RAIRO-Oper. Res. 44 (2010) 167–183
DOI: 10.1051/ro/2010013

RAIRO Operations Research
www.rairo-ro.org

DECOMPOSITION OF LARGE-SCALE STOCHASTIC OPTIMAL CONTROL PROBLEMS

Kengy Barty[1], Pierre Carpentier[2]
and Pierre Girardeau[3]

Abstract. In this paper, we present an Uzawa-based heuristic that is adapted to certain type of stochastic optimal control problems. More precisely, we consider dynamical systems that can be divided into small-scale subsystems linked through a static almost sure coupling constraint at each time step. This type of problem is common in production/portfolio management where subsystems are, for instance, power units, and one has to supply a stochastic power demand at each time step. We outline the framework of our approach and present promising numerical results on a simplified power management problem.

Keywords. Stochastic optimal control, decomposition methods, dynamic programming.

Mathematics Subject Classification. 93E20, 49M27, 49L20.

1. Introduction

Stochastic optimal control is concerned with finding strategies to manage dynamical systems in an optimal way, with respect to some cost function. The particularity of such optimization problems is that the optimization variables we deal with are random variables. Indeed, the dynamical systems we consider are

Received March 3, 2009. Accepted April 7, 2010.

[1] EDF R&D, 1, avenue du Général de Gaulle, 92141 Clamart Cedex, France

[2] ENSTA ParisTech, 32, boulevard Victor, 75739 Paris Cedex 15, France; pierre.carpentier@ensta.fr

[3] Université Paris-Est, CERMICS, École des Ponts, Champs sur Marne, 77455 Marne la Vallée Cedex 2, France, also with EDF R&D and ENSTA

Article published by EDP Sciences

partially driven by some exogenous noises and the objective function may also include such noises. Hence controls are random variables. Classical approaches such as Dynamic Programming and Stochastic Programming, that we briefly recall below, encounter difficulties when the system becomes large. The aim of this paper is to present a new heuristic to solve a class of such problems using price decomposition.

The problems we are studying are common in practice. For example, consider a physical system, say a set of numerous power units, that evolve depending on exogenous noises (water inflows, failures) and on controls (production levels). At each time step, an observation on the system arises and a control has to be chosen on the basis of the available information, namely the past observations (non-anticipativity constraint). The objective is to minimize the sum of the units' production costs over a given discretized time horizon, while satisfying a global demand constraint at each time step. This decision process hence consists of finding optimal strategies, *i.e.* functions that map, at each time t, the available information to the optimal decision with respect to the production cost.

As far as we know, most methods that have been proposed to decompose large-scale stochastic optimal control problems are based on Stochastic Programming (see [18,20]). This approach consists of representing the non-anticipativity constraints using a so-called scenario tree. Once discretized on such a structure, the problem is not stochastic anymore and various deterministic decomposition techniques have been used to solve it (see [16]). In this context, there are two main issues that are not easy to deal with. The first is concerned with the "distance" between the original problem and its deterministic reformulation [2,15], or how to draw a scenario tree in such a way that the solution of the discretized problem is an accurate estimate of the original one. [19] proves an upper bound on complexity for the Sample Average Approximation method (SAA), which seems to indicate that the number of scenarios needed to solve the true problem with a given accuracy grows exponentially with the time horizon. Some numerical experiments seem to confirm his conclusion (see [7,14]). The second issue is concerned with the way one can rebuild strategies from optimal commands obtained in the discretized problem [17].

On the other hand, when dealing with a Markov Decision Process, methods based on Dynamic Programming (DP) (see [4,5]) do provide a way to obtain strategies as feedback functions with respect to so-called state variables. Unfortunately, the well-known curse of dimensionality prevents us from using this approach straightforward on large-scale problems, because the computational burden increases exponentially with the state dimension. Numerous approximations have been proposed to tackle the difficulty. For instance, a popular idea in the field of hydro-power management, introduced by Turgeon in [22], consists of obtaining local strategies as a function of the local stock and the aggregated complementary stock. Another idea, namely Approximate Dynamic Programming (ADP), looks for the value functions (solutions of the DP equation) within a finite-dimensional space (see [3] or [8], Sect. 6.5). To be practically efficient, such an approach requires some a priori information about the problem, in order to define a well suited functional subspace. Indeed, there is no systematic means to choose the basis functions and several choices have been proposed [12,23].

