Duality of upper bounds in stochastic dynamic programming

Bernardo Freitas Paulo da Costa & Vincent Leclère

July 24, 2023

Abstract

For multistage stochastic programming problems with stagewise independent uncertainty, dynamic programming algorithms calculate polyhedral approximations for the value functions at each stage. The SDDP algorithm provides piecewise linear lower bounds, in the spirit of the L-shaped algorithm, and corresponding upper bounds took a longer time to appear. One strategy uses the *primal* dynamic programming recursion to build inner approximations of the value functions, while a second one builds lower approximations for the *conjugate* of the value functions. The resulting dynamic programming recursion for the conjugate value functions do not decompose over scenarios, which suggests a Lagrangian relaxation. We prove that this Lagrangian relaxation corresponds exactly to the inner upper bounds for a natural choice of multipliers.

Contents

1 Introduction

Multistage stochastic programming is a powerful framework that addresses decision-making problems in complex and dynamic systems with evolving uncertainties. With its ability to capture the dynamics of real-world systems and handle evolving uncertainties, multistage stochastic programming offers valuable insights and strategies for a wide range of applications, including energy management, financial portfolio optimization, and supply chain planning.

In this paper, we consider a linear multistage stochastic programming problem with recourse written as

$$\min_{\boldsymbol{x}_{[T]},\boldsymbol{y}_{[T]}} \quad \mathbb{E}\Big[\sum_{t=1}^{T} \boldsymbol{c}_t^\top \boldsymbol{y}_t\Big] \tag{1a}$$

$$\text{s.t.} \quad \boldsymbol{x}_0 = x_0, \tag{1b}$$

$$\boldsymbol{A}_t\boldsymbol{x}_t + \boldsymbol{B}_t\boldsymbol{x}_{t-1} + \boldsymbol{T}_t\boldsymbol{y}_t = \boldsymbol{d}_t \qquad \forall t \in [T], \tag{1c}$$

$$\boldsymbol{x}_t, \boldsymbol{y}_t \preceq \boldsymbol{\xi}_{[t]}, \qquad \forall t \in [T], \tag{1d}$$

where $[T]$ stands for $\{1,\dots,T\}$, and equality holds almost surely. We denote by $\boldsymbol{x}_{[T]}$, $\boldsymbol{y}_{[T]}$, and $\boldsymbol{\xi}_{[T]}$ the vectors $(\boldsymbol{x}_1,\dots,\boldsymbol{x}_T)$, $(\boldsymbol{y}_1,\dots,\boldsymbol{y}_T)$, and $(\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_T)$, respectively. Finally, $\boldsymbol{x}_t \preceq \boldsymbol{\xi}_{[t]}$ means that $\boldsymbol{x}_t$ is measurable with respect to $\boldsymbol{\xi}_{[t]}$, and the same holds for $\boldsymbol{y}_t$.

In most cases, linear multistage stochastic programming problems have an exponential complexity in the number of stages. That is why we assume that the noises $\boldsymbol{\xi}_t := (\boldsymbol{c}_t, \boldsymbol{A}_t, \boldsymbol{B}_t, \boldsymbol{T}_t, \boldsymbol{d}_t)$ are independent exogeneous random variables, opening the use of dynamic programming to solve the problem. More precisely, we introduce a sequence of *value functions* V_t satisfying the following Bellman recursion

$$V_{T+1} = 0, \tag{2a}$$

$$\dot{V}_t(x_{t-1},\xi_t) = \min_{x_t,y_t} \quad c_t^\top y_t + V_{t+1}(x_t) \quad \text{s.t.} \quad A_t x_t + B_t x_{t-1} + T_t y_t = d_t, \tag{2b}$$

$$V_t(x_{t-1}) = \mathbb{E}\big[\dot{V}_t(x_{t-1},\boldsymbol{\xi}_t)\big], \tag{2c}$$

and the value of Problem (1) is given by $V_1(x_0)$.

In the past 30 years, numerous algorithms have been proposed to leverage the Bellman recursion (2) to solve Problem (1), starting with the Stochastic Dual Dynamic Programming (SDDP) algorithm [PP91]. This has lead to a variety of Trajectory Following Dynamic Programming algorithms (see [FL23]) which have been successfully applied to large-scale multistage stochastic programming problems with various applications in the energy industry.

These algorithms leverage structural assumptions of the value functions V_t, namely convexity for SDDP, to iterativaly refine outer approximations of the value functions, and therefore obtain lower bounds. On the other hand, upper approximations have been often estimated statistically, by simulating a policy and averaging the resulting costs. Unfortunately, such statistical upper bounds are not tractable in nested risk-averse problems, where the expectation in (2) is replaced by a risk measure.

Recent works have offered alternative approaches to compute upper bounds, either through a primal recursion [PdMF13, BDZ17, GTW19], or leveraging duality [LCC+20, GSC23, dCL23]. The duality path consists in showing that the dual value functions follow a Bellman recursion similar to (2), and use it to determine outer approximations of the dual value functions:

$$\mathcal{D}_t(\pi_0) = \inf_{\boldsymbol{\lambda}, \boldsymbol{\pi}} \quad \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d}_t + \mathcal{D}_{t+1}(\boldsymbol{\pi})\right] \tag{3a}$$

$$\text{s.t.} \quad \mathbb{E}[\boldsymbol{B}_t^\top \boldsymbol{\lambda}] = \pi_0 \tag{3b}$$

$$\boldsymbol{\pi}_t + \boldsymbol{A}_t^\top \boldsymbol{\lambda} \geq 0 \tag{3c}$$

$$\boldsymbol{c}_t + \boldsymbol{T}_t^\top \boldsymbol{\lambda} \geq 0 \tag{3d}$$

Duality then turns these outer approximations into inner-approximation of the primal value functions. This was first done in [LCC+20] for the risk neutral setting, and then extended to risk-averse problems in [dCL23].

One of the main downsides of the dual approach is that the expectation constraint (3b) links the decision taken at stage t for the various realizations of ξ_t, contrary to the primal problem where the problems can be decoupled. Thus, a natural idea is to use a Lagrangian relaxation of the coupling constraint. In this paper, we explore the links between the inner approximations obtained through a primal inner update, a dual outer update, and a Lagrangian relaxed dual outer update.

More precisely, our contributions are as follows:

- We show that any Lagrangian relaxation of the dual recursion leads to valid upperbounds;
- We relate the relaxed dual operator to the Bellman primal operator;
- In particular, by choosing the Lagrange multiplier as the right primal points, we obtain the same upper bound as the primal inner approximation scheme
- Thus, we show that the primal inner approximation scheme can be linked to the Lagrangian relaxation of the dual outer approximation scheme.
- We extend the results to several variants of SDDP, including the risk-averse and the periodic settings.

The paper is organized as follows: section 2 recalls the classical SDDP algorithm, defining the Bellman operators and the update operators building outer approximations with cuts. It then discusses how to use the primal Bellman recursion to compute inner approximations of the value functions, and introduces the corresponding update operator and some specific algorithms for iteratively building inner approximations. section 3 discusses how to use duality to compute upper bounds. Indeed, the dual SDDP algorithm consists in using a Bellman recursion linking the conjugates of the value functions to run SDDP and get lower bounds for the dual. Using duality once more, we get primal lower bounds. We then discuss how Lagrangian relaxation decomposes the dual SDDP algorithm, and enlighten the links between the upper bounds constructed using primal and dual recursions. section 4 extends the results to the risk-averse setting, while section 5 briefly discusses the periodic setting. Finally, section 6 concludes the paper.

2 Inner and outer approximations of the value functions

We tackle multistage problem of the form (1) through dynamic programming methods computing inner and/or outer approximations. To do so, we first recall the classical SDDP algorithm, and then show how to compute inner approximations using convexity of the value functions. Finally, we some algorithms that maintain both an inner and outer approximations of the value functions.

2.1 Outer updates and the SDDP algorithm

As a motivation for our work, we recall the Stochastic Dual Dynamic Programming (SDDP) algorithm [PP91].

First, we define, for each $t \in [T]$, a Linear Bellman Operator (LBO) $\mathcal{B}_t$ associated to a two-stage stochastic problem with recourse cost Q, where Q is a proper polyhedral function[1], by:

$$\begin{aligned}
\mathcal{B}_t(Q) : x_{t-1} \mapsto \quad \inf_{\boldsymbol{x}_t, \boldsymbol{y}_t} \quad & \mathbb{E}\left[\boldsymbol{c}_t^\top \boldsymbol{y}_t + Q(\boldsymbol{x}_t)\right] \\
\text{s.t.} \quad & \boldsymbol{A}_t \boldsymbol{x}_t + \boldsymbol{B}_t x_{t-1} + \boldsymbol{T}_t \boldsymbol{y}_t = \boldsymbol{d}_t \quad \text{a.s.} \\
& \boldsymbol{x}_t, \boldsymbol{y}_t \geq 0 \quad \text{a.s.}
\end{aligned} \tag{4}$$

In particular, Equation (2) can be rewritten as

$$V_T = 0, \quad \text{and} \quad V_t = \mathcal{B}_t(V_{t+1}) \qquad \forall t \in [T]. \tag{5}$$

Further, the primal Bellman operator $\mathcal{B}_t$ can be decomposed per realization of the uncertainty ξ_t, *i.e.*, $\mathcal{B}_t(Q)(x_{t-1}) = \mathbb{E}\left[\dot{\mathcal{B}}(Q)(x_{t-1}, \xi_t)\right]$ for $t \in [T]$, where $\dot{\mathcal{B}}(Q)(x_{t-1}, \xi_t)$ is given as the optimal value of

$$\begin{aligned}
\min_{x_t, y_t} \quad & c_t^\top y_t + Q(x_t) \\
\text{s.t.} \quad & A_t x_t + B_t x_{t-1} + T_t y_t = d_t \\
& x_t, y_t \geq 0.
\end{aligned} \tag{$\dot{\mathfrak{B}}(Q)(x_{t-1}, \xi_t)$}$$

Note that solving the above linear problem, denoted $(\dot{\mathfrak{B}}(Q)(x_{t-1}, \xi_t))$, simultaneously returns the optimal out-state decision $x_t =: \mathcal{F}_t(Q)(x_{t-1}, \xi_t)$, the optimal value of the objective function $\dot{v} = \mathcal{B}_t(Q)(x_{t-1}, \xi)$ and a subgradient $\dot{\pi} \in \partial_x \dot{\mathcal{B}}_t(Q)(x_{t-1}, \xi)$ of the cost-to-go function evaluated at x_{t-1}.

