Noname manuscript No.
(will be inserted by the editor)

A Data-Driven Approach for a Class of Stochastic Dynamic Optimization Problems

Thuener Silva · Davi Valladão · Tito Homem-de-Mello

Received: date / Accepted: date

Abstract Dynamic stochastic optimization models provide a powerful tool to represent sequential decision-making processes. Typically, these models use statistical predictive methods to capture the structure of the underlying stochastic process without taking into consideration estimation errors and model misspecification. In this context, we propose a data-driven prescriptive analytics framework aiming to integrate the machine learning and dynamic optimization machinery in a consistent and efficient way to build a bridge from data to decisions. The proposed framework tackles a relevant class of dynamic decision problems comprising many important practical applications. The basic building blocks of our proposed framework are: (i) a Hidden Markov Model as a predictive (machine learning) method to represent uncertainty; and (ii) a distributionally robust dynamic optimization model as a prescriptive method that takes into account estimation errors associated with the predictive model and allows for control of the risk associated with decisions. Moreover, we present an evaluation framework to assess out-of-sample performance in rolling horizon schemes. A complete case study on dynamic asset allocation illustrates the proposed framework showing superior out-of-sample performance against selected benchmarks. The numerical results show the practical importance and applicability of the proposed framework since it

T. Silva
Industrial Engineering Department, Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Rua Marquês de São Vicente, 225, Gávea – Rio de Janeiro, RJ 22451-900, Brazil
Tel.: +55 21 3527.1284
E-mail: thuener@esp.puc-rio.br

D. Valladão
Industrial Engineering Department, Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Rua Marquês de São Vicente, 225, Gávea – Rio de Janeiro, RJ 22451-900, Brazil

T. Homem-de-Mello School of Business, Universidad Adolfo Ibáñez, Diagonal las Torres 2640, Peñalolén, Santiago, Chile

extracts valuable information from data to obtain robustified decisions with an empirical certificate of out-of-sample performance evaluation.

Keywords Stochastic programming · Distributionally robust dynamic optimization · Hidden Markov models · Risk constraints · Stochastic Dual Dynamic Programming

1 Introduction

Dynamic stochastic optimization models have a long history in operations research, with applications in many different areas. Such models represent a sequential decision-making process whereby information is revealed in stages, and decisions are made based on the information available up to that point. There are multiple ways to solve dynamic stochastic optimization models. The choice for a particular approach depends, of course, on the structure of the problem under study. Whatever the solution method being used, an assumption often made in the literature is that the distribution of the underlying stochastic process representing the uncertainty is known, perhaps by fitting a distribution (or moments) to available data, or by constructing a scenario tree from subjective probabilities, or by postulating a model according to the type of problem as in the case of financial models based on stochastic differential equations, to name a few examples. In some cases, even more sophisticated simulators can be used as long as they can generate random samples that accurately characterizes uncertainty.

Our goal in this paper is to depart from the common assumption of known distributions and to model the uncertainty directly from the data. This is accomplished by applying a machine learning approach to the problem. More specifically, we use a *Hidden Markov Model* (HMM) to learn the structure of the data. The HMM can indeed be viewed as a machine learning method in that it classifies the data according to *unobservable states*, and then estimates the transition probabilities between each pair of states. Since each (unobservable) state corresponds to situations where the underlying stochastic process behaves similarly, it is reasonable to assume that, conditionally on the state of the HMM, the process has a certain distribution—typically, a mixture of Gaussian distributions whereby the parameters are estimated from the data. We refer the reader to [33] for a comprehensive tutorial on HMM. In a sense, our paper complements the work of [5], who also present a data-driven approach for a class of dynamic stochastic optimization problems but based on different machine learning techniques. While that work allows explicitly for the presence of features in the data, it requires the data to be independent and identically distributed, whereas in our case we are more interested in the situation where there is correlation which is captured by the HMM.

Although HMMs have been studied for decades, it appears that their use in the context of optimization models is limited. In the case of dynamic *convex* stochastic optimization models, the resulting structure of the HMM allows us to employ a variation of the now well-established Stochastic Dual Dynamic

Programming algorithm (SDDP) to solve such problems. The SDDP method was proposed in the seminal paper of [27] for problems where the uncertainty is *stagewise independent* but it was later extended to the case where there is Markovian dependence, although such models require the user to input the transition probability matrices; see, for instance, [24, 31, 22]. In a nutshell, the algorithm consists of alternating forward and backward steps: the forward step generates a sample path of the process to obtain a corresponding sequence of solutions, and the backward step successively approximates the value function in each period by linear cuts. SDDP has two particularly attractive features: first, it can solve large-scale stochastic dynamic problems, and second, it provides a *policy* rather than just a numerical solution. Such features are accomplished through the development of a sequence piecewise-linear value function approximations in each stage. Once the approximations are built, one can evaluate the optimal policy for an arbitrary realization of the stochastic process by solving a sequence of linear programs. Models that use SDDP as a solution technique have indeed become very popular in the literature, with applications in many areas such as energy, finance and transportation.

One drawback of HMMs, however, is that since the probabilities of transition between pairs of states are estimated from the data, the solutions of the optimization model that uses such transition probabilities will be very much dependent on the observed data, leading to a problem of *overfitting*. Such dependence may then lead to poor *out-of-sample performance* of the resulting policies due to estimation errors and model misspecification. We prevent such phenomenon from happening by employing a distributionally robust optimization (DRO) approach that allows for variations in the estimated transition probability matrix of the HMM. Our DRO model leads to tractable formulations that do not increase the computational complexity of the model; moreover, they can be solved by a variation of the SDDP.

The concern about out-of-sample performance is, unfortunately, often overlooked in the stochastic optimization literature, particularly in the case of dynamic models—oftentimes the user simply implements the first-stage decision given by the model and then re-solves the model in every period in order to obtain new decisions. As we discuss in the paper, such procedure can be expensive and wasteful, as it discards the value function approximations obtained in previous steps of a rolling horizon scheme. To counter that effect, we propose an extra step in the out-of-sample evaluation that allows for an improvement of the current value function approximations for updated values of the previous decisions (initial conditions of the current problem). Such feature makes the policies generated by the algorithm easier to use and quicker to evaluate.

We illustrate our ideas with a dynamic asset allocation problem. Such problem consists of decision processes under uncertainty with complex characteristics embedding the investor's risk tolerance, transaction cost, and price dynamics. By building upon previous work, we propose a Data-Driven DRO approach that estimates an HMM for the return process and allows for ambiguity in the transition probability matrix with the thrust to enhance out-of-sample performance. In the numerical tests, we provide a comprehensive sensi-

tivity analysis of the robustness and risk-aversion parameters, and execute the robustness tuning procedure to select the appropriate robustness level. The results obtained for a hold-out testing dataset show that the resulting portfolio can yield excellent results, with enhanced out-of-sample performance over selected benchmarks, including the equal-weight-allocation strategy, which has been shown to be optimal under certain assumptions [25, 16] with a competitive out-of-sample (empirical) performance [9].

In summary, the contributions of this paper are the following:

(i) We consider a class of dynamic stochastic optimization problems with risk-aversion and, rather than specifying a distribution for the data as typical in the literature, we apply a machine learning approach—namely, a Hidden Markov Model—to learn the structure of the data, and incorporate that information into an optimization model;
(ii) To account for the estimation and misspecification errors resulting from the HMM estimation procedure, and to avoid excessive dependence of the model on the data, we propose a distributionally robust optimization approach to the problem whereby ambiguity is allowed in the Markov transition probability matrix estimated by the HMM;
(iii) By building from some ingredients from the literature, we present a variation of the SDDP algorithm that can solve the DRO problem mentioned in (ii). Moreover, we provide *deterministic* lower and upper bounds and prove that the gap between lower and upper bounds becomes zero after a finite number of iterations. While we present the methodology in the context of the DRO problem, the approach can be easily adapted to other settings such as the standard SDDP, or the SDDP with *nested* risk measures that can be linearized, such as CV@R.
(iv) We propose a rolling-horizon scheme for out-of-sample evaluation of the policies generated by the algorithm that make such policies easier to use and quicker to evaluate for a fixed robustness level; Also, we propose a robustness tuning procedure as a series of out-of-sample evaluation steps, whereby the robustness level with best out-of-sample performance is selected.
(v) A case study for an asset allocation problem is presented, demonstrating the benefits of the proposed approach over a benchmark from the literature. For this application, we propose an alternative novel deterministic lower-bound that exploits the structure of the problem.

2 A data-driven prescriptive analytics framework

We start by presenting a data-driven prescriptive analytics framework that integrates all the machine learning and optimization machinery in a consistent and efficient way to build a bridge from data to decisions. The basic building blocks of our proposed framework are (i) a predictive (machine learning) method to represent uncertainty, and (ii) a prescriptive (optimization)

model that takes into account estimation errors associated with the predictive model and allows for control of the risk associated with decisions. In our context, the predictive model is a Hidden Markov Model, and the prescriptive model is a distributionally robust dynamic optimization model with risk-based constraints that induces a (parameterized) level of robustness over the HMM transition probabilities given that they might be polluted with estimation errors.

In what follows, we describe these building blocks in more detail. Before that, however, we establish the notation for the dynamic stochastic optimization problem of interest. We consider a filtered probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where $\mathcal{F} = \mathcal{F}_T$ and $\mathcal{F}_0 \subseteq \ldots \subseteq \mathcal{F}_T$. The input is represented by a stochastic process $\{\boldsymbol{\xi}_t\}$ with values in $\mathbb{R}^m$ such that the sigma-algebra generated by $\{\boldsymbol{\xi}_s, K_s\}_{s=0}^t$ is contained in $\mathcal{F}_t$, and a Markov chain $\{K_t\}$ with unobservable states such that the following assumption holds:

Assumption 1. *The process $\{K_t\}$ is a hidden time-homogeneous Markov chain with finite state-space $\mathcal{K}$.*

In other words, we assume that the state given by K_t can not be observed or directly measured by the decision maker, but it respects the Markov property

$$P\left(K_t = k, \,|\, K_{t-1} = j, K_{t-2} = i, \ldots\right) = P\left(K_t = k, \,|\, K_{t-1} = j\right) = P_{jk},$$

where transition probability matrix P does not change over time.

Assumption 2. *The process $\{\boldsymbol{\xi}_t\}$ is an observable time-dependent vector-valued stochastic process that directly affects the performance of the underlying prescriptive model. The distribution of each $\boldsymbol{\xi}_t$ depends only on t and on the current (unobservable) state of the Markov chain $\{K_t\}$.*

Assumptions 1 and 2 are basic for HMMs and allow for modeling different "states" of the system. For instance, in the financial model discussed in Section 5, the process $\{\boldsymbol{\xi}_t\}$ corresponds to the financial returns of each asset. A typical HMM estimation would reveal that the Markov states correspond to the market being in a "bull", "regular", or "bear" state[1]. It is important to stress that such states are unobservable to the user — rather, they are learned directly from the data. In the financial example, we never observe directly the state of the market, but we can use the sequence of historical asset returns to estimate the transition probability matrix and the probability distribution of future returns conditioned to each Markov state.

Assumption 2 also implies that, conditionally on each given (unobservable) state of the Markov chain $\{K_t\}$, the underlying stochastic process $\{\boldsymbol{\xi}_t\}$ is stagewise independent. That is,

$$\begin{aligned} P\left(\boldsymbol{\xi}_t \in A, \boldsymbol{\xi}_{t+1} \in B \,|\, K_t = j, K_{t+1} = k\right) = \\ P\left(\boldsymbol{\xi}_t \in A \,|\, K_t = j\right) \; P\left(\boldsymbol{\xi}_{t+1} \in B \,|\, K_{t+1} = k\right). \end{aligned}$$

[1] A bull market refers to high returns while a bear market is associated low returns; the regular state indicates indicates that the market is neither in a bear nor in a bull state

The feasibility set in each time period consists of $\mathcal{F}_t$-adapted solutions $\mathbf{x}_t \in \mathbb{R}^n$ such that $\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$, where the set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ is given by linear inequalities which may involve $\mathbf{x}_{t-1}$ and $\boldsymbol{\xi}_t$. We will detail the formulation of the model shortly.

2.1 Predictive method: The Hidden Markov Model framework

Following standard practice in the literature, given historical data of the underlying stochastic process $\{\boldsymbol{\xi}_t\}$, the database is split into three parts: training, validation and test datasets. The training data are used to estimate the parameters of the HMM; for instance, the standard EM (expectation-maximization) algorithm [26] can be used to estimate means, variances, and covariances from data, as well as the nominal transition probability matrix. The validation data are used to tune some parameters of the optimization model, as described later in the paper. The final algorithm, with the tuned parameters, is then applied to the testing data.

In general, the HMM parameters estimated from data are the transition probabilities $\hat{p}_j(k) = P\left(K_{t+1} = k \mid K_t = j\right), \forall t = 0, \ldots, T-1$, and the coefficients Θ associated with conditional density function $p(\xi_t \mid K_t = k\,; \Theta)$. As the name suggests, the expectation-maximization (EM) estimation algorithm is an iterative procedure composed by two steps. Given initial values for $\hat{p}$ and Θ, the expectation step computes the probability of occurrence of Markov state j at time t given the uncertainty-realization trajectory $\boldsymbol{\xi}_{[1,T]} = \{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T\}$ and the coefficients Θ, i.e., $P(K_t = j \mid \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_T; \Theta), \forall t = 1, \ldots, T, j \in \mathcal{K}$. Then, the maximization step updates the values of the transition probability $\hat{p}$ and the coefficients Θ in order to maximize the likelihood of the observed data under the assumed hidden Markov structure. These two steps are repeated until a desired level of convergence is achieved.

Once the parameters are estimated, the HMM can then be used to classify the current state j of the Markov chain and therefore the (conditional) distribution of stochastic process $\{\boldsymbol{\xi}_t\}$, which due to Assumption 2 depends only on j. The current state classification is obtained by conducting statistical inference of the current state given all information so far. More specifically, we obtain from the HMM parameters the quantity $P\left(K_t = k \mid \boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_1\right), \forall k \in \mathcal{K}$, hereinafter referred to as the posterior probability of state k at time t, which is defined as

$$P\left(K_t = k \mid \boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_1\right) \;:=\; \frac{p(K_t = k, \boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_1)}{\sum_{j \in \mathcal{K}} p(K_t = j, \boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_1)}, \tag{1}$$

where $p(\boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_1, K_t = k)$ is the joint probability density function evaluated at the observed sample path and the current Markov state being k. This joint probability density function is obtained by an iterative procedure called the HMM forward pass, see [6]. Now, we can classify the current state

$$k_t^* \in \underset{k \in \mathcal{K}}{\arg\max}\; P\left(K_t = k \mid \boldsymbol{\xi}_t, \boldsymbol{\xi}_{t-1}, \ldots, \boldsymbol{\xi}_1\right) \tag{2}$$

as the most probable Markov state given all available information at time t.

2.2 Basic prescriptive method: Risk-constrained dynamic stochastic programming

In addition to the aforementioned constraints $\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$, we also consider risk-based constraints in our optimization model. The introduction of such constraints allows us to control the risk associated with decisions. The incorporation of risk control in dynamic stochastic optimization models has been the subject of considerable work in the literature since issues such as time consistency must be taken into account; see, e.g., [42, 37] for discussions. Following [37], "a policy is time consistent if and only if the future planned decisions are actually going to be implemented." As we shall see shortly, our risk-constrained model satisfies time consistency as it can be formulated in a recursive manner. This dynamic stochastic programming model will then be extended to a distributionally robust dynamic model in Section 2.3.