When dealing with large-scale optimization problems, the decomposition/coordination approach aims at finding a solution to the original problem by iteratively solving several smaller-dimensional subproblems. In the deterministic case, several types of decomposition have been proposed (*e.g.* by prices or by quantities) and unified in a general framework using the Auxiliary Problem Principle in [10]. In the open-loop stochastic case, *i.e.* when controls do not rely on any observation, [9] proposed to take advantage of both decomposition techniques and stochastic gradient algorithms. These techniques have been extended in the closed-loop stochastic case by [6], but so far they fail to provide decomposed state dependent strategies in the Markovian case. This is because a subproblem's optimal strategy depends on the state of the whole system, not only on the local state. In other words, decomposition approaches are meant to decompose the control space, namely the range of the strategy, but the numerical complexity of the problems we consider here also arises because of the dimensionality of the state space, that is to say the domain of the strategy.

We propose here a way to use price decomposition within the closed-loop stochastic case. The coupling constraints, namely the constraints preventing the problem from being naturally decomposed, are dualized using a Lagrange multiplier (price). At each iteration, the price decomposition algorithm solves each subproblem using the current prices, then uses the solutions to update the prices. In the stochastic context, prices are a random process whose dynamics are not available, so the subproblems do not in general fall into the Markovian setting. However, on a specific instance of this problem, [21] exhibits a dynamics for the optimal multipliers, and he shows that these dynamics are independent of the decision variables. Hence it is possible to come down to the Markovian framework and to use DP to solve the subproblems in this case. Following this idea, we propose to choose a parameterized dynamics for these multipliers so that solving subproblems using DP becomes possible. The update is then performed using a sampling/regression technique. Based on these ideas, the algorithm is twofold. On the one hand, it gives a lower bound for the value of the original problem by designing an approximate solution for the dual problem. On the other hand, it builds feedback functions which may allow us to design feasible strategies for the primal formulation. Such strategies will be built in the application we are concerned with in Section 6.

This paper is organized as follows. In Section 2 we describe the general type of problems we are concerned with in this paper. Then, in Section 3, we recall the Dynamic Programming equation and highlight the difficulties induced when considering large-scale problems. In Section 4, we present the classical price decomposition approach in Hilbert spaces and the difficulties encountered when dealing with stochastic optimal control problems. Based on these ingredients, we present in Section 5 a heuristic allowing us to solve subproblems using DP. We finally validate this approach on a simplified power management problem in Section 6.

2. Mathematical framework

Throughout this paper the random variables, defined over a probability space $(\Omega, \mathcal{A}, \mathbb{P})$, will be denoted using bold letters (*e.g.* $\boldsymbol{W} \in L^2(\Omega, \mathcal{A}, \mathbb{P}, \mathbb{W})$) whereas their realizations will be denoted using normal letters (*e.g.* $w \in \mathbb{W}$).

In this paper we consider a finite horizon stochastic optimal control problem, where T denotes the time horizon. Three types of random variables are involved in the problem, namely a state, a control, and a noise. The state $\boldsymbol{X}_t \in L^2(\Omega, \mathcal{A}, \mathbb{P}, \mathbb{R}^n)$ evolves with respect to dynamics depending on the control $\boldsymbol{U}_t \in L^2(\Omega, \mathcal{A}, \mathbb{P}, \mathbb{R}^m)$ and on some exogenous noise $\boldsymbol{\xi}_t \in L^2(\Omega, \mathcal{A}, \mathbb{P}, \mathbb{R}^p)$. Unlike deterministic optimal control problems, in the stochastic case the time "direction" is of particular importance. In order to fulfill the causality principle, the control at a given time step t only depends on the observation of noises prior to t. Moreover, we assume that the observation available at time t consists of all past noises. In order to mathematically represent such an information structure, we denote by $\mathcal{A}_t$ the σ-field generated at time t by past noises $(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t)$, so that the control $\boldsymbol{U}_t$ at time step t has to be measurable with respect to $\mathcal{A}_t$. These last constraints will be called the non-anticipativity constraints.