[1] The polyhedrality assumption is not needed to define Bellman operators. Convexity is required for exact linear cuts to be obtained through duality, and polyhedrality simplifies the constraints qualifications requirements.

Algorithm 1: Vanilla SDDP

Data: maximum number of iterations N, lower bounds $\underline{V}_t \leq V_t$

1 **for** $k = 1$ **to** N **do**
2   Set $x_0^k \leftarrow x_0$
3   **for** $t = 1$ **to** $T-1$ **do**   `// forward pass`
4   Randomly select $\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)$
5   Solve $(\dot{\mathfrak{B}}_t(\underline{V}_{t+1}^k)(x_{t-1}^k, \xi))$ for x_t^k
6   Set $\underline{V}_{T+1}^{k+1} = 0$
7   **for** $t = T$ **to** 1 **do**   `// backward pass`
8   **for** $\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)$ **do**
9   Solve $(\dot{\mathfrak{B}}(\underline{V}_{t+1}^{k+1})(x_{t-1}^k, \xi))$ for $(\dot{\underline{v}}_{t,\xi}^{k+1}, \dot{\pi}_{t,\xi}^{k+1})$
10   Set $\underline{v}_t^{k+1} = \sum_{\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)} \mathbb{P}(\boldsymbol{\xi}_t = \xi) \dot{\underline{v}}_{t,\xi}^{k+1}$
11   Set $\pi_t^{k+1} = \sum_{\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)} \mathbb{P}(\boldsymbol{\xi}_t = \xi) \dot{\pi}_{t,\xi}^{k+1}$
12   Set $\underline{V}_t^{k+1} = \max\Big(\underline{V}_t^k, v_t^{k+1} + (\cdot - x_t^{k+1})^\top \pi_t^{k+1}\Big)$

With this in mind, iteration k of the vanilla SDDP algorithm 1, consists in a forward pass where we compute the optimal out-state decision x_t^k for a sampled realization ξ_t^k of the uncertainty, and a backward pass where we compute the optimal value of the objective function v_t^k and a subgradient π_t^k of the cost-to-go function evaluated at x_{t-1}, for each stage t and realization ξ of the uncertainty $\boldsymbol{\xi}_t$. The backward phase thus updates an outer approximation of the cost-to-go functions V_t. The following definition formalizes this outer update.

Definition 1. *Let Q_t and Q_{t+1} be proper polyhedral functions, and $\mathcal{B}$ be an abstract linear Bellman operator. The outer update operator $\mathcal{U}_o$ is defined as*

$$\mathcal{U}_o(Q_t, Q_{t+1}, \mathcal{B}_t; x_t) := \max\big(Q_t \ , \ \mathcal{B}_t(Q_{t+1})(x_t) + \pi_t^\top(\cdot - x_t)\big) \tag{6}$$

where π_t is a subgradient of $\mathcal{B}_t(Q_{t+1})$ at x_t.

With these outer updates, lines 8 to 12 of algorithm 1 can be rewritten as

$$\underline{V}_t^{k+1} = \mathcal{U}_o(\underline{V}_t^k, \underline{V}_{t+1}^{k+1}, \mathcal{B}_t; x_t^k). \tag{7}$$

Remark 2 (Outer bound validity and representation)**.** Let $t \in [T]$. Note that, if $\underline{V}_{t+1} \leq V_{t+1}$, then $\mathcal{B}_t(\underline{V}_{t+1}) \leq \mathcal{B}_t(V_{t+1}) = V_t$, so that, if in addition $\underline{V}_t \leq V_t$ then the updated bound is still valid, *i.e.,* $\mathcal{U}_o(\underline{V}_t, \underline{V}_{t+1}, \mathcal{B}_t; x_t) \leq V_t$.

Further, if $\underline{V}_t$ is given as a collection of cuts (*i.e.,* an H-representation), then $\mathcal{U}_o(\underline{V}_t, \underline{V}_{t+1}, \mathcal{B}_t; x_t)$ is also given as a collection of cuts.

2.2 Primal inner approximation schemes

As we have seen, the vanilla SDDP algorithm relies on iteratively refining outer approximations of the value functions V_t based on the convexity of V_t. Still using convexity of V_t, we can also iteratively refine inner approximations of the value

Algorithm 2: IDP

Data: maximum number of iterations N, upperbounds $\overline{V}_t \leq V_t$

1 **for** $k = 1$ **to** N **do**
2 Set $x_0^k \leftarrow x_0$
3 **for** $t = 1$ **to** $T-1$ **do** `// forward pass`
4 Compute a trial point x_t^k
5 Set $\overline{V}_{T+1}^{k+1} = 0$
6 **for** $t = T$ **to** 1 **do** `// backward pass`
7 **for** $\xi \in \operatorname{supp}(\boldsymbol{\xi}_t)$ **do**
8 Solve $(\dot{\mathfrak{B}}(\overline{V}_{t+1}^{k+1})(x_{t-1}^k, \xi))$ for $\dot{\overline{v}}_{t,\xi}^{k+1}$
9 Set $\overline{v}_t^{k+1} = \sum_{\xi \in \operatorname{supp}(\boldsymbol{\xi}_t)} \mathbb{P}(\boldsymbol{\xi}_t = \xi) \dot{\overline{v}}_{t,\xi}^{k+1}$
10 Set $\overline{V}_t^{k+1} : x \mapsto \min_{\alpha \in \Delta^k} \left\{ \sum_{\kappa=1}^{k+1} \alpha_\kappa \overline{v}_t^\kappa \;\middle|\; \sum_{\kappa=1}^{k+1} \alpha_\kappa x_t^\kappa = x \right\}$ where Δ^k is the k-dimensional simplex

functions V_t, as presented in algorithm 2, where the trial point computation is not specified.

To efficiently express the backward pass of algorithm 2, we introduce the following operator for functions $f, g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$:

$$\begin{aligned} f \blacktriangledown g : x \mapsto \inf_{\lambda, y, z} \quad & \lambda f(y) + (1-\lambda) g(z) && \text{(8a)} \\ \text{s.t.} \quad & \lambda \in [0,1] && \text{(8b)} \\ & \lambda y + (1-\lambda) z = x. && \text{(8c)} \end{aligned}$$

In terms of epigraphs, we have $\operatorname{epi}(f \blacktriangledown g) = \overline{\operatorname{Conv}}(\operatorname{epi}(f) \cup \operatorname{epi}(g))$. In particular, $f \blacktriangledown g$ is convex and lower-semicontinuous. With this, we define the following inner update operator:

Definition 3. *Let Q_t and Q_{t+1} be proper polyhedral functions, and $\mathcal{B}$ be an abstract linear Bellman operator. The inner update operator $\mathcal{U}_i$ is defined as*

$$\mathcal{U}_i(Q_t, Q_{t+1}, \mathcal{B}; x_t) := Q_t \blacktriangledown \operatorname{pin}_{x_t, \mathcal{B}(Q_{t+1})(x_t)} \tag{9}$$

where

$$\operatorname{pin}_{\check{x},h} : x \mapsto \begin{cases} h & \text{if } x = \check{x} \\ +\infty & \text{otherwise.} \end{cases} \tag{10}$$

With this operator, lines 7 to 10 of algorithm 2 can be rewritten as

$$\overline{V}_t^{k+1} = \mathcal{U}_i(\overline{V}_t^k, \overline{V}_{t+1}^{k+1}, \mathcal{B}_t; x_t^k). \tag{11}$$

Remark 4. Let $t \in [T]$. Note that, if $\overline{V}_{t+1} \geq V_{t+1}$, then $\mathcal{B}_t(\overline{V}_{t+1}) \geq \mathcal{B}_t(V_{t+1}) = V_t$, so that, if in addition $\overline{V}_t \geq V_t$ then the updated bound is still valid, *i.e.,* $\mathcal{U}_i(\overline{V}_t, \overline{V}_{t+1}, \mathcal{B}_t; x_t) \geq V_t$.

Further, if $\overline{V}_t$ is given as the convex envelope of a minimum of pin functions(*i.e.,* a V-representation), then $\mathcal{U}_i(\overline{V}_t, \overline{V}_{t+1}, \mathcal{B}_t; x_t)$ is also given as a collection of pins.

Remark 5. The name "pin" combines the fact that the function $\mathrm{pin}_{x,h}$ "pins" the function V_t to be at most h at x, and also because its epigraph is a half-line going upwards from (x, h).

2.3 Algorithms using inner and outer bounds

In algorithm 2, the trial point selection is not specified.

To our knowledge, the first IDP algorithm[2] of this kind can be found in [PdMF13], where the authors propose to use as trial points the trajectories obtained during a standard SDDP procedure (see algorithm 1). More precisely, they run a standard SDDP procedure, keep the trajectories of the forward pass, and use them as trial points in a single backward pass of algorithm 2.

Some papers (like [BDZ17] and [GTW19] for the robust setting) have proposed to consider the possible out-state $x^k_{t,\xi}$, of a standard SDDP procedure and select the *problem-child*, that is the one maximizing the gap between the current upper and lower bounds, resulting in algorithm 3

Algorithm 3: RDDP

Data: maximum number of iterations N, upperbounds $\overline{V}_t \leq V_t$

for $k = 1$ **to** N **do**
- Set $x^k_0 \leftarrow x_0$
- **for** $t = 1$ **to** $T - 1$ **do** `// forward pass`
  - **for** $\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)$ **do**
    - Solve $(\dot{\mathfrak{B}}(\underline{V}^k_{t+1})(x^k_{t-1}, \xi))$ for $x^k_{t,\xi}$
  - Set x^k_t such that $(\overline{V}^k_{t+1} - \underline{V}^k_{t+1})(x^k_t) = \max_{\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)} (\overline{V}^k_{t+1} - \underline{V}^k_{t+1})(x^k_{t,\xi})$
- Set $\underline{V}^{k+1}_{T+1} = \overline{V}^{k+1}_{T+1} = 0$
- **for** $t = T$ **to** 1 **do** `// backward pass`
  - $\underline{V}^{k+1}_t = \mathcal{U}_o(\underline{V}^k_t, \underline{V}^{k+1}_{t+1}, \mathcal{B}_t; x^k_t)$
  - $\overline{V}^{k+1}_t = \mathcal{U}_i(\overline{V}^k_t, \overline{V}^{k+1}_{t+1}, \mathcal{B}_t; x^k_t)$

Note that the use of the outer approximation in the forward pass of algorithm 3 is mandatory to ensure convergence. Using the inner approximation in the forward pass would not allow to explore enough the state space to ensure convergence.