Before describing the risk-constrained dynamic model, we briefly recall the notion of *Conditional Value-at-Risk* (CV@R) risk measure defined in [35]. Given a random variable Z representing some quantity such that larger values are less favorable (for instance, losses), we write $\text{CV@R}_\beta[Z] = \min_{\eta\in\mathbb{R}} \left\{\eta + \frac{1}{1-\beta}\mathbb{E}\left[(Z-\eta)_+\right]\right\}$. This risk measure is concentrated on the right tail of the distribution of Z. When the random variable of interest is such that larger values are more favorable, then it is more appropriate to refer to acceptability functionals rather than to risk measures (see, e.g., [36]). We call this variable of interest "wealth" and denote it by W. Also, we denote by $\phi_\alpha[W]$ the acceptability functional corresponding to CV@R, i.e., $\phi_\alpha[W] := -\text{CV@R}_{1-\alpha}[-W]$, which can be written as (we omit the subscript α as it is fixed throughout the paper)[2]

$$\phi[W] \;=\; \max_{z\in\mathbb{R}} \left\{ z - \frac{1}{\alpha}\mathbb{E}\left[(z-W)_+\right] \right\}. \tag{3}$$

We return now to our model. In each period t, given a feasible solution $\mathbf{x}_t \in \mathbb{R}^n$ we define a function $g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})$ to represent the "wealth" resulting from the decision $\mathbf{x}_t$. Note that g_t depends on the random variable $\boldsymbol{\xi}_{t+1}$ which has not been realized yet at time t, but it does *not* depend on future values $\boldsymbol{\xi}_{t+2}, \ldots, \boldsymbol{\xi}_T$. To simplify the notation, let $W_{t+1} := g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})$. Then, we apply the acceptability functional ϕ defined in (3) to W_{t+1} conditionally on the current state j of the HMM, resulting in the quantity

$$\phi_{\hat{\mathbf{p}}_j}[W_{t+1}] \;:=\; \max_{z\in\mathbb{R}} \left\{ z - \frac{1}{\alpha} \sum_{k\in\mathcal{K}} \mathbb{E}\left[(z-W_{t+1})_+ \mid K_{t+1}=k\right] \hat{p}_j(k) \right\}. \tag{4}$$

Note that the expectation in (4) and hereinafter is associated with the subsequent random variable $\boldsymbol{\xi}_{t+1}$. With this notation, the risk-based constraint can

[2] Because of the direct one-to-one relationship between ϕ and $\text{CV@R}_{1-\alpha}$, in the paper we will often refer to ϕ as "CV@R", with the meaning understood from the context.

then be expressed as

$$\phi_{\hat{\mathbf{p}}_j}\left[g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right] \geq 0.$$

Finally, in each period t, given a feasible solution $\mathbf{x}_t \in \mathbb{R}^n$ and a realization of $\boldsymbol{\xi}_t$, a reward of $f_t(\mathbf{x}_t, \boldsymbol{\xi}_t)$ is accrued. For each state j of the HMM, the optimization model is then written as

$$Q_0^j := \max_{\mathbf{x}_0 \in \mathcal{X}_0} \quad f_0(\mathbf{x}_0) + \sum_{k \in \mathcal{K}} \mathbb{E}\left[Q_1^k(\mathbf{x}_0, \boldsymbol{\xi}_1) \middle| K_1 = k\right] \hat{p}_j(k) \tag{5}$$

$$\text{s.t.} \quad \phi_{\hat{\mathbf{p}}_j}\left[g_0(\mathbf{x}_0, \boldsymbol{\xi}_1)\right] \geq 0. \tag{6}$$

where $\mathcal{X}_0$ represents linear constraints on $\mathbf{x}_0$ and, for each $t = 1, \ldots, T-1$ and for each state j of the HMM,

$$Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) :=$$

$$\max_{\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)} f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) + \sum_{k \in \mathcal{K}} \mathbb{E}\left[Q_{t+1}^k(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}) \middle| K_{t+1} = k\right] \hat{p}_j(k) \tag{7}$$

$$\text{s.t.} \quad \phi_{\hat{\mathbf{p}}_j}\left[g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right] \geq 0. \tag{8}$$

The final-stage function Q_T is defined as

$$Q_T^j(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T) := \max_{\mathbf{x}_T \in \mathcal{X}_T(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T)} f_T(\mathbf{x}_T, \boldsymbol{\xi}_T). \tag{9}$$

Equations (5)-(9) define the risk-averse dynamic stochastic optimization we would like to solve. It is crucial to notice that, since the risk function is applied only locally in each period through the constraints $\phi_{\hat{\mathbf{p}}_j}\left[g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right] \geq 0$, time consistency is ensured (see [46]) — after all, the problem still has a nested form, which is a necessary condition for time-consistency (see, e.g., [42, 19]). Such constraints could be used, for example, to provide a convex approximation of probabilistic constraints of the form $\mathbb{P}\left(g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}) \leq 0\right) \leq \alpha$.

Observe that the policy resulting from solving the above recursive formulation requires the decision-maker to know the current state in each period, which contradicts the fact that such states are hidden. One way to circumvent this issue would be to consider the Markov posterior probabilities as state variables of the dynamic stochastic program; however, it would lead to a non-convex problem, which is generally intractable. In particular, the recent work by [10] considers a similar problem class with no risk constraints. The authors explore the saddle function structure and provide an efficient solution algorithm for that problem class. However, the solution methodology proposed by the authors is not suitable for our risk-constrained dynamic stochastic optimization, nor do they consider a DRO formulation. As discussed in Section 4, we circumvent that problem by applying a rolling horizon scheme and propose two alternative ways of using HMM posterior probability given in equation (1) to obtain the first-stage decisions: (i) by using the HMM classification techniques, i.e., by taking the most likely Markov state given by equation (2) to determine the current Markov state; and (ii) by modifying the first-stage problem by replacing the transition probability (conditioned to knowing the

current state) with the posterior probability (1), which is a function only of historical return realizations.

We will make the following assumptions for the remainder of the paper (for convenience we set $g_T \equiv 0$):

Assumption 3. *For any $t = 0, \ldots, T$ and any realization of $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{t+1}$, the functions $f_t(\cdot, \boldsymbol{\xi}_t)$ and $g_t(\cdot, \boldsymbol{\xi}_{t+1})$ are* affine.

Assumption 3 can be relaxed; for example, to the case where f_t and g_t are defined as the minimum of affine functions. Nevertheless, we keep the linear assumption for simplicity.

Assumption 4. *For any $t = 1, \ldots, T$, the set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ is non-empty for any $\mathbf{x}_{t-1}$ feasible in stage $t-1$, i.e., the problem has relatively complete recourse. The set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ consists of vectors $\mathbf{x}_t \in \mathbb{R}^n$ satisfying linear inequalities of the form $\mathbf{A}_t\mathbf{x}_t = \mathbf{b}_t - \mathbf{B}_t\mathbf{x}_{t-1}$ with $\mathbf{x}_t \geq 0$, where $\boldsymbol{\xi}_t = \{\mathbf{A}_t, \mathbf{b}_t, \mathbf{B}_t\}$. Also, the set $\mathcal{X}_0$ has the form $\{\mathbf{x}_0 \in \mathbb{R}^N_+ | \mathbf{A}_0\mathbf{x}_0 = \mathbf{b}_0\}$.*

Assumption 4 imposes a polyhedral structure on the set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$, which will be useful in the developments that follow.

2.3 Extended prescriptive method: A data-driven distributionally robust dynamic model

The HMM approach described in Section 2.1 has the advantage of learning directly from the data. However, HMM estimated probabilities are very sensitive to changes in the training data. Such sensitivity may cause considerable instability in the optimization model, with similar observed data leading to different performances of the corresponding optimal solutions. Moreover, poorly estimated probabilities will likely lead to poor out-of-sample performance of the solutions proposed by the model. As stated in Section 1, our primary goal is to provide a data-driven approach for dynamic stochastic optimization problems which performs well out of sample. Therefore, it is critical to address this estimation shortcoming.

Our approach to circumvent the estimation issue is to use a *distributionally robust optimization* (DRO) model for the problem. The idea of DRO is to construct an ambiguity set for the distributions of the random variables of the problem and then to optimize the worst-case within the ambiguity set. DRO problems have long been studied in the literature (albeit with a different terminology), starting from the seminal work of [39], then followed by [48] and later by [43], [41] and [17]. Much of the recent literature on this topic focuses on ways of constructing the ambiguity set (call it $\mathcal{P}$) that ensure tractability of the resulting problem. For example, in [8] the authors define $\mathcal{P}$ as the set of distributions that have a given mean and covariance matrix. Another popular approach is to define $\mathcal{P}$ as the set of distributions that are not "too far" from some reference distribution. Of course, such a notion requires defining

an appropriate way to measure the "distance" between distributions[3]. Several such distances exist, for instance, the Kantorovich and Wasserstein distances, the Kullback-Leibler divergence, and Chi-squared distance (and more generally *phi*-divergences), among others. This is a growing field with substantial current activity; we refer to [28, 4, 3, 25, 34] for some of the work in this area. The benefit of using DRO—under certain settings—to improve out-of-sample performance can be formally demonstrated; see [47].

Some recent works in the literature are directly related to the present paper as they also study DRO models for multistage stochastic programs: in [32] and [12], the authors use ambiguity sets differing in terms of the probability metrics used—respectively χ^2 and Wasserstein distances—and in both cases an adaptation of SDDP is provided to solve the resulting problem. These works assume ambiguity over the nominal stagewise independent probability distribution. Our work, on the other hand, uses the time dependence structure rendered by the HMM, but robustifies the optimal policy against ambiguity over the estimated transition matrix. Dealing with ambiguity only in the transition matrix is helpful since the number of points in the support (which is the number of states in the HMM) is small and thus the dimension of the DRO model is not very large. Moreover, we provide *deterministic both* lower and upper bounds for the objective function, as discussed in the subsequent sections, whereas those aforementioned works only provide lower bounds (for minimization problems).

In our DRO model, equations (7)-(8) are replaced with the following:

$$Q_t^j(\mathbf{x}_{t-1},\boldsymbol{\xi}_t) \;=$$

$$\max_{\mathbf{x}_t\in\mathcal{X}_t(\mathbf{x}_{t-1},\boldsymbol{\xi}_t)} f_t(\mathbf{x}_t,\boldsymbol{\xi}_t) + \left\{ \min_{\mathbf{p}_j\in\,\mathcal{P}_j} \sum_{k\in\mathcal{K}} \mathbb{E}\left[Q_{t+1}^k\big(\mathbf{x}_t,\boldsymbol{\xi}_{t+1}\big)\big|K_{t+1}=k\right] p_j(k) \right\} \quad (10)$$

$$\text{s.t.} \qquad \left\{ \min_{\mathbf{p}_j\in\,\mathcal{P}_j} \phi_{\mathbf{p}_j}\left[g_t(\mathbf{x}_t,\boldsymbol{\xi}_{t+1})\right] \right\} \geq 0 \quad (11)$$

and equations (5) and (6) are also replaced accordingly. In the above model, $\mathcal{P}_j$ is the ambiguity set for the distribution of the next state of the Markov chain, conditionally on the current state j, and is defined as

$$\mathcal{P}_j = \left\{ \mathbf{p}_j \in \mathbb{R}^{|\mathcal{K}|} \,\middle|\, \mathbf{1}^\top \mathbf{p}_j = 1,\ d(\mathbf{p}_j,\hat{\mathbf{p}}_j) \leq \Delta,\ \mathbf{p}_j \geq 0 \right\}, \quad (12)$$

where $d(\mathbf{p}_j,\hat{\mathbf{p}}_j)$ measures the total variation distance between $\mathbf{p}_j$ and $\hat{\mathbf{p}}_j$ (recall that $\hat{\mathbf{p}}_j$ is the vector of state-j probabilities estimated for the Markov chain), i.e.,

$$d(\mathbf{p}_j,\hat{\mathbf{p}}_j) := (1/2)\,\mathbf{1}^\top |\mathbf{p}_j - \hat{\mathbf{p}}_j|. \quad (13)$$

In the above formulation, the parameter Δ controls the level of ambiguity allowed in the model — a value of $\Delta = 0$ indicates that the estimated probabilities can be fully trusted, whereas a value of $\Delta = 1$ ignores the estimated

[3] Note that here we use "distance" as an abuse of terminology, since in some of these cases the function is not symmetric, i.e., $d(P,Q) \neq d(Q,P)$.

probabilities and simply optimizes with respect to the worst-case state of the HMM. Note that the use of ambiguity sets in the DRO formulation addresses model misspecification and estimation errors in the HMM transition probabilities, whereas the use of CV@R aims at measuring the risk of losses with respect to the scenarios. Therefore, there is no redundancy in using both techniques.

The use of a distributionally robust model for the transition probabilities of the Markov chain affects both the objective function and the CV@R constraint — note that the expression $\min_{\mathbf{p}_j \in \mathcal{P}_j} \{\cdot\}$ appears in both places. Using the same worst-case probability distribution for both the objective function and the CV@R constraint makes the separation between the inner and outer problems impossible [28]. Nevertheless, if we allow for two separated worst-case probability distributions (one for the objective function and other for the constraint), the problem becomes more tractable. This "separation" can be conceptually motivated by the idea that the objective function is a "worst case" expectation while the risk constraint must be feasible for any transition probability in the ambiguity set, i.e., $\phi_{\mathbf{p}_j}\left[g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right] \geq 0,\ \forall \mathbf{p}_j \in \mathcal{P}_j$. For convenience, we will use the same ambiguity set for both the objective function and the constraint, however, one could use different sets (for instance, defined with different Δs) to allow for different levels of robustness in each expression. Observe also that, while other probability distance functions could be used instead of the total variation distance in (13), the choice for the total variation is natural in this setting where there are only a finite number of Markov states. Moreover, as we shall see later, with the total variation distance the model can be efficiently solved because the robust counterpart is a linear optimization problem.

3 Solution methodology

In this section we propose an efficient way to solve the data-driven distributionally robust dynamic model: we develop a computationally tractable dual reformulation, and then we adapt the Stochastic Dual Dynamic Programming (SDDP) algorithm to suit the proposed model. Significant modifications are needed in SDDP. In particular, we highlight the development of a deterministic lower bound (for a maximization problem), which, while related to results recently proposed in the literature, is a novel result with a practical appeal. In the following subsections we describe these steps in detail.

3.1 A tractable dual reformulation

In this section, we present a tractable formulation of (10)-(11) based on the dual of the inner minimization problems in those equations. Consider initially the inner problem in (10). For a fixed $\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$, the dual formulation

of the inner problem (10) is

$$\begin{aligned}
\max_{\boldsymbol{\theta}^-,\boldsymbol{\theta}^+,\lambda,\eta} \quad & \sum_{k\in\mathcal{K}} \hat{p}_j(k)(\theta_k^+ - \theta_k^-) - \eta - 2\Delta\lambda \qquad (14)\\
\text{s.t.} \quad & -\theta_k^- + \theta_k^+ - \eta \le \mathbb{E}\left[Q_{t+1}^k(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}) \mid K_{t+1} = k\right], \quad \forall k \in \mathcal{K}\\
& \theta_k^- + \theta_k^+ - \lambda = 0, \quad \forall k \in \mathcal{K}\\
& \boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \lambda \ge 0.
\end{aligned}$$

By using a similar approach it is possible to construct a dual formulation to write (11) in a more tractable manner. Note that from (4) we can write

$$\phi_{\mathbf{p}_j}\left[g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right] = \max_{z\in\mathbb{R}} \left\{ h(z, \mathbf{p}_j) := z - \frac{1}{\alpha} \sum_{k\in\mathcal{K}} \mathbb{E}\left[\left(z - g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right)_+ \middle| K_{t+1} = k\right] p_j(k) \right\}.$$

The function h is concave in z and linear in $\mathbf{p}_j$. Thus, given that $\mathcal{P}_j$ is compact, we can deduce from Sion's minimax theorem [45] that $\min_{\mathbf{p}_j\in\mathcal{P}_j} \max_{z\in\mathbb{R}} h(z, \mathbf{p}_j) = \max_{z\in\mathbb{R}} \min_{\mathbf{p}_j\in\mathcal{P}_j} h(z, \mathbf{p}_j)$ and hence it follows that for any given z, $\min_{\mathbf{p}_j\in\mathcal{P}_j} h(z, \mathbf{p}_j)$ can be written as

$$\begin{aligned}
\min_{\mathbf{p}_j\ge 0, \mathbf{e}} \quad & z - \frac{1}{\alpha}\mathbb{E}\left[\left(z - g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right)_+ \middle| K_{t+1} = k\right] p_j(k) \qquad (15)\\
\text{s.t.} \quad & p_j(k) - e_k \le \hat{p}_j(k), \quad \forall k \in \mathcal{K} && : \widetilde{\theta}_k^-\\
& p_j(k) + e_k \ge \hat{p}_j(k), \quad \forall k \in \mathcal{K} && : \widetilde{\theta}_k^+\\
& \textstyle\sum_{k\in\mathcal{K}} e_k \le 2\Delta && : \widetilde{\lambda}\\
& \textstyle\sum_{k\in\mathcal{K}} p_j(k) = 1 && : \widetilde{\eta}
\end{aligned}$$

By writing the dual of (15) analogously to (14), it follows that the left-hand side of (11) can be written as the optimization problem

$$\begin{aligned}
\max_{z,\widetilde{\boldsymbol{\theta}}^-,\widetilde{\boldsymbol{\theta}}^+,\widetilde{\lambda},\widetilde{\eta}} \quad & z + \sum_{k\in\mathcal{K}} \hat{p}_j(k)(\widetilde{\theta}_k^+ - \widetilde{\theta}_k^-) - \widetilde{\eta} - 2\Delta\widetilde{\lambda} \qquad (16)\\
\text{s.t.} \quad & -\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\eta} \le -\frac{1}{\alpha}\mathbb{E}\left[\left(z - g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\right)_+ \middle| K_{t+1} = k\right], \quad \forall k \in \mathcal{K}\\
& \widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\lambda} = 0, \quad \forall k \in \mathcal{K}\\
& \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \widetilde{\lambda} \ge 0.
\end{aligned}$$