The global system consists of N units, whose dynamics and cost functions are mutually independent. More precisely, the state $\boldsymbol{X}_t$ (respectively the control $\boldsymbol{U}_t$) of the global system writes $\left(\boldsymbol{X}_t^1, \ldots, \boldsymbol{X}_t^N\right)$ with $\boldsymbol{X}_t^i \in L^2(\Omega, \mathcal{A}, \mathbb{P}, \mathbb{R}^{n_i})$ (respectively $\left(\boldsymbol{U}_t^1, \ldots, \boldsymbol{U}_t^N\right)$ with $\boldsymbol{U}_t^i \in L^2(\Omega, \mathcal{A}, \mathbb{P}, \mathbb{R}^{m_i})$) and $n = \sum_{i=1}^N n_i$ and $m = \sum_{i=1}^N m_i$, so that the global dynamics $\boldsymbol{X}_{t+1} = f_t\left(\boldsymbol{X}_t, \boldsymbol{U}_t, \boldsymbol{\xi}_{t+1}\right)$ can be written independently unit by unit: $\boldsymbol{X}_{t+1}^i = f_t^i\left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i, \boldsymbol{\xi}_{t+1}\right)$, $i = 1, \ldots, N$. In the same way, the global cost $L_t\left(\boldsymbol{X}_t, \boldsymbol{U}_t, \boldsymbol{\xi}_{t+1}\right)$ is equal to the sum of the local unit costs $L_t^i\left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i, \boldsymbol{\xi}_{t+1}\right)$, $i = 1, \ldots, N$. At the end of the time period, each unit i leads to a cost K^i that only depends on the final state $\boldsymbol{X}_T^i$ of the unit.

For now, the global problem can be stated independently unit by unit. The coupling between the units arises from a set of static $\mathbb{R}^d$-valued constraints, the constraint at time step t reading $\sum_{i=1}^N g_t^i\left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i\right) = 0$ (see Remark 2.2 for extensions to more enhanced relations). We suppose that all functions f_t^i, L_t^i and g_t^i are at least Borel measurable.

The initial state $\boldsymbol{X}_0$ is assumed to be known. Denoting $(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_T)$ by $\boldsymbol{X}$ and $(\boldsymbol{U}_0, \ldots, \boldsymbol{U}_{T-1})$ by $\boldsymbol{U}$, the problem we wish to solve is:

$$\min_{\boldsymbol{X}, \boldsymbol{U}} \quad \mathbb{E}\left(\sum_{t=0}^{T-1} \sum_{i=1}^{N} L_t^i\left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i, \boldsymbol{\xi}_{t+1}\right) + \sum_{i=1}^{N} K^i\left(\boldsymbol{X}_T^i\right)\right) \tag{2.1a}$$

$$\text{s.t.} \quad \boldsymbol{X}_{t+1}^i = f_t^i\left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i, \boldsymbol{\xi}_{t+1}\right), \qquad \forall t = 0, \ldots, T-1, \forall i = 1, \ldots, N, \tag{2.1b}$$

$$\boldsymbol{X}_0^i = x^i, \qquad \forall i = 1, \ldots, N, \tag{2.1c}$$

$$\sum_{i=1}^{N} g_t^i\left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i\right) = 0, \qquad \forall t = 0, \ldots, T-1, \tag{2.1d}$$

$$\boldsymbol{U}_t \text{ is } \mathcal{A}_t\text{-measurable}, \qquad \forall t = 0, \dots, T-1, \tag{2.1e}$$

$$\underline{x}_t \leq \boldsymbol{X}_t \leq \overline{x}_t, \qquad \forall t = 1, \dots, T, \tag{2.1f}$$

$$\underline{u}_t \leq \boldsymbol{U}_t \leq \overline{u}_t, \qquad \forall t = 0, \dots, T-1. \tag{2.1g}$$

There are three types of coupling in problem (2.1):

- The first one comes from the state dynamics (2.1b) that induce a temporal coupling, subsystem by subsystem.
- The second one arises from the static constraints (2.1d) that link together all the subsystems at each time step t.
- The third type of coupling comes from the non-anticipativity constraints (2.1e), which link together controls relying on the same noise history. If two realizations of the noise process are identical up to time t, then the same control has to be applied at time t on both realizations.

