\hat{x}_t, \hat{u}_t, \hat{\xi}_{tj}) + A_{tj}^\top \nabla \underline{V}_{t+1}^{k-1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj}) \right)$

12:         $\underline{V}_t^k \leftarrow \{\mathcal{V} \in \underline{V}_t^{k-1} : \mathcal{V} \geq g_t(x_t - \hat{x}_t) + v_t\}$

13:     **end for**

14:     Lower bound: $L_k = \underline{V}_1^k(\hat{x}_1)$

15: **end for**

16: Generate S sample paths ξ_j': $\{\xi_{tj}'\}_{1 \leq t \leq T}, j = 1, \cdots, S$, run forward step for each sample path $\xi_j', j = 1, \cdots, S$, obtain $(\hat{u}_{tj}, \hat{x}_{tj})_{1 \leq t \leq T}$ and $\hat{x}_{T+1,j}, j = 1, \cdots, S$ ▷ Evaluation

17: Set $\mathfrak{v}_{T+1,j} = c_{T+1}(\hat{x}_{T+1,j}), j = 1, \cdots, S$

18: **for** $t = T, \cdots, 1$ **do**

19:     $\hat{\theta}_t = \underset{\theta_t \in \Theta}{\arg\min} \, \frac{1}{S} \sum_{j=1}^{S} \Psi \left(c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}') + \underline{V}_{t+1}^K(A_{tj}'\hat{x}_t + B_{tj}'\hat{u}_t + b_{tj}'), \theta_t \right)$

20:     **for** $j = 1, \cdots, S$ **do**

21:         $\mathfrak{v}_{tj} = \Psi(c_t(\hat{x}_{tj}, \hat{u}_{tj}, \xi_{tj}') + \mathfrak{v}_{t+1,j}, \hat{\theta}_t)$

22:     **end for**

23: **end for**

24: $\bar{\mathfrak{v}}_1 = \frac{1}{S} \sum_{j=1}^{S} \mathfrak{v}_{1j}, \sigma^2 = \frac{1}{S-1} \sum_{j=1}^{S} (\mathfrak{v}_{1j} - \bar{\mathfrak{v}}_1)^2$

25: Statistical upper bound: $U_S = \bar{\mathfrak{v}}_1 + z_{1-\alpha}\sigma/\sqrt{S}$, where $\mathbb{P}(Z \geq z_{1-\alpha}) = \alpha, Z \sim \mathcal{N}(0,1)$.

at each stage) estimated from the historical data. Previous attempts to define a statistical upper bound have shown some of the challenges of this task. For example, the numerical results in [30] show that by formulating the problem as a risk-averse multistage stochastic program, the scale of the statistical upper bounds starts to explode when the number of stages T is more than 10.

We aim to demonstrate via the hydro-thermal planning problem, the effectiveness of the construction of the statistical upper bound proposed in Section 4. This suggests first to formulate the problem as a risk-averse optimal control model, and then to solve it by a variant of the SDDP algorithm, while preserving the number of stages, the states, and the data process in the original problem. More specifically, we construct the upper bound as explained in Section 4.2, detailed in Algorithm 1. We conduct experiments for risk measures of convex combination of expectation and AV@R and KL-divergence, as described in Examples 3.2 and 3.3, respectively. We solve both problems, and compute the corresponding statistical upper bounds, by an SDDP-type algorithm as described in Algorithm 1.

Implementation Details.

1. Convex combination of expectation and AV@R (Example 3.2): $((1 - \lambda)\mathbb{E}[\cdot] + \lambda\mathsf{AV@R}_\alpha(\cdot))$. For this risk measure, we perform tests with $\alpha = 0.05$ and $\lambda \in \{0, 0.5, 1\}$. When $\lambda = 0$, the problem becomes risk neutral, while $\lambda = 1$ corresponds to an extreme risk aversion.

   In this setting, at each backward step and in the evaluation procedure (line 9 and line 19 in Algorithm 1), $\hat{\theta}_t$ can be computed by arranging values $c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj}), j = 1, \cdots, N_t$, in the increasing order and taking the respective empirical $(1 - \alpha)$-quantile. Moreover, in order to obtain a fast converging deterministic lower bound, we adopt the biased-sampling technique proposed in [18].