Finally, by merging (14) with the outer maximization problem, adding the inequalities and variables of (16), and imposing that the objective function of

(16) is non-negative, we obtain the single-level reformulation of (10)-(11):

$$
\begin{aligned}
& Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) \;=\; \\
& \max_{\substack{\mathbf{x}_t, z, \boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \lambda, \\ \eta, \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \widetilde{\lambda}, \widetilde{\eta}}} \quad f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) + \sum_{k \in \mathcal{K}} \hat{p}_j(k)(\theta_k^+ - \theta_k^-) - \eta - 2\Delta\lambda && (17) \\
\text{s.t.} \;\; & z + \sum_{k \in \mathcal{K}} \hat{p}_j(k)(\widetilde{\theta}_k^+ - \widetilde{\theta}_k^-) - \widetilde{\eta} - 2\Delta\widetilde{\lambda} \geq 0 && (18) \\
& -\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\eta} + \frac{1}{\alpha}\mathbb{E}\left[\big(z - g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})\big)_+ \Big| K_{t+1} = k\right] \leq 0, \;\; \forall k \in \mathcal{K} && (19) \\
& -\theta_k^- + \theta_k^+ - \eta - \mathbb{E}\left[Q_{t+1}^k\big(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}\big) \mid K_{t+1} = k\right] \leq 0, \;\; \forall k \in \mathcal{K} && (20) \\
& \theta_k^- + \theta_k^+ - \lambda = 0, \;\; \forall k \in \mathcal{K} && (21) \\
& \widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\lambda} = 0, \;\; \forall k \in \mathcal{K} && (22) \\
& \boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \lambda, \widetilde{\lambda} \geq 0 && (23) \\
& \mathbf{x}_t \in \; \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t). && (24)
\end{aligned}
$$

Problem (17)-(24) is a multistage stochastic program, which is convex under Assumptions 3 and 4. It provides a conceptually tractable reformulation of (10)-(11). The word "conceptually" refers to the fact that such model cannot be directly implemented, for two reasons: first, inequalities (19) and (20) involve expectations and second, inequality (20) involves the unknown value function $Q_{t+1}^k\big(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}\big)$. The first difficulty can be dealt with employing sample average approximations. The second difficulty appears more daunting due to the curse of dimensionality. For instance, when no assumptions are made about the input process $\{\boldsymbol{\xi}_t\}$ there is a vast number of possible outcomes at each stage and the number of scenarios grows exponentially with the number of stages. As we shall see in Section 3.2, however, under Assumption 2 we can adapt the SDDP method to our setting, which allows us to approximate the value function $Q_{t+1}^k\big(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}\big)$ by piecewise-linear functions and so standard optimization methods can be used to solve the problem.

We can construct a sample average approximation of problem (17)-(24), which allows us to replace the expectations in (19) and (20) with averages of random realizations sampled from the "true" distributions. First, for each state $k \in \mathcal{K}$ of the Markov chain, we draw i.i.d. samples from the conditional distribution of $\boldsymbol{\xi}_{t+1}$ given $K_{t+1} = k$. We denote those samples by $\{\boldsymbol{\xi}_{t+1}^k(s)\}_{s \in \mathcal{S}_k}$. Next, define the probability $q_k(s)$ of scenario s conditional on state k of the Markov chain as

$$
q_k(s) \;:=\; \mathbb{P}\left(\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_{t+1}^k(s) \mid K_{t+1} = k\right).
$$

For instance, if the sample is generated via a Monte Carlo simulation or Latin Hypercube Sampling, then we would define equally probable scenarios $q_k(s) = 1/|\mathcal{S}_k|$, conditionally on the Markov state $K_{t+1} = k$. However, $q_k(s)$ might be defined differently if other technique, such as importance sampling, is used. Moreover, we introduce variables y_{ks} for each $k \in \mathcal{K}$ and each $s \in \mathcal{S}_k$ in order

to linearize the "plus" function in (19). Finally, the expected value function in (20) is expressed as

$$\mathcal{Q}_{t+1}^{k}(\mathbf{x}_t) := \sum_{s \in \mathcal{S}_k} Q_{t+1}^{k}(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}^{k}(s))\, q_k(s). \tag{25}$$

3.2 Modified Stochastic Dual Dynamic Programming algorithm

With a sample approximation of model (17) -(24) at hand, the only remaining issue is dealing with the value function in constraint (20). As discussed earlier, we adapt the SDDP method for this purpose. The SDDP algorithm is mainly characterized by two steps: a forward-in-time simulation and a backward-in-time recursion. The forward step generates trial solutions that are later used in the backward step to construct cutting-plane approximations of the future value function. When there is Markovian dependency, the forward step must generate (i) a path of states of the Markov chain and (ii) sample paths of the process $\{\boldsymbol{\xi}_t\}$ conditionally on each sampled state of the Markov chain. Then, trial solutions are created by solving the problem with the current value function approximations at each stage using the sampled processes. It is important to mention here that, in our context, the forward steps are generated using the *nominal* transition probability matrix given by the HMM; as we shall see in Section 3.4, such a property is crucial to prove convergence of the method. The backward step uses trial solutions and goes in the opposite time direction, from $t = T$ to $t = 1$, adding cuts to improve the outer approximation of the value function. In the context of a maximization problem, we can obtain a deterministic upper bound using the outer approximation generated by the SDDP backward procedure.

We remark that model (17) -(24) is not in the standard form of problems solved by SDDP since the value function appears in the constraint (20) rather in the objective function as customary in the literature. A similar situation arises in the model studied by [31], albeit in a somewhat different context since that paper deals with nested risk measures. Thus, for the sake of completeness, we detail the steps and show how to construct an upper (i.e., outer) approximation for the value function. In Section 3.3 we will discuss how to construct a lower (inner) approximation.

Suppose we are in iteration ν of the algorithm. Consider the sample approximation of problem (17) -(24) with constraint (20) replaced with an upper approximation $\overline{\mathcal{Q}}_{t+1}^{j,\nu}(\mathbf{x}_t)$ given by linear inequalities, and denote the optimal value of the approximated problem by $\widetilde{Q}_t^{j,\nu}(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$. We will detail shortly

how to construct $\overline{\mathcal{Q}}_{t+1}^{j,\nu}(\mathbf{x}_t)$. Then, we have, for $t = T-1$ to $t = 0$,

$$\widetilde{Q}_t^{j,\nu}(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) :=$$

$$\max_{\substack{\mathbf{x}_t, z, \mathbf{y}, \boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \lambda, \\ \eta, \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \widetilde{\lambda}, \widetilde{\eta}, \mathbf{u}}} \quad f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) + \sum_{k \in \mathcal{K}} \hat{p}_j(k)(\theta_k^+ - \theta_k^-) - \eta - 2\Delta\lambda \tag{26}$$

$$\text{s.t.} \quad z + \sum_{k \in \mathcal{K}} \hat{p}_j(k)(\widetilde{\theta}_k^+ - \widetilde{\theta}_k^-) - \widetilde{\eta} - 2\Delta\widetilde{\lambda} \ \geq \ 0 \tag{27}$$

$$-\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\eta} + \sum_{s \in \mathcal{S}_k} y_{ks} \frac{q_k(s)}{\alpha} \ \leq \ 0, \ \forall k \in \mathcal{K} \tag{28}$$

$$-\theta_k^- + \theta_k^+ - \eta - \overline{\mathcal{Q}}_{t+1}^{k,\nu}(\mathbf{x}_t) \ \leq \ 0, \qquad \forall k \in \mathcal{K} \tag{29}$$

$$\theta_k^- + \theta_k^+ - \lambda = 0, \qquad \forall k \in \mathcal{K} \tag{30}$$

$$\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\lambda} = 0, \qquad \forall k \in \mathcal{K} \tag{31}$$

$$z - g_t(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}^k(s)) - y_{ks} \ \leq \ 0, \qquad \forall k \in \mathcal{K}, \forall s \in \mathcal{S}_k \tag{32}$$

$$\mathbf{u} = \mathbf{x}_{t-1} \qquad : \pi_t^j(\boldsymbol{\xi}_t) \tag{33}$$

$$\mathbf{x}_t \in \mathcal{X}_t(\mathbf{u}, \boldsymbol{\xi}_t) \tag{34}$$

$$\boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \mathbf{y}, \lambda, \widetilde{\lambda} \geq 0. \tag{35}$$

For $t = T$ we have, at all iterations ν, the simpler problem

$$\widetilde{Q}_T^{j,\nu}(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T) := \max_{\mathbf{x}_T, \mathbf{u}} \quad f_T(\mathbf{x}_T, \boldsymbol{\xi}_T) \tag{36}$$

$$\text{s.t.} \quad \mathbf{u} = \mathbf{x}_{T-1} \qquad : \pi_T^j(\boldsymbol{\xi}_T) \tag{37}$$

$$\mathbf{x}_T \in \mathcal{X}_T(\mathbf{u}, \boldsymbol{\xi}_T). \tag{38}$$

Note that $\widetilde{Q}_T^{j,\nu}(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T) = Q_T^j(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T)$, i.e., there is no approximation in the last stage.

It is important to clarify the role of the auxiliary variable $\mathbf{u}$ introduced in the problem, which appears in constraints (33), (34), (37), and (38). This variable is just a generic artifact to obtain a subgradient[4] of the function $\widetilde{Q}_t^{j,\nu}(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ with respect to $\mathbf{x}_{t-1}$. From (33) and (37) we see that this subgradient is given by the dual variable $\pi_t^j(\boldsymbol{\xi}_t)$. Let $\hat{\mathbf{x}}_t^\nu$ be the optimal solution of (26)-(35) (and (36)-(38) in the case $t = T$) generated by the forward step in iteration ν for stage t.

In the backward step, we solve (26)-(35) (and (36)-(38) in the case $t = T$) for each $j \in \mathcal{K}$ and each scenario $\boldsymbol{\xi}_t^j = \boldsymbol{\xi}_t^j(s)$, $s \in \mathcal{S}_j$, with $\mathbf{x}_{t-1} = \hat{\mathbf{x}}_{t-1}^\nu$. Let $\pi_{t,s}^{j,\nu} := \pi_t^j(\boldsymbol{\xi}_t(s))$ denote the corresponding dual variable obtained from (33) (and from (37) in the case $t = T$). Then, we construct the Benders cut

$$\ell_t^{j,\nu}(\mathbf{x}_{t-1}) := \widetilde{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^\nu) + \left(\overline{\pi}_t^{j,\nu}\right)^\top (\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1}^\nu) \tag{39}$$

[4] In reality, it is a supergradient since the function is concave, but we will call it a subgradient as this terminology is more common in the literature.

for the function $\widetilde{\mathcal{Q}}_t^{j,\nu}(\cdot) := \sum_{s\in\mathcal{S}_j} \widetilde{Q}_t^{j,\nu}(\cdot, \boldsymbol{\xi}_{t+1}(s))\ q_j(s)$, using the average dual decision vector $\overline{\pi}_t^{j,\nu} = \sum_{s\in\mathcal{S}_j} \pi_{t,s}^{j,\nu}\ q_j(s)$. As discussed earlier, $\pi_{t,s}^{j,\nu}$ is a subgradient of $\widetilde{Q}_t^{j,\nu}(\cdot, \boldsymbol{\xi}_{t+1}(s))$ at $\hat{\mathbf{x}}_{t-1}^{\nu}$ and thus it follows that $\ell_t^{j,\nu}(\mathbf{x}_{t-1}) \geq \widetilde{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1})$ for all $\mathbf{x}_{t-1}$. Moreover, since the function $\mathcal{Q}_{t+1}^{k}(\mathbf{x}_t)$ in (20) is replaced with the upper approximation $\overline{\mathcal{Q}}_{t+1}^{k,\nu}(\mathbf{x}_t)$ in (29), it follows that $\widetilde{Q}_t^{j,\nu}(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{t+1}(s)) \geq Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{t+1}(s))$ for all $s \in \mathcal{S}_j$ and thus $\widetilde{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1}) \geq \mathcal{Q}_t^j(\mathbf{x}_{t-1})$.

We then update the upper approximation of the value function (so it can be used in period $t-1$) as

$$\overline{\mathcal{Q}}_t^{k,\nu}(\mathbf{x}_{t-1}) := \min_{i=1,\ldots,\nu} \ell_t^{k,i}(\mathbf{x}_{t-1}), \quad \forall k \in \mathcal{K}, \tag{40}$$

so we see that $\overline{\mathcal{Q}}_t^{k,\nu}(\mathbf{x}_{t-1}) \geq \widetilde{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1}) \geq \mathcal{Q}_t^j(\mathbf{x}_{t-1})$ for all $\mathbf{x}_{t-1}$. It follows that when we solve (26)-(35) in period $t-1$, by using (40) in constraint (29) we have the equivalent to the set of linear constraints

$$-\theta_k^- + \theta_k^+ - \eta\ - \ell_{t+1}^{k,i}(\mathbf{x}_t)\ \leq\ 0, \quad \forall k \in \mathcal{K}, \forall i = 1, \ldots, \nu. \tag{41}$$

Therefore, the outer approximation of the SAA problem can be represented as model (26)-(35), with constraint (29) replaced with inequalities (41). Note also that, because of Assumption 4, the constraints given by (34) are linear in $(\mathbf{x}_t, \mathbf{u})$. Thus, since $f_t(\cdot, \boldsymbol{\xi}_t)$ and $g_t(\cdot, \boldsymbol{\xi}_{t+1})$ are linear by Assumption 3, model (26)-(35) is just a linear program.

3.3 Deterministic lower bound

For standard SDDP applications, one can obtain a statistical lower bound by evaluating the current policy via Monte Carlo simulation and compute an estimator of the objective function, see details in [40]. However, it is not practical to obtain a statistical objective function assessment within the distributionally robust framework (10)-(11). The issue here is that, in order to evaluate the objective function in (10), we would need to know the optimal worst-case transition probability matrix in the corresponding inner problem, but this is not possible since we only have an approximation of value function Q_{t+1}^k. Thus, if we simulate scenarios using any (suboptimal) transition probability matrix, the statistical evaluation of the objective function will not be a valid lower bound.

Our approach is to explore an extended inner approximation of $Q_{t+1}^k(\cdot)$ to construct a valid lower bound to problem (17)-(24) . The standard inner approximation method uses a convex combination of evaluated trial points instead of the Benders cuts (outer approximation). This approach was first proposed by [29] who ensured the feasibility of the convex combination by pre-evaluating all vertices of the uncertainty support, e.g., a multidimensional hypercube. The contribution by [29] notwithstanding, the approach proposed

in that work is not efficient in practice since the number of vertices grows exponentially with the uncertainty dimension.

Consider the expected value function $\mathcal{Q}^j_{t+1}(\mathbf{x}_t)$ defined in (25), and suppose that in iteration ν of the algorithm we have a concave lower (inner) approximation $\underline{\mathcal{Q}}^{j,\nu}_{t+1}(\cdot)$ for $\mathcal{Q}^j_{t+1}(\cdot)$ given by linear inequalities. Let $\{\hat{\mathbf{x}}^i_t\}_{i=1,\ldots,\nu}$ denote the solutions obtained for each time $t \in \{1,\ldots,T\}$ from the previous ν forward steps of the algorithm. As in the case of the outer approximation discussed in Section 3.2, the algorithm goes backwards in time, from $t = T$ until $t = 0$, and $\underline{\mathcal{Q}}^{j,\nu}_t$ is constructed from $\underline{\mathcal{Q}}^{j,\nu}_{t+1}$. For $t = T$ we set $\underline{\mathcal{Q}}^{j,\nu}_T(\mathbf{x}_{T-1}) := \sum_{s\in\mathcal{S}_j} Q^j_T(\mathbf{x}_{T-1}, \boldsymbol{\xi}^j_T(s))\, q_j(s)$ in all iterations ν, where $Q^j_T(\cdot)$ is defined in (9). Let $\mathcal{R}$ denote the set of points satisfying constraint (11), and define, for $t < T$,

$$\widehat{Q}^{j,\nu}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) := \max_{\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1},\boldsymbol{\xi}_t)\cap\mathcal{R}} f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) + \left\{ \min_{\mathbf{p}_j \in \mathcal{P}_j} \sum_{k\in\mathcal{K}} \underline{\mathcal{Q}}^{k,\nu}_{t+1}(\mathbf{x}_t)\, p_j(k) \right\} \tag{42}$$

and $\widehat{\mathcal{Q}}^{j,\nu}_t(\mathbf{x}_{t-1}) := \sum_{s\in\mathcal{S}_j} \widehat{Q}^{j,\nu}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t(s))\, q_j(s)$. Note that, since $\underline{\mathcal{Q}}^{j,\nu}_{t+1}(\cdot)$ is piecewise-linear concave, it follows from Assumptions 3 and 4 that $\widehat{Q}^{j,\nu}_t(\cdot, \boldsymbol{\xi}_t)$ and $\widehat{\mathcal{Q}}^{j,\nu}_t(\cdot)$ are also piecewise-linear concave. Moreover, since $\underline{\mathcal{Q}}^{j,\nu}_{t+1}(\cdot)$ is a lower bound for $\mathcal{Q}^j_{t+1}(\cdot)$, by comparing (10)-(11) and (42) we see that $\widehat{Q}^{j,\nu}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) \leq Q^j_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ and thus it follows that

$$\widehat{\mathcal{Q}}^{j,\nu}_t(\mathbf{x}_{t-1}) \leq \mathcal{Q}^j_t(\mathbf{x}_{t-1}). \tag{43}$$

Consider now the function $\underline{\mathcal{Q}}^{j,\nu}_t(\mathbf{x}_{t-1})$ defined as

$$\begin{aligned} \underline{\mathcal{Q}}^{j,\nu}_t(\mathbf{x}_{t-1}) \;= \max_{\mathbf{x}',\boldsymbol{\mu}} \quad & \sum_{i=1}^{\nu} \mu_i\, \widehat{\mathcal{Q}}^{j,\nu}_t(\hat{\mathbf{x}}^i_{t-1}) - L\,\|\mathbf{x}'\|_1 \\ \text{s.t.} \quad & \sum_{i=1}^{\nu} \mu_i \hat{\mathbf{x}}^i_{t-1} + \mathbf{x}' = \mathbf{x}_{t-1} \\ & \sum_{i=1}^{\nu} \mu_i = 1 \\ & \boldsymbol{\mu} \geq 0 \end{aligned} \tag{44}$$

where L is a Lipschitz constant for $\mathcal{Q}^j_t(\cdot)$ under the 1-norm. Proposition 5 shows that $\underline{\mathcal{Q}}^{j,\nu}_t(\cdot)$ is indeed a valid lower bound for $\mathcal{Q}^j_{t+1}(\cdot)$.

