

Time Block Decomposition of Multistage Stochastic Optimization Problems

Pierre Carpentier

*UMA, ENSTA Paris, Institut Polytechnique de Paris, Palaiseau, France
pierre.carpentier@ensta-paris.fr*

Jean-Philippe Chancelier, Michel De Lara, Thomas Martin

CERMICS, Ecole des Ponts ParisTech, Marne-la-Vallée, France

Tristan Rigaut

Efficacity, Marne-la-Vallée, France

Dedicated to Roger J-B Wets on the occasion of his 85th birthday.

Received: May 26, 2022

Accepted: February 22, 2023

Multistage stochastic optimization problems are, by essence, complex as their solutions are functions of both stages and uncertainties. Their large scale nature makes decomposition methods appealing, like dynamic programming which is a sequential decomposition using a state variable defined at all stages. By contrast, in this paper we introduce the notion of state reduction by time blocks, that is, at stages that are not necessarily all the original stages. Then, we prove a dynamic programming equation with value functions that are functions of a state only at some stages. This equation crosses over time blocks, but involves a dynamic optimization inside each block. We illustrate our contribution by showing its potential in three applications in multistage stochastic optimization: mixing dynamic programming and stochastic programming, two-time-scale optimization problems, decision-hazard-decision optimization problems.

Keywords: Multistage stochastic optimization, dynamic programming, time scales, time block decomposition, decision-hazard-decision.

2010 Mathematics Subject Classification: 60J05 90C15 90C39 90C06 49M27.

1. Introduction

Solutions of multistage stochastic optimization problems are functions of both time and uncertainties. This makes such problems complex. However, their structure makes decomposition methods appealing to solve them [18]. One of the most common approaches are time decomposition (state-based resolution methods), like stochastic dynamic programming, in stochastic optimal control, and scenario decomposition, like progressive hedging, in stochastic programming. On the one hand, stochastic programming deals with an underlying random process taking a finite number of values, called scenarios [19]. Solutions are indexed by a scenario tree, the size of which increases exponentially with the number of stages (hence generally a few stages in practice). However, to overcome this obstacle, stochastic programming takes advantage of scenario decomposition methods (progressive hedging [17]). On

the other hand, stochastic control deals with a state model driven by a white noise, that is, the noise is made of a sequence of independent random variables. Under such assumptions, stochastic dynamic programming is able to handle many stages, as it offers reduction of the search for a solution among state feedbacks (instead of functions of the past noise) [2, 16].

In a word, dynamic programming is good at handling multiple stages – but at the price of assuming that noises are stagewise independent – whereas stochastic programming does not require such assumption, but can only handle a few stages. Could we take advantage of both methods? Is there a way to apply stochastic dynamic programming at a slow time scale – a scale at which noises could be considered statistically independent – crossing over fast time scale optimization problems where independence would not hold? This question is one of the motivations of this paper, and we indeed provide a method to decompose multistage stochastic optimization problems by time blocks. This decomposition method and the main result are, mathematically speaking, quite natural, but the main difficulty is notational. Indeed, the rigorous formulation of multistage stochastic optimization problems on so-called history spaces requires rather heavy notation.

Although specialists in stochastic optimal control and dynamic programming will find the results as natural and non surprising, or as part of folklore, the fact is that we have not been able to find references that treat the case of *a state defined only at a subset of stages*. This is why we set out to write this paper, without any real theoretical ambition, but with the objective that this result be established and can be used for applications using several forms of decomposition¹. This is also why we present three (theoretical) applications in multistage stochastic optimization: mixing dynamic programming and stochastic programming, two-time-scale optimization problems, decision-hazard-decision optimization problems.

As there are several ways to tackle the difficulties of dealing with a large number of time steps, we compare our approach with other ones. In this paper, we propose an *exact decomposition* of a multistage stochastic optimization problem *by time blocks* using *a state defined only at a subset of stages*, to be distinguished from either *time aggregation* or *approximate decomposition by timescales*, which both yield *approximate* problems. We discuss both now.

Time aggregation consists in grouping the time steps, that is, in considering a partition of the time steps in time blocks and “aggregating” variables and constraints in each time block. To our knowledge, this approach was initiated in [4] for stochastic linear programs. For such linear programs, it is indeed easily conceived that, by summing (“aggregating”) linear constraints, one obtains lower bounds for minimization problems. This approach was generalized in the paper [25] which puts forward a measure-theoretic framework with coarser and finer filtrations, and uses linear duality. Then, this was extended in [13] for stochastic convex programs, using filtrations and convex duality. The main idea can be sketched as follows: the coarser

¹The starting point of our reflections on this subject were conversations that three of us held with Roger Wets in Bogota in 2013. We discussed the interest and the way of mixing the techniques of scenario trees (to be able to take into account correlated noises) with the techniques of dynamic programming (to have a vision of the optimal future costs by means of value functions).

filtration is used to reduce the measurability of the decision variables, whereas the finer filtration is used to enlarge the measurability of the dual variables associated with the constraints, so that the optimal value of the problem obtained by using these two filtrations is an upper bound of the true optimal value; exchanging the role of the filtrations leads to a lower bound. Thus, with time aggregation, one obtains simpler problems that are lower and upper bounds for the original minimization stochastic problems, hence are approximations.

In *approximate decomposition by timescales*, one identifies several timescales in the original multistage stochastic optimization problem and then sets up an optimization problem for each timescale. It is approximate in that the connexion between the problems formulated for each timescale and the whole multistage problem is not explicit.

Approximate decomposition by timescales can be done in the context of dynamic programming, with the value functions obtained for a given timescale entering the final cost of the problem at the finer timescale. This approach gives a cascade of easier to solve optimization problems, and again corresponds to approximate the original problem. An example of this approach can be found in [8] where – for a problem involving both the control of the storage of a battery (5 minutes time steps) and the frequency regulation (2 seconds time steps) – is introduced a first hourly resource model whose resolution by dynamic programming leads to value functions used in a five minute storage model as final costs. The value functions, obtained by solving by dynamic programming this second model, are themselves used in a 2 seconds frequency model. Another possibility arises when the considered optimization problem displays a *periodical behavior*. In that case, a natural time block decomposition is given by the period of the system. In [20], by taking into account such a periodical pattern in the dynamic programming equations, one significantly reduces the computational effort to solve the problem using a fixed point approach. Finally, [15] presents a preliminary work on extending the Stochastic Dual Dynamic Programming approach to two-time-scale problems, such as those encountered in energy systems involving both long-term hydro storages and short-term battery storages.

Approximate decomposition by timescales can also be done in the framework of stochastic programming. In [12], the authors introduce a slow scenario tree, that is, a tree involving only the time stages of the slow time scale; but at each node of this slow scenario tree are attached fast time scale scenarios, which do not interfere with the other nodes of the slow scenario tree. The structure allows one to model and solve problems that need to combine strategic (long term) and operational (short term) uncertainty, without the explosion in the problem size that would follow from using a standard multistage model. The special situation where decisions are taken only at the slow time scale (whereas uncertainties occur at each time stage) is considered in [10]. The authors propose to build a scenario tree branching at the slow time stages, and designed using the theory of bridge processes between two consecutive nodes in order to represent the noise at the fast time scale.

The paper is organized as follows. In Section 2, we present stochastic dynamic programming with histories as a way to solve a stochastic optimal control problem formulated in discrete time. In Section 3, we revisit the notion of “state” by defining

state reduction by time blocks – that is, at time stages that are not necessarily all the original ones – and then we prove a reduced dynamic programming equation. This is the central contribution. Then, we illustrate our contribution by showing its potential in three cases. In Section 4, we show how to mix dynamic programming and stochastic programming. In Section 5, we detail how to handle problems with two time scales, and illustrate this with the crude oil procurement problem. In Section 6, we introduce what we call the decision-hazard-decision framework, and we provide a dynamic programming equation. We relegate technical results and proofs to Appendix A.

2. Stochastic dynamic programming with histories

In § 2.1, we present the setting to formulate multistage stochastic optimization problems over the so-called history space, with history feedbacks. Then, to prepare the main result in Sect. 3, we establish in § 2.2 a dynamic programming equation when the state is the history, that is, is made of the uncertainties and the controls prior to the current time (see the “canonical construction” in [11, p. 15]). Although quite natural, this equation is generally not written in the literature, as most frameworks in dynamic programming assume the *a priori* existence of a state.

We use the notation $\llbracket r, s \rrbracket = \{r, r+1, \dots, s-1, s\}$ for any two natural numbers r, s such that $r \leq s$. We will also use the shorter notation $r:s = \llbracket r, s \rrbracket$, for example in subscripts as in $h_{r:s}$. From now on, time is discrete and runs among the natural numbers $t \in \llbracket 0, T \rrbracket$, where $T \in \mathbb{N}^*$ is a positive natural number. Finally, we say that a function is *numerical* if it takes its values in $\overline{\mathbb{R}} = [-\infty, +\infty]$.

2.1. The Bertsekas-Shreve setting [3]

To obtain a stochastic dynamic programming with histories requires technical assumptions. Indeed, as Bertsekas and Shreve notice at the beginning of [3, §7.6]: “The dynamic programming algorithm is centered around infimization of functions, and this is intimately connected with projections of sets”; “Unfortunately, the projection of a Borel-measurable set need not be Borel-measurable. In Borel spaces, however, the projection of a Borel-measurable set is an analytic set”. They devote [3, §7.6] to the definition and study of analytic sets, and in [3, §7.7] define universally measurable functions, as well as lower semianalytic functions.

We call *Borel space* $(\mathbb{X}, \mathcal{B}_{\mathbb{X}})$ a Borel set $\mathbb{X}$ equipped with its Borel σ -field $\mathcal{B}_{\mathbb{X}}$ [3, Definition 7.7, p. 118]. By abuse of notation, we often speak of the Borel space $\mathbb{X}$. There exist two other interesting σ -fields: the *analytic σ -field* $\mathcal{A}_{\mathbb{X}}$ [3, Definition 7.19, p. 171]; the *universal σ -field* $\mathcal{U}_{\mathbb{X}}$ [3, Definition 7.18, p. 167]. We have the inclusions $\mathcal{B}_{\mathbb{X}} \subset \mathcal{A}_{\mathbb{X}} \subset \mathcal{U}_{\mathbb{X}}$ [3, p. 171].

For any Borel space $\mathbb{X}$, subset $X \subset \mathbb{X}$ and numerical function $\varphi : X \rightarrow \overline{\mathbb{R}}$, the function φ is said to be *lower semianalytic* [3, Definition 7.21] if the subset X is analytic ($X \in \mathcal{A}_{\mathbb{X}}$) and if the subset $\{x \in X \mid \varphi(x) < c\}$ is analytic for all $c \in \mathbb{R}$. We denote by $\mathbb{L}_+^0(\mathbb{X})$ the space of lower semianalytic nonnegative numerical functions over $\mathbb{X}$.

For a Borel space $\mathbb{X}$ (resp. $\mathbb{Y}$) equipped with the Borel σ -field $\mathcal{B}_{\mathbb{X}}$ (resp. $\mathcal{B}_{\mathbb{Y}}$), a mapping $f : \mathbb{X} \rightarrow \mathbb{Y}$ is said to be *universally measurable* [3, Definition 7.20, p. 171] (resp. *Borel-measurable*) if, for all $B \in \mathcal{B}_{\mathbb{Y}}$, $f^{-1}(B) \in \mathcal{U}_{\mathbb{X}}$ (resp. $f^{-1}(B) \in \mathcal{B}_{\mathbb{X}}$).

Histories and history spaces. For each time $t \in \llbracket 0, T-1 \rrbracket$, the control u_t takes its values in a Borel space $\mathbb{U}_t$. For each time $t \in \llbracket 0, T \rrbracket$, the uncertainty w_t takes its values in a Borel space $\mathbb{W}_t$. For $t \in \llbracket 0, T \rrbracket$, we define the *history space* $\mathbb{H}_t$ as the product Borel space [3, Proposition 7.13, p. 119]

$$\mathbb{H}_t = \mathbb{W}_0 \times \prod_{s=1}^t (\mathbb{U}_{s-1} \times \mathbb{W}_s) , \quad (1)$$

with the particular case $\mathbb{H}_0 = \mathbb{W}_0$. A generic element

$$h_t = (w_0, (u_{s-1}, w_s)_{s=1, \dots, t}) = (w_0, u_0, w_1, u_1, w_2, \dots, u_{t-2}, w_{t-1}, u_{t-1}, w_t) \in \mathbb{H}_t$$

is called a *history* at time t . For $1 \leq r \leq s \leq t$, we introduce the $(r : s)$ -*history subpart* $h_{r:s} = (u_{r-1}, w_r, \dots, u_{s-1}, w_s) \in \mathbb{H}_{r:s} = \prod_{\tau=r}^s (\mathbb{U}_{\tau-1} \times \mathbb{W}_\tau)$, so that we have $h_t = (h_{r-1}, h_{r:t})$.

History feedbacks. For $0 \leq r \leq t \leq T-1$, we define a $(r : t)$ -*history feedback* as a sequence $\{\gamma_s\}_{s \in \llbracket r, t \rrbracket}$ of universally measurable mappings $\gamma_s : \mathbb{H}_s \rightarrow \mathbb{U}_s$. We call $\Gamma_{r:t}$ the set of $(r : t)$ -history feedbacks. The history feedbacks reflect the following information structure. At the end of the time interval $[t-1, t[$, an uncertainty variable w_t is revealed. Then, at the beginning of the time interval $[t, t+1[$, a decision-maker chooses a control u_t contingent on no more than the past, giving the chronology

$$w_0 \rightsquigarrow u_0 \rightsquigarrow w_1 \rightsquigarrow u_1 \rightsquigarrow \dots \rightsquigarrow w_t \rightsquigarrow u_t \rightsquigarrow \dots \rightsquigarrow w_{T-1} \rightsquigarrow u_{T-1} \rightsquigarrow w_T . \quad (2)$$

Stochastic kernels. In what follows, given a Borel space $\mathbb{Y}$, $\Delta(\mathbb{Y})$ denotes the Borel space of probability measures over $\mathbb{Y}$ (see [3, Corollary 7.25.1]). Uncertainty is represented by a sequence $\{\rho_{t-1:t}\}_{t \in \llbracket 1, T \rrbracket}$ of Borel-measurable stochastic kernels (see [3, Definition 7.12 and Proposition 7.26, p. 134])

$$\rho_{t-1:t} : \mathbb{H}_{t-1} \rightarrow \Delta(\mathbb{W}_t) , \quad \forall t \in \llbracket 1, T \rrbracket . \quad (3)$$

Thus, for any past history $h_{t-1} \in \mathbb{H}_{t-1}$, we have that $\rho_{t-1:t}(h_{t-1}) \in \Delta(\mathbb{W}_t)$, the space of probability measures over $\mathbb{W}_t$. It is common practice (see [3, Definition 7.12, p. 134]) to use the notation $\rho_{t-1:t}(dw_t | h_{t-1})$ to denote this probability distribution, element of $\Delta(\mathbb{W}_t)$. So, the notation $|h_t$ is here to *evoke* a conditional distribution (of the next uncertainty knowing the past history), but it is not introduced as a conditional distribution, but simply as a way to express a *parametric dependence* (as explicitly said in [3, Definition 7.12, p. 134]). We could have indifferently written $\rho_{t-1:t}(dw_t, h_{t-1})$ or $\rho_{t-1:t}(dw_t; h_{t-1})$.