The problem-child node-selection procedure has very good theoretical properties, and sometimes good numerical properties as well. However, it is also natural to consider a stochastic version of this algorithm, that we call Stochastic Inner Dynamic Programming (SIDP), where the node-selection is done randomly, as described in algorithm 4.

[2] Actually a straigthforward extension were we compute more than one trial point per stage in the forward phase.

Algorithm 4: SIDP

Data: maximum number of iterations N, upperbounds $\overline{V}_t \le V_t$

for $k = 1$ **to** N **do**
- Set $x_0^k \leftarrow x_0$
- **for** $t = 1$ **to** $T-1$ **do** `// forward pass`
  - Randomly select $\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)$
  - Solve $(\dot{\mathfrak{B}}(\underline{V}_{t+1}^k)(x_{t-1}^k, \xi))$ for $x_{t,\xi}^k$
- Set $\underline{V}_{T+1}^{k+1} = \overline{V}_{T+1}^{k+1} = 0$
- **for** $t = T$ **to** 1 **do** `// backward pass`
  - $\underline{V}_t^{k+1} = \mathcal{U}_o(\underline{V}_t^k, \underline{V}_{t+1}^{k+1}, \mathcal{B}_t; x_t^k)$
  - $\overline{V}_t^{k+1} = \mathcal{U}_i(\overline{V}_t^k, \overline{V}_{t+1}^{k+1}, \mathcal{B}_t; x_t^k)$

3 Inner approximation through duality

In this section, we aim at obtaining inner approximations through duality. First, we recall that Fenchel duality can be used to obtain inner approximations, to the price of non-decomposable stage-problems. Lagrangian relaxation is then a natural way to decompose it, and we end by showing that a Lagrangian decomposition with well-chosen multipliers yields the same inner approximation as the primal inner approximation scheme of algorithm algorithm 4.

3.1 Inner approximation through duality

It has been shown in [LCC+20] that, for any proper polyhedral function V, we have

$$[\mathcal{B}(V)]^\star = \mathcal{B}^\ddagger(V^\star) \tag{12}$$

where, for any polyhedral function D we define

$$\begin{aligned}
\mathcal{B}^\ddagger(D) : \pi_0 \mapsto \inf_{\boldsymbol{\lambda}, \boldsymbol{\pi}} \quad & \mathbb{E}\Big[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi})\Big] && \text{(13a)}\\
\text{s.t.} \quad & \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 && \text{(13b)}\\
& \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \ge 0 && \text{(13c)}\\
& \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \ge 0 && \text{(13d)}
\end{aligned}$$

which is another LBO, which we denote $\mathcal{B}^\ddagger$, corresponding to a two-stage stochastic optimization problem with an *expectation constraint*. For the sake of completeness, we recall the proof of (12) in §B.2.

We can then apply the SDDP algorithm to (13), to obtain an outer approximation of $[V_t]^\star$, which we denote $\underline{U}_t$. This is presented in algorithm 5, although we are dropping some technicalities[3] to clarify the discussion.

The functions $\underline{U}_t^k$ obtained are lower bounds to $V_t^\star$. And, by polyhedrality of V_t, taking again the Fenchel transform shows that $V_t \le \big[\underline{U}_t^k\big]^\star$.

[3] Like initialisation of the dual state, the final value function, and compactification procedure required for SDDP to work.

Algorithm 5: Dual SDDP

Data: maximum number of iterations N, lowerbounds $\underline{U}_t \leq U_t$

for $k = 1$ **to** N **do**
  for $t = 1$ **to** $T - 1$ **do** `// forward pass`
    Randomly select $\xi \in \mathrm{supp}(\boldsymbol{\xi}_t)$
    Solve $(\mathfrak{B}_t^\ddagger(\underline{U}_{t+1}^k)(\pi_{t-1}^k)$ for $(\pi_{t,\xi}^k)_{\xi \in \Xi_t}$
    Set $\pi_t^k = \pi_{t,\xi}^k$
  for $t = T$ **to** 1 **do** `// backward pass`
    $\underline{U}_t^{k+1} = \mathcal{U}_o(\underline{U}_t^k, \underline{U}_{t+1}^{k+1}, \mathcal{B}_t^\ddagger; \pi_t^k)$.

Note that, contrary to algorithm 1, the forward pass requires to solve the stage-problem (13) coupled by the expectation constraint (13b). We discuss next how to decouple the problem via Lagrangian relaxation.

3.2 Lagrangian relaxation of dual problem

The primal problem defined in (4) can be decomposed in scenarios. On the other hand, the problem that arises through the dual Bellman operator $\mathcal{B}^\ddagger$ in (13) has a linking constraint $\mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0$. It is natural to look for a Lagrangian relaxation of this constraint in order to obtain smaller problems. The Lagrange multiplier we choose for the linking constraint will be denoted by $\hat{x}$, so we define:

Definition 6 (Relaxed version of dual Bellman operator)**.** *For any (convex) function D, we set*

$$\begin{aligned} \mathcal{B}^{\text{⸸}}(D;\hat{x})(\pi_0) := \inf_{\boldsymbol{\lambda},\boldsymbol{\pi}} \quad & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi})\right] + \hat{x}^\top \left(\pi_0 - \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}]\right) \\ s.t. \quad & \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\ & \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0 \end{aligned} \tag{14}$$

It will be convenient to fix $\hat{x}$ and consider the induced operator $D \mapsto \mathcal{B}^{\text{⸸}}(D;\hat{x})$ over functions, which we denote by $\mathcal{B}^{\text{⸸}}_{\hat{x}}$.

Observe that, for each $\hat{x}$, the function $\mathcal{B}^{\text{⸸}}_{\hat{x}}(D)$ in an affine *function of π_0.*

Since problem (14) is a Lagrangian relaxation, we have $\mathcal{B}^{\text{⸸}}(D;\hat{x})(\pi_0) \leq \mathcal{B}^\ddagger(D)(\pi_0)$. Therefore, we can use $\mathcal{B}^{\text{⸸}}_{\hat{x}}$ instead of $\mathcal{B}^\ddagger$ to perform *outer* (lower) updates, analogous to SDDP (Algorithm 1) and the Dual-SDDP:

Definition 7 (Relaxed dual update)**.** *Given two functions $\underline{U}_t$ and $\underline{U}_{t+1}$, a state π_t and a Lagrange multiplier $\hat{x}$, the relaxed dual update for the function $\underline{U}_t$ is just using $\mathcal{B}^{\text{⸸}}_{\hat{x}}$ instead of $\mathcal{B}^\ddagger$ in the outer update:*

$$\mathcal{U}_o(\underline{U}_t, \underline{U}_{t+1}, \mathcal{B}^{\text{⸸}}_{\hat{x}}; \pi_t).$$

With this in place, we derive a relation between inner and outer updates, respectively in the primal and dual problems:

Theorem 8. *Let F and G be closed, proper, convex functions, and $\mathcal{B}$ a linear Bellman operator. Then, for all $\hat{x}$ we have*

$$[\mathcal{U}_i(F, G, \mathcal{B}; \hat{x})]^\star = \mathcal{U}_o(F^\star, G^\star, \mathcal{B}^{\text{⸸}}_{\hat{x}}; \pi_0) \quad \forall \pi_0.$$

Proof. The inner update yields $F \blacktriangledown \operatorname{pin}_{\hat{x},\mathcal{B}(G)(\hat{x})}$, where $\operatorname{pin}_{\hat{x},\mathcal{B}(G)(\hat{x})}$ is the pin function $\mathbb{I}_{\hat{x}} + \mathcal{B}(G)(\hat{x})$, while the outer update yields $\max(F^\star, \psi)$, where ψ is the cut corresponding to $\mathcal{B}^{\downarrow}_{\hat{x}}(G^\star)$ at π_0. Actually, $\psi = \mathcal{B}^{\downarrow}_{\hat{x}}(G^\star)$, since the latter is an affine function, so the cut does not depend on the trial point π_0. By the conjugacy of max and $\blacktriangledown$ (see proposition 17 in the appendix), it is enough to show that $\psi = \left[\operatorname{pin}_{\hat{x},\mathcal{B}(G)(\hat{x})}\right]^\star$.

By strong duality (see proposition 19 in the appendix), the intercept of the affine function $\mathcal{B}^{\downarrow}_{\hat{x}}(G^\star)$ is precisely $-\mathcal{B}(G)(\hat{x})$. This, and the fact that the linear coefficient of $\mathcal{B}^{\downarrow}_{\hat{x}}(G^\star)$ is $\hat{x}$, shows that ψ is indeed the Fenchel conjugate of $\operatorname{pin}_{\hat{x},\mathcal{B}(G)(\hat{x})}$. ☐

3.3 Interpreting SIDP as a dual Lagrangian relaxation

In the inner approximation scheme of SIDP (Algorithm 4), one maintains a sequence of upper bounds $\overline{V}^k_t$ for each value function V_t, each one given by a finite collection of points $(z_{t,i}, v_{t,i})$, such that $V(z_{t,i}) \leq v_{t,i}$.