Proposition 5. *The function $\underline{\mathcal{Q}}^{j,\nu}_t(\cdot)$ defined in (44) is a piecewise-linear concave lower bound for $\mathcal{Q}^j_t(\cdot)$, whenever L is a Lipschitz constant for $\mathcal{Q}^j_t(\cdot)$ under the 1-norm.*

Proof. For $t = T$ we have $\underline{\mathcal{Q}}^{j,\nu}_T(\mathbf{x}_{T-1}) = \mathcal{Q}^j_T(\mathbf{x}_{T-1})$ by definition and so the statement is true. Suppose $t < T$. Define now a function $\overline{\mathcal{Q}}^j_t(\mathbf{x}_{t-1})$ similarly

to (44), but with the function $\widehat{\mathcal{Q}}_t^{j,\nu}$ replaced by the true value function $\mathcal{Q}_t^j$. From (43), it is clear that $\underline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1}) \leq \overline{\mathcal{Q}}_t^j(\mathbf{x}_{t-1})$. Thus, it suffices to show that $\overline{\mathcal{Q}}_t^j(\cdot)$ is lower bound for $\mathcal{Q}_{t+1}^j(\cdot)$. Let $(\boldsymbol{\mu}^*, \mathbf{x}^{*'})$ be an optimal solution to problem defining $\overline{\mathcal{Q}}_t^j(\mathbf{x}_{t-1})$. Define the quantity $\mathbf{x}^{*''} := \sum_{i=1}^{\nu} \mu_i^* \hat{\mathbf{x}}_{t-1}^i$, so we see that $\mathbf{x}_{t-1} = \mathbf{x}^{*'} + \mathbf{x}^{*''}$. Since $\mathcal{Q}_t^j(\cdot)$ is concave, we have that

$$\sum_{i=1}^{\nu} \mu_i^* \, \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^i) \;\leq\; \mathcal{Q}_t^j(\mathbf{x}^{*''}) \;\leq\; \mathcal{Q}_t^j(\mathbf{x}_{t-1}) + \zeta_{\mathbf{x}_{t-1}}^{\top}(\mathbf{x}^{*''} - \mathbf{x}_{t-1}), \tag{45}$$

where $\zeta_{\mathbf{x}_{t-1}}$ is any subgradient of $\mathcal{Q}_t^j(\cdot)$ at $\mathbf{x}_{t-1}$. It follows from the right-most inequality in (45) that

$$\begin{aligned}
\mathcal{Q}_t^j(\mathbf{x}_{t-1}) \;&\geq\; \mathcal{Q}_t^j(\mathbf{x}^{*''}) - \zeta_{\mathbf{x}_{t-1}}^{\top}(\mathbf{x}^{*''} - \mathbf{x}_{t-1}) \\
&\geq\; \mathcal{Q}_t^j(\mathbf{x}^{*''}) - |\zeta_{\mathbf{x}_{t-1}}^{\top}(\mathbf{x}^{*''} - \mathbf{x}_{t-1})| \\
&\geq\; \mathcal{Q}_t^j(\mathbf{x}^{*''}) - \|\zeta_{\mathbf{x}_{t-1}}\|_2 \, \|(\mathbf{x}^{*''} - \mathbf{x}_{t-1})\|_2 && (46) \\
&\geq\; \mathcal{Q}_t^j(\mathbf{x}^{*''}) - \|\zeta_{\mathbf{x}_{t-1}}\|_1 \, \|(\mathbf{x}^{*''} - \mathbf{x}_{t-1})\|_1 && (47) \\
&\geq\; \mathcal{Q}_t^j(\mathbf{x}^{*''}) - L \, \|\mathbf{x}^{*'}\|_1 && (48) \\
&\geq\; \sum_{i=1}^{\nu} \mu_i^* \, \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^i) - L \, \|\mathbf{x}^{*'}\|_1 && (49) \\
&=\; \overline{\mathcal{Q}}_t^j(\mathbf{x}_{t-1}). && (50)
\end{aligned}$$

The inequality in (46) in application of the Cauchy-Schwarz inequality, whereas the inequality in (47) follows from the well-known fact that $\|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1$ for any vector $\mathbf{x}$. Inequality (48) follows from the assumption that L is a Lipschitz constant for $\mathcal{Q}_t^j(\cdot)$ under the 1-norm and therefore the 1-norm of any subgradient of $\mathcal{Q}_t^j(\cdot)$ is bounded above by L. Finally, the inequality in (49) follows from the left-most inequality in (45), and (50) is the definition of $\overline{\mathcal{Q}}_t^j(\mathbf{x}_{t-1})$.

Consider again the function $\underline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1})$ defined in (44). As discussed earlier the function $\widehat{\mathcal{Q}}_t^{j,\nu}(\cdot)$ is piecewise-linear concave, and the function $-L\|\mathbf{x}\|_1$ is piecewise-linear concave as well. It follows that the function $\underline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1})$ defined in (44) is also piecewise-linear concave. □

Problem (44) enhances the formulation proposed by [29] in that it allows for the evaluation of the lower bound function at points that are not in the convex hull of the points previously generated by the algorithm, thereby avoiding the enumeration of the vertices of the uncertainty support as proposed in that work. Moreover, it is important to observe that the approach can be easily adapted to other settings such as the standard SDDP, or the SDDP with *nested* risk measures that can be linearized, such as CV@R. In the case of nested risk measures, the difficulty to obtain valid lower *and* upper bounds has long been recognized in the literature (see, e.g., [31, 44]).

We must also mention that the dual formulation of problem (44) can be written as

$$\begin{aligned} \underline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1}) \;= \min_{\psi,\boldsymbol{\zeta}} \quad & \psi + \boldsymbol{\zeta}^\top \mathbf{x}_{t-1} \\ \text{s.t.} \quad & \psi + \boldsymbol{\zeta}^\top \hat{\mathbf{x}}_{t-1}^i \geq \widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^i), \quad \forall i = 1,\ldots,\nu \\ & \|\boldsymbol{\zeta}\|_\infty \leq L, \end{aligned} \tag{51}$$

which corresponds to the lower bound function proposed by [1] (translated into the context of maximization problems). However, we argue that the primal formulation (44) facilitates the interpretation of the Lipschitz constant L since the decision variable $\mathbf{x}'$ can be interpreted as a slack vector which has nonzero components whenever $\mathbf{x}_{t-1}$ does not belong to the convex hull of $\{\hat{\mathbf{x}}_{t-1}^i\}_{i=1,\ldots,\nu}$. The slack vector $\mathbf{x}'$ then appears in the objective function with a sufficiently large penalty L. Such an approach opens the possibility of using other types of penalization of the slack variable $\mathbf{x}'$, which could be problem-dependent but provide tighter bounds. For instance, we explore the specific structure of the dynamic asset allocation problem presented in Section 5 to propose a modified lower bound with a proper penalization of the slack variable that does not require computing a Lipschitz constant. Finally, in order for the present paper to be self-contained we have chosen to provide a proof of Proposition 5 from first principles, applying different proof techniques than those used by [1] and [2].

We close this section by noting that $\widehat{Q}_t^{j,\nu}(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ in (42) can be computed as the solution of a linear program, similarly to (26)-(35) but with $\overline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1})$ in (29) replaced with $\underline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1})$. For more details about the deterministic lower and upper bounds algorithms see Appendix A.

3.4 Convergence

We establish now the convergence of our proposed approach. The following theorem shows that the gap between the deterministic upper and lower bounds becomes zero after finitely many iterations.

Theorem 6. *Consider the modified SDDP algorithm described in Section 3.2, with the upper bound $\overline{\mathcal{Q}}_{t+1}^{k,\nu}(\mathbf{x}_t)$ defined in (40). Consider the lower bound $\underline{\mathcal{Q}}_{t+1}^{k,\nu}(\mathbf{x}_t)$ defined in (44)). Suppose that the the transition probability matrix obtained from HMM is irreducible. Then, at some iteration ν, we have $\overline{\mathcal{Q}}_{t+1}^{k,\nu}(\hat{\mathbf{x}}_t^\nu) = \underline{\mathcal{Q}}_{t+1}^{k,\nu}(\hat{\mathbf{x}}_t^\nu)$ for some feasible solution $\{\hat{\mathbf{x}}_t^\nu\}_{t=0\ldots,T}$.*

Proof. The convergence of the outer approximation follows from the standard proof of convergence of the standard SDDP presented by [30]. In that paper, the authors show that the optimal solutions of the outer approximations converge to an optimal solution of the original problem in finitely many iterations, assuming that every scenario in the problem is eventually sampled in the forward pass. In our context, it follows from Assumptions 3 and 4 that

the objective function of the "true" discretized problem is concave piecewise linear, which is the setting in [30]. Thus, if the transition probability matrix obtained from HMM is irreducible, then it is possible to generate any scenario with nonzero probability and hence the proof of [30] can be applied.

For the inner approximation we can use an inductive step backwards from $t = T$ to $t = 1$. Suppose that, at some iteration ν, an optimal solution $\{\hat{\mathbf{x}}_t^\nu\}_{t=0,\ldots,T}$ of the outer problem is also an optimal solution of the original problem—as discussed above, one such solution is guaranteed to be found based on the arguments of [30]. That is, we have that $\overline{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^\nu) = \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^\nu)$, $t = T, \ldots, 1$. We will show by induction that $\underline{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^\nu) = \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^\nu)$, $t = T, \ldots, 1$, which immediately implies that the gap between upper and lower bounds is equal to zero.

As discussed in the proof of Proposition 5, for $t = T$ we have $\underline{\mathcal{Q}}_T^{j,\nu}(\mathbf{x}_{T-1}) = \mathcal{Q}_T^j(\mathbf{x}_{T-1})$ for all $\mathbf{x}_{T-1}$, so in particular the equality holds at $\mathbf{x}_{T-1} = \hat{\mathbf{x}}_{T-1}^\nu$. Suppose now that it holds for $t+1 \leq T$. That is, we have $\underline{\mathcal{Q}}_{t+1}^{j,\nu}(\mathbf{x}_t) = \mathcal{Q}_{t+1}^j(\mathbf{x}_t)$ for $\mathbf{x}_t = \hat{\mathbf{x}}_t^\nu$ and, from Proposition 5, $\underline{\mathcal{Q}}_{t+1}^{j,\nu}(\mathbf{x}_t) \leq \mathcal{Q}_{t+1}^j(\mathbf{x}_t)$ for $\mathbf{x}_t \neq \hat{\mathbf{x}}_t^\nu$. It follows that $\hat{\mathbf{x}}_t^\nu$ is a maximizer of the problem in (42) when $\widehat{Q}_t^{j,\nu}(\cdot, \boldsymbol{\xi}_t)$ is calculated at $\mathbf{x}_{t-1} = \hat{\mathbf{x}}_{t-1}^\nu$ and thus we have that $\widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^\nu) = \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^\nu)$. Hence, when calculating $\underline{\mathcal{Q}}_t^{k,\nu}(\cdot)$ at $\hat{\mathbf{x}}_{t-1}^\nu$, by concavity of $\mathcal{Q}_t^j(\cdot)$ the maximization problem in (44) puts weight $\mu_\nu = 1$ and thus we have $\underline{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^\nu) = \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^\nu)$. □

4 Assessing out-of-sample performance in a rolling horizon scheme

Most SDDP applications use a rolling horizon scheme to mitigate the end-effect of the terminal time stage. One way to interpret this usage is that the actual problem has an infinite horizon and is approximated by a finite horizon model with many time stages such that the "end of the world" has a small influence on the first stage decision. This is the case for long term energy planning, portfolio selection and asset-liability management problems, to name a few. In this section, we establish a generic out-of-sample evaluation framework and develop an acceleration scheme for the particular case of time-homogeneous models where the parameters of the problem (i.e. the functions $f_t(\mathbf{x}_t, \boldsymbol{\xi}_t) = f(\mathbf{x}_t, \boldsymbol{\xi}_t)$ and $g_t(\mathbf{x}_t, \boldsymbol{\xi}_t) = g(\mathbf{x}_t, \boldsymbol{\xi}_t)$, and the coefficients in the set $\mathcal{X}_t$) do not depend on the time period.

The framework for the rolling horizon scheme in a general setting can be described as follows. Consider a implementation horizon of length H and let $t_1, \ldots, t_H$ denote the times at which the model is solved and the corresponding first-stage optimal solution is implemented. A suitable way to emulate the actual decision making process is to concatenate five steps for a given time $t \in \{t_1, \ldots, t_H\}$: (i) the HMM parameters are estimated via the EM (expectation-maximization) algorithm using as input the sequence of observed uncertainty realization $(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t)$; (ii) an SAA version of the problem is generated as in Section 3.2; (iii) a Markov state classification is performed. A simple classification method can be described as follows: consider the state

with highest posterior probability of occurrence given the historical path of the process $\{\boldsymbol{\xi}_t\}$; then, use that state as the initial one, cf. equation (2). In step (iv), the SDDP algorithm for the problem with T stages is run until convergence (according to Algorithm 1) for problem (17)-(24) assuming a given previous implemented decision $\mathbf{x}_{t-1}$ and the current uncertainty realization $\boldsymbol{\xi}_t$. Note that, in step (iv), the SDDP is "trained" assuming observed Markov states with the current state defined by the HMM classification. In step (v), the first-stage decision $\mathbf{x}_t$ is implemented, the time t is updated and we go to step (i). Note that this procedure is computationally intensive since a SDDP is run until convergence for each time step of the simulation. Finally, for the implementation of optimal policy $\mathbf{x}_t$ in step (v) it is necessary to use a method—step (iii)— to infer the initial state of the Markov chain (recall that such states are not observable).

For the time-homogeneous case, it is appropriate to use only the first stage problem to implement every decision in the rolling horizon scheme. This is motivated by the fact that the problem structure does not depend on the period. In this context, we propose a relatively fast evaluation framework that is divided into two parts: estimation and sampling, and out-of-sample evaluation. In the estimation and sampling part, the training dataset is used as input for the EM algorithm to estimate the HMM parameters, i.e., nominal transition probabilities and conditional probability distributions of the uncertain vector. Those conditional distributions are sampled using Latin Hypercube Sampling (LHS)—which typically performs better than Monte Carlo sampling method, as shown in [18]—to construct the SAA scenario tree. For an out-of-sample evaluation, a rolling horizon scheme is used over the testing dataset to simulate historical (out-of-sample) performance. In essence, we follow three steps for a given time t: (i) the Markov state classification is performed using (2); (ii) a SDDP is run until convergence (again, according to Algorithm 1) for problem (26)-(35) assuming a observable Markov chain, the current state defined by the HMM classifier, a given previous stage decision $\mathbf{x}_{t-1}$ and the current uncertainty realization $\boldsymbol{\xi}_t$; (iii) the first stage decision $\mathbf{x}_t$ is implemented, the time t is updated and we go to step (i). Note that the steps are very similar to the initial decision process laid out earlier; however, in the out-of-sample evaluation, we do not re-estimate the parameters of the HMM, nor do we generate a new SAA version of the model. Thus, the convergence of SDDP in step (ii) should be much faster as it can use the value function approximations constructed in the previous steps as described below.