2. KL-divergence (Example 3.3). For this risk measure, we conduct experiments for $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-8}, 10^{-12}\}$, which corresponds to problems with different levels of risk aversion. In particular, when $\epsilon = 10^{-12}$, the problem is essentially a risk neutral problem, up to some numerical error.

   In this case, at steps indicated by line 9 and line 19 in Algorithm 1, the following (one-dimensional) convex program:

   $$\hat{\lambda}_t = \arg\min_{\lambda_t > 0}\{\lambda_t \epsilon + \lambda_t \ln \mathbb{E}_{P_t}\left[e^{Z_t/\lambda_t}\right]\}, \tag{5.32}$$

   where $Z_t := \{c_t(\hat{x}_t, \hat{u}_t, \xi_{tj}) + \underline{V}_{t+1}(A_{tj}\hat{x}_t + B_{tj}\hat{u}_t + b_{tj})\}_{1 \leq j \leq N_t}$, was solved using `Scipy` solver.

Results. For risk measure $(1 - \lambda)\mathbb{E}[\cdot] + \lambda\mathsf{AV@R}_\alpha(\cdot)$, with $\lambda = 0.5$, in order to examine the trend of the statistical upper bound, we compute the upper bound for the problem at every 10 iterations with a sample of size $S = 10$, by running 10 forward passes in parallel. Figure 1 displays the evolution of the deterministic lower bounds and the statistical upper bounds for the hydro-thermal planning problem for 3000 iterations. We can see from the figure that the statistical upper bound oscillates significantly for the first 500 iterations and then gradually stabilizes within narrow fluctuations. Table 1 reports, for different choices of λ, the statistical upper bounds obtained from Monte Carlo simulation using 3000 samples, along with the deterministic

lower bounds and the relative gap ($\frac{\text{upper bound} - \text{lower bound}}{\text{lower bound}}$) at iteration 3000. From the results, it seems that the relative gap of the problem is not very sensitive to the level of risk aversion.
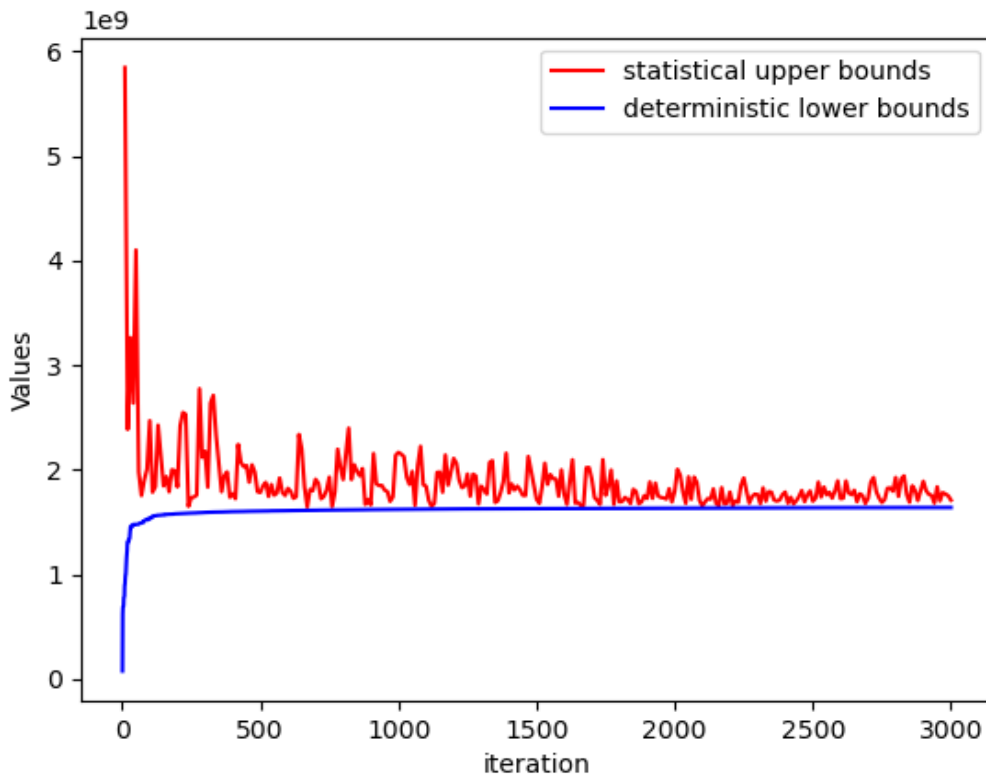

Figure 1: Evolution of lower and upper bounds for convex combination of expectation and AV@R problem when $\lambda = 0.5$.