For the dual problem, we have an outer approximation scheme, where each value function $[V_t]^\star$ is bounded below by $\underline{U}^k_t$. Now, suppose that these bounds are conjugate to each other:

$$\left[\overline{V}^k_t\right]^\star = \underline{U}^k_t,$$

which is natural since an upper bound gets transformed to a lower bound via Fenchel conjugacy. We have:

Proposition 9 (Conjugacy-preserving update)**.** *Under the hypothesis that* $\left[\overline{V}^k_t\right]^\star = \underline{U}^k_t$ *for all* t*, performing an update of the inner approximation* $\overline{V}^k_t$ *using the primal problem evaluated at* $\hat{x}_{t-1}$ *corresponds to performing an update of the outer approximation* $\underline{U}^k_t$ *via the dual problem relaxed with* $\hat{x}_{t-1}$*, in the sense that* $\left[\overline{V}^{k+1}_t\right]^\star = \underline{U}^{k+1}_t$.

Algorithmically, adding one vertex to the inner approximation of the primal value function through primal Bellman operator is equivalent to adding one cut to outer approximation of the dual value function with the relaxed dual Bellman operator, where the Lagrange multiplier is chosen as the evaluation point in the primal.

Proof. The first part is an application of theorem 8 on the conjugate of the primal update. If one makes a "forward" update of the value functions in both problems, this follows from $\overline{V}^{k+1}_t = \mathcal{U}_i(\overline{V}^k_t, \overline{V}^k_{t+1}, \mathcal{B}; \hat{x}_{t-1})$ and $\underline{U}^{k+1}_t = \mathcal{U}_o(\underline{U}^k_t, \underline{U}^k_{t+1}, \mathcal{B}^{\downarrow}_{\hat{x}}; \pi_0)$. If, on the other hand, one performs a "backward" update, then we have $\overline{V}^{k+1}_t = \mathcal{U}_i(\overline{V}^k_t, \overline{V}^{k+1}_{t+1}, \mathcal{B}; \hat{x}_{t-1})$ and $\underline{U}^{k+1}_t = \mathcal{U}_o(\underline{U}^k_t, \underline{U}^{k+1}_{t+1}, \mathcal{B}^{\downarrow}_{\hat{x}}; \pi_0)$, and the result follows from backwards induction on t.

The second part results from the conjugacy between cuts and vertices (see corollary 18 in the appendix). ☐

Applying this result through all iterations of the primal algorithm with outer and inner approximations, on the one hand, and the combined primal-dual algorithm with relaxed dual updates, we obtain

Theorem 10. *If the starting outer approximations $\underline{U}_t^0$ for $[V_t]^\star$ are conjugate to $\overline{V}_t^0$, the inner approximations of V_t, and if at every iteration of the primal-dual algorithm the update of the dual bounding functions is performed via a relaxation at the point where the inner-outer algorithm adds a vertex to the inner approximation, then $\underline{U}_t^k = \left[V_t^k\right]^\star$ for all iterations.*

In particular, the upper bounds obtained via the primal-relaxed dual method are the same as those from the inner-outer primal method.

3.4 Relatively complete recourse for the dual

The dual problem from eq. (13) may lack relatively complete recourse. One possible way of addressing this is to consider that the original primal problem has explicit bounds on the decision variables. This is a natural assumption for the state variables x, since SDDP usually needs bounded states for the forward steps to be well-defined; as for the y variables, such bounds can be often derived from knowledge of the support of the uncertainty.

With these bounds, the primal recursion becomes

$$\begin{aligned}
\mathcal{B}_t(Q) : x_{t-1} \mapsto \inf_{\boldsymbol{x}_t, \boldsymbol{y}_t} \quad & \mathbb{E}\left[\boldsymbol{c}_t^\top \boldsymbol{y}_t + Q(\boldsymbol{x}_t)\right] \\
\text{s.t.} \quad & \boldsymbol{A}_t \boldsymbol{x}_t + \boldsymbol{B}_t x_{t-1} + \boldsymbol{T}_t \boldsymbol{y}_t = \boldsymbol{d}_t \quad \text{a.s.} \\
& 0 \leq \boldsymbol{x}_t \leq \overline{x}_t \quad \text{a.s.} \\
& 0 \leq \boldsymbol{y}_t \leq \overline{y}_t \quad \text{a.s.}
\end{aligned} \tag{15}$$

and the corresponding dual recursion will be

$$\begin{aligned}
\mathcal{B}^\ddagger(D) : \pi_0 \mapsto \inf_{\boldsymbol{\lambda}, \boldsymbol{\pi}, \boldsymbol{\zeta}_x, \boldsymbol{\zeta}_y} \quad & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + \overline{x}_t^\top \boldsymbol{\zeta}_x + \overline{y}_t^\top \boldsymbol{\zeta}_y + D(\boldsymbol{\pi})\right] \\
\text{s.t.} \quad & \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 \\
& \boldsymbol{\zeta}_x + \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\
& \boldsymbol{\zeta}_y + \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0
\end{aligned} \tag{16}$$

This problem has relatively complete recourse as soon as $\boldsymbol{B}^\top$ has full range, which is the same as saying that $V_t(x)$ is not constant along a direction, which is reasonable to assume.

3.5 State bounds and Lipschitz regularization

As mentionned above, it is often important that the state variables remain bounded along the iterations of the algorithm. In the dual, this is achieved by adding a box constraint on the Lagrange multipliers π to the formulation in eq. (16).

For the primal inner approximation scheme, one also needs to ensure that the inner approximations are defined everywhere. Indeed, given a trial point $\hat{x}$ and an approximation $\overline{V}^k$, it could happen that the dynamic constraint defining the next state z leads to a point outside the domain of $\overline{V}^k$. This could be the case, for example, for the first iteration of the algorithm, when $\overline{V}^0$ is just defined at a single point. Therefore, one can add a regularization term to the Bellman operator, which will ensure that the resulting function is defined everywhere. A natural choice is to add a Lipschitz regularization term, so that one actually evaluates $\mathcal{B}(\overline{V}^k \mathbin{\square} L \cdot n)$, where L is the Lipschitz constant, and $n(x) = \|x\|_1$.

Now, since the conjugate of the infimal convolution is the pointwise sum of the conjugates, we have

$$\left[\mathcal{B}(\overline{V}^k \,\square\, L \cdot n)\right]^\star = \mathcal{B}^\ddagger\left(\left[\overline{V}^k\right]^\star + [L \cdot n]^\star\right). \tag{17}$$

The conjugate of the Lipschitz regularization term is the indicator function of the set $\{\pi \mid \|\pi\|_\infty \leq L\}$, and therefore corresponds to adding a box constraint to the dual problem. This is precisely the regularization needed in the dual problem, to ensure that the dual state variables π remain bounded along the iterations of the algorithm.

4 The risk-averse setting

In this section, we show an analogous result for the recursion of coperspective functions introduced in [dCL23]. The formulation of the risk measure in that paper is

$$\rho(Z) := \sup_{q \in Q} \mathbb{E}^q[Z], \tag{18}$$

where Q is a polyhedral set of probability measures, and the corresponding risk-averse Bellman operator $\mathcal{B}(V)$ is

$$\begin{aligned} \mathcal{B}(V)(x_0) = \inf_{\boldsymbol{x},\boldsymbol{y}} \quad & \rho\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] \\ \text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \\ & \boldsymbol{x}, \boldsymbol{y} \geq 0. \end{aligned} \tag{19}$$

Defining the *coperspective* of a function f as the perspective of its Fenchel conjugate, which, for $\gamma_0 > 0$ is given by:

$$f^\boxtimes(\pi_0, \gamma_0) := \sup_{x_0} \pi_0^\top x_0 - \gamma_0 f(x_0),$$

we obtain a relation analogous to the one in (12):

$$[\mathcal{B}(V)]^\boxtimes = \mathcal{B}^\boxtimes(V^\boxtimes),$$

where the corresponding coperspective Bellman operator $\mathcal{B}^\boxtimes$ is:

$$\begin{aligned} \mathcal{B}^\boxtimes(D)(\pi_0, \gamma_0) = \inf_{\boldsymbol{\lambda},\boldsymbol{\pi}} \quad & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi}, \boldsymbol{\gamma})\right] \\ \text{s.t.} \quad & \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 \\ & \boldsymbol{\gamma} \in \gamma_0 Q \\ & \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\ & \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0. \end{aligned} \tag{20}$$

The dual recursion induced by (20) presents both incoming states π_0 and γ_0 in linking constraints. The expectation constraint corresponding to π_0 can be relaxed with a primal trajectory $\hat{x}$, as in the risk-neutral case. It remains to show how to relax the constraint corresponding to γ_0 with a primal parameter, and that these choices lead to a similar conjugacy result for primal inner updates.

4.1 The AV@R case

We illustrate our approach first with the AV@R case. One advantage of using AV@R is the Rockafellar-Uryashev representation

$$\rho_\alpha(\boldsymbol{Z}) := \inf_\theta \left\{\theta + \frac{1}{1-\alpha}\mathbb{E}\left[(\boldsymbol{Z}-\theta)_+\right]\right\}. \tag{21}$$

Thus, the dynamic programming equation in the AV@R case becomes:

$$\begin{array}{rlll} \mathcal{B}(V)(x_0) = & \inf\limits_{\boldsymbol{x},\boldsymbol{y};\theta,\boldsymbol{u},\boldsymbol{z}} & \theta + \frac{1}{1-\alpha}\mathbb{E}[\boldsymbol{u}] & \\ & \text{s.t.} & \theta + \boldsymbol{u} \geq \boldsymbol{z} & [\boldsymbol{\delta}] \\ & & \boldsymbol{z} \geq \boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x}) & [\boldsymbol{\gamma}] \\ & & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} & [\boldsymbol{\lambda}] \\ & & \boldsymbol{x},\boldsymbol{y},\boldsymbol{u} \geq 0 & [\boldsymbol{\mu},\boldsymbol{\nu},\boldsymbol{\eta}]\ , \end{array} \tag{22}$$

where we indicate Lagrange multipliers for each constraint. Since all decision variables are random vectors, except for the parameter θ, we use the expectation inner product to derive the Lagrangian:

$$\theta + \frac{1}{1-\alpha}\mathbb{E}[\boldsymbol{u}] + \mathbb{E}[\boldsymbol{\delta}(\boldsymbol{z}-\theta-\boldsymbol{u}) + \boldsymbol{\gamma}(\boldsymbol{c}^\top\boldsymbol{y} + V(\boldsymbol{x}) - \boldsymbol{z})] \\ + \mathbb{E}[\boldsymbol{\lambda}^\top(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_{t-1} + \boldsymbol{T}\boldsymbol{y} - \boldsymbol{d})] - \mathbb{E}[\boldsymbol{\mu}^\top\boldsymbol{x} + \boldsymbol{\nu}^\top\boldsymbol{y} + \boldsymbol{\eta}\boldsymbol{u}].$$