Given that HMM parameters are fixed, the value function for each state and period remains the same and can be reused over the rolling horizon scheme. However, the value function might not be well approximated given the updated value of the initial condition $\mathbf{x}_{t-1}$. Therefore, we use the current approximation of the value function of the first stage to perform a convergence test using deterministic upper and lower bounds (see Appendix A) to evaluate the gap given the updated initial condition. If the gap is not sufficiently small, we restart the SDDP algorithm to improve the value function until it achieves a satisfactory gap. Once the algorithm converges, a current first-stage solution

is obtained and implemented. The whole procedure is now repeated one-step ahead, given the previous optimal decision and the currently observed uncertainty realization. The whole evaluation process iterates until it reaches the last period to be simulated. This process is described in Figure 1 assuming a fixed value for Δ.

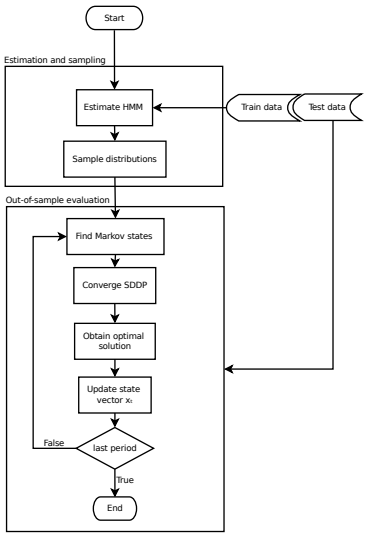


Fig. 1: Flowchart for backtesting using distributionally robust SDDP framework.

Determining an appropriate value of Δ *a priori* is difficult in general. Some papers in the DRO literature compute the level of ambiguity based on the number of data points (see, e.g., [25] and [7]). However, this type of procedure assumes that the data points are independent and identically distributed (i.i.d.), an assumption that is likely not to hold in the settings we are considering in the present paper. In our approach, the HMM approximates the dynamics of the stochastic process and the ambiguity set accounts not only for estimation errors (which go to zero with the number of data points), but also for model misspecification. We suggest choosing Δ via a robustness tuning procedure that selects the value of Δ (among a relatively small number of candidates) with the best out-of-sample performance. For that, we split our data in: training, validation and testing datasets. In this context, the robustness tuning is a series of out-of-sample evaluation steps in the the validation dataset followed by a final out-of-sample evaluation step in the testing (hold-out) dataset. This is indeed the approach we used in the case study presented in the next section.

5 Case study: a risk-constrained dynamic asset allocation model

In this section, we illustrate an application of the framework laid out in the previous sections to an asset allocation problem. The model learns the asset returns from the data and solves a dynamic optimization problem where the goal is to maximize the expected final wealth, taking into account the transaction costs in each period. Other papers use learning approaches for this problem; for example, a regret-optimization approach is applied in [21] to find the best (single-period) portfolio choice using historical data as input. We build upon the work of [46], which allows us to use their results as a benchmark since that paper does not deal with out-of-sample performance. In subsection 5.1, we recap the stochastic model for asset returns while in subsection 5.2, we present an equivalent formulation for the risk constrained dynamic asset allocation model proposed by [46].

5.1 The HMM learning methodology for asset returns

The uncertain returns $\mathbf{r_t}$ are represented by a Hidden Markov Model (HMM). In the context of the financial market, HMM methodology is frequently used to model asset returns [13, 14, 23]. Such paradigm postulates that the probability distribution of asset returns depends only on the current state of the market that evolve according to a discrete-time finite-state Markov Chain. Such states, however, cannot be observed, hence the need for a Hidden Markov Model. Conditionally on each state, the log-returns are independent and identically distributed, with distribution given by a multivariate Gaussian whose parameters are estimated from data. This modeling choice is suitable for financial time series since it empirically reproduces most of the stylized facts for asset return series [38]. As before, we denote by K_t the (random) Markov state at time t, by $\mathcal{K}$ the set of states of the Markov chain and by $\hat{P}$ the corresponding estimated transition matrix with dimension $|\mathcal{K}| \times |\mathcal{K}|$, with $\hat{p}_j(k)$ denoting the probability to transition from state j to state k.

5.2 A CV@R-constrained dynamic asset allocation model

The model proposed in [46] is a multistage stochastic program that maximizes, in each stage, the future value function that represents the conditional expectation of the terminal wealth, subject to a CV@R constraint. Using the notation defined in (5)-(9), that model can be written as follows. Given an initial wealth W_0 and the stochastic return process $\mathbf{r}_t$, we denote $\boldsymbol{\xi}_t = (\mathbf{1} + \mathbf{r}_t)$

and solve, for each possible initial state j, the problem

$$Q_0^j := \max_{\mathbf{x}_0 \in \mathbb{R}_+^{N+1}} \sum_{k \in \mathcal{K}} \mathbb{E}\left[Q_1^k(\mathbf{x}_0, \boldsymbol{\xi}_1) \middle| K_{t+1} = k\right] \hat{p}_j(k) \tag{52}$$

$$\text{s.t.} \quad \phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_1^\top \mathbf{x}_0\right] \geq (1-\gamma) W_0 \tag{53}$$

$$(\mathbf{1} + \tilde{\mathbf{c}})^\top \mathbf{x}_0 = W_0 \tag{54}$$

where $\tilde{\mathbf{c}}$ is the vector containing the transaction cost rate for each asset, and Q_t^j (for $t = 1, \ldots, T-1$) is defined recursively as

$$Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) = \max_{\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)} \sum_{k \in \mathcal{K}} \mathbb{E}\left[Q_{t+1}^k(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}) \middle| K_{t+1} = k\right] \hat{p}_j(k) \tag{55}$$

$$\text{s.t.} \quad \phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_{t+1}^\top \mathbf{x}_t\right] \geq (1-\gamma)\left(\boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}\right), \tag{56}$$

while the end-of-horizon function $Q_T^j(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T) = \boldsymbol{\xi}_T^\top \mathbf{x}_{T-1}$, defines the terminal wealth.

In particular, we assume a risk-free asset indexed by $i = 0$ with null return, i.e., $P(r_{0,t} = 0) = 1$ and, consequently, $P(\xi_{0,t} = 1) = 1$, for all $t \in \{1, \ldots, T\}$. We only assume positive transaction cost rates for the risky assets by defining $\tilde{\mathbf{c}} = (0, \mathbf{c})^\top$. Moreover, to simplify the discussion below we assume that all risky assets have the same transaction cost rate c, so we have $\mathbf{c} = (c, c, \ldots, c)^\top$. In this context, the set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ is defined as

$$\left\{ \mathbf{x}_t \in \mathbb{R}_+^{N+1} \;\middle|\; \exists \mathbf{b}_t, \mathbf{d}_t \in \mathbb{R}_+^N : \begin{array}{l} x_{0,t} + (\mathbf{1} + \mathbf{c})^\top \mathbf{b}_t - (\mathbf{1} - \mathbf{c})^\top \mathbf{d}_t = x_{0,t-1} \\ x_{i,t} - b_{i,t} + d_{i,t} = \xi_{i,t}\, x_{i,t-1}, \quad \forall i \in \mathcal{A}. \end{array} \right\} \tag{57}$$

where $x_{0,t}$ refers to a risk-free asset (cash) allocation while $x_{i,t}$ for $i > 0$ refers to risky asset allocations.

From (57), we see that the allocations $\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ satisfy the equation

$$\mathbf{1}^\top \mathbf{x}_t = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1} - \mathbf{c}^\top (\mathbf{b}_t + \mathbf{d}_t), \tag{58}$$

that is, the amount of money available at time t is the return of the investment made at time $t-1$, minus the transaction costs of assets that were bought ($\mathbf{b}_t$) and sold ($\mathbf{d}_t$).

A few words about the above model are in order. First, notice that the objective functions in (52) and (55) maximize the expected future value of the allocation in each period, where the expectation is taken with respect to both the returns and the Markov states. Constraint (54) reflects the fact that the transaction costs are incurred before the returns are realized; thus, assuming that the initial wealth W_0 is in cash, the initial allocation $\mathbf{1}^\top \mathbf{x}_0$ plus the corresponding purchase costs must be equal to that amount. This constraint is generalized to an arbitrary time period t by means of the set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ defined in (57), which accounts for the transaction costs resulting from both purchases and sales of assets (note that $b_{i,t}$ and $d_{i,t}$ are never simultaneously

positive as nothing is gained from buying and selling the same asset in a given time period).

Note that in problem (55)-(56) the values of $\boldsymbol{\xi}_t$ and $\mathbf{x}_{t-1}$ are given. Thus, the wealth $W_t = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}$—prior to discounting transaction costs, cf. (58)—in period t is just a constant and hence by the translation-invariant property of coherent risk measures we have that $\text{CV@R}_{1-\alpha}[W_t - W_{t+1}] = W_t + \text{CV@R}_{1-\alpha}[-W_{t+1}] = W_t - \phi_{\hat{\mathbf{p}}^j}[W_{t+1}]$. It follows that constraint (56) can be written as $\text{CV@R}_{1-\alpha}[W_t - W_{t+1}] \leq \gamma W_t$. That is, the constraint limits the loss between periods t and $t+1$ to a percentage of the wealth at time t (note that constraint (53) applies the same idea at $t = 0$). The parameter γ can then be interpreted as the level of risk-aversion of the decision-maker: at one extreme ($\gamma = 0$) we have $\text{CV@R}_{1-\alpha}[W_t - W_{t+1}] \leq 0$ which in particular implies that $P(W_{t+1} < w_t \,|\, W_t = w_t) \leq \alpha$, i.e., the probability of a loss between periods t and $t+1$ must be very low; at the other extreme ($\gamma = 1$) we do not impose any risk constraints and so when there are no transaction costs the optimal portfolio will invest only in the asset(s) with highest expected return at each time t ("all eggs in the same basket").

5.3 A novel lower bound for the dynamic asset allocation problem

Motivated by the primal inner-approximation presented in Section 3.3, we use the particular structure of the dynamic asset allocation problem to propose a novel upper bound exploring a convex combination of pre-evaluated points and a proper penalty function for values outside the associated convex hull. With this result at hand, we use the standard SDDP upper bound (outer approximation) to efficiently compute a deterministic optimality gap. Throughout this section the function $Q_{t+1}^j(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})$ corresponds to the DRO version of problem (52)-(56), defined as in (10)-(11) and its equivalent formulation (17)-(24). Recall also the expected value function $\mathcal{Q}_{t+1}^k(\mathbf{x}_t)$ defined in (25).

As shown in [46], the asset allocation problem (52)-(56) has relatively complete recourse whenever $\gamma \geq c$. Indeed, if the maximum allowed loss γ is at least the transaction cost rate, it is always feasible to sell all risky assets and adopt a risk-free strategy with null return: $x_{o,t} = W_t$, and $x_{i,t} = 0$, $\forall i \in \mathcal{A}$. Moreover, this feasible and simple strategy has a straightforward value function since the objective function, i.e., the terminal wealth W_T, is equal to the current wealth ($W_t = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}$) minus the total transaction cost of selling the risky assets ($\mathbf{c}^\top(\mathbf{b}_t + \mathbf{d}_t)$), where $d_{i,t} = \xi_{i,t}\, x_{i,t-1}$ and $b_{i,t} = 0$ for every $i \in \mathcal{A}$. This is shown formally in Proposition 7 below.

Proposition 7. *Suppose that the parameter γ that appears on the right-hand side of (53) satisfies $\gamma \geq c$, where c is the transaction cost rate. Then, $Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) \geq (1-c)\, \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}$ for all Markov states j.*

Proof. Consider a fixed time period t. As the previous allocation vector $\mathbf{x}_{t-1}$ and the realization $\boldsymbol{\xi}_t$ are given as parameters of Q_t^j, we let $W_t = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}$ denote the wealth right before buying and selling decisions at time t. Now, define

the risk-free (sub-optimal) policy where all risky assets are sold at time t and the risk-free investment (with zero return) is held until the end of the horizon T. We use the superscript notation $\mathbf{x}_t^{rf}$, $\mathbf{b}_t^{rf}$ and $\mathbf{d}_t^{rf}$ to denote the values of these decision variables under the risk-free policy. Formally, the risk-free policy amounts to imposing that $d_{i,t}^{rf} = \xi_{i,t}\, x_{i,t-1}$, $b_{i,t}^{rf} = 0$ for every $i \in \mathcal{A}$, and also $\mathbf{b}_\tau^{rf} = \mathbf{d}_\tau^{rf} = \mathbf{0}$ for all $\tau = t+1, \ldots, T$. Using (58), the amount of money invested in the risk-free asset after buying and selling decisions at t is given by $x_{0,t}^{rf} = \mathbf{1}^\top \mathbf{x}_t^{rf} = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1} - \mathbf{c}^\top(\mathbf{b}_t^{rf} + \mathbf{d}_t^{rf}) = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1} - c \sum_{i\in\mathcal{A}} \xi_{i,t}\, x_{i,t-1}$.

Since the risk-free asset has null return (i.e., $r_{0,t+1} = 0$ and, consequently, $\xi_{0,t+1} = 1$), the subsequent wealths can be calculated as $W_\tau = \boldsymbol{\xi}_\tau^\top \mathbf{x}_{\tau-1}^{rf} = x_{0,\tau-1}^{rf} = x_{0,t}^{rf}, \forall \tau \in \{t+1, \ldots, T\}$. Therefore, we have the terminal wealth $W_T = x_{0,t}^{rf} = \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1} - c \sum_{i\in\mathcal{A}} \xi_{i,t}\, x_{i,t-1}$, which corresponds to the objective value of the risk-free (suboptimal) policy. Hence,

$$Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) \geq \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1} - c \left(\sum_{i\in\mathcal{A}} \xi_{i,t}\, x_{i,t-1} \right) \geq (1-c)\, \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}.$$

□

Similarly to the developments in Section 3.3, at iteration ν of the algorithm we have a concave lower (inner) approximation $\underline{\mathcal{Q}}_{t+1}^{k,\nu}(\cdot)$ for $\mathcal{Q}_{t+1}^k(\cdot)$ given by linear inequalities. We then define the function $\widehat{Q}_t^j(\hat{\mathbf{x}}_{t-1}^i, \boldsymbol{\xi}_t)$ as in (42) and compute its expectation $\widehat{\mathcal{Q}}_t^j(\hat{\mathbf{x}}_{t-1}^i) = \mathbb{E}\left[\widehat{Q}_t^j(\hat{\mathbf{x}}_{t-1}^i, \boldsymbol{\xi}_t) \middle| K_t = j\right]$.

Recall that $\{\hat{\mathbf{x}}_{t-1}^i\}_{i=1,\ldots,\nu}$ denote the solutions obtained from the previous iterations of the algorithm. We add an initial point $\hat{\mathbf{x}}_t^0 = \mathbf{0}$ for all t. Since $\hat{\mathbf{x}}_t^0 = \mathbf{0}$ corresponds to having no wealth at all, it is clear that $\widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^0) := \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^0) = 0$ for every iteration ν, every time stage t and every Markov state j. We now devise a novel lower bound for the asset allocation problem.

Proposition 8. *Let $\overline{\boldsymbol{\xi}}_{t,j} = \mathbb{E}[\boldsymbol{\xi}_t | K_t = j]$ denote the conditional expectations of the returns. Suppose $\gamma \geq c$. Then, the function*

$$\begin{aligned}
\underline{\mathcal{Q}}_t^{j,\nu}(\mathbf{x}_{t-1}) \;:=\; \max_{\mathbf{x}', \boldsymbol{\mu}} \quad & \sum_{i=0}^{\nu} \mu_i\, \widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^i) + (1-c)\, \overline{\boldsymbol{\xi}}_{t,j}^\top \mathbf{x}' \\
s.t. \quad & \sum_{i=0}^{\nu} \mu_i \hat{\mathbf{x}}_{t-1}^i + \mathbf{x}' = \mathbf{x}_{t-1} \\
& \sum_{i=0}^{\nu} \mu_i = 1 \\
& \boldsymbol{\mu}, \mathbf{x}' \geq 0.
\end{aligned} \tag{59}$$

is a lower bound for $\mathcal{Q}_t^j(\mathbf{x}_{t-1})$.