We define, for any history feedback $\{\gamma_s\}_{s \in \llbracket t, T-1 \rrbracket} \in \Gamma_{t:T-1}$, a new sequence of Borel-measurable stochastic kernels $\rho_{t:T}^\gamma : \mathbb{H}_t \rightarrow \Delta(\mathbb{H}_T)$, that capture the transitions between histories when the dynamics $h_{s+1} = (h_s, u_s, w_{s+1})$ is driven by $u_s = \gamma_s(h_s)$ for all s in $\llbracket t, T-1 \rrbracket$ (see Definition A.1 in Appendix A for the detailed construction of $\rho_{t:T}^\gamma$). Note that $\rho_{t:T}^\gamma$ generates a probability distribution on the space $\mathbb{H}_T$ of histories over the whole timespan $\llbracket 0, T \rrbracket$.

Cost function. The cost criterion to be minimized is a nonnegative² and lower semianalytic numerical function

$$j : \mathbb{H}_T \rightarrow [0, +\infty] . \quad (4)$$

Notice that (4) does not represent a cost at final time, but a cost function of the whole history $h_T = (w_0, u_0, w_1, u_1, \dots, w_{T-1}, u_{T-1}, w_T) \in \mathbb{H}_T$. As h_T contains *all* past controls and uncertainties, a function $j : \mathbb{H}_T \rightarrow [0, +\infty]$ covers the most general case. For instance, the function j can have the special form of a sum of time block costs, like in Equation (16).

2.2. Stochastic dynamic programming equation with histories

Family of optimization problems. We consider the following family of optimization problems, indexed by t in $\llbracket 0, T-1 \rrbracket$ and parameterized by the history $h_t \in \mathbb{H}_t$: for all t in $\llbracket 0, T-1 \rrbracket$, we define the minimum value

$$V_t(h_t) = \inf_{\gamma_{t:T-1} \in \Gamma_{t:T-1}} \int_{\mathbb{H}_T} j(h'_T) \rho_{t:T}^\gamma(dh'_T | h_t) , \quad \forall h_t \in \mathbb{H}_t , \quad (5a)$$

and we also define

$$V_T(h_T) = j(h_T) , \quad \forall h_T \in \mathbb{H}_T . \quad (5b)$$

The numerical function $V_t : \mathbb{H}_t \rightarrow [0, +\infty]$ is called the *value function* at time t .

Next, we show how the sequence $\{V_t\}_{t \in \llbracket 0, T \rrbracket}$ of value functions can be used to solve, via dynamic programming, the optimization problem of interest, that is, the one starting at $t = 0$, whose value is (recall that $h_0 = w_0$)

$$V_0(w_0) = \inf_{\gamma_{0:T-1} \in \Gamma_{0:T-1}} \int_{\mathbb{H}_T} j(h'_T) \rho_{0:T}^\gamma(dh'_T | w_0) . \quad (6)$$

Bellman operators and dynamic programming. We show that the value functions in (5) are *Bellman functions*, in that they are solution of a Bellman or dynamic programming equation.

Theorem 2.1. *We suppose to be in the setting of § 2.1. For t in $\llbracket 0, T-1 \rrbracket$, we define the Bellman operator $\mathcal{B}_{t+1:t}$ by, for all $\varphi \in \mathbb{L}_+^0(\mathbb{H}_{t+1})$ and for all $h_t \in \mathbb{H}_t$,*

$$(\mathcal{B}_{t+1:t}\varphi)(h_t) = \inf_{u_t \in \mathbb{U}_t} \int_{\mathbb{W}_{t+1}} \varphi(h_t, u_t, w_{t+1}) \rho_{t:t+1}(dw_{t+1} | h_t) . \quad (7a)$$

Then, the Bellman operators are such that

$$\mathcal{B}_{t+1:t} : \mathbb{L}_+^0(\mathbb{H}_{t+1}) \rightarrow \mathbb{L}_+^0(\mathbb{H}_t) , \quad (7b)$$

and the value functions V_t defined in (5) are lower semianalytic and satisfy the Bellman equation, or (stochastic) dynamic programming equation,

$$V_T = j , \quad V_t = \mathcal{B}_{t+1:t} V_{t+1} , \quad \text{for } t \in \llbracket 0, T-1 \rrbracket . \quad (8)$$

² We could also consider a cost criterion $j : \mathbb{H}_t \rightarrow \overline{\mathbb{R}}$, either bounded function, or uniformly bounded below function. However, for the sake of simplicity, we will deal in the sequel with nonnegative numerical functions. The case $j(h_T) = +\infty$ materializes joint constraints between uncertainties and controls in h_T .

The proof is sketched in Appendix A. This theorem is inspired by [3, Chap. 8], with the feature that the state x_t is, in our case, the canonical history h_t , with the canonical dynamics $h_{t+1} = (h_t, u_t, w_{t+1})$. This quite general dynamic programming result is the basis of all future developments in this paper. Although the recalls and statements presented in this Section 2 are mostly straightforward consequences of results already established in the literature, the developments are indispensable to tackle time block decomposition in the forthcoming Section 3.

3. State reduction by time blocks and dynamic programming

In standard approaches to solve, by dynamic programming, a stochastic optimal control problem formulated in discrete time, either a state is given for all times (as in [3], [11], [16] and [24]), or no state is given (as in [9],[26]). In this paper, our approach is intermediate, in that a state is possibly obtained, but only at certain times. Thus, in this section, we consider the question of reducing the history using a compressed “state” variable. Differing with traditional practice, such a variable may not be available at any time $t \in \llbracket 0, T \rrbracket$, but at some specified times $0 = t_0 < \dots < t_N = T$.

We have recalled in Section 2 that the history h_t is itself a state variable with associated canonical dynamics $h_{t+1} = (h_t, u_t, w_{t+1})$. However, the size of this canonical state increases with time t , which is an unpleasant feature for dynamic programming – quickly leading to the well-known curse of dimensionality – hence the practical need to introduce a (ideally low dimensional) state space, at least at some specified times, as done in this paper. As already said in the introduction, the main difficulty in achieving this goal is notational.

In § 3.1, we start by introducing the notion of state reduction on a single time block. In § 3.2, we move to state reduction on multiple consecutive time blocks and we give the corresponding dynamic programming equations across time blocks. In § 3.3, we conclude on how we obtain reduced optimal feedbacks.

3.1. State reduction on a single time block

We first present the case where the reduction only occurs at two times denoted by r and t , and such that $0 \leq r < t \leq T$.

Definition 3.1. Let $\mathbb{X}_r$ and $\mathbb{X}_t$ be two Borel *state spaces*, θ_r and θ_t be two Borel-measurable *reduction mappings*

$$\theta_r : \mathbb{H}_r \rightarrow \mathbb{X}_r, \quad \theta_t : \mathbb{H}_t \rightarrow \mathbb{X}_t, \quad (9a)$$

and $f_{r:t}$ be a Borel-measurable *dynamics*

$$f_{r:t} : \mathbb{X}_r \times \mathbb{H}_{r+1:t} \rightarrow \mathbb{X}_t. \quad (9b)$$

The triplet $(\theta_r, \theta_t, f_{r:t})$ is called a *state reduction across* $\llbracket r, t \rrbracket$ if we have³

$$\theta_t((h_r, h_{r+1:t})) = f_{r:t}(\theta_r(h_r), h_{r+1:t}), \quad \forall h_t \in \mathbb{H}_t. \quad (9c)$$

³ Notice that, if only the couple $(\theta_r, f_{r:t})$ is given, we can define the reduction mapping θ_t by (9c), and thus obtain a triplet $(\theta_r, \theta_t, f_{r:t})$ which is a state reduction across $\llbracket r, t \rrbracket$.

Finally, the state reduction $(\theta_r, \theta_t, f_{r:t})$ is said to be *compatible* with the sequence $\{\rho_{s-1:s}\}_{r+1 \leq s \leq t}$ of Borel-measurable stochastic kernels (3) if

- there exists a Borel-measurable *reduced stochastic kernel* $\tilde{\rho}_{r:r+1} : \mathbb{X}_r \rightarrow \Delta(\mathbb{W}_{r+1})$, such that the stochastic kernel $\rho_{r:r+1}$ in (3) can be factored, for all $h_r \in \mathbb{H}_r$, as

$$\rho_{r:r+1}(dw_{r+1} \mid h_r) = \tilde{\rho}_{r:r+1}(dw_{r+1} \mid \theta_r(h_r)),$$

- for all s in $\llbracket r+2, t \rrbracket$, there exists a Borel-measurable *reduced stochastic kernel* $\tilde{\rho}_{s-1:s} : \mathbb{X}_r \times \mathbb{H}_{r+1:s-1} \rightarrow \Delta(\mathbb{W}_s)$, such that the stochastic kernel $\rho_{s-1:s}$ can be factored, for all $h_{s-1} \in \mathbb{H}_{s-1}$, as

$$\rho_{s-1:s}(dw_s \mid (h_r, h_{r+1:s-1})) = \tilde{\rho}_{s-1:s}(dw_s \mid (\theta_r(h_r), h_{r+1:s-1})).$$

The above definition is similar to the sufficient statistics idea in stochastic control: the state variable, which summarizes the history, is sufficient for the controller to design its control policy ([23, p. 19], [3, Definition 10.6], [22]). However, sufficient statistics in the stochastic control literature are defined at the original time stages. By contrast, Definition 3.1 – and the coming Definition 3.3 – consider a notion of sufficient statistics *only for a subset of time stages*.

According to Definition 3.1, the triplet $(\theta_r, \theta_t, f_{r:t})$ is a state reduction across $\llbracket r, t \rrbracket$ if and only if the diagram in the left part of Figure 3.1 is commutative; it is compatible if and only if the diagram in the middle part of Figure 3.1 is commutative.

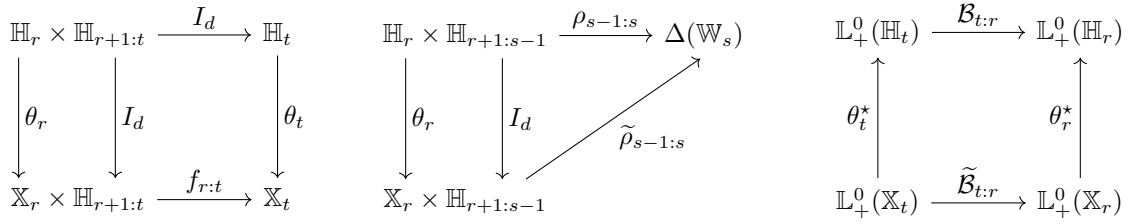


Figure 3.1: Commutative diagrams in case of state reduction

The following proposition is the key ingredient to formulate dynamic programming equations with a reduced state.

Proposition 3.2. *Under the assumptions of § 2.1, we define the Bellman operator across $\llbracket t, r \rrbracket$, $\mathcal{B}_{t:r} : \mathbb{L}_+^0(\mathbb{H}_t) \rightarrow \mathbb{L}_+^0(\mathbb{H}_r)$ by*

$$\mathcal{B}_{t:r} = \mathcal{B}_{r+1:r} \circ \cdots \circ \mathcal{B}_{t:t-1}, \quad (10)$$

where the one time step operators $\mathcal{B}_{s:s-1}$, for s in $\llbracket r+1, t \rrbracket$ are defined in (7a).

Suppose that there exists a state reduction $(\theta_r, \theta_t, f_{r:t})$ that is compatible with the sequence $\{\rho_{s-1:s}\}_{s \in \llbracket r+1, t \rrbracket}$ of stochastic kernels (3). Then, there exists a reduced Bellman operator across $\llbracket t, r \rrbracket$, $\tilde{\mathcal{B}}_{t:r} : \mathbb{L}_+^0(\mathbb{X}_t) \rightarrow \mathbb{L}_+^0(\mathbb{X}_r)$, such that

$$(\tilde{\mathcal{B}}_{t:r} \tilde{\varphi}_t) \circ \theta_r = \mathcal{B}_{t:r}(\tilde{\varphi}_t \circ \theta_t), \quad \forall \tilde{\varphi}_t \in \mathbb{L}_+^0(\mathbb{X}_t). \quad (11)$$

For any $\tilde{\varphi}_t \in \mathbb{L}_+^0(\mathbb{X}_t)$ and for any $x_r \in \mathbb{X}_r$, we have that

$$\begin{aligned} (\tilde{\mathcal{B}}_{t:r} \tilde{\varphi}_t) &= \inf_{u_r \in \mathbb{U}_r} \int_{\mathbb{W}_{r+1}} \tilde{\rho}_{r:r+1}(dw_{r+1} \mid x_r) \\ &\quad \inf_{u_{r+1} \in \mathbb{U}_{r+1}} \int_{\mathbb{W}_{r+2}} \tilde{\rho}_{r+1:r+2}(dw_{r+2} \mid x_r, u_r, w_{r+1}) \quad \cdots \\ &\quad \inf_{u_{t-1} \in \mathbb{U}_{t-1}} \int_{\mathbb{W}_t} \tilde{\rho}_{t-1:t}(dw_t \mid x_r, u_r, w_{r+1}, \dots, u_{t-2}, w_{t-1}) \\ &\quad \tilde{\varphi}_t(f_{r:t}(x_r, u_r, w_{r+1}, \dots, u_{t-1}, w_t)) . \end{aligned} \quad (12)$$

The formula (12) represents a nested sequence of infima of integrals (with respect to different stochastic kernels).

The proof of Proposition 3.2 is given in Appendix A. Proposition 3.2 can be interpreted as follows. Denoting by $\theta_t^* : \mathbb{L}_+^0(\mathbb{X}_t) \rightarrow \mathbb{L}_+^0(\mathbb{H}_t)$ the operator defined by $\theta_t^*(\tilde{\varphi}_t) = \tilde{\varphi}_t \circ \theta_t$ for any $\tilde{\varphi}_t \in \mathbb{L}_+^0(\mathbb{X}_t)$, the relation (11) rewrites as $\theta_r^* \circ \tilde{\mathcal{B}}_{t:r} = \mathcal{B}_{t:r} \circ \theta_t^*$, that is, Proposition 3.2 states that the diagram in the right part of Figure 3.1 is commutative.

3.2. State reduction on multiple consecutive time blocks and dynamic programming equations

Proposition 3.2 can easily be extended to the case of multiple consecutive time blocks $\llbracket t_i, t_{i+1} \rrbracket$, with $N \in \mathbb{N}^*$, $i \in \llbracket 0, N-1 \rrbracket$ and $0 = t_0 < \dots < t_N = T$.

Definition 3.3. Let $\{\mathbb{X}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ be a family of Borel state spaces, $\{\theta_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ be a family of Borel-measurable reduction mappings $\theta_{t_i} : \mathbb{H}_{t_i} \rightarrow \mathbb{X}_{t_i}$, and $\{f_{t_i:t_{i+1}}\}_{i \in \llbracket 0, N-1 \rrbracket}$ be a family of Borel-measurable dynamics

$$f_{t_i:t_{i+1}} : \mathbb{X}_{t_i} \times \mathbb{H}_{t_{i+1}:t_{i+1}} \rightarrow \mathbb{X}_{t_{i+1}} .$$

The family $(\{\mathbb{X}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{\theta_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{f_{t_i:t_{i+1}}\}_{i \in \llbracket 0, N-1 \rrbracket})$ is called a *state reduction across the consecutive time blocks* $\llbracket t_i, t_{i+1} \rrbracket$, where $i \in \llbracket 0, N-1 \rrbracket$, if every triplet $(\theta_{t_i}, \theta_{t_{i+1}}, f_{t_i:t_{i+1}})$ is a state reduction across $\llbracket t_i, t_{i+1} \rrbracket$, for $i \in \llbracket 0, N-1 \rrbracket$.

The state reduction across the consecutive time blocks $\llbracket t_i, t_{i+1} \rrbracket$ is said to be *compatible* with the family $\{\rho_{s-1:s}\}_{s \in \llbracket 1, T \rrbracket}$ of stochastic kernels given in (3) if every triplet $(\theta_{t_i}, \theta_{t_{i+1}}, f_{t_i:t_{i+1}})$ is compatible with the family $\{\rho_{s-1:s}\}_{s \in \llbracket t_i+1, t_{i+1} \rrbracket}$, for $i \in \llbracket 0, N-1 \rrbracket$.