Eliminating the multipliers $\boldsymbol{\nu}$ and $\boldsymbol{\eta}$, we obtain, as there is no duality gap, the dual formulation of $\mathcal{B}(V)(x_0)$:

$$\begin{array}{rlll} \mathcal{B}(V)(x_0) = & \sup\limits_{\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\mu}} & \mathbb{E}\left[\boldsymbol{\lambda}^\top(\boldsymbol{B}x_0 - \boldsymbol{d}) - V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{A}^\top\boldsymbol{\lambda};\boldsymbol{\gamma})\right] & \\ & \text{s.t.} & \mathbb{E}[\boldsymbol{\delta}] = 1 & [\theta] \\ & & \boldsymbol{\delta} \leq \frac{1}{1-\alpha} & [\boldsymbol{u}] \\ & & \boldsymbol{\gamma} = \boldsymbol{\delta} & [\boldsymbol{z}] \\ & & \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top\boldsymbol{\lambda} \geq 0 & [\boldsymbol{y}] \\ & & \boldsymbol{\mu},\boldsymbol{\delta},\boldsymbol{\gamma} \geq 0 & \end{array} \tag{23}$$

where we replace $\inf\limits_{\boldsymbol{x}}(\boldsymbol{A}^\top\boldsymbol{\lambda} - \boldsymbol{\mu})^\top\boldsymbol{x} + \boldsymbol{\gamma}V(\boldsymbol{x})$ by $-V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{A}^\top\boldsymbol{\lambda};\boldsymbol{\gamma})$.

For the coperspective calculation, we will need to evaluate $\gamma_0\mathcal{B}(V)(x_0)$. Pushing the multiplier γ_0 inside the objective function, we obtain

$$\begin{array}{rll} \gamma_0\mathcal{B}(V)(x_0) = & \sup\limits_{\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\mu}} & \mathbb{E}\left[\gamma_0\boldsymbol{\lambda}^\top(\boldsymbol{B}x_0 - \boldsymbol{d}) - \gamma_0 V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{A}^\top\boldsymbol{\lambda};\boldsymbol{\gamma})\right] \\ & \text{s.t.} & \mathbb{E}[\boldsymbol{\delta}] = 1 \\ & & \boldsymbol{\delta} \leq \frac{1}{1-\alpha} \\ & & \boldsymbol{\gamma} = \boldsymbol{\delta} \\ & & \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top\boldsymbol{\lambda} \geq 0 \\ & & \boldsymbol{\mu},\boldsymbol{\delta},\boldsymbol{\gamma} \geq 0 \end{array} \tag{24}$$

$$\begin{array}{rll} = & \sup\limits_{\tilde{\boldsymbol{\lambda}},\tilde{\boldsymbol{\gamma}},\tilde{\boldsymbol{\delta}},\tilde{\boldsymbol{\mu}}} & \mathbb{E}\left[\tilde{\boldsymbol{\lambda}}^\top(\boldsymbol{B}x_0 - \boldsymbol{d}) - V^{\boxtimes}(\tilde{\boldsymbol{\mu}} - \boldsymbol{A}^\top\tilde{\boldsymbol{\lambda}};\tilde{\boldsymbol{\gamma}})\right] \\ & \text{s.t.} & \mathbb{E}[\tilde{\boldsymbol{\delta}}] = \gamma_0 \\ & & \tilde{\boldsymbol{\delta}} \leq \frac{\gamma_0}{1-\alpha} \\ & & \tilde{\boldsymbol{\gamma}} = \tilde{\boldsymbol{\delta}} \\ & & \tilde{\boldsymbol{\gamma}}\boldsymbol{c} + \boldsymbol{T}^\top\tilde{\boldsymbol{\lambda}} \geq 0 \\ & & \tilde{\boldsymbol{\mu}},\tilde{\boldsymbol{\delta}},\tilde{\boldsymbol{\gamma}} \geq 0 \end{array} \tag{25}$$

where we multiply every decision variable by γ_0, so that $\tilde{\boldsymbol{\lambda}} = \gamma_0 \boldsymbol{\lambda}$, ..., and use the fact that $V^{\boxtimes}$ is positively homogeneous.

Now, we can derive the coperspective of $\mathcal{B}(V)$, using once again strong duality to interchange inf and sup:

$$
\begin{aligned}
\left[\mathcal{B}(V)\right]^{\boxtimes}(\pi_0, \gamma_0) &= \sup_{x_0} \pi_0^\top x_0 - \gamma_0 \mathcal{B}(V)(x_0) && (26)\\
&= \sup_{x_0} \pi_0^\top x_0 + \inf_{\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\mu}} \ \mathbb{E}\left[-\boldsymbol{\lambda}^\top(\boldsymbol{B}x_0 - \boldsymbol{d}) + V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{A}^\top\boldsymbol{\lambda}; \boldsymbol{\gamma})\right] \\
&\qquad\qquad \text{s.t.} \quad \mathbb{E}[\boldsymbol{\delta}] = \gamma_0 \\
&\qquad\qquad\qquad\;\; 0 \leq \boldsymbol{\delta} \leq \tfrac{\gamma_0}{1-\alpha} \\
&\qquad\qquad\qquad\;\; \boldsymbol{\gamma} = \boldsymbol{\delta} \\
&\qquad\qquad\qquad\;\; \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top\boldsymbol{\lambda} \geq 0 \\
&\qquad\qquad\qquad\;\; \boldsymbol{\mu} \geq 0 && (27)\\
&= \inf_{\boldsymbol{\lambda},\boldsymbol{\gamma},\boldsymbol{\delta},\boldsymbol{\mu}} \ \mathbb{E}\left[\boldsymbol{\lambda}^\top\boldsymbol{d} + V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{A}^\top\boldsymbol{\lambda}; \boldsymbol{\gamma})\right] && (28)\\
&\qquad \text{s.t.} \quad \mathbb{E}[\boldsymbol{B}^\top\boldsymbol{\lambda}] = \pi_0 \\
&\qquad\qquad\;\; \mathbb{E}[\boldsymbol{\delta}] = \gamma_0 \\
&\qquad\qquad\;\; \boldsymbol{\delta} \leq \tfrac{\gamma_0}{1-\alpha} \\
&\qquad\qquad\;\; \boldsymbol{\gamma} = \boldsymbol{\delta} \\
&\qquad\qquad\;\; \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top\boldsymbol{\lambda} \geq 0 \\
&\qquad\qquad\;\; \boldsymbol{\mu}, \boldsymbol{\gamma}, \boldsymbol{\delta} \geq 0
\end{aligned}
$$

Note that, besides the constraint $\mathbb{E}\left[\boldsymbol{B}^\top\boldsymbol{\lambda}\right] = \pi_0$, the optimization problem corresponding to the coperspective Bellman operator contains a further expectation constraint: $\mathbb{E}[\boldsymbol{\delta}] = \gamma_0$. In the same way as it was natural to relax the dual state constraint with a primal trajectory, we again notice that this constraint has a natural multiplier given by the primal parameter θ. The value of θ can be evaluated by decomposition, since the primal problem can be solved independently for each realization of ξ_t, and then the corresponding θ can be calculated using the optimal costs-to-go $\boldsymbol{z}$ in (21).

Also, notice that the expectation constraint $\mathbb{E}[\boldsymbol{\delta}] = \gamma_0$ is only a part of the constraint $\boldsymbol{\gamma} \in \gamma_0 Q$: indeed, the latter also includes the constraints $0 \leq \boldsymbol{\delta} \leq \frac{\gamma_0}{1-\alpha}$ and the (trivial) constraint $\boldsymbol{\gamma} = \boldsymbol{\delta}$. However, one only needs to relax the expectation constraint in order to obtain a scenario decomposition for the dual problem.

We won't prove the conjugacy of primal inner and relaxed dual outer updates for the AV@R risk measure here. Instead, we will provide a more general risk-averse setting in the next section, which we will use to derive a general conjugacy result.

4.2 Parametric risk measures

We consider the case where the risk measure $\rho(Z)$ is given by

$$\rho(Z) := \inf_{\theta \in \Theta} \mathbb{E}\left[\Psi(Z; \theta)\right],$$

for a fixed underlying probability, Ψ an *homogeneous* convex function of (z, θ), and Θ a convex cone. We also assume that Ψ is chosen so that the resulting risk measure ρ becomes translation-invariant. With these hypothesis, such a risk measure is coherent (convex, positively homogeneous).

From now on, we assume that ρ is well-defined for $Z \in L^p$, with $1 \le p < \infty$, so that the dual space is L^q for $1/p + 1/q = 1$

Proposition 11. *For such risk measures ρ, we have:*

1. $\rho(Z) = \sup_{q\in Q} \mathbb{E}[q \cdot Z]$, *where* $Q = \partial\rho(0)$*;*
2. $\sup_{Z,\theta\in\Theta} \mathbb{E}[\gamma \cdot Z - \Psi(Z;\theta)] = \mathbb{I}_Q(\gamma)$ *for* $\gamma \in L^q$*.*

Proof. The first result follows from homogeneity of ρ.