We ultimately suppose that noises $\boldsymbol{\xi}_t$ are independent (white noise). We are thus in the Markovian case and it is well known that the optimal control, which is a priori a function of all the past noises, only depends on the current state [5].

Remark 2.1 (white noise assumption)**.** If the noises $\boldsymbol{\xi}_t$ are not independent[1] but still have known dynamics, one can always include the necessary noise history in the state to come back to the Markovian case. Unfortunately, this usually leads to a higher state dimension, and hence a higher numerical complexity in the DP framework, as will be explained in Section 3.

Remark 2.2 (coupling constraints involving noises)**.** It is possible to replace the static coupling constraint $\sum_{i=1}^{N} g_t^i \left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i\right) = 0$ by $\sum_{i=1}^{N} g_t^i \left(\boldsymbol{X}_t^i, \boldsymbol{U}_t^i\right) = \boldsymbol{D}_t$, where $\boldsymbol{D}_t$ is a random variable representing for instance a global demand. However, expressions are then harder to write: in the Markovian case, *i.e.* when the $\boldsymbol{D}_t$'s are independent one from another, $\boldsymbol{D}_t$ is observed before choosing the control at time t, so optimal controls must depend on both the state $\boldsymbol{X}_t$ and the noise $\boldsymbol{D}_t$.

3. Stochastic dynamic programming

In order to solve stochastic optimal control problems in the Markovian framework, Bellman proposed in [4] the Dynamic Programming (DP) method. It consists of introducing value functions $V_t : \mathbb{R}^n \to \overline{\mathbb{R}}$ that represent the expected optimal cost when starting from state x at time t. In the case of problem (2.1), it reads:

$$V_t\left(x\right) = \min_{\boldsymbol{X}, \boldsymbol{U}} \mathbb{E}\left(\sum_{s=t}^{T-1} \sum_{i=1}^{N} L_s^i \left(\boldsymbol{X}_s^i, \boldsymbol{U}_s^i, \boldsymbol{\xi}_{s+1}\right) + \sum_{i=1}^{N} K^i \left(\boldsymbol{X}_T^i\right) \middle| \boldsymbol{X}_t = x \right), \tag{3.1}$$

[1] and also the noises $\boldsymbol{D}_t$ introduced in Remark 2.2.

subject to the same constraints as in (2.1) and using the convention that if the optimization problem (3.1) is not feasible, then $V_t(x) = +\infty$. The value functions are usually computed in a recursive manner using the DP equation:

$$V_T(x) = \sum_{i=1}^{N} K^i\left(x^i\right), \tag{3.2a}$$

and, for $t = 1, \dots, T-1$:

$$V_t(x) = \min_{u \in [\underline{u}_t, \overline{u}_t]} \mathbb{E}\left(\sum_{i=1}^{N} L_t^i\left(x^i, u^i, \boldsymbol{\xi}_{t+1}\right) + V_{t+1}\left(f_t\left(x, u, \boldsymbol{\xi}_{t+1}\right)\right)\right), \tag{3.2b}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} g_t^i\left(x^i, u^i\right) = 0. \tag{3.2c}$$

Unlike Stochastic Programming methods, a major advantage of DP is that it provides the control $\boldsymbol{U}_t$ as a feedback function on the state variable $\boldsymbol{X}_t$:

$$\boldsymbol{U}_t = \Phi_t\left(\boldsymbol{X}_t\right).$$

Except on very simple examples, equation (3.2) cannot be solved analytically, and many numerical methods have been proposed. A common practice is to discretize the state space and estimate the expectations using Monte Carlo sampling. Unfortunately, as was mentioned in Section 1, we are facing the curse of dimensionality: the complexity of DP grows exponentially with respect to the state space dimension.