Remark 3.4. (Composed state dynamics as a reduction mapping) There is a practical case where state reductions can readily be obtained, namely, when the model is given by controlled state dynamics driven by noises. In that case, we are given a sequence $\{\mathbb{X}_s\}_{s \in \llbracket 0, T \rrbracket}$ of Borel state spaces and a sequence $\{f_{s:s+1}\}_{s \in \llbracket 0, T-1 \rrbracket}$ of Borel-measurable dynamics

$$f_{s:s+1} : \mathbb{X}_s \times \mathbb{U}_s \times \mathbb{W}_{s+1} \rightarrow \mathbb{X}_{s+1} . \quad (13)$$

For any time $s \in \llbracket 0, T-1 \rrbracket$, we define the composition $f_{0:s+1} = f_{s:s+1} \circ f_{s-1:s} \circ \dots \circ f_{0:1}$ with the abuse of notation that the composition is performed on the state argument. Setting $\mathbb{W}_0 = \mathbb{X}_0$, we obtain that $f_{0:s+1} : \mathbb{H}_{s+1} \rightarrow \mathbb{X}_{s+1}$ is a Borel-measurable mapping from the history space $\mathbb{H}_{s+1}$ taking values in the state space $\mathbb{X}_{s+1}$.

Given a natural number $N > 0$ and an increasing sequence $0 = t_0 < \dots < t_N = T$ of times, we now define the sequence $\{\theta_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ of Borel-measurable reduction mappings by $\theta_{t_i} = f_{0:t_i} : \mathbb{H}_{t_i} \rightarrow \mathbb{X}_{t_i}$ for $i > 0$, and by $\theta_0 = I_d$ (the identity mapping on $\mathbb{W}_0$) for $i = 0$. Moreover, given i and $j \in \llbracket 0, N \rrbracket$, with $i < j$ we obtain, for all $h_{t_j} \in \mathbb{H}_{t_j}$, that

$$\theta_{t_j}(h_{t_j}) = \theta_{t_j}((h_{t_i}, h_{t_i+1:t_j})) = f_{t_i:t_j}(\theta_{t_i}(h_{t_i}), h_{t_i+1:t_j}), \quad (14)$$

with $f_{t_i:t_j} = f_{t_j-1:t_j} \circ f_{t_j-2:t_j-1} \circ \dots \circ f_{t_i:t_i+1}$ which gives the state reduction Equation (9c). $\square$

Remark 3.5. (Block independent exogenous noises and stochastic kernels) There is a practical case where *compatible* state reductions can readily be obtained. Assume that the sequence $\{\rho_{s-1:s}\}_{s \in \llbracket 1, T \rrbracket}$ of stochastic kernels in (3) are mappings whose arguments do not include the control part (that is, depend at most on the history uncertainty part (see (44a)). If we interpret stochastic kernels as (conditional) distributions of noises (random process), this means that the system dynamics are driven by an exogenous noise process, say $\{\mathbf{W}_t\}_{t \in \llbracket 1, T \rrbracket}$.

Assume moreover that the stochastic kernels give rise to noises that are independent block by block, in the sense that the random vectors $\mathbf{W}_0, (\mathbf{W}_t)_{t \in \llbracket 1, t_1 \rrbracket}, (\mathbf{W}_t)_{t \in \llbracket t_1+1, t_2 \rrbracket}, \dots, (\mathbf{W}_t)_{t \in \llbracket t_i+1, t_{i+1} \rrbracket}, \dots, (\mathbf{W}_t)_{t \in \llbracket t_{N-2}+1, t_{N-1} \rrbracket}, (\mathbf{W}_t)_{t \in \llbracket t_{N-1}+1, t_N \rrbracket}$ are stochastically independent. Then, from Definitions 3.1 and 3.3, we deduce that any state reduction across the same time blocks is compatible with the stochastic kernels. $\square$

Assuming now the existence of a state reduction across the consecutive time blocks $\llbracket t_i, t_{i+1} \rrbracket$ compatible with the sequence of stochastic kernels (3), we obtain the existence of a *sequence of reduced Bellman operators* across the time blocks $\llbracket t_i, t_{i+1} \rrbracket$ as an immediate consequence of multiple applications of Proposition 3.2, that is, $\tilde{\mathcal{B}}_{t_{i+1}:t_i} : \mathbb{L}_+^0(\mathbb{X}_{t_{i+1}}) \rightarrow \mathbb{L}_+^0(\mathbb{X}_{t_i})$, $i \in \llbracket 0, N-1 \rrbracket$, such that we have, for any function $\tilde{\varphi}_{t_{i+1}} \in \mathbb{L}_+^0(\mathbb{X}_{t_{i+1}})$, the equality $(\tilde{\mathcal{B}}_{t_{i+1}:t_i} \tilde{\varphi}_{t_{i+1}}) \circ \theta_{t_i} = \mathcal{B}_{t_{i+1}:t_i}(\tilde{\varphi}_{t_{i+1}} \circ \theta_{t_{i+1}})$.

We now consider the family of optimization problems defined by the associated value functions (5). Thanks to the state reductions, we can enounce the following two theorems which establish dynamic programming equations *across* consecutive time blocks. The first one, Theorem 3.6, states a dynamic programming equation for an optimization problem in Mayer form (that is, just involving a final cost). The second one, Theorem 3.7, is more general as it involves both instantaneous costs and a final cost. As it is well known that the second case can be reduced to a Mayer form through a state augmentation, the proof of Theorem 3.7 easily follows from the proof of Theorem 3.6.

Theorem 3.6. (Time block decomposition for the Mayer form) *We assume to be in the setting of §2.1. Suppose that a state reduction $(\{\mathbb{X}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{\theta_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{f_{t_i:t_{i+1}}\}_{i \in \llbracket 0, N-1 \rrbracket})$ exists across the consecutive time blocks $\llbracket t_i, t_{i+1} \rrbracket$, satisfying $0 = t_0 < \dots < t_N = T$, which is compatible with the sequence $\{\rho_{s-1:s}\}_{s \in \llbracket 1, T \rrbracket}$ of stochastic kernels given in (3).*

Suppose that there exists a reduced cost criterion $\tilde{j} : \mathbb{X}_T \rightarrow [0, +\infty]$, which is a nonnegative lower semianalytic function and is such that the cost function j in (4)

can be factored as $j = \tilde{j} \circ \theta_T$. We define the sequence of reduced value functions $\{\tilde{V}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$, where $\tilde{V}_{t_i} : \mathbb{X}_{t_i} \rightarrow [0, +\infty]$ for $i \in \llbracket 0, N \rrbracket$, by

$$\tilde{V}_{t_N} = \tilde{j} \text{ and } \tilde{V}_{t_i} = \tilde{\mathcal{B}}_{t_{i+1}:t_i} \tilde{V}_{t_{i+1}}, \quad \forall i \in \llbracket 0, N-1 \rrbracket, \quad (15)$$

where the reduced Bellman operators $\{\tilde{\mathcal{B}}_{t_{i+1}:t_i}\}_{i \in \llbracket 0, N-1 \rrbracket}$ across the time intervals $\{\llbracket t_i, t_{i+1} \rrbracket\}_{i \in \llbracket 0, N-1 \rrbracket}$ are given in (12). Then, the sequence $\{V_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ in (5) satisfies $V_{t_i} = \tilde{V}_{t_i} \circ \theta_{t_i}$, for all $i \in \llbracket 0, N \rrbracket$.

Proof. The proof is an immediate consequence of multiple applications of Theorem 2.1 and Proposition 3.2. $\square$

Finally, we consider the special case where the criterion $j : \mathbb{H}_T \rightarrow [0, +\infty]$ is factored as

$$j(h_T) = \sum_{i=0}^{N-1} \ell_{t_i}(\theta_{t_i}(h_{t_i}), h_{t_{i+1}:t_{i+1}}) + \ell_{t_N}(\theta_{t_N}(h_{t_N})), \quad (16)$$

where the numerical functions $\{\ell_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ are nonnegative lower semianalytic, with $\ell_{t_i} : \mathbb{X}_{t_i} \times \mathbb{H}_{t_{i+1}:t_{i+1}} \rightarrow [0, +\infty]$ for $i \in \llbracket 0, N \rrbracket$.

The associated optimization problems, indexed by $i \in \llbracket 0, N-1 \rrbracket$ and parameterized by $h_{t_i} \in \mathbb{H}_{t_i}$, are given by

$$V_{t_i}(h_{t_i}) \quad (17a)$$

$$= \inf_{\gamma_{t_i:T-1} \in \Gamma_{t_i:T-1}} \int_{\mathbb{H}_T} \left(\sum_{j=i}^{N-1} \ell_{t_j}(\theta_{t_j}(h'_{t_j}), h'_{t_{j+1}:t_{j+1}}) + \ell_{t_N}(\theta_{t_N}(h'_{t_N})) \right) \rho_{t_i:T}^\gamma(dh'_T \mid h_{t_i}),$$

and, for $i = N$,

$$V_{t_N}(h_{t_N}) = \ell_{t_N}(\theta_{t_N}(h_{t_N})). \quad (17b)$$

These Bellman equations are a special case of Equations (5) when the cost criterion j is given by (16). It is left to the reader to prove that the following theorem holds true⁴.

Theorem 3.7. (Taking care of instantaneous costs in addition to final cost) *Suppose that the assumptions of Theorem 3.6 are satisfied, but for the cost criterion $j : \mathbb{H}_T \rightarrow [0, +\infty]$ defined by Equation (16).*

We define the sequence of reduced value functions $\{\tilde{V}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$, $\tilde{V}_{t_i} : \mathbb{X}_{t_i} \rightarrow [0, +\infty]$ for $i \in \llbracket 0, N \rrbracket$, by

$$\tilde{V}_{t_N} = \ell_{t_N} \text{ and } \tilde{V}_{t_i} = \bar{\mathcal{B}}_{t_{i+1}:t_i} \tilde{V}_{t_{i+1}}, \quad \forall i \in \llbracket 0, N-1 \rrbracket, \quad (18)$$

where the reduced Bellman operator $\bar{\mathcal{B}}_{t_{i+1}:t_i}$ across the interval $\llbracket t_i, t_{i+1} \rrbracket$ are given, for any $i \in \llbracket 0, N-1 \rrbracket$, for any $\tilde{\varphi}_{t_{i+1}} \in \mathbb{L}_+^0(\mathbb{X}_{t_{i+1}})$ and for any $x_{t_i} \in \mathbb{X}_{t_i}$, by

⁴The proof uses [3, Lemma 7.30 (3,4)] on the stability of lower semianalytic numerical functions under addition and under right composition with a Borel-measurable mapping.

$$\begin{aligned}
(\bar{\mathcal{B}}_{t_{i+1}:t_i} \tilde{\varphi}_{t_{i+1}})(x_{t_i}) &= \inf_{u_{t_i} \in \mathbb{U}_{t_i}} \int_{\mathbb{W}_{t_i+1}} \tilde{\rho}_{t_i:t_{i+1}}(dw_{t_{i+1}} \mid x_{t_i}) \\
&\quad \inf_{u_{t_i+1} \in \mathbb{U}_{t_i+1}} \int_{\mathbb{W}_{t_i+2}} \tilde{\rho}_{t_i+1:t_{i+2}}(dw_{t_{i+2}} \mid x_{t_i}, u_{t_i}, w_{t_{i+1}}) \quad \dots \\
&\quad \inf_{u_{t_{i+1}-1} \in \mathbb{U}_{t_{i+1}-1}} \int_{\mathbb{W}_{t_{i+1}}} \tilde{\rho}_{t_{i+1}-1:t_{i+1}}(dw_{t_{i+1}} \mid x_{t_i}, u_{t_i}, w_{t_{i+1}}, \dots, u_{t_{i+1}-2}, w_{t_{i+1}-1}) \\
&\quad \left(\ell_{t_i}(x_{t_i}, u_{t_i}, w_{t_{i+1}}, \dots, u_{t_{i+1}-1}, w_{t_{i+1}}) \right. \\
&\quad \left. + \tilde{\varphi}_{t_{i+1}}(f_{t_i:t_{i+1}}(x_{t_i}, u_{t_i}, w_{t_{i+1}}, \dots, u_{t_{i+1}-1}, w_{t_{i+1}})) \right). \quad (19)
\end{aligned}$$

Then, the sequence $\{V_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ in Equations (17) satisfies $V_{t_i} = \tilde{V}_{t_i} \circ \theta_{t_i}$, for all $i \in \llbracket 0, N \rrbracket$.

Here again, Formula (19) represents a nested sequence of infima of integrals (with respect to different stochastic kernels).

Of course, solving Equation (15) or Equation (19) can be as difficult as solving the original Bellman equation. However, the interest of such time block decomposition will be illustrated on different applications in Sect. 4, Sect. 5 and Sect. 6.

3.3. State reduction on multiple consecutive time blocks and reduced optimal feedbacks

As in the classical dynamic programming framework [3, p. 190], we recover the property that the search of an optimal policy among all policies (*history* feedbacks) can be limited to the search of an optimal *state* feedback. This is the most important result in practice.

Proposition 3.8. *Under the assumptions of Theorem 3.6, the reduced value functions $\{\tilde{V}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ defined in (15) are equal to the minimum value of the following optimization problems, parameterized by the reduced history (state) $x_{t_i} \in \mathbb{X}_{t_i}$*

$$\tilde{V}_{t_i}(x_{t_i}) = \inf_{\gamma_{t_i:T-1} \in \Gamma_{t_i:T-1}^{x_{t_i}}} \int_{\mathbb{H}_{t_{i+1}:T}} \bar{j}(x_{t_i}, h'_{t_{i+1}:T}) \tilde{\varrho}_{t_i:T}^{\gamma}(dh'_{t_{i+1}:T} \mid x_{t_i}, h_{t_{i+1}:t}), \quad (20a)$$

for all $x_{t_i} \in \mathbb{X}_{t_i}$, and

$$\tilde{V}_T(x_T) = \tilde{j}(x_T), \quad \forall x_T \in \mathbb{X}_T, \quad (20b)$$

where the mapping $\bar{j}$ is given by $\bar{j} = \tilde{j} \circ f_{t_{N-1}:t_N} \circ f_{t_{N-2}:t_{N-1}} \circ \dots \circ f_{t_i:t_{i+1}}$ (with, as already noted, the abuse of notation that the composition is performed on the state argument), where $\tilde{\varrho}_{t_i:T}^{\gamma}$ is the reduced stochastic kernel (see Definition 3.1) associated with the kernel $\varrho_{t_i:T}^{\gamma}$, the kernel $\varrho_{r:t}^{\gamma}$ being given in the factorization of the kernel $\rho_{r:t}^{\gamma}$, namely $\rho_{r:t}^{\gamma}(dh'_t \mid h_r) = \delta_{h_r}(dh'_r) \otimes \varrho_{r:t}^{\gamma}(dh'_{r+1:t} \mid h_r)$ given by (48), δ being the Dirac measure, and where $\Gamma_{t_i:T-1}^{x_{t_i}}$ is the set of $(t_i:T-1)$ -reduced history feedbacks, that is, the set of sequences $\{\gamma_s\}_{s \in \llbracket t_i, T-1 \rrbracket}$ of universally measurable mappings $\gamma_s : \mathbb{X}_{t_i} \times \mathbb{H}_{t_{i+1}:s} \rightarrow \mathbb{U}_s$.