	$(1 - \lambda)\mathbb{E}[\cdot] + \lambda\mathsf{AV@R}_\alpha(\cdot)$		
λ	Deterministic lower bound $(\times 10^9)$	Statistical upper bound $(\times 10^9)$	Gap(%)
0.0	0.345	0.348	0.97
0.5	1.640	1.672	1.93
1.0	6.669	7.003	5.02

Table 1: Convergence of convex combination of expectation and AV@R problem for different λ.

Table 2 reports results for the KL-divergence problem. The statistical upper bounds are computed by Monte Carlo simulation using 3000 samples, the lower bound and the relative gap, are computed as well for difference values of ϵ. All results in the table are obtained when the problems are solved for 3000 iterations. We observe that when ϵ increases, the relative gap becomes larger.

14

KL-divergence			
ϵ	Deterministic lower bound $(\times 10^9)$	Statistical upper bound $(\times 10^9)$	Gap(%)
10^{-1}	4.894	5.959	21.76
10^{-2}	4.202	4.659	10.89
10^{-3}	3.991	4.306	7.88
10^{-8}	3.246	3.324	2.42
10^{-12}	0.339	0.342	1.03

Table 2: Convergence of KL-divergence problem for different ϵ.

6 Concluding remarks

There are two somewhat different reasons for the gap between the considered statistical upper and deterministic lower bounds. One reason is the optimality gap similar to the risk neutral case. The additional gap, as compared to the risk neutral setting, appears because Jensens inequality is employed in derivations (4.30). This gap tends to increase as the function $\Psi(\cdot, \theta)$ becomes more "nonlinear". This can be clearly seen in Table 2, the gap increases with increase of ϵ, and also in Table 1 as the problem becomes more risk-averse.

When the function Ψ is not polyhedral, as for instance in the setting of ϕ-divergence example, the procedure requires solving nonlinear optimization programs. This could be inconvenient since nonlinear optimization solvers should be used. In the considered example of KL-divergence, this requires solving one-dimensional nonlinear programs, which does not pose a significant problem. In general, in order to keep the procedure to linear programming solvers, the Q-factor approach, discussed in section 7.2 of the Appendix, can be used. Note however that the Q-factor approach involves increasing the state space which could significantly slow down the convergence of the algorithm.

References

[1] Ben-Tal A. and Teboulle M. An old-new concept of convex risk measures: The optimized certainty equivalent. *Mathematical Finance*, 17:449–476, 2007.

[2] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Mathematical Finance*, 9:203–228, 1999.

[3] M. Bandarra and V. Guigues. Single cut and multicut stochastic dual dynamic programming with cut selection for multistage stochastic linear programs: Convergence proof and numerical experiments. *Computational Management Science*, 18(2):125–148, 2021.

[4] G. Bayraksan and D. K. Love. Data-driven stochastic programming using phi-divergences. *Tutorials in Operations Research, INFORMS*, pages 1563–1581, 2015.

[5] A. Ben-Tal and M. Teboulle. Penalty functions and duality in stochastic programming via phi-divergence functionals. *Mathematics of Operations Research*, 12:224–240, 1987.

[6] D.P. Bertsekas and S.E. Shreve. *Stochastic Optimal Control, The Discrete Time Case.* Academic Press, New York, 1978.

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

[8] J.R. Birge and F. Louveaux. *Introduction to Stochastic Programming.* Springer Series in Operations Research. Springer, 2011.

[9] B.F.P. da Costa and V. Leclere. Dual sddp for risk-averse multistage stochastic programs. *arXiv*, 2021.

[10] L. Ding, S. Ahmed, and A. Shapiro. A python package for multi-stage stochastic programming. *Optimization online*, 2019.

[11] H. Föllmer and A. Schied. *Stochastic Finance: An Introduction in Discrete Time.* Walter de Gruyter, Berlin, 2nd edition, 2004.

[12] V. Guigues. SDDP for some interstage dependent risk-averse problems and application to hydro-thermal planning. *Computational Optimization and Applications*, 57:167–203, 2014.