Moreover, equation (3.2) is not decomposable in the sense that it cannot be replaced by the solving of N DP equations depending only on the local state x^i. Indeed, even if V_T is a sum of functions depending on the local state x^i as in (3.2a), this additive property does not hold for the preceding time steps because of the coupling constraint (3.2c). Hence, looking for the value function as a sum of functions depending only on the local state would lead to suboptimal strategies. In other words, the local state of a subsystem is not sufficient to take the optimal local decision; some global information about the system is necessary.

Nonetheless, DP remains a seductive approach for small-scale problems since it provides a way to obtain feedback functions. Based on a decomposition scheme presented in Section 4, we will describe in Section 5 a heuristic approach in which problem (2.1) is decomposed into small-scale subproblems that we solve using DP.

4. Price decomposition

Let us recall some results about the classical Uzawa algorithm [1], which aims at iteratively getting round the static coupling constraint (2.1d). When the cost function is additive, this algorithm is also referred to as the price decomposition approach (see [10] for further details). Let us first introduce the Lagrangian of problem (2.1):

$$\mathcal{L}\left(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda}\right)=\mathbb{E}\left(\sum_{t=0}^{T-1}\sum_{i=1}^{N}\left(L_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i,\boldsymbol{\xi}_{t+1}\right)+\boldsymbol{\lambda}_t^\top g_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i\right)\right)+\sum_{i=1}^{N}K^i\left(\boldsymbol{X}_T^i\right)\right),$$

with $\boldsymbol{\lambda}_t \in L^2\left(\Omega,\mathcal{A},\mathbb{P},\mathbb{R}^d\right)$ the Lagrange multiplier associated to the coupling constraint (2.1d) and $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_0,\dots,\boldsymbol{\lambda}_{T-1})$. When the Lagrangian has a saddle point, we know from classical duality theory in optimization [13] that problem (2.1) is equivalent to:[2]

$$\begin{aligned}
\max_{\boldsymbol{\lambda}}\min_{\boldsymbol{X},\boldsymbol{U}}\quad & \mathcal{L}\left(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda}\right) && \text{(4.1a)}\\
\text{s.t.}\quad & \boldsymbol{X}_{t+1}^i = f_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i,\boldsymbol{\xi}_{t+1}\right), \qquad \forall t=0,\dots,T-1, \forall i=1,\dots,N, && \text{(4.1b)}\\
& \boldsymbol{U}_t \text{ is } \mathcal{A}_t\text{-measurable}, \qquad \forall t=0,\dots,T-1, && \text{(4.1c)}\\
& \underline{x}_t \le \boldsymbol{X}_t \le \overline{x}_t, \qquad \forall t=1,\dots,T, && \text{(4.1d)}\\
& \underline{u}_t \le \boldsymbol{U}_t \le \overline{u}_t, \qquad \forall t=0,\dots,T-1. && \text{(4.1e)}
\end{aligned}$$

Recall that the Lagrange multiplier $\boldsymbol{\lambda}_t$ can be interpreted as the marginal price one should pay for satisfying the coupling constraint (2.1d). Because of the $\mathcal{A}_t$-measurability of the variables involved in this constraint and of the properties of conditional expectation, it is easy to see that we can always choose $\boldsymbol{\lambda}_t$ to be $\mathcal{A}_t$-measurable.