Proof. First, note that problem (59) is always feasible. Indeed, given that $\widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^0) = 0$, the solution $\mu_0 = 1$ and $\mathbf{x}' = \mathbf{x}_{t-1}$ recovers the lower bound in Proposition 7. Then, for any feasible $\mu_0, \ldots, \mu_\nu$ and $\mathbf{x}'$, we have that

$$\sum_{i=0}^{\nu} \mu_i \, \widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^i) + (1-c)\, \overline{\boldsymbol{\xi}}_{t,j}^{\top} \mathbf{x}' \;\leq\; \sum_{i=0}^{\nu} \mu_i \, \mathcal{Q}_t^j(\hat{\mathbf{x}}_{t-1}^i) + \mathcal{Q}_t^j(\mathbf{x}') \tag{60}$$

$$\leq\; \mathcal{Q}_t^j\left(\sum_{i=0}^{\nu} \mu_i \, \hat{\mathbf{x}}_{t-1}^i\right) + \mathcal{Q}_t^j(\mathbf{x}') \tag{61}$$

$$\leq\; \mathcal{Q}_t^j(\mathbf{x}_{t-1}). \tag{62}$$

The inequality (60) holds since $\widehat{\mathcal{Q}}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^i) \leq \mathcal{Q}_t^{j,\nu}(\hat{\mathbf{x}}_{t-1}^i)$, and $(1-c)\, \overline{\boldsymbol{\xi}}_{t,j}^{\top} \mathbf{x}' \leq \mathcal{Q}_t^j(\mathbf{x}')$, according to Proposition 7. Additionally, we use concavity to ensure inequality (61) while (62) is guaranteed since $\mathcal{Q}_t^j$ is positively homogeneous (proof in C), and therefore superadditive. □

5.4 Numerical results

To analyze how our approach behaves in practice, we test the model with realistic data. The data sets used in the experiments come from Kenneth R. French data set[5]. The stocks from NYSE, AMEX, and NASDAQ are represented by capitalization-weighted indexes for each industry sector. We use monthly data of five industrial portfolios ("Cnsmr", "Manuf", "HiTec", "Hlth" and "Other"). For simplicity, we use excess returns, i.e., the incremental return over the risk-free asset. This way, the risk-free asset presents $r_{0,t} = 0, \forall t = 1, \ldots, T$.

The framework was implemented in Julia language 0.6, using `JuMP` [11] and CPLEX 12.7.1.0 to solve linear programming problems. All experiments were conducted on Intel Xeon E5-2680 2.7 GHz with 128GB RAM machine, while reported computational times are associated with single-core usage. The `hmmlearn 0.2.0`[6] library was used to construct the return distributions assuming that, conditional to each Markov state, log (excess) returns follow multivariate Gaussian distributions.

5.4.1 Results for the predictive model.

The training dataset comprises 444 months[7] (prior to January 2007), while the dataset for historical simulation uses 96 months (from January 2007 to Setember 2014) to validate the proposed framework. Following [46], we select three Markov states[8] and, conditional to each state, 750 return realizations

[5] http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data library.html

[6] https://github.com/hmmlearn/hmmlearn

[7] The HMM training uses as much data as possible to represent a variety of market situations.

[8] In this particular instance we use the results of [46], but in general a cross-validation procedure could be used.

$K_t \backslash K_{t+1}$	1	2	3
1	69.28	28.58	2.14
2	58.23	40.66	1.11
3	0.77	8.79	90.44

Table 1: Markov transition matrix in percentage.

State	Cnsmr	Manuf	HiTec	Hlth	Other
1	1.17 (0.14)	1.13 (0.10)	0.91 (0.12)	1.29 (0.13)	1.21 (0.14)
2	1.86 (0.20)	1.81 (0.15)	2.08 (0.18)	1.29 (0.16)	1.94 (0.19)
3	-1.24 (0.51)	-0.93 (0.39)	-1.81 (0.46)	-0.81 (0.35)	-1.60 (0.50)

Table 2: Mean percentage (standard-deviation in parenthesis) of asset returns conditional to each Markov state.

obtained using *Latin Hypercube Sampling* of multivariate Gaussian distributions to construct the Sample Average Approximations of the problem. All simulations start with \$1 in the risk-free asset, therefore if the strategy ends the simulation with \$2 it implies an accumulated excess return of 100%. Figure 2 illustrates the posterior probability of each Markov state as in (1), and the solid line indicates the simulated wealth of the equal-weighted portfolio (as a proxy for the general behavior of the market) with cumulative return on the right axis.

Fig. 2: Markov states and equal-weight portfolio wealth.

A closer look at the Markov transition matrix Table 1, the individual asset returns and the corresponding standard deviations in Table 2, in conjunction with Figure 2, allows us to infer how HMM is classifying the historical data and how to interpret the states. State 1 has low positive returns and a low probability of transitioning to state 3. State 2 has a high probability of transitioning to state 1, it also has higher returns than state 1, and it has more volatility. State 3 has negative returns, is almost absorbent with 90% chance to transition to itself, and has almost no probability of transition to state 1.

Therefore, states 2 and 3 can be seen as bull and bear states, respectively. It is more difficult to infer the role of state 1. It seems to be a less volatile regular state since it is the most probable state during the whole simulation (Figure 2).

5.4.2 Results for the prescriptive model.

As discussed in Section 4, we implemented the algorithm in a rolling-horizon fashion. The horizon (number of periods) in each problem is $T = 16$ months with monthly decisions. To illustrate the convergence of the deterministic lower and upper bounds established in Theorem 6—with the lower bound calculated as in Proposition 7—Figure 3 depicts the value of the bounds for an arbitrary run of the algorithm with $\Delta = 0.3$ and $\gamma = 0.07$ for a maximum of 5,000 iterations. In this example the final values of the deterministic lower and upper bounds were respectively 0.003388 and 0.003393, corresponding to an optimality gap of 0.1457%. In practice, we fixed an optimality relative gap of 1% as a stopping criterion. The time to converge the SDDP algorithm for the 1% gap was almost 3 hours for each period (or month). However, by applying the accelerated rolling-horizon procedure described in Section 4 that uses value function approximations constructed in the previous steps, from the second iteration onward the algorithm took less than 30 minutes per period.

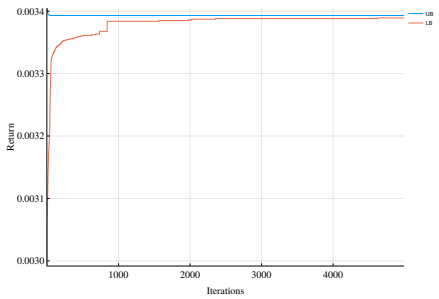


Fig. 3: Deterministic lower and upper bounds for $\Delta = 0.3$ and $\gamma = 0.07$ starting from iteration 100.

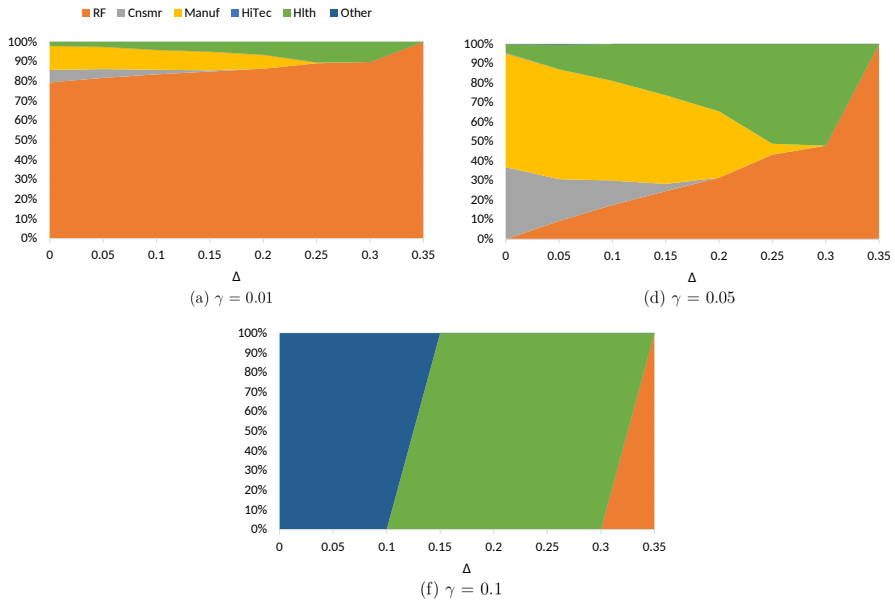
For the out-of-sample evaluation described in Section 4, we start with an estimated HMM, an SAA of the original problem, and the output of the algorithm after running until convergence—comprising a first stage problem and set of T future value functions. We shall denote the periods of the testing dataset $t \in \{t_1, \ldots, t_H\}$ and define $R_t = \frac{W_t - W_{t-1}}{W_{t-1}}$ the portfolio percentage profit given by the proposed strategy at time t (recall that $W_t := \boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}^*$ is the corresponding wealth, prior to discounting transaction costs, cf. (58)). It is important to reiterate that the implemented decisions $\mathbf{x}_t^*$ are obtained as the first stage solution of a T-stage problem given the current (inferred)

Markov state—the most probable state given all information available up to t, cf. (2). In order to compare the out-of-sample performance of the different models, we use the "ex-post" average return of the portfolio strategy, $\hat{R}_{EP} := \frac{1}{H}\sum_{t=1}^{H} R_t$, and the "ex-post" CV@R of the returns, defined as $-\hat{\phi}_{EP}$, where $\hat{\phi}_{EP} = \max_{z\in\mathbb{R}} \left\{ z - \frac{1}{H\alpha}\sum_{t\in\{t_1,\ldots,t_H\}} (z - R_t)_+ \right\}$ following the expression in (4). Note that a comparison of the ex-post CV@R with the parameter γ can be interpreted as an out-of-sample evaluation of constraint (56), since as remarked earlier, that constraint can be written as $\text{CV@R}_{1-\alpha}[W_t - W_{t+1}] \leq \gamma W_t$, i.e., $-\phi_{\hat{\mathbf{p}}_j}[R_{t+1}] \leq \gamma$. Despite the differences with the ex-ante counterparts, ex-post metrics are widely used, especially within the context of financial markets.

For a better assessment of out-of-sample performance, several experiments were done with different combinations of γ and Δ. This can be viewed as a cross-validation procedure. It is important to stress the difference between these parameters. The former quantifies the decision-maker level of risk aversion (cf. (56)), whereas the latter establishes the confidence in the estimated distribution (cf. (12)). In this context, γ restricts the possible decisions, however even if the risk restriction is met, the confidence (or lack thereof) in the estimated probabilities ($\hat{\mathbf{p}}$) will still impact the optimal portfolio decision.

We illustrate the compound effect of the ambiguity aversion (Δ) and the risk aversion (γ) coefficients over the optimal allocation. The effect of these coefficients can be seen in Figure 4, where we present the optimal portfolio on a particular date as a function of Δ for a few values of γ. We see, for example, that for a low value of γ — i.e., a more risk-averse decision-maker — the optimal portfolio is less sensitive to variations in Δ, as the optimal portfolio puts a high percentage on the risk-free asset regardless of the value of Δ. For a slightly less risk-averse decision-maker ($\gamma = 0.05$) the optimal portfolio is diversified, with the components changing according to Δ. Note that for values of Δ larger than 0.35, the ambiguity set includes all distributions and so the min-max problem will always assume the worst possible state, so it is not surprising that the optimal portfolio from that value of Δ on puts everything on the risk-free asset. With $\gamma = 0.1$, we essentially have a risk-neutral decision-maker, and so for most values of Δ, the optimal portfolio consists only of the asset with the largest expected return, though that asset changes based on the level of confidence on the parameters of the HMM given by Δ.

To further analyze this distinction between the parameters and how the Markov state impacts the final portfolio we show the portfolio allocation during the simulation for specific γ and Δ values. This comparison is depicted in Figure 5 where we present the allocation policy for two Δ values, 0.0 and 0.2, with $\gamma = 0.07$ for the whole simulation period. The choice for $\gamma = 0.07$ is because it is the ex-post CV@R corresponding to the equal-weight portfolio. The left axis shows the allocation in percentage of each asset, and the right axis shows the wealth for our DRO model and the equal-weight portfolio for comparison purposes. Some observations can be made: first, notice that during the period between 2008-2009 (which corresponds to the subprime crisis) the optimal portfolios for both values of Δ learn from the HMM that the market



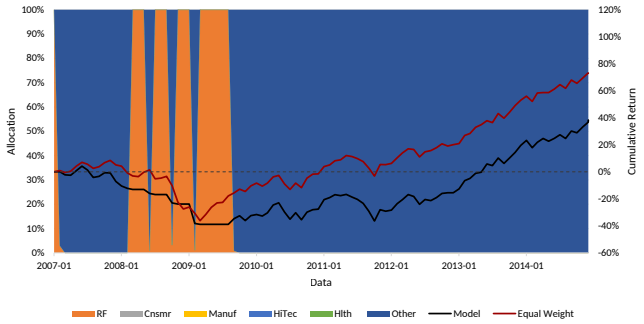
(a) $\gamma = 0.01$ (d) $\gamma = 0.05$ (f) $\gamma = 0.1$

Fig. 4: Allocation for different Δ values with γ equal to 0.01, 0.05 and 0.1.

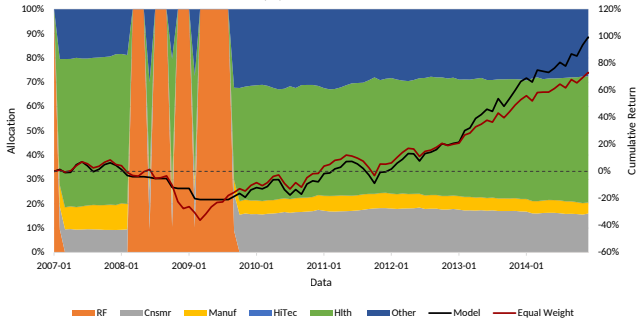
is in a bear state (cf. Figure 2) and thus allocate almost everything into the risk-free asset. Second, while the equal-weight portfolio clearly dominates the portfolio for the case $\Delta = 0$, it is outperformed by the robustified portfolio with $\Delta = 0.2$ as the latter strategy yields better protection during the "bear" times and provides good diversification and good returns during the remaining periods.

The risk-return curves for different values of γ and Δ are shown in Figure 6. Naturally, for portfolios with the same risk, the ones with more returns are preferred. Whereas, for portfolios with the same returns, the ones with less risk are preferred. In the figure, each line corresponds to one value of Δ, whereas each dot corresponds to one value of γ. We see that the efficient frontier consists of portfolios corresponding to Δ around 0.25-0.3, regardless of γ. This, it appears—based on these experiments—that the right choice for Δ ensures good performance regardless of the decision-maker risk tolerance. The use of γ, however, is still important for sensitivity purposes, as we can see that lower values of Δ combined with high values of γ can yield portfolios with inferior performance. Notice also that, as remarked earlier, values of Δ above 0.35 lead to excessive robustness as there is no trust in the HMM parameters, and thus the optimal portfolio consists only of the risk-free asset. That is, it is important to collect sufficient data, so one has *some* confidence in the HMM parameters, but it is better not to trust such parameters blindly.

As shown by other authors [9, 25, 16], the equal-weight portfolio is a good benchmark strategy to compare with as it has competitive out-of-sample per-



(a) $\Delta = 0.00$



(b) $\Delta = 0.20$

Fig. 5: Allocation during the simulation for Δ values 0.00 and 0.20 with $\gamma = 0.07$

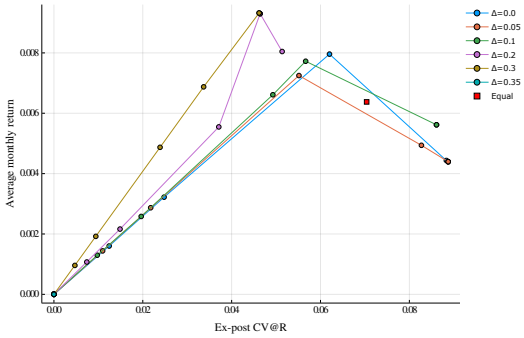


Fig. 6: Out-of-sample monthly average return and CV@R.

formance, especially when the model faces extreme uncertainty or when the transaction costs are high. In Figure 6, we see that the equal-weight portfolio is dominated by most strategies. It confirms the superior out-of-sample performance of our proposed model, except in the cases where the ex-post CV@R

is high, although that can only occur when the pre-specified risk tolerance γ is high, so the decision-maker is nearly risk-neutral.

Finally, in order to assess the effect of Δ on the quality of the portfolio, we need to use a metric that summarizes risk and return. The ICV@R [15] was inspired by the Sharpe ratio in that it measures return by unit of risk. It is computed as the ratio between the average return $\hat{R}_{EP}$ and the deviation between the average return and the average tail $\hat{\phi}_{EP}$ [20], that is, $ICV@R := \frac{\hat{R}_{EP}}{\hat{R}_{EP} - \hat{\phi}_{EP}}$.

Figure 7 depicts the values of the ICV@R index for various values of Δ and γ. We see that the ICV@R function is in most cases monotonically non-decreasing for $\Delta \leq 0.325$ regardless of the value of γ, which suggests again that in order to have better out-of-sample performance one should use higher (but not too high) Δ values. Moreover, we see again that values of Δ around 0.25-0.325 yield the highest values of ICV@R (and thus the best portfolios according to this criterion) regardless of the value of γ. The figure also shows that the equal-weight portfolio is dominated by our distributionally robust approach for all cases with $\Delta \in [0.125, 0.325]$.