The second follows from the subdifferential characterization of the risk-set. Optimizing over θ yields:

$$\sup_{Z,\theta\in\Theta} \mathbb{E}[\gamma \cdot Z - \Psi(Z;\theta)] = \sup_{Z} \mathbb{E}[\gamma \cdot Z] - \rho(Z) = \rho^\star(\gamma) \tag{29}$$

If $\gamma \in Q$, then $\rho(Z) \ge \mathbb{E}[\gamma \cdot Z]$ for all Z ($\rho(0) = 0$ by homogeneity), so the supremum is attained at $Z = 0$. Otherwise, there exists Z such that $\rho(Z) < \mathbb{E}[\gamma \cdot Z]$; by homogeneity of ρ, this difference can be made arbitrarily large by scaling Z. $\square$

In this setting, proceeding analogouly to the AV@R case, the dual problem of

$$\begin{aligned} \mathcal{B}(V)(x_0) = \; & \inf_{\boldsymbol{x},\boldsymbol{y}} \quad \rho\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] && (30)\\ & \text{s.t.} \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \\ & \qquad\;\; \boldsymbol{x}, \boldsymbol{y} \ge 0 \\ = \; & \inf_{\boldsymbol{x},\boldsymbol{y};\theta,\boldsymbol{z}} \quad \mathbb{E}\left[\Psi(\boldsymbol{z};\theta)\right] && (31)\\ & \text{s.t.} \quad \boldsymbol{z} \ge \boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x}) \\ & \qquad\;\; \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \\ & \qquad\;\; \boldsymbol{x}, \boldsymbol{y} \ge 0 \end{aligned}$$

is given by

$$\begin{aligned} \mathcal{B}(V)(x_0) = \; & \sup_{\boldsymbol{\gamma},\boldsymbol{\lambda},\boldsymbol{\mu}} \quad \mathbb{E}\left[\boldsymbol{\lambda}^\top(\boldsymbol{B}x_0 - \boldsymbol{d}) - V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{\lambda}^\top \boldsymbol{A}, \gamma)\right] && (32)\\ & \qquad\quad + \inf_{\boldsymbol{z},\theta\in\Theta} \mathbb{E}\left[\Psi(\boldsymbol{z},\theta) - \boldsymbol{\gamma}\cdot\boldsymbol{z}\right] \\ & \text{s.t.} \quad \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \ge 0 \\ & \qquad\;\; \boldsymbol{\mu}, \boldsymbol{\gamma} \ge 0 \\ = \; & \sup_{\boldsymbol{\gamma},\boldsymbol{\lambda},\boldsymbol{\mu}} \quad \mathbb{E}\left[\boldsymbol{\lambda}^\top(\boldsymbol{B}x_0 - \boldsymbol{d}) - V^{\boxtimes}(\boldsymbol{\mu} - \boldsymbol{\lambda}^\top \boldsymbol{A}, \gamma)\right] && (33)\\ & \text{s.t.} \quad \gamma \in Q \\ & \qquad\;\; \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \ge 0 \\ & \qquad\;\; \boldsymbol{\mu}, \boldsymbol{\gamma} \ge 0 \end{aligned}$$

using Proposition 11. We can still apply strong duality if the problem is feasible, and mild regularity conditions on the risk generating function Ψ because then the remaining nonlinear constraint can be made strict increasing $\boldsymbol{z}$.

Therefore, the coperspective Bellman operator is

$$
\begin{aligned}
\mathcal{B}^{\boxtimes}(D)(\pi_0,\gamma_0) = \quad & \inf_{\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\gamma}} && \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi},\boldsymbol{\gamma})\right] && (34)\\
& \text{s.t.} && \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 \\
& && \boldsymbol{\gamma} \in \gamma_0 Q \\
& && \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\
& && \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0 \\
= \quad & \inf_{\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\gamma}} && \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi},\boldsymbol{\gamma})\right] && (35)\\
& && \quad + \sup_{\boldsymbol{z},\theta\in\Theta} \mathbb{E}\left[\boldsymbol{\gamma}\cdot \boldsymbol{z} - \gamma_0 \Psi(\boldsymbol{z},\theta)\right] \\
& \text{s.t.} && \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 \\
& && \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\
& && \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0
\end{aligned}
$$

Analogous to the case of AV@R, we again have a primal parameter θ which can be used to relax the coupling constraint $\boldsymbol{\gamma} \in \gamma_0 Q$, using the formulation in eq. (35). Also performing the Lagrangian relaxation of the expectation constraint $\mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0$, we obtain a decomposable dual problem:

$$
\begin{aligned}
\mathcal{B}^{\boxtimes}_{\hat{x},\hat{\theta}}(D)(\pi_0,\gamma_0) = \quad & \inf_{\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\gamma}} && \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi},\boldsymbol{\gamma})\right] + \hat{x}^\top(\pi_0 - \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}]) && (36)\\
& && \quad + \sup_{\boldsymbol{z}} \mathbb{E}\left[\boldsymbol{\gamma}\cdot \boldsymbol{z} - \gamma_0 \Psi(\boldsymbol{z},\hat{\theta})\right] \\
& \text{s.t.} && \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\
& && \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0.
\end{aligned}
$$

Again, it is easy to see that the function defined above is affine in π_0. If D is homogeneous in (π,γ), as it would be the case when $D = V^{\boxtimes}$, then the function is also affine in γ_0, and $\mathcal{B}^{\boxtimes}_{\hat{x},\hat{\theta}}$ maps homogeneous functions to homogeneous functions.

To complete the link between primal and dual recursions, we only need to show that the coefficient of γ_0 is indeed $-\mathcal{B}(V)(\hat{x})$, if $\hat{\theta}$ is chosen appropriately. Indeed:

Theorem 12. *Let V be a proper polyhedral function, and ρ a Ψ-generated risk measure. Let $\boldsymbol{t} = \dot{\mathcal{B}}(V)(\hat{x},\boldsymbol{\xi})$ be the optimal cost in each scenario ξ of the primal problem, and let $\hat{\theta} \in \arg\min_{\theta\in\Theta} \mathbb{E}\left[\Psi(\boldsymbol{t},\theta)\right]$. Then*

$$
\mathcal{B}^{\boxtimes}_{\hat{x},\hat{\theta}}(V^{\boxtimes})(\pi_0,\gamma_0) = \hat{x}^\top \pi_0 - \gamma_0 \mathcal{B}(V)(\hat{x}). \tag{37}
$$

Proof. From the discussion above, we only need to show that

$$
\begin{aligned}
\inf_{\boldsymbol{\lambda},\boldsymbol{\pi},\boldsymbol{\gamma}} \quad & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + V^{\boxtimes}(\boldsymbol{\pi},\boldsymbol{\gamma})\right] - \hat{x}^\top \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] \; = -\rho\left[\dot{\mathcal{B}}(V)(\hat{x},\boldsymbol{\xi})\right] && (38)\\
& \quad + \sup_{\boldsymbol{z}} \mathbb{E}\left[\boldsymbol{\gamma}\cdot \boldsymbol{z} - \Psi(\boldsymbol{z},\hat{\theta})\right] \\
\text{s.t.} \quad & \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\
& \boldsymbol{\gamma}\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0.
\end{aligned}
$$

where we set γ_0 to one. We again introduce the multipliers $\boldsymbol{x}$ and $\boldsymbol{y}$, interchange inf and sup by strong duality, and obtain that the left-hand side is equal to

$$
\begin{aligned}
\sup_{\boldsymbol{x},\boldsymbol{y}\geq 0} \inf_{\boldsymbol{\pi},\boldsymbol{\gamma}} \quad & \mathbb{E}\left[-\boldsymbol{\gamma}\boldsymbol{c}^\top \boldsymbol{y} - \boldsymbol{\pi}^\top \boldsymbol{x} + V^{\boxtimes}(\boldsymbol{\pi},\boldsymbol{\gamma}) + \sup_{\boldsymbol{z}} \mathbb{E}\left[\boldsymbol{\gamma}\cdot \boldsymbol{z} - \Psi(\boldsymbol{z},\hat{\theta})\right]\right] && (39)\\
\text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\hat{x} + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d},
\end{aligned}
$$

where we already eliminated the multiplier $\boldsymbol{\lambda}$. Starting with the infimum over $\boldsymbol{\pi}$, we get

$$\inf_{\boldsymbol{\pi}} \left[-\boldsymbol{\pi}^\top \boldsymbol{x} + V^{\boxtimes}(\boldsymbol{\pi}, \gamma) \right] = -\gamma V(\boldsymbol{x}),$$

and we now interchange $\inf_{\boldsymbol{\gamma}}$ and $\sup_{\boldsymbol{z}}$:

$$\sup_{\boldsymbol{z}} \inf_{\boldsymbol{\gamma}} \left[-\boldsymbol{\gamma} \boldsymbol{c}^\top \boldsymbol{y} - \boldsymbol{\gamma} V(\boldsymbol{x}) + \boldsymbol{\gamma} \cdot \boldsymbol{z} - \Psi(\boldsymbol{z}, \hat{\theta}) \right] = \begin{array}{rl} \sup_{\boldsymbol{z}} & -\Psi(\boldsymbol{z}, \hat{\theta}) \\ \text{s.t.} & \boldsymbol{z} = \boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x}) \end{array}$$

Substituting back in (39), we obtain

$$\begin{array}{rl} \sup\limits_{\boldsymbol{x}, \boldsymbol{y} \geq 0} \sup\limits_{\boldsymbol{z}} & \mathbb{E}\left[-\Psi(\boldsymbol{z}, \hat{\theta}) \right] \\ \text{s.t.} & \boldsymbol{z} = \boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x}) \\ & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\hat{x} + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \end{array} \tag{40}$$

which, upon interchanging the minus sign with the supremum, yields precisely the right-hand side of (37), since $\hat{\theta}$ realizes the risk measure ρ. □