Let us introduce the dual function $\psi\left(\boldsymbol{\lambda}\right) := \min_{\boldsymbol{X},\boldsymbol{U}}\mathcal{L}\left(\boldsymbol{X},\boldsymbol{U},\boldsymbol{\lambda}\right)$ subject to constraints (4.1b)–(4.1e). The key point of the price decomposition algorithm is that computing $\psi\left(\boldsymbol{\lambda}\right)$ is much easier than solving the original problem (2.1). Indeed, one can write:

$$\begin{aligned}
\psi\left(\boldsymbol{\lambda}\right) &= \min_{\boldsymbol{X},\boldsymbol{U}}\mathbb{E}\left(\sum_{t=0}^{T-1}\sum_{i=1}^{N}\left(L_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i,\boldsymbol{\xi}_{t+1}\right)+\boldsymbol{\lambda}_t^\top g_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i\right)\right)+\sum_{i=1}^{N}K^i\left(\boldsymbol{X}_T^i\right)\right),\\
&= \sum_{i=1}^{N}\min_{\boldsymbol{X}^i,\boldsymbol{U}^i}\mathbb{E}\left(\sum_{t=0}^{T-1}\left(L_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i,\boldsymbol{\xi}_{t+1}\right)+\boldsymbol{\lambda}_t^\top g_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i\right)\right)+K^i\left(\boldsymbol{X}_T^i\right)\right),
\end{aligned}$$

so that we replace the solving of an optimization problem with variables $(\boldsymbol{X},\boldsymbol{U})$ by the solving of N subproblems with variables $\left(\boldsymbol{X}^i,\boldsymbol{U}^i\right)$.

[2] That is problems (2.1) and (4.1) have the same optimal value; see Remark 4.1 for further details.

Given $\boldsymbol{\lambda}^k$, an iteration of the price decomposition algorithm first solves the N subproblems:

$$\begin{aligned}
\min_{\boldsymbol{X}^i,\boldsymbol{U}^i} \quad & \mathbb{E}\left(\sum_{t=0}^{T-1}\left(L_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i,\boldsymbol{\xi}_{t+1}\right)+{\boldsymbol{\lambda}_t^k}^\top g_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i\right)\right)+K^i\left(\boldsymbol{X}_T^i\right)\right) && (4.2a)\\
\text{s.t.} \quad & \boldsymbol{X}_{t+1}^i = f_t^i\left(\boldsymbol{X}_t^i,\boldsymbol{U}_t^i,\boldsymbol{\xi}_{t+1}\right), \qquad \forall t=0,\dots,T-1, && (4.2b)\\
& \boldsymbol{U}_t^i \text{ is } \mathcal{A}_t\text{-measurable}, \qquad \forall t=0,\dots,T-1, && (4.2c)\\
& \underline{x}_t^i \le \boldsymbol{X}_t^i \le \overline{x}_t^i, \qquad \forall t=1,\dots,T, && (4.2d)\\
& \underline{u}_t^i \le \boldsymbol{U}_t^i \le \overline{u}_t^i, \qquad \forall t=0,\dots,T-1. && (4.2e)
\end{aligned}$$

The Lagrange multiplier $\boldsymbol{\lambda}^k$ is then updated using a gradient-like algorithm. Under standard assumptions[3], the gradient of ψ is:

$$\nabla_{\boldsymbol{\lambda}_t}\psi\left(\boldsymbol{\lambda}^k\right)=\sum_{i=1}^N g_t^i\left(\boldsymbol{X}_t^{i,k+1},\boldsymbol{U}_t^{i,k+1}\right), \qquad (4.3)$$

where $\boldsymbol{X}_t^{i,k+1}$ and $\boldsymbol{U}_t^{i,k+1}$ are the solutions of problem (4.2).

Remark 4.1 (Uzawa's algorithm convergence)**.** Iterations involving the resolution of all subproblems (4.2) and the update of $\boldsymbol{\lambda}$ using (4.3) exactly correspond to Uzawa's algorithm. Under classical assumptions [13] including smoothness and strong convexity of the objective function, the sequences $\{\boldsymbol{U}_t^{i,k}\}_{k\in\mathbb{N}}$ converge toward the solution of problem (2.1).

At first sight, problem (4.2) looks like a stochastic optimal control problem with control $\boldsymbol{U}_t^i$ and state $\boldsymbol{X}_t^i$, the solution of which would be a local feedback on $\boldsymbol{X}_t^i$. This contradicts the fact that the solution of problem (2.1) is a feedback function on the whole state $\left(\boldsymbol{X}_t^1,\dots,\boldsymbol{X}_t^N\right)$. In order to understand where this contradiction comes from, one has to highlight the role of $\boldsymbol{\lambda}$ in problem (4.2).