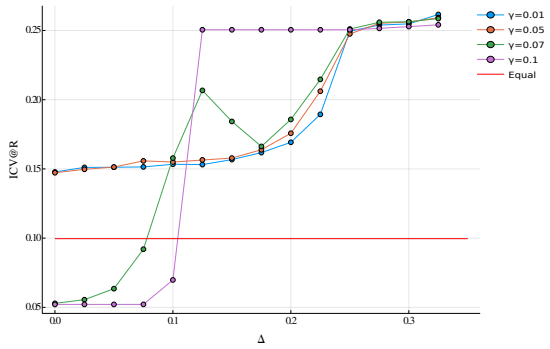


Fig. 7: Out-of-sample ICV@R and Δ for each γ.

5.5 Testing procedure

As a final step in our case study, we applied the algorithm to testing (hold-out) data using a suitable robustness level $\Delta = 0.3$ chosen according to the out-of-sample performances in the validation procedure (see Figure 6). The testing data comprises the period from May-2019 to April-2020. The choice of the testing period was due to two goals: first, we wanted to leave some space between the validation data and the testing data to avoid any contamination; second, since we wanted to fully test the capacity of the model to react to adverse situations, we deliberately chose a period when the market suffered huge losses, as it was the case in March 2020 which coincided with the explosion

of the COVID-19 pandemics. The risk aversion parameter was set to $\gamma = 0.07$, a value comparable to the equal-weight portfolio risk. Figure 8 depicts the optimal allocation strategy found by the algorithm for the model with $\Delta = 0.3$. We see that the model invested on a mix of Health ("Hlth") and risk-free assets for most of the year, except at the beginning and at the end of this testing period where it anticipated a potential downturn and consequently moved the allocation to 100% risk-free assets.

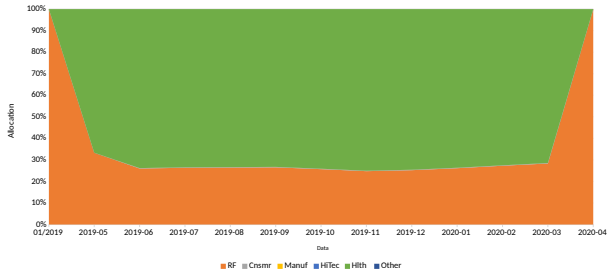


Fig. 8: Out-of-sample May-2019 to April-2020 allocation for HMM-DRO policy with $\Delta = 0.3$.

Figure 9 depicts two pieces of information related to the testing period. In the figure, the labels on the horizontal axis correspond to the situation at the end of each month. The shaded areas indicate the posterior probability of each Markov state as in (1), following the left vertical axis. We see that for the most of the testing period the HMM classified the market overwhelmingly as a mix of "regular" and "bull" states, until March 2020 when it turned the classification into a "bear" state as the COVID-19 crisis expanded worldwide. Such behavior suggests that the HMM was effective in learning the states of the market directly from the data.

The other piece of information displayed in Figure 9 is a comparison among a number of policies, as it shows the accumulated monthly return (following the right vertical axis) during the testing period for (i) the HMM-based policy with $\Delta = 0$; (ii) the HMM-DRO-based policy with $\Delta = 0.3$; (iii) the policy that is obtained using an SDDP model with no HMM; and (iv) the passive equal-weight strategy. We see that the strategy given by our HMM-DRO approach outperformed the equal-weight strategy for most of the year, except for the last month when the equal weight-strategy recovered more quickly from the downturn in March and benefited from the market rebound in April 2020. Moreover, the HMM-DRO policy outperformed the pure-HMM and the no-HMM ones. We see that although the pure-HMM and no-HMM policies performed reasonably well up to the point where the crisis started, after that point both policies suffered huge losses and never recovered. In contrast, the robust policy with $\Delta = 0.3$ was able to weather the effects of the crisis much better. The graph also shows the merits of the HMM approach: indeed, the no-

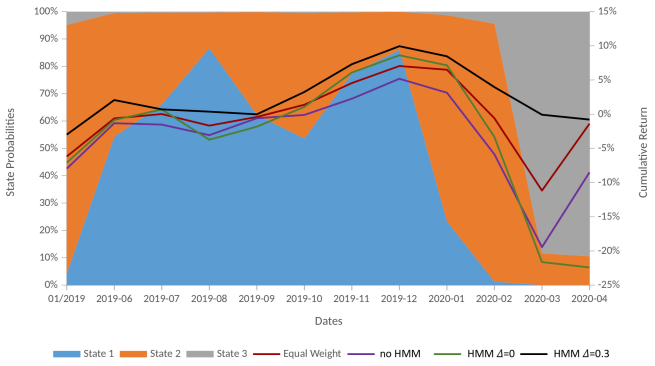


Fig. 9: Out-of-sample May-2019 to April-2020 performance comparison among policies and states of the HMM.

HMM policy had the worst performance of all for most of the period. Overall, the analysis suggests that, by using the HMM to learn about the different states of the market and by taking into account the estimation errors of the HMM, the proposed approach can indeed yield competitive performance during normal times and provide better protection during downturns.

6 Conclusions

The evolution of computing power, new theoretical results, and the development of specialized software tools have made stochastic dynamic optimization models widely applicable in recent years. In our opinion, however, the increase in the utilization of such models has not been accompanied by a similar development in the treatment of data. For most applications reported in the literature, some form for the underlying stochastic process $\{\boldsymbol{\xi}_t\}$ is assumed (after some study of available data), the problem is solved, and the optimal solution given by the model is implemented. Our goal is in this paper is to bring this practice closer to a new reality of data-driven problems, where information can be inferred automatically from the data via some machine learning technique, and an independent validation procedure is applied in order to evaluate the decisions yielded by the model.

To accomplish our goal, we have presented a framework for data-driven distributionally robust dynamic decision models with a particular structure that is applicable in many different contexts. Our approach combines a Hidden Markov Model (HMM) as the predictive engine with a dynamic Distributionally Robust Optimization (DRO) model as the prescriptive methodology. Notwithstanding the HMM flexibility to approximately capture the dynamics of a variety of stochastic processes, it is subject to estimation errors as well as model misspecification. Therefore, a distributionally robust dynamic optimization model is a suitable choice to embody the uncertainty dynamics represented

by the HMM and at the same time to robustify decisions against the uncertainty over the HMM parameters. We have provided a tractable reformulation of the optimization problem and shown that we can adapt the well-known Stochastic Dual Dynamic Programming (SDDP) algorithm to solve the proposed model. Along the way, we have developed a deterministic lower bound (for a maximization problem), which, although related to recent literature, is a novel result that can be generalized to other multi-stage problems. Moreover, the bound has a practical appeal by allowing for user-defined simple policies evaluations to improve computational tractability and solution efficiency, especially when taken together with the deterministic upper bound provided by SDDP.

For a fixed robustness level, we have presented an evaluation framework to assess the out-of-sample performance of the optimal policies yielded by the model in a rolling horizon scheme. We have also introduced an acceleration scheme in case of computationally intensive problems, which is applicable when the problem structure does not depend on the time period. A robustness tuning procedure was proposed as a series of out-of-sample evaluation steps, whereby the robustness level with the best out-of-sample performance is selected. We have illustrated the power and flexibility of the proposed data-driven prescriptive analytics framework with a complete case study on dynamic asset allocation. The numerical results show superior out-of-sample performance against selected benchmarks on a hold-out testing dataset. The case study reiterates the practical importance and applicability of the proposed framework since it emulates the actual decision process of a dynamic asset allocation problem, extracting valuable information from data to obtain robust decisions with an empirical certificate of suitable out-of-sample performance.

While the case study focuses on one type of problem (dynamic asset allocation), we believe that the framework presented in the paper can be useful in other contexts as well. For example, in long-term energy planning—a type of problem for which SDDP has been extensively used—the stochastic input process (e.g., water inflows, solar radiation) could be inferred directly from the data, using machine learning techniques as described in this paper. We believe that the presented work raises important questions for future development on the integration of machine learning methods and dynamic optimization under uncertainty, and hope it will stimulate further research in this area.

Acknowledgements First author gratefully acknowledges the support provided by Funenseg, Brazil. The second author gratefully acknowledges the support of CNPq 302338/2017-9. The third author acknowledges the support of grant FONDECYT 1171145, Chile.

References

1. Baucke, R.: Risk aversion in multistage stochastic optimisation problems. Ph.D. thesis, Operations Research - University of Auckland (2018). URL https://researchspace.auckland.ac.nz/handle/2292/45173
2. Baucke, R., Downward, A., Zakeri, G.: A deterministic algorithm for solving stochastic minimax dynamic programmes (2018). AVailable on *Optimization Online*

3. Bayraksan, G., Love, D.K.: Data-driven stochastic programming using phi-divergences. In: Tutorials in Operations Research, pp. 1–19. INFORMS (2015)
4. Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., Rennen, G.: Robust solutions of optimization problems affected by uncertain probabilities. Management Science **59**(2), 341–357 (2013)
5. Bertsimas, D., McCord, C.: From predictions to prescriptions in multistage optimization problems. arXiv preprint arXiv:1904.11637 (2019)
6. Bishop, C.M.: Pattern recognition and machine learning. Springer (2006)
7. Blanchet, J., Kang, Y., Murthy, K.: Robust Wasserstein profile inference and applications to machine learning. Journal of Applied Probability **56**(3), 830–857 (2019)
8. Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper Res **58**(3), 595–612 (2010)
9. DeMiguel, V., Garlappi, L., Uppal, R.: Optimal versus naive diversification: How inefficient is the 1/n portfolio strategy? The review of Financial studies **22**(5), 1915–1953 (2007)
10. Dowson, O., Morton, D.P., Pagnoncelli, B.K.: Partially observable multistage stochastic programming. Optimization Online (2019). URL http://www.optimization-online.org/DB{_}HTML/2019/03/7141.html
11. Dunning, I., Huchette, J., Lubin, M.: Jump: A modeling language for mathematical optimization. SIAM Review **59**(2), 295–320 (2017)
12. Duque, D., Morton, D.P.: Distributionally robust stochastic dual dynamic programming (2019). Available on *Optimization Online*
13. Elliott, R.J., Van der Hoek, J.: An application of hidden markov models to asset allocation problems. Finance and Stochastics **1**(3), 229–238 (1997)
14. Elliott, R.J., Siu, T.K.: Strategic asset allocation under a fractional hidden markov model. Methodology and Computing in Applied Probability **16**(3), 609–626 (2014)
15. Fernandes, B., Street, A., Valladão, D., Fernandes, C.: An adaptive robust portfolio optimization model with loss constraints based on data-driven polyhedral uncertainty sets. European Journal of Operational Research **255**(3), 961–970 (2016)
16. Georg Ch Pflug, A.P., Wozabal, D.: The 1/n investment strategy is optimal under high model ambiguity. Journal of Banking & Finance **36**(2), 410–417 (2012)
17. Goh, J., Sim, M.: Distributionally robust optimization and its tractable approximations. Oper Res **58**, 902–917. (2010)
18. Homem-de-Mello, T., De Matos, V.L., Finardi, E.C.: Sampling strategies and stopping criteria for stochastic dual dynamic programming: a case study in long-term hydrothermal scheduling. Energy Systems **2**(1), 1–31 (2011)
19. Homem-de-Mello, T., Pagnoncelli, B.K.: Risk aversion in multistage stochastic programming: A modeling and algorithmic perspective. European Journal of Operational Research **249**(1), 188–199 (2016)
20. Kalinchenko, K., Uryasev, S., Rockafellar, R.T.: Calibrating risk preferences with the generalized capital asset pricing model based on mixed conditional value-at-risk deviation. The Journal of Risk **15**(1), 45 (2012)
21. Lim, A.E., Shanthikumar, J.G., Vahn, G.Y.: Robust portfolio choice with learning in the framework of regret: Single-period case. Management Science **58**(9), 1732–1746 (2012)
22. Löhndorf, N., Shapiro, A.: Modeling time-dependent randomness in stochastic dual dynamic programming. European Journal of Operational Research **273**(2), 650 – 661 (2019)
23. Mamon, R.S., Elliott, R.J. (eds.): Hidden Markov Models in Finance. International Series in Operations Research & Management Science. Springer US, Boston, MA (2014)
24. Mo, B., Gjelsvik, A., Grundt, A.: Integrated risk management of hydro power scheduling and contract management. Power Systems, IEEE Transactions on **16**(2), 216–221 (2001)
25. Mohajerin Esfahani, P., Kuhn, D.: Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. Mathematical Programming pp. 1–52 (2017)
26. Moon, T.K.: The expectation-maximization algorithm. IEEE Signal Processing Magazine **13**(6), 47–60 (1996)
27. Pereira, M.V.F., Pinto, L.M.V.G.: Multi-stage stochastic optimization applied to energy planning. Math. Program. **52**(2), 359–375 (1991)

28. Pflug, G., Wozabal, D.: Ambiguity in portfolio selection. Quantitative Finance **7**(4), 435–442 (2007)
29. Philpott, A., de Matos, V., Finardi, E.: On solving multistage stochastic programs with coherent risk measures. Operations Research **61**(4), 957–970 (2013)
30. Philpott, A.B., Guan, Z.: On the convergence of stochastic dual dynamic programming and related methods. Operations Research Letters **36**(4), 450–455 (2008)
31. Philpott, A.B., de Matos, V.L.: Dynamic sampling algorithms for multi-stage stochastic programs with risk aversion. European Journal of Operational Research **218**(2), 470–483 (2012)
32. Philpott, A.B., de Matos, V.L., Kapelevich, L.: Distributionally robust SDDP. Computational Management Science **15**(3-4), 431–454 (2018)
33. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)
34. Rahimian, H., Bayraksan, G., Homem-de-Mello, T.: Identifying effective scenarios in distributionally robust stochastic programs with total variation distance. Mathematical Programming pp. 1–38 (2018)
35. Rockafellar, R.T., Uryasev, S.: Optimization of conditional value-at-risk. Journal of risk **2**, 21–42 (2000)
36. Roorda, B., Schumacher, J.M., Engwerda, J.: Coherent acceptability measures in multiperiod models. Mathematical Finance **15**(4), 589–612 (2005)
37. Rudloff, B., Street, A., Valladão, D.M.: Time consistency and risk averse dynamic decision models: Definition, interpretation and practical consequences. European Journal of Operational Research **234**(3), 743–750 (2014)
38. Rydén, T., Teräsvirta, T., Åsbrink, S.: Stylized facts of daily return series and the hidden markov model. Journal of applied econometrics **13**(3), 217–244 (1998)
39. Scarf, H.: A min-max solution of an inventory problem. In: K. Arrow, S. Karlin, H. Scarf (eds.) Studies in the Mathematical Theory of Inventory and Production,, pp. 201–209. Stanford University Press, Stanford, CA, (1958)
40. Shapiro, A.: Analysis of stochastic dual dynamic programming method. European Journal of Operational Research **209**(1), 63–72 (2011)
41. Shapiro, A., Ahmed, S.: On a class of minimax stochastic programs. Siam J Optimiz **14**(4), 1237–1249 (2004)
42. Shapiro, A., Dentcheva, D., Ruszczyński, A.: Lectures on stochastic programming : modeling and theory, 2nd edn. SIAM (2014)
43. Shapiro, A., Kleywegt, A.: Minimax analysis of stochastic problems. Optim Meth and Software **17**, 523–542 (2002)
44. Shapiro, A., Tekaya, W., da Costa, J.P., Soares, M.P.: Risk neutral and risk averse stochastic dual dynamic programming method. European Journal of Operational Research **224**(2), 375–391 (2013)
45. Sion, M., et al.: On general minimax theorems. Pacific Journal of mathematics **8**(1), 171–176 (1958)
46. Valladão, D., Silva, T., Poggi, M.: Time-consistent risk-constrained dynamic portfolio optimization with transactional costs and time-dependent returns. Annals of Operations Research **282**(1-2), 379–405 (2019)
47. Van Parys, B.P., Mohajerin Esfahani, P., Kuhn, D.: From data to decisions: Distributionally robust optimization is optimal. arXiv preprint arXiv:1704.04118 (2017)
48. Žáčková, J.: On minimax solutions of stochastic linear programming problems. Časopis pro Pěstování Matematiky **91**(4), 423–430 (1966)

A Algorithms

In this section, we present detailed information about the SDDP implementation. The first algorithm (Algorithm 1) describes the central part of our distributionally robust SDDP, followed by the descriptions of the forward (Algorithm 2), backward (Algorithm 3) steps, and deterministic lower bound evaluation (Algorithm 4).