[13] V. Guigues. Dual dynamic programing with cut selection: Convergence proof and numerical experiments. *European Journal of Operational Research*, 258:47–57, 2017.

[14] V. Guigues and W. Römisch. Sampling-based decomposition methods for multistage stochastic programs based on extended polyhedral risk measures. *SIAM Journal on Optimization*, 22:286–312, 2012.

[15] V. Guigues, A. Shapiro, and Y. Cheng. Duality and sensitivity analysis of multistage linear stochastic programs. *Optimization online*, 2019.

[16] G. Infanger and D. Morton. Cut sharing for multistage stochastic linear programs with interstage dependency. *Math. Program.*, 75:241–256, 1996.

[17] V. Leclere, P. Carpentier, J-P. Chancelier, A. Lenoir, and F. Pacaud. Exact converging bounds for stochastic dual dynamic programming via fenchel duality. *SIAM Journal on Optimization*, 30:1223–1250, 2020.

[18] R.P. Liu and A. Shapiro. Risk neutral reformulation approach to risk averse stochastic programming. *European Journal of Operational Research*, 286:21–31, 2020.

[19] N. Lohndorf and A. Shapiro. Modeling time-dependent randomness in stochastic dual dynamic programming. *European Journal of Operational Research*, 273:650–661, 2019.

[20] A. Muller and D. Stoyan. *Comparison Methods for Stochastic Models and Risks.* Wiley Series in Probability and Statistics. John Wiley Sons Ltd., Chichester, 2002.

[21] M.V.F. Pereira and L.M.V.G. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical programming*, 52(1-3):359–375, 1991.

[22] A. Philpott, V. de Matos, and E. Finardi. Improving the performance of stochastic dual dynamic programming. journal of computational and applied mathematics. *Journal of Computational and Applied Mathematics*, 290:196 – 208, 2015.

[23] A. B. Philpott and Z. Guan. On the convergence of stochastic dual dynamic programming and related methods. *Operations Research Letters*, 36:450–455, 2008.

[24] A.B. Philpott, V.L. de Matos, and E. Finardi. On solving multistage stochastic programs with coherent risk measures. *Operations Research*, 61(4):957–970, 2013.

[25] A.R. De Queiroz and D.P. Morton. Sharing cuts under aggregated forecasts when decomposing multi-stage stochastic programs. *Operations Research Letters*, 41:311–316, 2013.

[26] R. T Rockafellar. *Conjugate Duality and Optimization.* Society for Industrial and Applied Mathematics, Philadelphia, 1974.

[27] R. Tyrrell Rockafellar. *Convex analysis.* Princeton Mathematical Series. Princeton University Press, 1970.

[28] M. Shaked and J.G. Shanthikumar. *Stochastic Orders and their Applications. .* Probability and Mathematical Statistics. Academic Press Inc., Boston, MA, 1994.

[29] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory.* SIAM, Philadelphia, third edition, 2021.

[30] A. Shapiro and L. Ding. Upper bound for optimal value of risk averse multistage problems. *Technical report, Georgia Tech*, 2016.

[31] A. Shapiro, W. Tekaya, J.P. da Costa, and M. Pereira Soares. Risk neutral and risk averse stochastic dual dynamic programming method. *European Journal of Operational Research*, 224:375–391, 2013.

[32] Alexander Shapiro, Wajdi Tekaya, Joari Paulo da Costa, and Murilo Pereira Soares. Risk neutral and risk averse stochastic dual dynamic programming method. *European journal of operational research*, 224(2):375–391, 2013.

7 Appendix

7.1 Controls

Consider the setting where the control set depends on the state variables. That is, consider the following extension of problem (2.1) - (2.2):

$$\min_{u_t \in \mathcal{U}_t(x_t)} \quad \mathbb{E}^\pi \left[\sum_{t=1}^T c_t(x_t, u_t, \xi_t) + c_{T+1}(x_{T+1}) \right], \tag{7.1}$$

$$\text{s.t.} \quad x_{t+1} = F_t(x_t, u_t, \xi_t), \ t = 1, ..., T, \tag{7.2}$$

where $\mathcal{U}_t : \mathbb{R}^{n_t} \rightrightarrows \mathbb{R}^{m_t}$ is a (measurable) point to set mapping, $t = 1, ..., T$. By changing the cost functions to $\bar{c}_t(x_t, u_t, \xi_t) := c_t(x_t, u_t, \xi_t) + \mathbb{I}_{\mathcal{U}_t(x_t)}(u_t)$, we can write problem (7.1) - (7.2) in the following form