Proof. Using Theorem 3.6, we have that, for all $i \in \llbracket 0, N \rrbracket$, $V_{t_i} = \tilde{V}_{t_i} \circ \theta_{t_i}$, with $\tilde{V}_{t_i}$ satisfying the Bellman equation (15). For establishing that $\tilde{V}_{t_i}$ is a value function satisfying Equation (20), we now prove that, in the definition of V_{t_i} in Equation (5),

we can replace the space $\Gamma_{t_i:T-1}$ of history feedbacks by the space $\Gamma_{t_i:T-1}^{x_{t_i}}$ of state feedbacks. We proceed as follows. Following [3, Chapter 8], we use the Bellman equation (19) to obtain ϵ -minimizers for each problem (5). As ϵ -minimizers are obtained by recursively solving Equations (19), they are obtained by solving (up to ϵ) parametric optimization problems. Thus, we easily get, using [3, Proposition 8.3, p. 200], that an ϵ -minimizer at time $t \in \llbracket t_i, t_{i+1} \rrbracket$ is a universally measurable function of $(\theta(h_{t_i}), h_{t_{i+1}:t})$. From this last fact, we get that – in the value function definition of V_{t_i} given in Equation (5) – the space $\Gamma_{0:T-1}$ can be replaced by the space of feedbacks given by universally measurable functions of the ordered pair $(\theta(h_{t_i}), h_{t_{i+1}:t})$ without changing the value function.

Finally, when considering Equation (5) – with this restricted space of state feedbacks, and considered at time t_i for $i \in \llbracket 0, N \rrbracket$ — we obtain that the cost to be minimized is now parameterized by $\theta(h_{t_i})$ — as it is the case for the cost to be integrated and also for the stochastic kernels induced by the state feedbacks. By setting $x_i = \theta(h_{t_i})$, the obtained optimization problem is the right hand side of (20), that we call $\tilde{V}_{t_i}(x_{t_i})$, and we have that, for all $i \in \llbracket 0, N \rrbracket$, $V_{t_i} = \tilde{V}_{t_i} \circ \theta_{t_i}$. It remains to prove that $\tilde{V}_{t_i} = \tilde{\tilde{V}}_{t_i}$, for all $i \in \llbracket 0, N \rrbracket$. By a proof similar to the one of Theorem 3.6, we show that the sequence $\{\tilde{\tilde{V}}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ satisfies the Bellman equation (15). By uniqueness, the sequence $\{\tilde{\tilde{V}}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ coincides with the sequence $\{\tilde{V}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$. $\square$

4. Mixing dynamic programming and stochastic programming

As a first application of the formalism developed in § 3.2, we show how dynamic programming and stochastic programming can be mixed (which was the original motivation for the paper, see Footnote 1). The proof of the following proposition is a straightforward application of Theorem 3.7 combined with Remark 3.5.

Proposition 4.1. *Suppose that the assumptions of Theorem 3.6 are satisfied. We consider multiple consecutive time blocks $\llbracket t_i, t_{i+1} \rrbracket$, with $N \in \mathbb{N}^*$, $i \in \llbracket 0, N-1 \rrbracket$ and $0 = t_0 < \dots < t_N = T$, and we assume that*

- *a state reduction $(\{\mathbb{X}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{\theta_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{f_{t_i:t_{i+1}}\}_{i \in \llbracket 0, N-1 \rrbracket})$ exists across the consecutive time blocks $\{\llbracket t_i, t_{i+1} \rrbracket\}_{i \in \llbracket 0, N-1 \rrbracket}$,*
- *the noises are exogeneous and time block independent, that is, the elements of the sequence $\{\rho_{s-1:s}\}_{s \in \llbracket 1, T \rrbracket}$ in (3) are, for all $i \in \llbracket 0, N-1 \rrbracket$ and $r \in \llbracket t_i, t_{i+1} \rrbracket$, of the form*

$$\rho_{r:r+1} : \mathbb{W}_{t_i} \times \dots \times \mathbb{W}_r \rightarrow \Delta(\mathbb{W}_{r+1}) , \quad (21)$$

which means that the distribution of the uncertainty w_{r+1} is only function of the past uncertainties $(w_{t_i}, \dots, w_r)$ within the time block,

- *the cost criterion $j : \mathbb{H}_T \rightarrow [0, +\infty]$ can be factored as*

$$j(h_T) = \sum_{i=0}^{N-1} \ell_{t_i}(\theta_{t_i}(h_{t_i}), h_{t_{i+1}:t_{i+1}}) + \ell_{t_N}(\theta_{t_N}(h_{t_N})) , \quad (22)$$

where the numerical functions $\{\ell_{t_i}\}_{i \in \llbracket 0, N \rrbracket}$ are nonnegative lower semianalytic, with $\ell_{t_i} : \mathbb{X}_{t_i} \times \mathbb{H}_{t_{i+1}:t_{i+1}} \rightarrow [0, +\infty]$ for $i \in \llbracket 0, N \rrbracket$.

Then, the multistage stochastic optimization problem (6) can be solved by the following algorithm.

Initialization. Define $\tilde{V}_{t_N} = \ell_{t_N} : \mathbb{X}_T \rightarrow [0, +\infty]$.

Backward recursion. Suppose that the function $\tilde{V}_{t_{i+1}} : \mathbb{X}_{t_{i+1}} \rightarrow [0, +\infty]$ is known at index $i + 1 \in \llbracket 1, N \rrbracket$. Then, for each state $x_{t_i} \in \mathbb{X}_{t_i}$ (for instance on a grid approximating the set $\mathbb{X}_{t_i}$, or on $\mathbb{X}_{t_i}$ itself when finite and small enough), compute the previous Bellman value function $\tilde{V}_{t_i}$ at index i as

$$\begin{aligned} \tilde{V}_{t_i}(x_{t_i}) = & \inf_{u_{t_i} \in \mathbb{U}_{t_i}} \int_{\mathbb{W}_{t_i+1}} \rho_{t_i:t_i+1}(\mathrm{d}w_{t_i+1} \mid w_{t_i}) \\ & \inf_{u_{t_i+1} \in \mathbb{U}_{t_i+1}} \int_{\mathbb{W}_{t_i+2}} \rho_{t_i+1:t_i+2}(\mathrm{d}w_{t_i+2} \mid w_{t_i}, w_{t_i+1}) \quad \cdots \\ & \inf_{u_{t_{i+1}-1} \in \mathbb{U}_{t_{i+1}-1}} \int_{\mathbb{W}_{t_{i+1}}} \rho_{t_{i+1}-1:t_{i+1}}(\mathrm{d}w_{t_{i+1}} \mid w_{t_i}, w_{t_i+1}, \dots, w_{t_{i+1}-1}) \\ & \left(\ell_{t_i}(x_{t_i}, u_{t_i}, w_{t_i+1}, \dots, u_{t_{i+1}-1}, w_{t_{i+1}}) \right. \\ & \left. + \tilde{V}_{t_{i+1}}(f_{t_i:t_{i+1}}(x_{t_i}, u_{t_i}, w_{t_i+1}, \dots, u_{t_{i+1}-1}, w_{t_{i+1}})) \right). \quad (23) \end{aligned}$$

Final step. Compute $V_0(w_0) = \tilde{V}_{t_0}(\theta_{t_0}(w_{t_0}))$.

In many practical situations, all the uncertainty sets $\mathbb{W}_0, \dots, \mathbb{W}_T$ are finite and the computation in (23) is tractable by using *stochastic programming* and *scenario tree* techniques, which do not require stagewise independence of the noises. We are thus able to take advantage of both the dynamic programming world and the stochastic programming world:

- use dynamic programming at a selection of time stages (for instance, at those of the slow time scale) and across the corresponding time blocks (for instance, across consecutive slow time stages), when noises are stochastically independent block by block; that yields Bellman value functions only for the chosen selection of time stages (for instance, at the slow time scale);
- use stochastic programming inside time blocks (for instance, at fast time scale, within two consecutive slow time stages); the fast time scale final cost function of a block is given by the Bellman value function computed at the slow time scale which corresponds to the terminal time stage of the block; no stagewise independence assumption is required within time blocks (for instance, for the short time scale noises).

Remark 4.2. As a special case, it is straightforward to check that the triplet $(\{\mathbb{W}_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{\theta_{t_i}\}_{i \in \llbracket 0, N \rrbracket}, \{f_{t_i:t_{i+1}}\}_{i \in \llbracket 0, N-1 \rrbracket})$, with

- the reduction mapping θ_{t_i} given by $\theta_{t_i}(h_{t_i}) = w_{t_i}$ for all $i \in \llbracket 0, N \rrbracket$,
- the dynamics $f_{t_i:t_{i+1}}$ given by $f_{t_i:t_{i+1}}(w_{t_i}, h_{t_{i+1}:t_{i+1}}) = w_{t_{i+1}}$, for all $i \in \llbracket 0, N-1 \rrbracket$.

is a state reduction across the consecutive time blocks $\llbracket t_i, t_{i+1} \rrbracket$, $i \in \llbracket 0, N-1 \rrbracket$ which is compatible with the sequence of stochastic kernels given by Equation (21). Thus, Proposition 4.1 applies.

But, in this special case, the optimal controls can be computed in parallel with respect to time blocks, as the term $\tilde{V}_{t_{i+1}}(w_{t_{i+1}})$ is a constant in (23). What is interesting in (23) is the added fact that the optimal strategy which was, a priori,

searched as feedbacks depending on the whole history is in fact made up of independent strategies, each defined on a single time block and made up of feedbacks depending only on the block history (the history within the block). $\square$

Numerical illustration. To numerically illustrate the mixing between dynamic programming and stochastic programming, we consider a toy optimization problem over a time span $\llbracket 0, T \rrbracket$, where T is an even natural number (for instance $T = 24$ for an hourly period problem during a day). The problem involves a storage, the state x_t of which is driven by a dynamics involving a control variable u_t and a noise variable w_{t+1} . We assume that the noises during the first half time span, that is, for $t \in \llbracket 1, T/2 \rrbracket$, are independent of the noises during the second half time span, that is, for $t \in \llbracket T/2, T \rrbracket$. We also assume that each noise variable w_t can only take two possible values, so that the whole uncertainty process can be represented by a binary tree.

In this problem, we consider the two consecutive time blocks $\llbracket 0, T/2 \rrbracket$ and $\llbracket T/2, T \rrbracket$, and the state reduction is given in a straightforward manner by the variable x_t (as explained in Remark 3.4). Thus, we are able to compute Bellman functions by the algorithm given in Proposition 4.1. We illustrate the algorithm for the time horizon $T = 24$.

- The Bellman function $\tilde{V}_{24}$ is given by the final cost function of the problem.
- The Bellman function $\tilde{V}_{12}$ is approximated by discretization and it is computed on a grid involving n points $(x^1, \dots, x^n)$. For $i \in \llbracket 1, n \rrbracket$, each value $\tilde{V}_{12}(x^i)$ is obtained by solving a stochastic programming problem on the time span $\llbracket 12, 24 \rrbracket$, that is, on a tree involving 2^{12} leaves (as each noise variable w_t can only take two possible values).
- The optimal cost of the optimization problem is $\tilde{V}_0(x_0)$, obtained again by stochastic programming on the time span $\llbracket 0, 12 \rrbracket$, that is, by solving a stochastic optimization problem on a tree involving 2^{12} leaves, the final cost being given by the function $\tilde{V}_{12}$.

Gathering the calculations performed by this algorithm, we obtain that solving the global problem by mixing dynamic programming and stochastic programming is done by solving $(n + 1)$ stochastic optimization programs on scenario trees, each involving 2^{12} leaves. The total number of leaves to explore – when solving the problem by this mixing method – is $(n + 1)2^{12} \approx 4(n + 1)10^3$, which gives an estimation of the algorithm computational effort.

This mixing method is to be compared with a pure scenario tree method, that is, when the problem is solved by a stochastic optimization program on a scenario tree over the whole time horizon on 24 hours, the total number of leaves to explore being $2^{24} \approx 1.6 \cdot 10^7$. Even using a fairly fine state discretization grid, for example a grid containing 100 points, the resolution by mixing dynamic programming and stochastic programming – when compared to the pure stochastic programming approach – leads to a quite significant gain, namely a factor $1.6 \times 10^7 / 4(100 + 1) \times 10^3 \approx 40$ in our case.

We performed numerical experiments with a single computer equipped with 12 Intel Core i5-10500 CPU and 16 GB of RAM. We used the LP package of the solver Gurobi

9.51. Apart from the solver, all our code has been implemented with the Julia language and the JuMP modeler. As we failed to obtain a solution for the original problem on a tree for the horizon $T = 24$, we performed numerical tests for shorter horizons, hence for smaller numbers of time steps. For every $T \in \{12, 14, 16, 18, 20, 24\}$, we considered that a state reduction existed at time $T/2$. The results are gathered in Table 4.1, and show that the computational time – that is, the CPU time needed to create the LP model by JuMP and to solve it by Gurobi – needed by the pure scenario tree method is very rapidly increasing with the number of time steps, whereas the computational time needed by the mixing method grows very slowly with the number of time steps, at least for the different horizons under consideration. Finally, note that the mixing method can be easily parallelized since the computation of the n values $\{\tilde{V}_{T/2}(x^i)\}_{i \in \llbracket 1, n \rrbracket}$ of the Bellman function $\tilde{V}_{T/2}$ can be performed in parallel, reducing the CPU time by a factor $(n + 1)/2$, that is, approximately 50 in our case.

Horizon T	Mixing method	Pure scenario tree method
12	6.5 s	0.5 s
14	6.5 s	5.0 s
16	6.6 s	71.2 s
18	6.8 s	1009.5 s
20	6.8 s	137,296.0 s
24	7.2 s	optimization failed

Table 4.1: Computational time (in seconds) needed for solving the problem by both the mixing method and the pure scenario tree method, for different horizons

5. Two-time-scale optimization problems

As a second application of the formalism developed in § 3.2, we show how to tackle cases of two-time-scale optimization problems. Indeed, some decisions problems naturally involve two different time scales, because of the timing of decisions – as for example long term investment decision and short term monitoring of physical devices.

In § 5.1 and § 5.2 we detail the structure and we formulate the two-time-scale optimization problems that we consider. In § 5.3, we show how to decompose such problems by time blocks. In § 5.4, we illustrate the approach on a crude oil procurement problem.

5.1. Structure of a two-time-scale optimization problem

We provide the data for a two-time-scale multistage optimization problem.

Two time scales. The slow time scale is represented by a finite totally ordered set (chain) $(S, \preceq)$ as follows – where s^+ denotes the *successor* of $s \in S$ and s^- its *predecessor*, and where we use the notation $t \prec t'$ for $t \preceq t'$ and $t \neq t'$ –

$$\min S = \underline{s} \prec \cdots \prec s^- \prec s \prec s^+ \prec \cdots \prec \bar{s} = \max S, \quad (24a)$$

and the fast time scale by a finite totally ordered set $(F, \preceq)$:

$$\min F = \underline{f} \prec \cdots \prec f^- \prec f \prec f^+ \prec \cdots \prec \bar{f} = \max F. \quad (24b)$$

For instance, $S = \{Mo, Tu, We, Th, Fr, Sa, Su\}$ may represent days, whereas $F = \llbracket 1, 24 \rrbracket$ may represent hours within a day. In some problems, we might even take $F = \llbracket 0, 24 \rrbracket$ to handle the fact that two decisions (one slow and one fast) are taken at midnight, hence an additional fast time step 0.