Algorithm 1 Distributionally robust SDDP

Require: K_0, $\mathbf{x}_0$ and $\{\overline{\mathcal{Q}}_t^j, \underline{\mathcal{Q}}_t^j\}_{t=1,k=1}^{T,K}$(Init. future value functions)
  Initialize: $UB \leftarrow \overline{Q}_0^{K_0}, LB \leftarrow 0, GAP = \frac{UB-LB}{UB}$ and
  Create subproblems $\{\overline{Q}_t^j, \underline{Q}_t^j\}_{t=1,k=1}^{T,K}$
  while $GAP > \epsilon$ **do**
    Generate a sample path: $\{\boldsymbol{\xi}_t, K_t\}_{t=1}^{T}$
    $\{\hat{\mathbf{x}}_t\}_{t=1}^{T} \leftarrow$ **Forward Step** (Algorithm 2)
    $\{\overline{Q}_t^j\}_{t=1,j=1}^{T,K} \leftarrow$ **Backward Step** (Algorithm 3)
    $UB \leftarrow \overline{Q}_0^{K_0}(\mathbf{x}_0) + \mathbf{1}^\top \mathbf{x}_0$
    $\{\underline{Q}_t^j\}_{t=1,j=1}^{T,K} \leftarrow$ **Det. Lower Bound** (Algorithm 4)
    $LB \leftarrow \underline{Q}_0^{K_0}(\mathbf{x}_0) + \mathbf{1}^\top \mathbf{x}_0$
  end while

Forward step (Algorithm 2) consists of finding trial points $\{\hat{\mathbf{x}}_t\}_{t=1}^{T}$ using current subproblems $\{\overline{Q}_t^j, \underline{Q}_t^j\}_{t=1,k=1}^{T,K}$. Notice that, inside the subproblems, both in forward and backward algorithms, it is only used the outer approximation of the future value function $\{\overline{\mathcal{Q}}_t^j\}_{t=1,k\in\mathcal{K}}^{T}$. The procedure to update the inner approximation is described in (Algorithm 4).

Algorithm 2 Forward Step

Require: K_0, $\mathbf{u}_0$, a sample path $\{\boldsymbol{\xi}_t, K_t\}_{t=1}^{T}$ and $\{\overline{Q}_t^j\}_{t=1,k\in\mathcal{K}}^{T}$
  for $t \in \{1, \ldots, T\}$ **do**
    $\hat{\mathbf{x}}_t \leftarrow$ solution of $\overline{Q}_t^{K_t}(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$
  end for
  Return $\{\hat{\mathbf{x}}_t\}_{t=1}^{T}$

Backward step (Algorithm 3) updates the current outer approximation of the future value function inside the subproblems.

Algorithm 3 Backward Step

Require: $\{\hat{\mathbf{x}}_t\}_{t=1}^{T}$ and $\{\overline{Q}_t^j\}_{t=1,k\in\mathcal{K}}^{T}$
  for $t \in \{T,\dots,1\}$ **do**
    for $j \in \mathcal{K}$ **do**
      for $s \in S_k$ **do**
        Solve (63)-(75)
        Compute dual variable $\pi_{t,s}^{j}$ of (63)-(75)
      end for
      $\overline{\mathcal{Q}}_t^k(\mathbf{x}_{t-1}) \leftarrow \min_{i=1,\dots,\nu+1} \ell_t^{k,i}(\mathbf{x}_{t-1})$
      Update $\overline{\mathcal{Q}}_t^k(\mathbf{x}_{t-1})$ inside $\overline{Q}_{t-1}^k(\mathbf{x}_{t-2}\boldsymbol{\xi}_{t-1})$
    end for
  end for
  Return $\{\overline{Q}_t^j\}_{t=1,k\in\mathcal{K}}^{T}$

Unlike the backward step, which is a Benders decomposition, in the deterministic lower bound algorithm we update the model using the column generation method. The method adds variables, at each iteration, to the outer approximation of the future value function $\{\underline{\mathcal{Q}}_t^j\}_{t=1,k\in\mathcal{K}}^{T}$. To facilitate the understanding of the lower bound construction, we present model (63)-(75), which is similar to the problem (26)-(35) with $\mathfrak{Q}_t^j(\mathbf{x}'') = -\mathbf{c}^\top\mathbf{x}''$. However, here we are using the lower approximation of the future value function describes in Section 3.3.

$$\underline{Q}_t^j(\mathbf{x}_{t-1},\boldsymbol{\xi}_t) :=$$

$$\max_{\substack{\mathbf{x}_t,z,\mathbf{y},\boldsymbol{\theta}^-,\boldsymbol{\theta}^+,\lambda,\\ \eta,\widetilde{\boldsymbol{\theta}}^-,\widetilde{\boldsymbol{\theta}}^+,\widetilde{\lambda},\widetilde{\eta},\mathbf{u}\\ \boldsymbol{\lambda}',\mathbf{x}',\mathbf{x}''}} \quad f_t(\mathbf{x}_t,\boldsymbol{\xi}_t) + \sum_{k\in\mathcal{K}} \hat{p}_j(k)(\theta_k^+ - \theta_k^-) - \eta - 2\Delta\lambda \tag{63}$$

$$\text{s.t.} \quad z + \sum_{k\in\mathcal{K}} \hat{p}_j(k)(\widetilde{\theta}_k^+ - \widetilde{\theta}_k^-) - \widetilde{\eta} - 2\Delta\widetilde{\lambda} \;\geq\; 0 \tag{64}$$

$$-\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\eta} + \sum_{s\in\mathcal{S}_k} y_{ks}\,\frac{q_k(s)}{\alpha} \;\leq\; 0, \qquad \forall k\in\mathcal{K} \tag{65}$$

$$-\theta_k^- + \theta_k^+ - \eta - \underline{\mathcal{Q}}_{t+1}^k(\mathbf{x}_t) \;\leq\; 0, \qquad \forall k\in\mathcal{K} \tag{66}$$

$$\underline{\mathcal{Q}}_{t+1}^k(\mathbf{x}_t) - \sum_{i\,\in\,\mathcal{I}_t} \lambda_i' \frac{1}{2}\underline{\mathcal{Q}}_{t+1}^k(\hat{\mathbf{x}}_t^i) + \frac{1}{2}\mathbf{c}^\top\mathbf{x}'' = 0, \quad \forall k\in\mathcal{K} \tag{67}$$

$$\mathbf{x}' + \mathbf{x}'' = \mathbf{x}_t \tag{68}$$

$$\sum_{i\,\in\,\mathcal{I}_t} \lambda_i'\,\hat{\mathbf{x}}_t^i - \mathbf{x}' = 0 \tag{69}$$

$$\sum_{i\,\in\,\mathcal{I}_t} \lambda_i' = 1 \tag{70}$$

$$\theta_k^- + \theta_k^+ - \lambda = 0, \qquad \forall k\in\mathcal{K} \tag{71}$$

$$\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\lambda} = 0, \qquad \forall k\in\mathcal{K} \tag{72}$$

$$z - g_t(\mathbf{x}_t,\boldsymbol{\xi}_{t+1}^k(s)) - y_{ks} \;\leq\; 0, \qquad \forall k\in\mathcal{K}, \forall s\in\mathcal{S}_k \tag{73}$$

$$\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1},\boldsymbol{\xi}_t) \tag{74}$$

$$\boldsymbol{\theta}^-,\boldsymbol{\theta}^+,\widetilde{\boldsymbol{\theta}}^-,\widetilde{\boldsymbol{\theta}}^+,\mathbf{y},\lambda,\widetilde{\lambda},\boldsymbol{\lambda}',\mathbf{x}',\mathbf{x}'' \geq 0. \tag{75}$$

Algorithm 4 Deterministic Lower Bound (Inner Problem)

Require: $\{\hat{\mathbf{x}}_t\}_{t=1}^T$ and $\{\underline{\mathcal{Q}}_t^j\}_{t=1,k\in\mathcal{K}}^T$
  for $t \in \{T,\dots,1\}$ **do**
    for $j \in \mathcal{K}$ **do**
      for $s \in \{1,\dots,S\}$ **do**
        if $t \leq T$ **then**
          Add column $\lambda_{i=|\mathcal{I}_t|+1}$ to $\boldsymbol{\lambda}'$ in (63)-(75) with coefficients $\underline{\mathcal{Q}}_{t+1}^j(\hat{\mathbf{x}}_\mathbf{t})$, $\hat{\mathbf{x}}_t$ and 1 in constraints (68) and (69) and (70) respectively.
        end if
        $\underline{\mathcal{Q}}_t^j(\hat{\mathbf{x}}_\mathbf{t-1}) \leftarrow \sum_{s\in\mathcal{S}_j} Q_t^j(\hat{\mathbf{x}}_{t-1}, \boldsymbol{\xi}_t(s))q_j(s)$
      end for
    end for
  end for
  Return $\{\underline{\mathcal{Q}}_t^k\}_{t=1,k\in\mathcal{K}}^T$

B Implementation Remarks

The convergence of SDDP may be rather slow when using models that admit the wealth at the last stage. This can also lead to numerical instability. To have better performance, we evaluate the immediate return and transactional costs and take them into account at each stage decision, making it easy to estimate the immediate influence of the current decision in the objective function.

It is straightforward that the model with immediate returns is equivalent to the one that considers only the wealth in the last stage, see [46] for more details. Below is the complete distributionally robust optimization model with immediate returns and transactional costs on the objective function is detailed

$$
\begin{aligned}
&Q_t^j(\mathbf{x}_{t-1}, \mathbf{r}_t) = \\
&\max_{\substack{\mathbf{x}_t, z, \mathbf{y}, \boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \lambda, \\ \eta, \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \widetilde{\lambda}, \widetilde{\eta}, \mathbf{b}_t, \mathbf{d}_t}} \quad & \sum_{s\in\mathcal{S}_k} \left(\mathbf{r}_{t+1}(s)^\top \mathbf{x}_t\right) q_k(s) - \mathbf{c}^\top(\mathbf{b}_t + \mathbf{d}_t) + \sum_{k\in\mathcal{K}} \hat{p}_j(k)(\theta_k^+ - \theta_k^-) - \eta - 2\Delta\lambda \\
\text{s.t.} \quad & z + \sum_{k\in\mathcal{K}} \hat{p}_j(k)(\widetilde{\theta}_k^+ - \widetilde{\theta}_k^-) - \widetilde{\eta} - 2\Delta\widetilde{\lambda} - \mathbf{c}^\top(\mathbf{b}_t + \mathbf{d}_t) \geq -\gamma((\mathbf{1} + \mathbf{r}_t)^\top \mathbf{x}_{t-1}) \\
& -\widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\eta} + \sum_{s\in\mathcal{S}_k} y_{ks} \frac{q_k(s)}{\alpha} \leq 0, && \forall k \in \mathcal{K} \\
& -\theta_k^- + \theta_k^+ - \eta - \overline{\mathcal{Q}}_{t+1}^k(\mathbf{x}_t) \leq 0, && \forall k \in \mathcal{K} \\
& \theta_k^- + \theta_k^+ - \lambda = 0, && \forall k \in \mathcal{K} \\
& \widetilde{\theta}_k^- + \widetilde{\theta}_k^+ - \widetilde{\lambda} = 0, && \forall k \in \mathcal{K} \\
& z - \mathbf{r}_{t+1}(s)^\top \mathbf{x}_t - y_{ks} \leq 0, && \forall k \in \mathcal{K}, \forall s \in \mathcal{S}_k \\
& x_{i,t} - b_{i,t} + d_{i,t} = (1 + r_{i,t}) x_{i,t-1}, && \forall i \in \mathcal{A} \\
& x_{0,t} + (\mathbf{1} + \mathbf{c})^\top \mathbf{b}_t - (\mathbf{1} - \mathbf{c})^\top \mathbf{d}_t = x_{0,t-1} \\
& \mathbf{x}_t, \boldsymbol{\theta}^-, \boldsymbol{\theta}^+, \widetilde{\boldsymbol{\theta}}^-, \widetilde{\boldsymbol{\theta}}^+, \mathbf{y}, \mathbf{b}_t, \mathbf{d}_t, \lambda, \widetilde{\lambda} \geq 0. && (76)
\end{aligned}
$$

C Positively homogeneous proof for DRO dynamic asset allocation

Let us define the DRO dynamic asset allocation

$$Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) \;=\; \max_{\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)} \left\{ \min_{\mathbf{p}_j \in \mathcal{P}_j} \sum_{k \in \mathcal{K}} \mathbb{E}\left[Q_{t+1}^k\big(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}\big) \Big| K_{t+1} = k \right] \hat{p}_j(k) \right\}$$
$$\text{s.t.} \qquad \phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_{t+1}^\top \mathbf{x}_t \right] \;\geq\; (1-\gamma)\big(\boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}\big), \quad \forall \mathbf{p}_j \in \mathcal{P}_j,$$

where the end-of-horizon function $Q_T^j(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T) = \boldsymbol{\xi}_T^\top \mathbf{x}_{T-1}$, defines the terminal wealth. We also define the set $\mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ is defined as

$$\left\{ \mathbf{x}_t \in \mathbb{R}_+^{N+1} \;\middle|\; \exists \mathbf{b}_t, \mathbf{d}_t \in \mathbb{R}_+^N : \begin{array}{l} x_{0,t} + (\mathbf{1}+\mathbf{c})^\top \mathbf{b}_t - (\mathbf{1}-\mathbf{c})^\top \mathbf{d}_t = x_{0,t-1} \\ x_{i,t} - b_{i,t} + d_{i,t} = \xi_{i,t}\, x_{i,t-1}, \quad \forall i \in \mathcal{A} \end{array} \right\}.$$

Proposition 9. *The function $Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$ is positively homogeneous with respect to $\mathbf{x}_{t-1}$.*

Proof. Let $\upsilon > 0$ denote a positive constant. Let us also define $\widetilde{\mathbf{x}_t} = \upsilon\, \mathbf{x}_t$ and $\widetilde{\mathbf{x}}_{t-1} = \upsilon\, \mathbf{x}_{t-1}$. First note that if, and only if, $\widetilde{\mathbf{x}}_t \in \mathcal{X}_t(\widetilde{\mathbf{x}}_{t-1}, \boldsymbol{\xi}_t)$, then $\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$. Moreover, if, and only if, $\phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_{t+1}^\top \widetilde{\mathbf{x}}_t \right] \geq (1-\gamma)\big(\boldsymbol{\xi}_t^\top \widetilde{\mathbf{x}}_{t-1}\big)$, then $\phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_{t+1}^\top \mathbf{x}_t \right] \geq (1-\gamma)\big(\boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}\big)$, for any $\mathbf{p}_j \in \mathcal{P}_j$. Also, for $t = T$, we have that

$$Q_T^j(\widetilde{\mathbf{x}}_{T-1}, \boldsymbol{\xi}_T) = \boldsymbol{\xi}_T^\top \widetilde{\mathbf{x}}_{T-1} = \upsilon\,(\boldsymbol{\xi}_T^\top \mathbf{x}_{T-1}) = \upsilon\, Q_T^j(\mathbf{x}_{T-1}, \boldsymbol{\xi}_T)$$

Employing the inductive hypothesis $Q_{t+1}^j(\widetilde{\mathbf{x}}_t, \boldsymbol{\xi}_{t+1}) = \upsilon\, Q_{t+1}^j(\mathbf{x}_t, \boldsymbol{\xi}_{t+1})$, we have that

$$Q_t^j(\widetilde{\mathbf{x}}_{t-1}, \boldsymbol{\xi}_t) \;=\; \max_{\widetilde{\mathbf{x}}_t \in \mathcal{X}_t(\widetilde{\mathbf{x}}_{t-1}, \boldsymbol{\xi}_t)} \left\{ \min_{\mathbf{p}_j \in \mathcal{P}_j} \sum_{k \in \mathcal{K}} \mathbb{E}\left[Q_{t+1}^k\big(\widetilde{\mathbf{x}}_t, \boldsymbol{\xi}_{t+1}\big) \Big| K_{t+1} = k \right] \hat{p}_j(k) \right\}$$
$$\text{s.t.} \qquad \phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_{t+1}^\top \widetilde{\mathbf{x}}_t \right] \;\geq\; (1-\gamma)\big(\boldsymbol{\xi}_t^\top \widetilde{\mathbf{x}}_{t-1}\big), \quad \forall \mathbf{p}_j \in \mathcal{P}_j,$$

$$= \max_{\mathbf{x}_t \in \mathcal{X}_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)} \left\{ \min_{\mathbf{p}_j \in \mathcal{P}_j} \sum_{k \in \mathcal{K}} \mathbb{E}\left[\upsilon\, Q_{t+1}^k\big(\mathbf{x}_t, \boldsymbol{\xi}_{t+1}\big) \Big| K_{t+1} = k \right] \hat{p}_j(k) \right\}$$
$$\text{s.t.} \qquad \phi_{\hat{\mathbf{p}}_j}\left[\boldsymbol{\xi}_{t+1}^\top \mathbf{x}_t \right] \;\geq\; (1-\gamma)\big(\boldsymbol{\xi}_t^\top \mathbf{x}_{t-1}\big), \quad \forall \mathbf{p}_j \in \mathcal{P}_j,$$

$$= \upsilon\, Q_t^j(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t)$$

□