$$\min_{u_t \in \mathbb{R}^{m_t}} \quad \mathbb{E}^\pi \left[\sum_{t=1}^T \bar{c}_t(x_t, u_t, \xi_t) + c_{T+1}(x_{T+1}) \right], \tag{7.3}$$

$$\text{s.t.} \quad x_{t+1} = F_t(x_t, u_t, \xi_t), \ t = 1, ..., T. \tag{7.4}$$

In order to maintain convexity of the value functions, we need to verify convexity in (x_t, u_t) of the cost functions $\bar{c}_t(x_t, u_t, \xi_t)$, i.e., to verify convexity of the indicator functions $\psi_t(x_t, u_t) := \mathbb{I}_{\mathcal{U}_t(x_t)}(u_t)$. Note that $\psi_t(x_t, u_t) = 0$ if $u_t \in \mathcal{U}_t(x_t)$, and $\psi_t(x_t, u_t) = +\infty$ otherwise, i.e., $\psi_t(\cdot, \cdot)$ is the indicator function of the set $A_t := \{(x_t, u_t) : u_t \in \mathcal{U}_t(x_t)\}$. Therefore $\psi_t(x_t, u_t)$ is convex iff the set A_t is a convex subset of $\mathbb{R}^{n_t} \times \mathbb{R}^{m_t}$. In particular, suppose that $\mathcal{U}_t(x_t) := \{u_t : g_{ti}(x_t, u_t) \leq 0, \ i = 1, ..., k\}$ for given functions $g_{ti} : \mathbb{R}^{n_t} \times \mathbb{R}^{m_t} \to \mathbb{R}$. Then the set A_t is convex if the functions $g_{ti}(\cdot, \cdot)$ are convex.

We also should be able to compute a subgradient of $\psi_t(x_t, u_t)$ with respect to x_t. Assuming that the set A_t is convex and closed, the subdifferential of $\psi_t(x_t, u_t)$ at a point $(\bar{x}_t, \bar{u}_t) \in A_t$ is given by the normal cone to A_t at the point $(\bar{x}_t, \bar{u}_t)$.

7.2 Q-factor approach

The following is a counterpart of the Q-factor approach popular in the SOC applications. Consider the dynamic equations (4.23) and define

$$Q_t(x_t, u_t, \theta_t) := \mathbb{E}_{P_t} \left[\Psi \big(c_t(x_t, u_t, \xi_t) + V_{t+1}(A_t x_t + B_t u_t + b_t), \theta_t \big) \right]. \tag{7.5}$$

We have that

$$V_t(x_t) = \inf_{u_t \in \mathcal{U}_t, \theta_t \in \Theta} Q_t(x_t, u_t, \theta_t),$$

and hence the dynamic equations (4.23) can be written in terms of $Q_t(x_t, u_t, \theta_t)$ as

$$Q_t(x_t, u_t, \theta_t) = \mathbb{E}_{P_t} \left[\Psi \Big(c_t(x_t, u_t, \xi_t) + \inf_{u_{t+1} \in \mathcal{U}_{t+1}, \theta_{t+1} \in \Theta} Q_{t+1}\big(A_t x_t + B_t u_t + b_t, u_{t+1}, \theta_{t+1}\big), \theta_t \Big) \right]. \tag{7.6}$$

The cutting planes, SDDP type, algorithm can be applied directly to functions $Q_t(x_t, u_t, \theta_t)$ rather than to the value functions $V_t(x_t)$. In the backward step of the algorithm, subgradients with respect to x_t, u_t and θ_t, of the current approximations of the functions $Q_t(x_t, u_t, \theta_t)$, should be computed. An advantage of that approach is that the calculation of these subgradients does

not require solving *nonlinear optimization* programs even if the function Ψ is not polyhedral[7]. On the other hand, this Q-factor approach involves increasing the state space from x_t to (x_t, u_t, θ_t), which could make the convergence of the algorithm considerably slower.

---

[7]The function Ψ is not polyhedral, for example, in the ϕ-divergence case. In that case the SDDP algorithm, applied to the value functions $V_t(x_t)$, requires solving nonlinear programs of the form (3.17).