Unified timeline. We define the unified timeline of the decision problem in two steps. First, we equip the product set $S \times F$ with the following lexicographic order:

$$\begin{aligned} (\underline{s}, \underline{f}) \prec \cdots \prec (s^-, \bar{f}) \prec (s, \underline{f}) \prec (s, \underline{f}^+) \prec \cdots \\ \cdots \prec (s, \bar{f}^-) \prec (s, \bar{f}) \prec (s^+, \underline{f}) \prec \cdots \prec (\bar{s}, \bar{f}). \end{aligned} \quad (25)$$

More formally, we denote by $(s, f)^+$ the successor of (s, f) in $S \times F \setminus \{(\bar{s}, \bar{f})\}$, with

$$(s, f)^+ = \begin{cases} (s, f^+) & \text{if } f \neq \bar{f}, \\ (s^+, \underline{f}) & \text{if } f = \bar{f}. \end{cases} \quad (26a)$$

Similarly, we denote by $(s, f)^-$ the predecessor of (s, f) in $S \times F \setminus \{(\underline{s}, \underline{f})\}$, with

$$(s, f)^- = \begin{cases} (s, f^-) & \text{if } f \neq \underline{f}, \\ (s^-, \bar{f}) & \text{if } f = \underline{f}. \end{cases} \quad (26b)$$

Second, we add to the product set $S \times F$ an extra time denoted by $(\underline{s}^-, \bar{f})$, corresponding to the extra slow time $\underline{s}^-$, which is such that $(\underline{s}, \underline{f})^- = (\underline{s}^-, \bar{f})$. We denote by $\bar{S}$ the set $\{\underline{s}^-\} \cup S$ and by $\bar{S} \times \bar{F}$ the set $(\underline{s}^-, \bar{f}) \cup (S \times F)$, also called the *extended timeline* when equipped with an order $\preceq$ as follows (where we use the notation $(s, f) \prec (s', f')$ for $(s, f) \preceq (s', f')$ and $(s, f) \neq (s', f')$)

$$\begin{aligned} (\underline{s}^-, \bar{f}) \prec (\underline{s}, \underline{f}) \prec \cdots \prec (s^-, \bar{f}) \prec (s, \underline{f}) \prec (s, \underline{f}^+) \prec \cdots \\ \cdots \prec (s, \bar{f}^-) \prec (s, \bar{f}) \prec (s^+, \underline{f}) \prec \cdots \prec (\bar{s}, \bar{f}). \end{aligned} \quad (27)$$

The two-time-scale optimization problem will be formulated on the extended timeline $\bar{S} \times \bar{F}$, which we trivially identify with the time set $\llbracket 0, T \rrbracket$, where $T = |S| \times |F|$.

Decisions. We suppose given

- a family $\{\mathbb{U}_s^s\}_{s \in \bar{S} \setminus \{\bar{s}\}}$ of *slow time scale decision Borel spaces*, and a family $\{\mathbb{W}_s^s\}_{s \in S}$ of *slow time scale uncertainty Borel spaces*,
- a family $\{\mathbb{U}_{(s,f)}^{sf}\}_{(s,f) \in S \times (F \setminus \{\bar{f}\})}$ of *fast time scale decision Borel spaces*, and a family $\{\mathbb{W}_{(s,f)}^{sf}\}_{(s,f) \in S \times (F \setminus \{\bar{f}\})}$ of *fast time scale uncertainty Borel spaces*.

Dynamics. We suppose given a family $\{\mathbb{X}_{(s,f)}^{sf}\}_{(s,f) \in S \times F}$ of *fast time scale state Borel spaces*. For the sake of simplicity, we set $\mathbb{X}_s^s = \mathbb{X}_{(s,\bar{f})}^{sf}$ for all $s \in S$. Thus, the slow time $s \in S$ is identified with the two scale time $(s, \bar{f})$, as illustrated in Figure 5.1. We also suppose given a family $\{\mathcal{G}_s^s\}_{s \in \bar{S} \setminus \{\bar{s}\}}$ of *slow time scale dynamics*

Borel-measurable mappings, that represent the evolution “driven at the slow time scale” given, for $s \in \bar{S} \setminus \{\bar{s}\}$, by⁵

$$\begin{aligned} \mathcal{G}_s^s : \mathbb{X}_s^s \times \mathbb{U}_s^s \times \mathbb{W}_{s+}^s &\rightarrow \mathbb{X}_{(s+,f)}^{sf}, \\ (x_s^s, u_s^s, w_{s+}^s) &\mapsto x_{(s+,f)}^{sf} = \mathcal{G}_s^s(x_s^s, u_s^s, w_{s+}^s). \end{aligned} \quad (28a)$$

We suppose given a family $\{\mathcal{G}_{(s,f)}^{sf}\}_{(s,f) \in S \times (F \setminus \{\bar{f}\})}$ of *fast time scale dynamics Borel-measurable mappings*, that represent the evolution “driven at the fast time scale” given, for all $s \in S$ and $f \in F \setminus \{\bar{f}\}$, by

$$\begin{aligned} \mathcal{G}_{(s,f)}^{sf} : \mathbb{X}_{(s,f)}^{sf} \times \mathbb{U}_{(s,f)}^{sf} \times \mathbb{W}_{(s,f)+}^{sf} &\rightarrow \mathbb{X}_{(s,f)+}^{sf}, \\ (x_{(s,f)}^{sf}, u_{(s,f)}^{sf}, w_{(s,f)+}^{sf}) &\mapsto x_{(s,f)+}^{sf} = \mathcal{G}_{(s,f)}^{sf}(x_{(s,f)}^{sf}, u_{(s,f)}^{sf}, w_{(s,f)+}^{sf}). \end{aligned} \quad (28b)$$

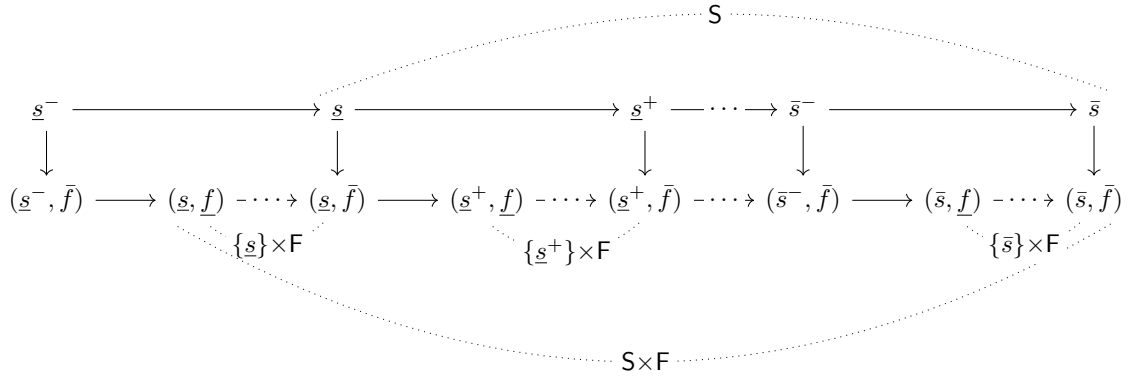


Figure 5.1: The product timeline with an extra starting point $(\bar{s}^-, \bar{f})$.

Cost functions We suppose given a family $\{\Lambda_s\}_{s \in \bar{S} \setminus \{\bar{s}\}}$ of *slow time scale nonnegative lower semianalytic cost functions*, with

$$\Lambda_{s-} : \mathbb{X}_{s-}^s \times \mathbb{U}_{s-}^s \times \mathbb{W}_{s-}^s \times \underbrace{\prod_{f \in F \setminus \{\bar{f}\}} (\mathbb{X}_{(s,f)}^{sf} \times \mathbb{U}_{(s,f)}^{sf} \times \mathbb{W}_{(s,f)+}^{sf})}_{\text{interval }]s^-, s[= \{(s, f), \dots, (s, \bar{f}^-)\}} \rightarrow [0, +\infty], \quad (29)$$

for $s \in S$, and a *slow time scale nonnegative lower semianalytic final cost function* $\Lambda_{\bar{s}} : \mathbb{X}_{\bar{s}}^s \rightarrow [0, +\infty]$ that make up, by summation, an intertemporal cost

$$\sum_{s \in S} \Lambda_{s-}(x_{s-}^s, u_{s-}^s, w_{s-}^s, (x_{(s,f)}^{sf}, u_{(s,f)}^{sf}, w_{(s,f)+}^{sf})_{f \in F \setminus \{\bar{f}\}}) + \Lambda_{\bar{s}}(x_{\bar{s}}^s). \quad (30)$$

Stochastic kernels. Finally, we suppose given a family of *constant slow time scale Borel-measurable stochastic kernels* $\{\rho_{s;s+}^s\}_{s \in \bar{S} \setminus \{\bar{s}\}}$

$$\rho_{s;s+}^s \in \Delta(\mathbb{W}_{s+}^s), \quad \forall s \in \bar{S} \setminus \{\bar{s}\}, \quad (31a)$$

and, for each $s \in S$, a family $\{\rho_{(s,f):(s,f)+}^{sf}\}_{f \in F \setminus \{\bar{f}\}}$ of *fast time scale Borel-measurable stochastic kernels*

⁵ We stress that the slow time scale dynamics (28a) yields as output the first fast state of the slow period (and not the next slow state). Thus, the slow time scale dynamics (28a) is *not* a dynamics from one slow state to the next slow state.

$$\rho_{(s,f):(s,f)^+}^{\text{sf}} : \mathbb{W}_s^{\text{s}} \times \underbrace{\prod_{\substack{f'=\underline{f}^+ \\ \text{in interval }]s^-,s[}}^f \mathbb{W}_{(s,f')}^{\text{sf}} \longrightarrow \Delta(\mathbb{W}_{(s,f)^+}^{\text{sf}}), \quad \forall s \in \mathbf{S}, \quad \forall f \in \mathbf{F} \setminus \{\bar{f}\}, \quad (31b)$$

with the convention that the Cartesian products of spaces in Equation (31b) reduce to nothing when the upper index of the Cartesian product is strictly lower than the corresponding lower index.

Note that, for a given $s \in \mathbf{S}$, each fast time scale stochastic kernel $\rho_{(s,f):(s,f)^+}^{\text{sf}}$, only depends on the noises of the slow time interval $[s^-, s[= \{(s^-, \bar{f}), (s, \underline{f}), \dots, (s, \bar{f}^-)\}$.

The (constant) assumption (31a) and the (single block) assumption (31b) correspond to stochastic independence between time blocks, and will be useful in the proof of Proposition 5.1.

5.2. Formulation of a two-time-scale optimization problem on the product timeline

To apply Theorem 3.6, we introduce sets associated with the extended timeline (27) by

$$\mathbb{X}_{(s,f)} = \begin{cases} \mathbb{X}_s^{\text{s}} & \text{if } f = \bar{f} \\ \mathbb{X}_{(s,f)}^{\text{sf}} & \text{if } f \neq \bar{f} \end{cases}, \quad \forall (s, f) \in \overline{\mathbf{S} \times \mathbf{F}}, \quad (32a)$$

$$\mathbb{U}_{(s,f)} = \begin{cases} \mathbb{U}_s^{\text{s}} & \text{if } f = \bar{f} \\ \mathbb{U}_{(s,f)}^{\text{sf}} & \text{if } f \neq \bar{f} \end{cases}, \quad \forall (s, f) \in \overline{\mathbf{S} \times \mathbf{F}} \setminus \{(\bar{s}, \bar{f})\}, \quad (32b)$$

$$\mathbb{W}_{(s,f)}^{\text{sf}} = \begin{cases} \mathbb{W}_s^{\text{s}} & \text{if } f = \underline{f} \\ \mathbb{W}_{(s,f)}^{\text{sf}} & \text{if } f \neq \underline{f} \end{cases}, \quad \forall (s, f) \in \mathbf{S} \times \mathbf{F}, \quad (32c)$$

with the particular case of the extra initial slow time

$$\mathbb{W}_{(\underline{s}^-, \bar{f})}^{\text{sf}} = \mathbb{X}_{\underline{s}^-}^{\text{s}}, \quad (32d)$$

and a family of state dynamics $\mathcal{G}_{(s,f)} : \mathbb{X}_{(s,f)} \times \mathbb{U}_{(s,f)} \times \mathbb{W}_{(s,f)^+} \rightarrow \mathbb{X}_{(s,f)^+}$ defined by

$$\mathcal{G}_{(s,f)} = \begin{cases} \mathcal{G}_s^{\text{s}} & \text{if } f = \bar{f} \\ \mathcal{G}_{(s,f)}^{\text{sf}} & \text{if } f \neq \bar{f} \end{cases}, \quad \forall (s, f) \in \overline{\mathbf{S} \times \mathbf{F}} \setminus \{(\bar{s}, \bar{f})\}. \quad (33)$$

From these spaces, we deduce the history spaces and the histories for all $(s, f) \in \overline{\mathbf{S} \times \mathbf{F}}$

$$\mathbb{H}_{(s,f)} = \mathbb{W}_{(\underline{s}^-, \bar{f})} \times \prod_{(\underline{s}, \underline{f}) \preceq (s', f') \preceq (s, f)} (\mathbb{U}_{(s', f')^-} \times \mathbb{W}_{(s', f')}) , \quad (34a)$$

$$h_{(s,f)} = \left(w_{(\underline{s}^-, \bar{f})}, (u_{(s', f')^-}, w_{(s', f')})_{(\underline{s}, \underline{f}) \preceq (s', f') \preceq (s, f)} \right) \quad (34b)$$

and, for suitable indices, the partial history sets and the partial histories

$$\mathbb{H}_{(s,f):(s',f')} = \prod_{(s,f) \preceq (s'',f'') \preceq (s',f')} (\mathbb{U}_{(s'',f'')}^- \times \mathbb{W}_{(s'',f'')}) , \quad (35a)$$

$$h_{(s,f):(s',f')} = ((u_{(s'',f'')}^-, w_{(s'',f'')})_{(s,f) \preceq (s'',f'') \preceq (s',f')}) . \quad (35b)$$

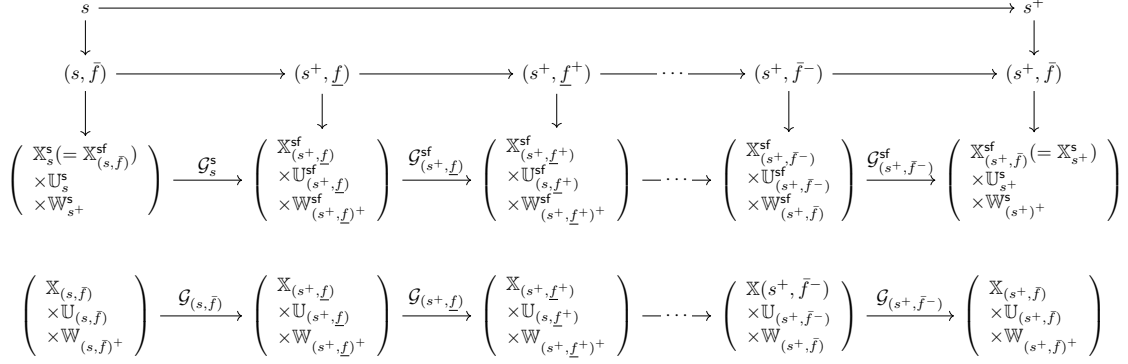


Figure 5.2: Original dynamics and their reformulation on the product timeline on the slow time interval $[s, s^+[\equiv \{(s, \bar{f}), (s^+, \underline{f}), \dots, (s^+, \bar{f}^-)\}$.

The cost criterion formulated in Equation (30) combined with state dynamics leads to a (nonnegative lower semianalytic) cost criterion $j : \mathbb{H}_{(\bar{s}, \bar{f})} \rightarrow [0, +\infty]$.

Based on the stochastic kernels (31a) and (31b), we introduce stochastic kernels $\rho_{(s,f):(s,f)^+}$ associated with the extended timeline (27), for each $(s, f) \in \bar{S} \times \bar{F} \setminus \{\bar{s}, \bar{f}\}$, by $\rho_{(s,f):(s,f)^+} : \mathbb{H}_{(s,f)} \rightarrow \Delta(W_{(s,f)^+})$ with

$$\rho_{(s,f):(s,f)^+}(dw_{(s,f)^+} \mid h_{(s,f)}) = \begin{cases} \rho_{s:s^+}^s(dw_{s^+}^s) & \text{if } f = \bar{f} , \\ \rho_{(s,f):(s,f)^+}^{sf}(dw_{(s,f)^+}^{sf} \mid w_s^s, w_{(s,\underline{f})}^{sf}, \dots, w_{(s,f)}^{sf}) & \text{if } f \neq \bar{f} \end{cases}$$

Note that, for $f \neq \bar{f}$, the stochastic kernels $\rho_{(s,f):(s,f)^+} : \mathbb{H}_{(s,f)} \rightarrow \Delta(W_{(s,f)^+})$, only depend on the partial history uncertainty part from $(s, \underline{f})$ to (s, f) , and not on the (past) controls.