5 Periodic setting

Another important setting in dynamic programming, going back to MDPs, is the periodic setting for infinite-horizon problems with discount factor β over each time step. For simplicity, we deal with the time-invariant (1-periodic) case, where the matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{T}$, and vectors $\boldsymbol{c}$ and $\boldsymbol{d}$ have the same distribution at every stage. Thus, we drop the time indices for these quantities, and the Bellman operator (41) is also time-invariant. Then an infinite, 1-periodic Bellman recursion

$$Q_t(x_{t-1}) = \mathcal{B}(Q_{t+1})(x_{t-1}) = \begin{array}{rl} \inf\limits_{\boldsymbol{x}_t, \boldsymbol{y}_t} & \mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y}_t + \beta Q_{t+1}(\boldsymbol{x}_t) \right] \\ \text{s.t.} & \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}x_{t-1} + \boldsymbol{T}\boldsymbol{y}_t = \boldsymbol{d} \\ & \boldsymbol{x}_t, \boldsymbol{y}_t \geq 0 \end{array} \tag{41}$$

becomes a fixed-point problem:

$$Q_t = \mathcal{B}(Q_{t+1}) = Q_{t+1}. \tag{42}$$

By Fenchel conjugacy, we deduce a dual Bellman recursion, which is only slightly different from the one we obtained from the finite-horizon one since it now includes the discount factor β:

$$D_t(\pi_{t-1}) = \mathcal{B}^\ddagger(D_{t+1})(\pi_{t-1}) = \begin{array}{rl} \sup\limits_{\boldsymbol{\pi}_t, \boldsymbol{\lambda}_t} & \mathbb{E}\left[\boldsymbol{d}^\top \boldsymbol{\lambda}_t + \beta D_{t+1}(\boldsymbol{\pi}_t) \right] \\ \text{s.t.} & \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}_t] = \pi_{t-1} \\ & \beta \boldsymbol{\pi}_t + \boldsymbol{A}^\top \boldsymbol{\lambda}_t \geq 0 \\ & \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda}_t \geq 0. \end{array} \tag{43}$$

Notice that, besides discounting the value function D_{t+1}, the dual Bellman recursion (43) also includes the discount factor β in one of the constraints. This also leads to a fixed-point equation:

$$D_t = \mathcal{B}^\ddagger(D_{t+1}) = D_{t+1}. \tag{44}$$

5.1 Lagrangian relaxation

Now, we can define the relaxed Bellman operator:

$$\begin{aligned} \mathcal{B}^{\downarrow}(D;\hat{x})(\pi_0) = \inf_{\boldsymbol{\lambda},\boldsymbol{\pi}} \quad & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi})\right] + \hat{x}^\top\left(\pi_0 - \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}]\right) \\ \text{s.t.} \quad & \beta\boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\ & \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0 \end{aligned} \tag{45}$$

which, as in the finite-horizon case, is a Lagrangian relaxation of the dual Bellman operator, and an affine function of π_0. Proceeding analogously to the finite-horizon case, we have:

$$\mathcal{B}^{\downarrow}(V^\star;\hat{x})(\pi_0) = \hat{x}^\top \pi_0 - \beta\mathcal{B}(V;\hat{x}), \tag{46}$$

for proper polyhedral functions V.

This shows that, in the same way as in the finite-horizon case, the cuts used in the relaxed dual updates are conjugate to the "pin functions" of primal inner updates. Therefore, the upper bounds constructed from either method are the same, provided they start from conjugate initial conditions and the relaxation $\hat{x}$ for the dual update is the trial point where the pin function will be evaluated in the primal update.

6 Conclusion

We have reviewed two different approaches for calculating deterministic upper bounds for multistage stochastic problems. The first one consists on applying the dynamic programming recursion on upper bounds of the value functions, producing a sequence of inner approximations. The second one builds, by convex duality, a dynamic programming recursion for the conjugates of the value functions, and deduces the upper bounds from outer approximations. The optimization problems in the dual approach have a coupling constraint, which prevents their decomposition as in the primal recursion. We then showed that a natural Lagrangian relaxation of the dual recursion corresponds to inner approximations of the primal value function.

This relation suggests that, on the basis of the number of iterations, the dual algorithm could converge faster than its dual relaxation, and therefore faster than the backwards primal inner approximations, which should be further investigated.

A Convex analysis

A (convex) polyhedron can be defined as the intersection of a finite number of half-spaces. A polyhedral function is a function whose epigraph is a polyhedron.

Theorem 13 (Minkowski-Weyl's theorem for polyhedra, see [Zie12, 1.2] or [Fuk16, Thm 3.9])**.** *For $P \subset \mathbb{R}^d$, the following statements are equivalent :*

1. *There exist $(a_i)_{i\in[q]} \in (\mathbb{R}^d)^q$ and $(b_i)_{i\in\mathbb{R}^q}$ such that $P := \left\{x \in \mathbb{R}^d \mid a_i^\top x \leq b_i, \forall i \in [q]\right\}$.*

2. *There exist finite families of vectors v_i and r_j in $\mathbb{R}^d$ such that* $P = \mathrm{Conv}(v_1, \ldots, v_s) + \mathrm{Cone}(r_1, \ldots, r_t)$

In particular, P is a polyhedron if and only if it satisfies one of these statements.

Definition 14. *V and H representation of polyhedral functions. TODO: look-up reference.*

We do not need nor use minimality of representation.

A.1 Properties of the Fenchel transform

Definition 15. *The Fenchel transform of a function $f : \mathbb{R}^n \to \mathbb{R}$ is $f^\star(\pi) := \sup_{x \in \mathbb{R}^n} \pi^\top x - f(x)$.*

Proposition 16 (Properties of the Fenchel transform)**.**

- *Any function $f^\star$ obtained by the Fenchel transform is convex.*
- *The Fenchel transform is monotonic decreasing: if $f \leq g$, then $f^\star \geq g^\star$.*
- *For a closed, proper, convex function, $[f^\star]^\star = f$.*

Moreover:

Proposition 17 (Conjugacy of max and convex hull)**.** *Let f and g be closed, proper, convex functions in R^n. Then*

$$[\max(f, g)]^\star = f^\star \blacktriangledown g^\star \tag{47}$$

$$[f \blacktriangledown g]^\star = \max(f^\star, g^\star) \tag{48}$$

In particular, this yields a natural correspondence between H-representations and V-representations for the conjugate:

Corollary 18. *If f is a polyhedral function given as a maximum of cuts,*

$$\begin{aligned} f(x) = \quad \min_\theta \quad & \theta \\ s.t. \quad & \theta \geq \pi_j^\top x + b_j \quad \forall j, \end{aligned}$$

then its conjugate is given by vertices of the epigraph:

$$\begin{aligned} f^\star(\pi) = \quad \min_{\alpha \geq 0} \quad & \textstyle\sum_j \alpha_j b_j \\ s.t. \quad & \textstyle\sum_j \alpha_j \pi_j = \pi \\ & \textstyle\sum_j \alpha_j = 1 \end{aligned}$$

A.2 Further conjugacy results

Proposition 19. *Assume that D is a closed, proper, convex function. Then*

$$\mathcal{B}^\dagger_{\hat{x}}(D)(\pi_0) = \hat{x}^\top \pi_0 - \mathcal{B}(D^\star)(\hat{x}). \tag{49}$$

In particular, if D is conjugate to a closed, proper, convex function V, then

$$\mathcal{B}^\dagger_{\hat{x}}(D)(\pi_0) = \hat{x}^\top \pi_0 - \mathcal{B}(V)(\hat{x}). \tag{50}$$

Proof. Recall that $\mathcal{B}^{\natural}_{\hat{x}}(D)$ is an affine function; the linear term $\hat{x}^\top \pi_0$ is immediate from (14). So we only need to prove that $\mathcal{B}^{\natural}_{\hat{x}}(D)(0) = -\mathcal{B}(D^\star)(\hat{x})$. It is natural to introduce multipliers $\boldsymbol{z}$ and $\boldsymbol{y}$, respectively, for the two constraints in (14), yielding

$$\begin{aligned} \inf_{\boldsymbol{\lambda},\boldsymbol{\pi}} \quad & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi})\right] - \hat{x}^\top \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] \\ & + \sup_{\boldsymbol{y},\boldsymbol{z}\geq 0} \mathbb{E}\left[-(\boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda})^\top \boldsymbol{z}\right] + \mathbb{E}\left[-(\boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda})^\top \boldsymbol{y}\right] \end{aligned} \tag{51}$$

Simplifying, then interchanging inf and sup by strong duality:

$$\begin{aligned} \mathcal{B}^{\natural}_{\hat{x}}(D)(0) = & \inf_{\boldsymbol{\lambda},\boldsymbol{\pi}} \sup_{\boldsymbol{y},\boldsymbol{z}\geq 0} \quad -\mathbb{E}[\boldsymbol{\lambda}^\top \boldsymbol{B}\hat{x}] + \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + D(\boldsymbol{\pi})\right] && (52) \\ & \qquad -\mathbb{E}\left[\boldsymbol{\pi}^\top \boldsymbol{z} + \boldsymbol{\lambda}^\top \boldsymbol{A}\boldsymbol{z}\right] - \mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y} + \boldsymbol{\lambda}^\top \boldsymbol{T}\boldsymbol{y}\right] \\ = & \sup_{\boldsymbol{y},\boldsymbol{z}\geq 0} \inf_{\boldsymbol{\lambda},\boldsymbol{\pi}} \quad -\mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y}\right] + \mathbb{E}\left[V^\star(\boldsymbol{\pi}) - \boldsymbol{\pi}^\top \boldsymbol{z}\right] && (53) \\ & \qquad -\mathbb{E}\left[\boldsymbol{\lambda}^\top (\boldsymbol{A}\boldsymbol{z} + \boldsymbol{B}\hat{x} + \boldsymbol{T}\boldsymbol{y} - \boldsymbol{d})\right] \\ = & \sup_{\boldsymbol{y},\boldsymbol{z}\geq 0} \quad -\mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y}\right] - \mathbb{E}\left[D^\star(\boldsymbol{z})\right] && (54) \\ & \quad \text{s.t.} \quad \boldsymbol{A}\boldsymbol{z} + \boldsymbol{B}\hat{x} + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \\ = & -\mathcal{B}(D^\star)(\hat{x}). \end{aligned}$$

□

Recalling that $\mathcal{B}^\ddagger$ was defined so that $\mathcal{B}^\ddagger(V^\star) = [\mathcal{B}(V)]^\star$, we see from definition 15 that $\mathcal{B}^\ddagger(D) = \sup_{\hat{x}} \mathcal{B}^{\natural}_{\hat{x}}(D)$ for closed, proper, convex functions D.