The components of the problem are now formulated on the extended timeline $\bar{S} \times \bar{F}$, already identified with the time set $\llbracket 0, T \rrbracket$. Thus, we are in the framework of § 2.1 and we aim at solving an optimization problem as formulated in Equation (6).

5.3. Two-time-scale decomposition

The existence of Bellman equations for a two-time-scale optimization problem is given by the following proposition.

Proposition 5.1. *Consider a two-time-scale optimization problem as formulated in § 5.1 and § 5.2. The optimization problem (6) has a solution given by a dynamic*

programming equation at the slow scale. More precisely, let $\{V_s\}_{s \in \bar{S}}$ be given by $V_{\bar{s}} = \Lambda_{\bar{s}}$ and, for $s \in \bar{S} \setminus \{\bar{s}\}$, by the backward induction⁶

$$\begin{aligned} V_s(x_s^s) = & \inf_{u_s^s \in \mathbb{U}_s^s} \int_{\mathbb{W}_{s+}^s} \rho_{s:s+}^s(dw_{s+}^s) \\ & \inf_{u_{(s^+, \underline{f})}^{sf} \in \mathbb{U}_{(s^+, \underline{f})}^{sf}} \int_{\mathbb{W}_{(s^+, \underline{f})}^{sf}} \rho_{(s^+, \underline{f}):(s^+, \underline{f}^+)}^{sf}(dw_{(s^+, \underline{f})}^{sf} \mid w_{s+}^s) \cdots \\ & \inf_{u_{(s^+, \bar{f}^-)}^{sf} \in \mathbb{U}_{(s^+, \bar{f}^-)}^{sf}} \int_{\mathbb{W}_{(s^+, \bar{f})}^{sf}} \rho_{(s^+, \bar{f}^-):(s^+, \bar{f})}^{sf}(dw_{(s^+, \bar{f})}^{sf} \mid w_{s+}^s, w_{(s^+, \underline{f})}^{sf}, \dots, w_{(s^+, \bar{f}^-)}^{sf}) \\ & \left(\Lambda_s(x_s^s, u_s^s, w_{s+}^s, \dots, u_{(s^+, \bar{f}^-)}^{sf}, w_{(s^+, \bar{f})}^{sf}) \right. \\ & \left. + V_{s+}(\mathcal{G}_{s:s+}(x_s^s, u_s^s, w_{s+}^s, \dots, u_{(s^+, \bar{f}^-)}^{sf}, w_{(s^+, \bar{f})}^{sf})) \right), \quad (36) \end{aligned}$$

where $\mathcal{G}_{s:s+}$ is the composition $\mathcal{G}_{s:s+} = \mathcal{G}_{(s^+, \bar{f}^-)}^{sf} \circ \dots \circ \mathcal{G}_{(s^+, \underline{f})}^{sf} \circ \mathcal{G}_s^s$ associated with the state dynamics defined in (28). Then, the value of the optimization problem (6) is given by $V_{s-}(x_s^s)$, where the initial condition x_s^s corresponds to w_0 in (6), as stated by (32d).

Proof. The proof is an application of Theorem 3.7 with the help of Remarks 3.4 and 3.5. First, we have re-framed in § 5.2 the two-time-scale optimization problems described in § 5.1 in the formalism of § 2.1 with the help of the extended timeline (27). Second, as we are given state dynamics (33) on the extended timeline and thanks to Remark 3.4, we obtain a state reduction at times $\{(s, \bar{f})\}_{s \in \bar{S}}$ by composition of the state dynamics. Moreover, as the slow time scale stochastic kernels given by Equation (31a) are constant, the state reduction across the slow time scale is compatible with the stochastic kernels (see Remark 3.5). We are thus able to apply Theorem 3.7 and obtain the slow time scale Bellman recursion (36) as a special case of Equation (19). $\square$

The slow time scale Bellman equation (36) is as difficult to solve as the Bellman equation on the extended timeline. However, the interest of (36) lies elsewhere. Imagine that one is able to obtain, in a relatively easy way, lower $\underline{V}_s$ and upper $\overline{V}_s$ approximations of V_s in (36). Then, in order to obtain optimal controls for the optimization problem (6), one can proceed as follows. By replacing the last term V_{s+} of (36) by either $\underline{V}_{s+}$ or $\overline{V}_{s+}$, one can now solve a (lower or upper) surrogate of Equation (36) and thus obtain the optimal controls on the time interval $[s, s^+ [= \{(s, \bar{f}), (s^+, \underline{f}), \dots, (s^+, \bar{f}^-)\}$. For instance, one could use scenario decomposition methods, like progressive hedging [17], that do not require statistical independence of noises within the slow time interval $[s, s^+]$. Thus, the two-time-scale stochastic optimization problem as formulated in § 5.1 and § 5.2 can be approximatively solved, from below and from above, by a mix of slow time scale dynamic programming and of (for example) progressive hedging (or any other method, including dynamic programming).

⁶Here again, the formula (36) represents a nested sequence of infima of integrals (with respect to different stochastic kernels).

5.4. Illustration with the crude oil procurement problem

This illustration stems from a research work done in partnership with TotalEnergies, in the context of a PhD thesis [14]. Crude oil procurement is the part of the oil supply chain that sits between the production of crude oil and its processing in a refinery. The goal of procurement is to purchase crude oil from various suppliers around the world and having it delivered in time to the refinery to be processed. As illustrated in Figure 5.3, every month (on the bottom line) a refinery receives crudes that have been bought during the 8 previous weeks (on the upper line).

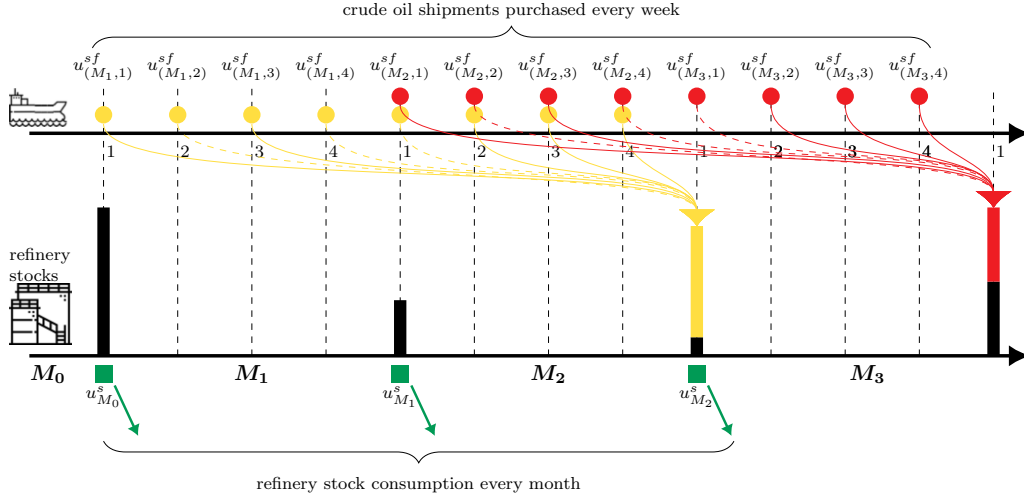


Figure 5.3: Procurement of crude oil over 3 months M_1 , M_2 and M_3 , where a circle $\circ$ denotes purchase decisions and a square $\square$ denotes consumption decisions

The problem naturally displays two time scales. On the one hand, deliveries to the refinery are made at the beginning of each month, and crude consumption is set once a month. On the other hand, crude oil shipments can be purchased at the frequency of the week; every week, a selection of shipments is presented to the decision-maker who must decide which shipments to purchase. Following the construction of the extended timeline in (27), we represent by the sequence

$$\begin{aligned}
 (M_0, 5) &\prec (M_1, 1) \prec (M_1, 2) \prec (M_1, 3) \prec (M_1, 4) \prec (M_1, 5) \\
 &\prec (M_2, 1) \prec (M_2, 2) \prec (M_2, 3) \prec (M_2, 4) \prec (M_2, 5) \\
 &\prec (M_3, 1) \prec (M_3, 2) \prec (M_3, 3) \prec (M_3, 4) \prec (M_3, 5)
 \end{aligned} \tag{37}$$

the timeline associated with Figure 5.3 (notice that we consider that a month is made of 4 weeks). The initial time stage $(M_0, 5)$ corresponds to the additional time $(\underline{s}^-, \bar{f})$ in (27). The times $(M_1, 5)$ and $(M_2, 5)$ both represent the “end of the month” when a consumption decision (slow scale decision u_s^s on the bottom line of Figure 5.3) is taken.

We now illustrate how the crude oil procurement problem can be put in the form of a two-time-scale optimization problem such as presented in §5.1. For this purpose, we proceed to the identifications in Table 5.1.

Notations from § 5.1	Crude oil procurement
S	set of months during which we manage the refinery; in Figure 5.3, $S = \{M_1, M_2, M_3\}$
F	set of weeks in each month; in Figure 5.3, $F = \{1, 2, 3, 4, 5\}$
$\mathbb{U}_s^s$	set of crude oil consumptions
$\mathbb{W}_{s^+}^s$	set of product prices
$\mathbb{U}_{(s,f)}^{sf}$	set of crude shipments purchased in week (s, f)
$\mathbb{W}_{(s,f)^+}^{sf}$	set of crude oil prices in week (s, f)
$\mathcal{G}_{(s,f)}^{sf}$	accumulation of shipments purchased in (s, f)
$\mathcal{G}_s^s$	delivery of orders and consumption of crude oil for the month s^+
Λ_s	operational costs during the month s (crude oil purchases during s - earnings from production)
Λ_{M_4}	end cost valuation of the buffers and stocks in the refinery before the beginning of the month M_4

Table 5.1: Identification of the elements introduced in § 5.1 with elements of the crude oil procurement problem

We call s -buffer (resp. s^- -buffer), the temporary stock that is created at the beginning of the month s (resp. s^-) and that will be delivered two months after. For instance, in Figure 5.3, the yellow disks represent the M_1 -buffer and the red disks represent the M_2 -buffer. We introduce the state variable

$$x_{(s,f)}^{sf} = \{s^- \text{-buffer}, s \text{-buffer}, \text{refinery stocks}\},$$

together with the accumulation dynamics $\mathcal{G}_{(s,f)}^{sf}$ for the buffers, and the accumulation dynamics $\mathcal{G}_s^s$ for the stocks. Regarding the criterion to minimize, it is an intertemporal cost like in (30) with slow time scale cost functions Λ_{s^-} made of minus the purchases of crude oil plus the selling of finished products inside a week. Supposing that the products prices are independent month by month, we represent this assumption by a family of constant stochastic kernels $\{\rho_{s^-,s^+}^s\}_{s \in \bar{S} \setminus \{\bar{s}\}}$. By contrast, we do not assume that the crude prices are independent week by week, and the possible dependency is modeled by stochastic kernels $\rho_{(s,f)^-, (s,f)^+}^{sf}$ for $f \in F \setminus \{\bar{f}\}$.

Now that all the elements from § 5.1 have been identified, Proposition 5.1 enables us to write a dynamic programming equation such as (36) at the scale of the month, without losing the time-dependency of crude prices inside the month.

6. Decision-hazard-decision optimization problems

In multistage stochastic optimization, the *decision-hazard-decision* (DHD) framework corresponds to the case where, at the beginning of each time interval, a decision is taken without knowing the uncertainty that will materialize at the end of the time

interval (decision-hazard), and, at the end of the time interval, a *recourse* decision is possible knowing this uncertainty (hazard-decision). The reader is referred to [5, §1.2.1] for the notions of decision-hazard and hazard-decision in stochastic optimal control. A discussion about these notions and an application in stochastic thermal scheduling can also be found in [21].

In § 6.1, we provide motivation for the decision-hazard-decision framework. In § 6.2, we formalize the decision-hazard-decision framework and we provide a dynamic programming equation.

6.1. Motivation for the decision-hazard-decision framework

We illustrate our motivation to develop a formalism for the decision-hazard-decision framework with a single dam management problem. We suppose given a stochastic process $\{\mathbf{A}_t\}_{t \in \llbracket 1, T \rrbracket}$ on a probability space. We can model the dynamics of the water volume in a dam by

$$\mathbf{S}_{t+1} = \min\{S^\sharp, \mathbf{S}_t - \mathbf{Q}_t + \mathbf{A}_{t+1}\}, \quad \forall t \in \llbracket 0, T-1 \rrbracket, \quad (38)$$

where $S^\sharp$ is the maximal dam volume, $\mathbf{S}_t$ is the volume (stock) of water at the beginning of period $[t, t+1]$, $\mathbf{A}_{t+1}$ is the inflow water volume (rain, etc.) during $[t, t+1]$, $\mathbf{Q}_t$ is the turbinéd outflow volume during $[t, t+1]$. The control variable $\mathbf{Q}_t$ is decided at the *beginning* of period $[t, t+1]$, chosen such that $0 \leq \mathbf{Q}_t \leq \mathbf{S}_t$, supposed to depend on the stock $\mathbf{S}_t$ but not on the inflow water $\mathbf{A}_{t+1}$ (as $\mathbf{A}_{t+1}$ takes place during $[t, t+1]$, hence materializes at $t+1$, hence the time index $t+1$). The min operation in Equation (38) ensures that the dam volume always remains below its maximal capacity $S^\sharp$, but induces a nonlinearity in the dynamics. This nonlinear dynamics is an obstacle to apply stochastic dual dynamic programming (SDDP).

Alternatively, we can model the dynamics of the water volume in a dam by

$$\mathbf{S}_{t+1} = \mathbf{S}_t - \mathbf{Q}_t + \mathbf{A}_{t+1} - \mathbf{R}_{t+1}, \quad \forall t \in \llbracket 0, T-1 \rrbracket, \quad (39)$$

where the new control variable $\mathbf{R}_{t+1}$ is the spilled volume, decided at the *end* of period $[t, t+1]$, supposed to depend both on the stock $\mathbf{S}_t$ and on the inflow water $\mathbf{A}_{t+1}$, and chosen such that $0 \leq \mathbf{S}_t - \mathbf{Q}_t + \mathbf{A}_{t+1} - \mathbf{R}_{t+1} \leq S^\sharp$. Thus, with the formulation (39), we “pay the price” to add one control $\mathbf{R}_{t+1}$, but we obtain a linear model instead of the nonlinear model (38). This is especially interesting when using the stochastic dual dynamic programming (SDDP) algorithm, for which the linearity of the dynamics is used to obtain the convexity properties required by the algorithm.

6.2. Decision-hazard-decision framework and dynamic programming equation

Let $\{\mathbb{U}_t^\sharp\}_{t \in \llbracket 0, T-1 \rrbracket}$ (head or “before” controls), $\{\mathbb{U}_t^\flat\}_{t \in \llbracket 1, T \rrbracket}$ (tail or “after” controls), $\{\mathbb{W}_t\}_{t \in \llbracket 1, T \rrbracket}$ (uncertainties) and $\{\mathbb{X}_t\}_{t \in \llbracket 0, T \rrbracket}$ (states) be sequences of Borel spaces.

Let also be given Borel-measurable dynamics mappings

$$f_t : \mathbb{X}_t \times \mathbb{U}_t^\sharp \times \mathbb{W}_{t+1} \times \mathbb{U}_{t+1}^\flat \rightarrow \mathbb{X}_{t+1}, \quad \forall t \in \llbracket 0, T-1 \rrbracket, \quad (40a)$$

nonnegative lower semianalytic instantaneous cost functions

$$L_t : \mathbb{X}_t \times \mathbb{U}_t^h \times \mathbb{W}_{t+1} \times \mathbb{U}_{t+1}^b \rightarrow [0, +\infty] , \quad \forall t \in \llbracket 0, T-1 \rrbracket , \quad (40b)$$

and a nonnegative lower semianalytic final cost function

$$K : \mathbb{X}_T \rightarrow [0, +\infty] . \quad (40c)$$

Finally, we suppose given a Borel probability space $(\Omega, \mathcal{F}, \mathbb{P})$, a random variable $\mathbf{X}_0 : \Omega \rightarrow \mathbb{X}_0$, and a stochastic process $\{\mathbf{W}_t\}_{t \in \llbracket 1, T \rrbracket}$ (the noise process), where $\mathbf{W}_t : \Omega \rightarrow \mathbb{W}_t$ for $t \in \llbracket 1, T \rrbracket$.

Thus equipped, we consider the following multistage stochastic optimization problem

$$\inf_{\{(\mathbf{U}_t^h, \mathbf{U}_{t+1}^b)\}_{t \in \llbracket 0, T-1 \rrbracket}} \mathbb{E} \left[\sum_{t=0}^{T-1} L_t(\mathbf{X}_t, \mathbf{U}_t^h, \mathbf{W}_{t+1}, \mathbf{U}_{t+1}^b) + K(\mathbf{X}_T) \right] , \quad (41a)$$

$$\sigma(\mathbf{U}_t^h) \subset \sigma(\mathbf{X}_0, \mathbf{W}_1, \dots, \mathbf{W}_t) , \quad \forall t \in \llbracket 0, T-1 \rrbracket , \quad (41b)$$

$$\sigma(\mathbf{U}_t^b) \subset \sigma(\mathbf{X}_0, \mathbf{W}_1, \dots, \mathbf{W}_t) , \quad \forall t \in \llbracket 1, T \rrbracket , \quad (41c)$$

$$\mathbf{X}_{t+1} = f_t(\mathbf{X}_t, \mathbf{U}_t^h, \mathbf{W}_{t+1}, \mathbf{U}_{t+1}^b) , \quad \forall t \in \llbracket 0, T-1 \rrbracket . \quad (41d)$$

Thus, in the above setting, during the time interval between two time steps, the decision-maker makes two decisions. At the end of the time interval $[t-1, t[$, a random variable $\mathbf{W}_t$ is revealed, and then, at the beginning of the time interval $[t, t+1[$, the decision-maker makes a *head decision* $\mathbf{U}_t^h$. What is new – in comparison with the classical decision-hazard framework – is that, at the end of the time interval $[t, t+1[$, when a next random variable $\mathbf{W}_{t+1}$ is revealed, the decision-maker has the possibility to make a *tail decision* $\mathbf{U}_{t+1}^b$. This latter decision $\mathbf{U}_{t+1}^b$ can be thought as a *recourse* variable for a two stage stochastic optimization problem that would take place inside the time interval $[t, t+1[$. Note that, because of the term $(\mathbf{U}_t^h, \mathbf{U}_{t+1}^b)$ in the cost function L_t and in the dynamics f_t , considering the pair $(\mathbf{U}_t^h, \mathbf{U}_t^b)$ as the control variable at time t would not satisfy the assumptions of Theorem 3.7.

Proposition 6.1. *If the a random variable $\mathbf{X}_0$ is deterministic with value x_0 , and if the noise process $\{\mathbf{W}_t\}_{t \in \llbracket 1, T \rrbracket}$ is white, that is, is made of independent random variables, then the value $V_0(x_0)$ of the multistage stochastic optimization problem (41) is given by the dynamic programming backward induction*

$$V_T(x) = K(x) , \quad (42a)$$

and, for $t \in \llbracket 0, T-1 \rrbracket$,

$$V_t(x) = \inf_{u_t^h \in \mathbb{U}_t^h} \mathbb{E} \left[\inf_{u_{t+1}^b \in \mathbb{U}_{t+1}^b} \left\{ L_t(x, u_t^h, \mathbf{W}_{t+1}, u_{t+1}^b) + V_{t+1}(f_t(x, u_t^h, \mathbf{W}_{t+1}, u_{t+1}^b)) \right\} \right] . \quad (42b)$$

Proof. As the statement is made with random variables, whereas the theory has been developed with stochastic kernels, we make the link as follows: for each time $t \in \llbracket 1, T \rrbracket$, the stochastic kernel $\rho_{t-1:t}$ in (3) is the probability distribution of the

random variable $\mathbf{W}_t$. This done, the proof is an application of Theorem 3.7, as follows (we just sketch the procedure, as the detailed proof can be found in the preprint [6]).

We rename the uncertainty sets $\{\mathbb{W}_t\}_{t \in \llbracket 1, T \rrbracket}$ as $\{\mathbb{W}_t^b\}_{t \in \llbracket 1, T \rrbracket}$ and we introduce for each time $t \in \llbracket 1, T \rrbracket$ a spurious uncertainty variable w_t^b taking values in a singleton set $\mathbb{W}_t^b = \{\bar{w}_t^b\}$, so that we obtain the following sequence of events

$$\underbrace{w_0^b}_{=x_0} \rightsquigarrow u_0^b \rightsquigarrow w_1^b \rightsquigarrow u_1^b \rightsquigarrow \underbrace{w_1^b}_{\text{spurious}} \rightsquigarrow u_1^b \rightsquigarrow w_2^b \rightsquigarrow u_2^b \rightsquigarrow \underbrace{w_2^b}_{\text{spurious}} \rightsquigarrow u_2^b \rightsquigarrow \dots$$

$$\rightsquigarrow w_{T-1}^b \rightsquigarrow u_{T-1}^b \rightsquigarrow \underbrace{w_{T-1}^b}_{\text{spurious}} \rightsquigarrow u_{T-1}^b \rightsquigarrow w_T^b \rightsquigarrow u_T^b \rightsquigarrow \underbrace{w_T^b}_{\text{spurious}} .$$

Proceeding this way, we have doubled the timeline as each time t has been “duplicated” in the ordered pair (t, b) and $(t, \bar{b})$. With this, we embed the decision-hazard-decision structure as a particular case of multiple consecutive time blocks (of size 2) as in § 3.2. $\square$

Eq. (42) can be solved using the stochastic dual dynamic programming (SDDP) algorithm provided that lower semicontinuity and convexity of the value functions are preserved. This is ensured first by assuming linearity in the dynamics – a feature that may be obtained by modeling the problem in the decision-hazard-decision framework as illustrated in § 6.1 – and second by assuming convexity and lower semicontinuity for the cost functions as well as compactness for the existence of optimal controls.

7. Conclusion and perspectives

As said in the introduction, decomposition methods are appealing to tackle multi-stage stochastic optimization problems, as they are naturally large scale. The most common approaches are time decomposition (and state-based resolution methods, like stochastic dynamic programming, in stochastic optimal control), and scenario decomposition (like progressive hedging in stochastic programming).

This paper is part of a general research program that consists in *mixing* different decomposition bricks. Space decomposition methods have been investigated in [1] and [7]. Here, we have tackled the issue of using time block decomposition in such a way that stochastic dynamic programming is used at the slow time scale with an appropriate white noise assumption, whereas stochastic programming methods such as progressive hedging can be used at the fast time scale where such an independence assumption does not hold. This approach paves the way of mixing time decomposition with scenario decomposition. For this purpose, we have revisited the notion of state, and have provided a way to perform time decomposition but only across specified time blocks.

Acknowledgements. We thank Roger Wets for fruitful discussions about the possibility of mixing stochastic dynamic programming with progressive hedging.

A. Technical details and proofs

We suppose to be in the framework of § 2.1. We introduce the notations

$$\mathbb{W}_{r:t} = \prod_{s=r}^t \mathbb{W}_s, \quad 0 \leq r \leq t \leq T, \quad \mathbb{U}_{r:t} = \prod_{s=r}^t \mathbb{U}_s, \quad 0 \leq r \leq t \leq T-1. \quad (43)$$

Let $0 \leq r \leq s \leq t \leq T$. From a history $h_t \in \mathbb{H}_t$, we extract the $(r : s)$ -history uncertainty part

$$[h_t]_{r:s}^{\mathbb{W}} = (w_r, \dots, w_s) = w_{r:s} \in \mathbb{W}_{r:s}, \quad 0 \leq r \leq s \leq t, \quad (44a)$$

the $(r : s)$ -history control part (notice that the indices are special)

$$[h_t]_{r:s}^{\mathbb{U}} = (u_{r-1}, \dots, u_{s-1}) = u_{r-1:s-1} \in \mathbb{U}_{r-1:s-1}, \quad 1 \leq r \leq s \leq t. \quad (44b)$$

Flows. Let r and t be given such that $0 \leq r < t \leq T$. For a $(r : t-1)$ -history feedback $\gamma = \{\gamma_s\}_{s \in \llbracket r, t \rrbracket - 1} \in \Gamma_{r:t-1}$, we define the flow $\Phi_{r:t}^\gamma$ by

$$\Phi_{r:t}^\gamma : \mathbb{H}_r \times \mathbb{W}_{r+1:t} \rightarrow \mathbb{H}_t \quad (45a)$$

$$(h_r, w_{r+1:t}) \mapsto (h_r, \gamma_r(h_r), w_{r+1}, \gamma_{r+1}(h_r, \gamma_r(h_r), w_{r+1}), w_{r+2}, \dots, \gamma_{t-1}(h_{t-1}), w_t).$$

Otherwise stated, the flow is given by

$$\Phi_{r:t}^\gamma(h_r, w_{r+1:t}) = (h_r, u_r, w_{r+1}, u_{r+1}, w_{r+2}, \dots, u_{t-1}, w_t), \quad (45b)$$

$$\text{with} \quad h_s = (h_r, u_r, w_{r+1}, \dots, u_{s-1}, w_s), \quad r < s \leq t, \quad (45c)$$

$$\text{and} \quad u_s = \gamma_s(h_s), \quad r \leq s \leq t-1. \quad (45d)$$

When $0 \leq r = t \leq T$, we put $\Phi_{r:r}^\gamma : \mathbb{H}_r \rightarrow \mathbb{H}_r$, $h_r \mapsto h_r$.

With this convention, the expression $\Phi_{r:t}^\gamma$ makes sense when $0 \leq r \leq t \leq T$. The mapping $\Phi_{r:t}^\gamma$ gives the history at time t as a function of the initial history h_r at time r and of the history feedbacks $\{\gamma_s\}_{s \in \llbracket r, t \rrbracket - 1} \in \Gamma_{r:t-1}$. An immediate consequence of this definition are the *flow properties*:

$$\Phi_{r:t+1}^\gamma(h_r, w_{r+1:t+1}) = \left\{ \Phi_{r:t}^\gamma(h_r, w_{r+1:t}), \gamma_t(\Phi_{r:t}^\gamma(h_r, w_{r+1:t})), w_{t+1} \right\}, \quad (46a)$$

$$0 \leq r \leq t \leq T-1,$$

$$\Phi_{r:t}^\gamma(h_r, w_{r+1:t}) = \Phi_{r+1:t}^\gamma \left\{ (h_r, \gamma_r(h_r), w_{r+1}), w_{r+2:t} \right\}, \quad 0 \leq r < t \leq T. \quad (46b)$$

We recall that $\mathbb{L}_+^0(\mathbb{H}_t)$ denotes the space of lower semianalytic nonnegative numerical functions over $\mathbb{H}_t$.

Definition A.1. Let r and t be given such that $0 \leq r \leq t \leq T$.

- When $0 \leq r < t \leq T$, for a $(r : t-1)$ -history feedback $\gamma = \{\gamma_s\}_{s \in \llbracket r, t-1 \rrbracket} \in \Gamma_{r:t-1}$, and for a given family $\{\rho_{s-1:s}\}_{s \in \llbracket r+1, t \rrbracket}$ of Borel-measurable stochastic kernels $\rho_{s-1:s} : \mathbb{H}_{s-1} \rightarrow \Delta(\mathbb{W}_s)$, $s \in \llbracket r+1, t \rrbracket$, we define a Borel-measurable stochastic kernel $\rho_{r:t}^\gamma : \mathbb{H}_r \rightarrow \Delta(\mathbb{H}_t)$ such that, for any numerical function $\varphi \in \mathbb{L}_+^0(\mathbb{H}_t)$, we have that

$$\begin{aligned} & \int_{\mathbb{H}_t} \varphi(h'_r, h'_{r+1:t}) \rho_{r:t}^\gamma(dh'_t \mid h_r) \\ &= \int_{\mathbb{W}_{r+1:t}} \varphi(\Phi_{r:t}^\gamma(h_r, w_{r+1:t})) \prod_{s=r+1}^t \rho_{s-1:s}(dw_s \mid \Phi_{r:s-1}^\gamma(h_r, w_{r+1:s-1})). \end{aligned} \quad (47)$$

- When $0 \leq r = t \leq T$, we define $\rho_{r:r}^\gamma : \mathbb{H}_r \rightarrow \Delta(\mathbb{H}_r)$ by $\rho_{r:r}^\gamma(dh'_r \mid h_r) = \delta_{h_r}(dh'_r)$ where δ represents the Dirac measure.

The stochastic kernels $\rho_{r:t}^\gamma$ on $\mathbb{H}_t$, given by (47), are of the form

$$\rho_{r:t}^\gamma(dh'_t | h_r) = \rho_{r:t}^\gamma(dh'_r dh'_{r+1:t} | h_r) = \delta_{h_r}(dh'_r) \otimes \varrho_{r:t}^\gamma(dh'_{r+1:t} | h_r), \quad (48)$$

where, for each $h_r \in \mathbb{H}_r$, the probability distribution $\varrho_{r:t}^\gamma(dh'_{r+1:t} | h_r)$ only charges the histories visited by the flow from $r + 1$ to t . The construction of the stochastic kernels $\rho_{r:t}^\gamma$ is developed in [3, p. 190] for relaxed history feedbacks and obtained by using [3, Proposition 7.45, p. 175].