B Linear Bellman operators

B.1 Abstract Linear Bellman operators

As a way of unifying equations (2) and (3), we introduce *abstract Bellman operators.*

Formally speaking, an abstract Bellman operator $\mathcal{B}$ is a mapping of $\mathcal{L}(\mathbb{R}^m, \bar{\mathbb{R}})$ to $\mathcal{L}(\mathbb{R}^n, \bar{\mathbb{R}})$, where $\mathcal{L}(\mathbb{R}^m, \bar{\mathbb{R}})$ is the set of extended real-valued functions on $\mathbb{R}^m$. An abstract linear Bellman operator is an abstract Bellman operator $\mathcal{B}$ that can be written as

$$\begin{aligned} \mathcal{B}(V) : x_0 \mapsto \inf_{\boldsymbol{x},\boldsymbol{y}} \quad & \mathbb{E}_{\mathbb{P}}\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] && (55a) \\ \text{s.t.} \quad & \mathcal{A}(\boldsymbol{x}) + \mathcal{T}(\boldsymbol{y}) = \mathcal{D}(x_0) && (55b) \\ & \boldsymbol{x} \geq 0, \boldsymbol{y} \geq 0, && (55c) \end{aligned}$$

where $\mathcal{A}$, $\mathcal{D}$ and $\mathcal{T}$ are affine operators from a space of random vectors into $\mathbb{R}^d$, and $\mathbb{P}$ a reference probability measure. To fix ideas, assume that the reference probability measure is finitely supported (*i.e.*, $\mathrm{supp}(\boldsymbol{\xi}) = (\xi_j)_{j\in[|\Xi|]}$). Then, the random variables $\boldsymbol{x}$ (resp. $\boldsymbol{y}$) can be represented by vectors $\vec{x} = (\boldsymbol{x}(\xi_1), \dots, \boldsymbol{x}(\xi_{|\Xi|})$ of length $n_x \times |\Xi|$ concatenating the values of $\boldsymbol{x}$ for each realization ξ_j of the uncertainty. The same goes for $\boldsymbol{y}, \boldsymbol{x}$ and $\boldsymbol{d}$. Accordingly, the affine operators $\mathcal{A}$, $\mathcal{D}$ and $\mathcal{T}$ can be represented by matrices M_A, M_D and M_T of adequate size, such that the constraint (55b) can be written as

$M_A\vec{x}+M_T\vec{y}=M_D x_0+\vec{d}$. Thus, an abstract linear Bellman operator, evaluated at any x_0, for a discrete reference probability, can be written as a standard linear program:

$$\begin{aligned}
\mathcal{B}(V)(x_0) := \inf_{\vec{x},\vec{y}} \quad & \sum_{j=1}^{|\Xi|} \mathbb{P}(\boldsymbol{\xi}=\xi_j)\left[c^\top \vec{y}_j + V(\vec{x}_j)\right] && \text{(56a)}\\
\text{s.t.} \quad & M_A\vec{x}+M_T\vec{y}=\vec{d}-M_B x_0 && \text{(56b)}\\
& \vec{x}\geq 0, \vec{y}\geq 0, && \text{(56c)}
\end{aligned}$$

where, for ease of notation, $\vec{x}_j$ corresponds to the value taken by $\boldsymbol{x}$ for the realization ξ_j of the uncertainty.

Remark 20 (Example of abstract linear Bellman operators)**.** To make things more concrete, note that if $\mathcal{B}$ is the linear Bellman operator associated with the problem (1), we can see that M_A, M_B and M_T are block-diagonal, representing the fact that the constraints are independent for each realization of the uncertainty. However, the dual Bellman operators appearing in Problems (3) are not block-diagonal, as the problem are coupled through the expectation constraint, in which case the matrices M_A, M_B and M_T are not block-diagonal anymore, but instead have a L-shaped structure, i.e.,

$$M_A = \begin{pmatrix}
\boldsymbol{A}_t^\top(\boldsymbol{\xi}_1) & & & \\
 & \boldsymbol{A}_t^\top(\boldsymbol{\xi}_2) & & \\
 & & \ddots & \\
 & & & \boldsymbol{A}_t^\top(\boldsymbol{\xi}_{|\Xi|}) \\
\mathbb{P}(\boldsymbol{\xi}=\xi_1)\boldsymbol{B}_t^\top(\boldsymbol{\xi}_1) & \mathbb{P}(\boldsymbol{\xi}=\xi_2)\boldsymbol{B}_t^\top(\boldsymbol{\xi}_2) & \dots & \mathbb{P}(\boldsymbol{\xi}=\xi_{|\Xi|})\boldsymbol{B}_t^\top(\boldsymbol{\xi}_{|\Xi|})
\end{pmatrix}$$

Proposition 21. *Let $\mathcal{B}$ be an abstract linear Bellman operator associated with a finitely supported reference distribution. Then, $\mathcal{B}$ is monotous, convexity preserving and polyhedrality preserving,* i.e.,

i) if $V \leq W$, then $\mathcal{B}(V)\leq\mathcal{B}(W)$,

ii) if V is convex, then $\mathcal{B}(V)$ is convex,

iii) if V is polyhedral, then $\mathcal{B}(V)$ is polyhedral.

Proof. Obvious from the matricial expression (56). □

B.2 Expression of dual Bellman operators

$$
\begin{aligned}
[\mathcal{B}(V)]^\star(\pi_0) &= \sup_{x_0} \ \pi_0^\top x_0 - \mathcal{B}(V)(x_0) && (57)\\
&= \sup_{x_0} \ \pi_0^\top x_0 - \begin{array}[t]{rl} \inf\limits_{\boldsymbol{x},\boldsymbol{y}\geq 0} & \mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] \\ \text{s.t.} & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \end{array} && (58)\\
&= \begin{array}[t]{rl} \sup\limits_{x_0} \sup\limits_{\boldsymbol{x},\boldsymbol{y}\geq 0} & \pi_0^\top x_0 - \mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] \\ \text{s.t.} & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y} = \boldsymbol{d} \end{array} && (59)\\
&= \sup_{x_0} \sup_{\boldsymbol{x},\boldsymbol{y}\geq 0} \ \pi_0^\top x_0 - \mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] && (60)\\
&\qquad + \inf_{\boldsymbol{\lambda}} \ \mathbb{E}[\boldsymbol{\lambda}^\top(\boldsymbol{d} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{B}x_0 - \boldsymbol{T}\boldsymbol{y})] \\
&= \inf_{\boldsymbol{\lambda}} \ \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d}\right] + \sup_{x_0} \sup_{\boldsymbol{x},\boldsymbol{y}\geq 0} \ \mathbb{E}[-\boldsymbol{\lambda}^\top(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}x_0 + \boldsymbol{T}\boldsymbol{y})] \\
&\qquad + \pi_0^\top x_0 - \mathbb{E}\left[\boldsymbol{c}^\top \boldsymbol{y} + V(\boldsymbol{x})\right] && (61)\\
&= \begin{array}[t]{rl} \inf\limits_{\boldsymbol{\lambda}} & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d}\right] + \sup\limits_{\boldsymbol{x}\geq 0} \mathbb{E}[-\boldsymbol{\lambda}^\top \boldsymbol{A}\boldsymbol{x} - V(\boldsymbol{x})] \\ \text{s.t.} & \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 \\ & \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0 \end{array} && (62)\\
&= \begin{array}[t]{rl} \inf\limits_{\boldsymbol{\lambda},\boldsymbol{\pi}} & \mathbb{E}\left[\boldsymbol{\lambda}^\top \boldsymbol{d} + V^\star(\boldsymbol{\pi})\right] \\ \text{s.t.} & \mathbb{E}[\boldsymbol{B}^\top \boldsymbol{\lambda}] = \pi_0 \\ & \boldsymbol{\pi} + \boldsymbol{A}^\top \boldsymbol{\lambda} \geq 0 \\ & \boldsymbol{c} + \boldsymbol{T}^\top \boldsymbol{\lambda} \geq 0 \end{array} && (63)
\end{aligned}
$$

by, in order:

1. The definition of Fenchel dual;
2. The definition of $\mathcal{B}(V)$;
3. Interchanging $(-\inf)$ to $(\sup -)$;
4. Introducing the Lagrange multiplier $\boldsymbol{\lambda}$;
5. Strong duality to interchange inf and sup;
6. Eliminating x_0 and $\boldsymbol{y}$;
7. The definition of $V^\star$

References

[BDZ17] Regan Baucke, Anthony Downward, and Golbon Zakeri. A deterministic algorithm for solving multistage stochastic programming problems. *Optimization Online*, pages 1–25, 2017.

[dCL23] Bernardo Freitas Paulo da Costa and Vincent Leclère. Dual sddp for risk-averse multistage stochastic programs. *Operations Research Letters*, 51(3):332–337, 2023.

[FL23] Maël Forcier and Vincent Leclere. Convergence of trajectory following dynamic programming algorithms for multistage stochastic problems without finite support assumptions. *Journal of Convex Analysis*, page to appear, 2023.

[Fuk16] Komei Fukuda. Lecture: Polyhedral computation, spring 2016. 2016.

[GSC23] Vincent Guigues, Alexander Shapiro, and Yi Cheng. Duality and sensitivity analysis of multistage linear stochastic programs. *European Journal of Operational Research*, 308(2):752–767, 2023.

[GTW19] Angelos Georghiou, Angelos Tsoukalas, and Wolfram Wiesemann. Robust dual dynamic programming. *Operations Research*, 67(3):813–830, 2019.

[LCC$^+$20] Vincent Leclère, Pierre Carpentier, Jean-Philippe Chancelier, Arnaud Lenoir, and François Pacaud. Exact converging bounds for stochastic dual dynamic programming via fenchel duality. *SIAM Journal on Optimization*, 30(2):1223–1250, 2020.

[PdMF13] Andy Philpott, Vitor de Matos, and Erlon Finardi. On solving multistage stochastic programs with coherent risk measures. *Operations Research*, 61(4):957–970, 2013.

[PP91] Mario VF Pereira and Leontina MVG Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical programming*, 52:359–375, 1991.

[Zie12] Günter M. Ziegler. *Lectures on polytopes*, volume 152. Springer Science & Business Media, 2012.