Proposition A.2. *The family $\{\rho_{s:t}^\gamma\}_{s \in \llbracket r, t \rrbracket}$ of stochastic kernels of Definition A.1 has the flow property:*

$$\rho_{s:t}^\gamma(dh'_t | h_s) = \int_{\mathbb{W}_{s+1}} \rho_{s:s+1}(dw_{s+1} | h_s) \rho_{s+1:t}^\gamma(dh'_t | (h_s, \gamma_s(h_s), w_{s+1})), \quad \forall s < t. \quad (49)$$

Proof. Let $s < t$. For any $\varphi \in \mathbb{L}_+^0(\mathbb{H}_t)$, we have that

$$\begin{aligned} & \int_{\mathbb{H}_t} \varphi(h'_s, h'_{s+1:t}) \rho_{s:t}^\gamma(dh'_t | h_s) \\ &= \int_{\mathbb{W}_{s+1:t}} \varphi(\Phi_{s:t}^\gamma(h_s, w_{s+1:t})) \int_{\mathbb{W}_{s+1:t}} \prod_{s'=s+1}^t \rho_{s'-1:s'}(dw_{s'} | \Phi_{s:s'-1}^\gamma(h_s, w_{s+1:s'-1})) \end{aligned} \quad (50a)$$

by Definition (47)

$$\begin{aligned} &= \int_{\mathbb{W}_{s+1:t}} \varphi(\Phi_{s:t}^\gamma(h_s, w_{s+1:t})) \rho_{s:s+1}(dw_{s+1} | h_s) \\ & \quad \int_{\mathbb{W}_{s+1:t}} \prod_{s'=s+2}^t \rho_{s'-1:s'}(dw_{s'} | \Phi_{s:s'-1}^\gamma(h_s, w_{s+1:s'-1})) \\ &= \int_{\mathbb{W}_{s+1:t}} \varphi(\Phi_{s+1:t}^\gamma((h_s, \gamma_s(h_s), w_{s+1}), w_{s+2:t})) \rho_{s:s+1}(dw_{s+1} | h_s) \\ & \quad \prod_{s'=s+2}^t \rho_{s'-1:s'}(dw_{s'} | \Phi_{s+1:s'-1}^\gamma((h_s, \gamma_s(h_s), w_{s+1}), w_{s+2:s'-1})) \\ & \quad \text{(by the flow property (46b))} \\ &= \int_{\mathbb{W}_{s+1}} \rho_{s:s+1}(dw_{s+1} | h_s) \int_{\mathbb{W}_{s+2:t}} \varphi(\Phi_{s+1:t}^\gamma((h_s, \gamma_s(h_s), w_{s+1}), w_{s+2:t})) \\ & \quad \prod_{s'=s+2}^t \rho_{s'-1:s'}(dw_{s'} | \Phi_{s+1:s'-1}^\gamma((h_s, \gamma_s(h_s), w_{s+1}), w_{s+2:s'-1})) \end{aligned}$$

by Fubini's Theorem

$$\begin{aligned} &= \int_{\mathbb{W}_{s+1}} \rho_{s:s+1}(dw_{s+1} | h_s) \\ & \quad \int_{\mathbb{H}_t} \varphi((h'_s, \gamma_s(h'_s), w'_{s+1}), h'_{s+2:t}) \rho_{s+1:t}^\gamma(dh'_t | (h_s, \gamma_s(h_s), w_{s+1})) \end{aligned}$$

by Definition (47)

$$= \int_{\mathbb{H}_t} \varphi((h'_s, \gamma_s(h'_s), w'_{s+1}), h'_{s+2:t}) \quad (50b)$$

$$\int_{\mathbb{W}_{s+1}} \rho_{s:s+1}(dw_{s+1} \mid h_s) \rho_{s+1:t}^\gamma(dh'_t \mid (h_s, \gamma_s(h_s), w_{s+1})) \quad (50c)$$

by Fubini Theorem. As the two expressions (50a) and (50c) are equal for any $\varphi \in \mathbb{L}_+^0(\mathbb{H}_t)$, we deduce the flow property (49). $\square$

Proof of Theorem 2.1. We only give a sketch of the proof, as it is a variation on different results of [3], the framework of which we follow.

We are in the setting of [3, Chap. 7 and 8]. We take the history space $\mathbb{H}_t$ for state space, and the state dynamics

$$f(h_t, u_t, w_{t+1}) = (h_t, u_t, w_{t+1}) = h_{t+1} \in \mathbb{H}_{t+1} = \mathbb{H}_t \times \mathbb{U}_t \times \mathbb{W}_{t+1}. \quad (51)$$

Then, the family $\{\rho_{s-1:s}\}_{s \in \llbracket 1, T \rrbracket}$ of Borel-measurable stochastic kernels (3) gives a family of disturbance kernels (vocabulary of [3, p 189]) that do not depend on the current control. The criterion to be minimized (4) is a function of the history at time T , thus of the state at time T . We consider the finite horizon model with final cost corresponding to the optimization problem defined by the associated value function (6):

$$\begin{aligned} V_0(w_0) &= \inf_{\gamma_{0:T-1} \in \Gamma_{0:T-1}} \int_{\mathbb{H}_T} j(h'_T) \rho_{0:T}^\gamma(dh'_T \mid w_0) \\ &= \inf_{\gamma_{0:T-1} \in \Gamma_{0:T-1}} \int_{\mathbb{W}_{1:T}} j(\Phi_{0:T}^\gamma(w_{0:T})) \prod_{s=1}^T \rho_{s-1:s}(dw_s \mid \Phi_{0:s-1}^\gamma(w_{0:s-1})), \quad (\text{by (47)}) \end{aligned}$$

where the flows $\Phi_{0:s}^\gamma$ for $s \in \llbracket 0, T-1 \rrbracket$ are defined by Equation (45a), and where we are minimizing over the so-called state-feedbacks. Then, the proof of Theorem 2.1 follows from the results developed in Chap. 7, 8 and 10 of [3] in a Borel setting.

The Bellman operators in (7a) satisfy (7b) because, by Lemma 7.30(4) and Propositions 7.47 and 7.48 in [3], we have that $\varphi \in \mathbb{L}_+^0(\mathbb{H}_{t+1}) \implies \mathcal{B}_{t+1:t} \varphi \in \mathbb{L}_+^0(\mathbb{H}_t)$, for t in $\llbracket 0, T-1 \rrbracket$.

Since we are considering a finite horizon model with a final cost, we detail the steps needed to use the results of [3, Chap. 8]. The final cost at time T can be turned into an instantaneous cost at time $T-1$ by inserting the state dynamics (51) in the final cost. Getting rid of the disturbance in the expected cost by using the disturbance kernel is standard practice. Then, we can turn this non-homogeneous finite horizon model into a finite horizon model with homogeneous dynamics and costs by following the steps of [3, Chap. 10]. Using [3, Proposition 8.2, p. 198], we obtain that the family of optimization problems defined by the associated value functions (5), when minimizing over the relaxed state feedbacks, satisfies the Bellman equation (8); we

⁷ More precisely, the property (7b) is a consequence of the properties that i) the Bellman operator (7a) corresponds to the operator T (with $g = 0$ and $\alpha = 1$) in [3, Definition 8.5, p. 195] ii) that $T(J)$ is lower semianalytic whenever J also is, as explained right above [3, Lemma 8.2, p. 196].

conclude with [3, Proposition 8.4, p. 203] which covers the minimization over state feedbacks.

The Bellman equation (8) is a consequence of [3, Proposition 8.2, p. 198].

To finish, Theorem 2.1 is valid under the general Borel assumptions of [3, Chap. 8] and with the specific (F^+) assumption needed for [3, Proposition 8.4, p. 203]; this last assumption is fulfilled here since we have assumed that the cost criterion (4) is nonnegative. $\square$

Proof of Proposition 3.2. We suppose to be in the framework of § 2.1.

We are in the setting of [3, Chap. 7, Chap. 8]. Let $\tilde{\varphi}_t : \mathbb{X}_t \rightarrow [0, +\infty]$ be a given lower semianalytic nonnegative numerical function, and let $\varphi_t : \mathbb{H}_t \rightarrow [0, +\infty]$ be

$$\varphi_t = \tilde{\varphi}_t \circ \theta_t. \quad (52)$$

Let $\varphi_r : \mathbb{H}_r \rightarrow [0, +\infty]$ be the lower semianalytic nonnegative numerical function obtained by applying the Bellman operator $\mathcal{B}_{t:r}$ across $\llbracket t, r \rrbracket$ (see (10)) to the lower semianalytic nonnegative numerical function φ_t :

$$\varphi_r = \mathcal{B}_{t:r}\varphi_t = \mathcal{B}_{r+1:r} \circ \cdots \circ \mathcal{B}_{t:t-1}\varphi_t. \quad (53)$$

By [3, Lemma 7.30(3), p. 178] – on the stability of lower semianalytic functions under right composition with a Borel-measurable mapping – we get that the nonnegative numerical function φ_r is lower semianalytic. We show that there exists a lower semianalytic nonnegative numerical function $\tilde{\varphi}_r : \mathbb{X}_r \rightarrow [0, +\infty]$ such that

$$\varphi_r = \tilde{\varphi}_r \circ \theta_r. \quad (54)$$

First, we show by backward induction that, for all $s \in \llbracket r, t \rrbracket$, there exists a lower semianalytic nonnegative numerical function $\bar{\varphi}_s$ such that $\varphi_s(h_s) = \bar{\varphi}_s(\theta_r(h_r), h_{r+1:s})$. Second, we prove that the function $\tilde{\varphi}_r = \bar{\varphi}_r$ satisfies (54) and is lower semianalytic.

(1) For $s = t$, we have, by (52) and by (9c), that

$$\varphi_t(h_t) = \tilde{\varphi}_t(\theta_t(h_t)) = \tilde{\varphi}_t(f_{r:t}(\theta_r(h_r), h_{r+1:t})),$$

so that the nonnegative numerical function $\bar{\varphi}_t$ is given by $\tilde{\varphi}_t \circ f_{r:t}$. By [3, Lemma 7.30(3), p. 178], $\bar{\varphi}_t$ is a lower semianalytic numerical function.

(2) Assume that, at $s + 1$, the result holds true, that is,

$$\varphi_{s+1}(h_{s+1}) = \bar{\varphi}_{s+1}(\theta_r(h_r), h_{r+1:s+1}),$$

where the numerical function $\bar{\varphi}_{s+1}$ is nonnegative lower semianalytic. Then, by (53),

$$\begin{aligned} \varphi_s(h_s) &= (\mathcal{B}_{s+1:s}\varphi_{s+1})(h_s) \\ &= \inf_{u_s \in \mathbb{U}_s} \int_{\mathbb{W}_{s+1}} \varphi_{s+1}((h_s, u_s, w_{s+1})) \rho_{s:s+1}(dw_{s+1} \mid h_s) \\ &\quad \text{(by definition (7a) of the Bellman operator)} \\ &= \inf_{u_s \in \mathbb{U}_s} \int_{\mathbb{W}_{s+1}} \bar{\varphi}_{s+1}((\theta_r(h_r), (h_{r+1:s}, u_s, w_{s+1}))) \inf_{u_s \in \mathbb{U}_s} \int_{\mathbb{W}_{s+1}} \rho_{s:s+1}(dw_{s+1} \mid h_s) \\ &\quad \text{(by the induction assumption)} \end{aligned}$$

$$\begin{aligned}
&= \inf_{u_s \in \mathbb{U}_s} \int_{\mathbb{W}_{s+1}} \bar{\varphi}_{s+1}((\theta_r(h_r), (h_{r+1:s}, u_s, w_{s+1}))) \\
&\quad \inf_{u_s \in \mathbb{U}_s} \int_{\mathbb{W}_{s+1}} \tilde{\rho}_{s:s+1}(dw_{s+1} \mid (\theta_r(h_r), h_{r+1:s})) \\
&\quad \quad \quad \text{(by compatibility of the stochastic kernel)} \\
&= \bar{\varphi}_s(\theta_r(h_r), h_{r+1:s}),
\end{aligned}$$

where

$$\bar{\varphi}_s(x_r, h_{r+1:s}) = \inf_{u_s \in \mathbb{U}_s} \int_{\mathbb{W}_{s+1}} \bar{\varphi}_{s+1}((x_r, (h_{r+1:s}, u_s, w_{s+1}))) \tilde{\rho}_{s:s+1}(dw_{s+1} \mid (x_r, h_{r+1:s})).$$

By [3, p.196] (right before Lemma 8.2), we get that the numerical function $\bar{\varphi}_s$ is nonnegative lower semianalytic. Thus, we have shown that the result holds true at time s .

The induction implies that, at time r , the expression of $\varphi_r(h_r)$ is $\varphi_r(h_r) = \bar{\varphi}_r(\theta_r(h_r))$, since the term $h_{r+1:r}$ vanishes. Choosing $\tilde{\varphi}_r = \bar{\varphi}_r$ gives the expected result. $\square$

References

- [1] K. Barty, P. Carpentier, P. Girardeau: *Decomposition of large-scale stochastic optimal control problems*, RAIRO Oper. Res. 44/3 (2010) 167–183.
- [2] R. E. Bellman: *Dynamic Programming*, Princeton University Press, Princeton (1957).
- [3] D. P. Bertsekas, S. E. Shreve: *Stochastic Optimal Control: The Discrete-Time Case*, Athena Scientific, Belmont (1996).
- [4] J. R. Birge: *Aggregation bounds in stochastic linear programming*, Math. Programming 31 (1985) 25–41.
- [5] P. Carpentier, J.-P. Chancelier, G. Cohen, M. De Lara: *Stochastic Multi-Stage Optimization. At the Crossroads between Discrete Time Stochastic Control and Stochastic Programming*, Springer, Berlin (2015).
- [6] P. Carpentier, J.-P. Chancelier, M. De Lara, T. Martin, T. Rigaut: *Time Block Decomposition of Multistage Stochastic Optimization Problems*, preprint hal-01757113 (2023).
- [7] P. Carpentier, J.-P. Chancelier, M. De Lara, F. Pacaud: *Mixed spatial and temporal decompositions for large-scale multistage stochastic optimization problems*, J. Optim. Theory Appl. 186/3 (2020) 985–1005.
- [8] B. Cheng, W. B. Powell: *Co-optimizing battery storage for the frequency regulation and energy arbitrage using multi-scale dynamic programming*, IEEE Trans. Smart Grid 9/3 (2016) 1997–2005.
- [9] I. V. Evstigneev: *Measurable selection and dynamic programming*, Math. Oper. Res. 1/3 (1976) 267–272.
- [10] M. Glanzer, G. Ch. Pflug: *Multiscale stochastic optimization: modeling aspects and scenario generation*, Comp. Optimization Appl. 75/1 (2020) 1–34.
- [11] O. Hernández-Lerma, J.-B. Lasserre: *Discrete-Time Markov Control Processes: Basic Optimality Criteria*, Stochastic Modelling and Applied Probability Vol. 30, Springer, New York (1996).

- [12] M. Kaut, K. T. Midthun, A. S. Werner, A. Tomasgard, L. Hellemo, M. Fodstad: *Multi-horizon stochastic programming*, Comp. Management Sci. 11/1 (2014) 179–193.
- [13] D. Kuhn: *Aggregation and discretization in multistage stochastic programming*, Math. Programming 113 (2008) 61–94.
- [14] T. Martin: *Stochastic Optimization for the Procurement of Crude Oil in Refineries*, PhD Thesis, École des Ponts ParisTech, Paris (2021).
- [15] R. Porteiro, A. Ferragut, F. Paganini: *Towards multi-timescale energy provisioning using stochastic dual dynamic programming*, in: IEEE 9th Power, Instrumentation and Measurement Meeting (EPIM) (2018) 1–6.
- [16] M. L. Puterman: *Markov Decision Processes*, Wiley, New York (1994).
- [17] R. T. Rockafellar, R. J-B Wets: *Scenarios and policy aggregation in optimization under uncertainty*, Math. Oper. Res. 16/1 (1991) 119–147.
- [18] A. Ruszczyński: *Decomposition methods*, in: *Stochastic Programming*, Handbooks in Operations Research and Management Science Vol 10, Elsevier, Amsterdam (2003) 141–211.
- [19] A. Shapiro, D. Dentcheva, A. Ruszczyński: *Lectures on Stochastic Programming: Modeling and Theory*, MPS/SIAM Series on Optimization Vol. 9, Philadelphia (2009).
- [20] A. Shapiro, L. Ding: *Periodical multistage stochastic programs*, SIAM J. Optimization 30/3 (2020) 2083–2102.
- [21] A. Street, D. Valladao, A. Lawson, A. Velloso: *Assessing the cost of the hazard-decision simplification in multistage stochastic hydrothermal scheduling*, Applied Energy 280 (2020), art. no. 115939.
- [22] J. Subramanian, A. Mahajan: *Approximate information state for partially observed systems*, in: IEEE 58th Conference on Decision and Control (CDC) (2019) 1629–1636.
- [23] P. Whittle: *Optimization over Time. Vol. 1: Dynamic Programming and Stochastic Control*, Wiley, New York (1982).
- [24] H. S. Witsenhausen: *A standard form for sequential stochastic control*, Math. Systems Theory 7/1 (1973) 5–11.
- [25] S. E. Wright: *Primal-dual aggregation and disaggregation for stochastic linear programs*, Math. Oper. Res. 19/4 (1994) 893–908.
- [26] S. Yüksel: *A universal dynamic program and refined existence results for decentralized stochastic control*, SIAM J. Control Optim. 58/5 (2020) 2711–2739.