5. Dual approximate dynamic programming

Let us take a closer look at problem (4.2). First suppose that $\boldsymbol{\lambda}$ is a white noise process. Then problem (4.2) lies in the Markovian framework with state $\boldsymbol{X}_t^i$ and noise $(\boldsymbol{\xi}_t,\boldsymbol{\lambda}_t)$. The optimal control $\boldsymbol{U}_t^i$ depends only on the local state $\boldsymbol{X}_t^i$ and one can apply stochastic dynamic programming to solve this small-scale optimal control problem. Unfortunately, we do not know anything about the time correlations of the price process $\boldsymbol{\lambda}$.

Let us now consider the general case. Defining $\left(\boldsymbol{X}_t^i,\boldsymbol{\lambda}_1,\dots,\boldsymbol{\lambda}_t\right)$ as the state at time t, problem (4.2) falls in the Markovian setting. In particular, the optimal

[3] See [11] for results on the differentiability of the dual function ψ.

control $\boldsymbol{U}_t^i$ is $\left(\boldsymbol{X}_t^i, \boldsymbol{\lambda}_1, \dots, \boldsymbol{\lambda}_t\right)$-measurable. However DP in this context proves numerically intractable because the state dimension increases with respect to time.

Consider now an intermediate case, and suppose that the dual variable $\boldsymbol{\lambda}$ has a short memory dynamics, for instance that $\boldsymbol{\lambda}_{t+1}$ only depends on $\boldsymbol{\lambda}_t$ and $\boldsymbol{\xi}_{t+1}$:

$$\boldsymbol{\lambda}_{t+1} = h_t\left(\boldsymbol{\lambda}_t, \boldsymbol{\xi}_{t+1}\right). \tag{5.1}$$

Using $\left(\boldsymbol{X}_t^i, \boldsymbol{\lambda}_t\right)$ as the state variable at time t, problem (4.2) falls in the Markovian setting. The state dimension does not increase with respect to time anymore and is hopefully small so that problem (4.2) can be solved using DP.

In a very specific instance of problem (2.1), namely:

$$\begin{aligned}
\min_{\boldsymbol{X},\boldsymbol{U}} \quad & \mathbb{E}\left(\sum_{t=0}^{T-1}\sum_{i=1}^{N}\frac{c_i}{2}\left(\boldsymbol{U}_t^i\right)^2 + \sum_{i=1}^{N}\frac{\gamma_i}{2}\left(\boldsymbol{X}_T^i - \boldsymbol{X}_0^i\right)^2\right) \\
\text{s.t.} \quad & \boldsymbol{X}_{t+1}^i = \boldsymbol{X}_t^i - \boldsymbol{U}_t^i + \boldsymbol{A}_{t+1}^i, \qquad \forall t = 0, \dots, T-1, \\
& \sum_{i=1}^{n}\boldsymbol{U}_t^i = \boldsymbol{D}_t, \qquad \forall t = 0, \dots, T-1, \\
& \boldsymbol{U}_t \text{ is } \mathcal{A}_t\text{-measurable}, \qquad \forall t = 0, \dots, T-1,
\end{aligned} \tag{5.2}$$

with $\boldsymbol{\xi}_t = \left(\boldsymbol{D}_t, \boldsymbol{A}_t^1, \dots, \boldsymbol{A}_t^N\right)$, [21] has brought to light such an intermediate case. Here the dimension of the state $\boldsymbol{X}_t^i$ (respectively of the control $\boldsymbol{U}_t^i$) in the subsystem i is $n_i = 1$ (respectively $m_i = 1$), for $i = 1, \dots, N$. The result is the following.

Proposition 5.1. *If $\left(\boldsymbol{D}_t, \boldsymbol{A}_t^1, \dots, \boldsymbol{A}_t^N\right)_{t=0,\dots,T}$ is a white noise process and if there exists $\alpha \in \mathbb{R}^+$ such that $\gamma_i = \alpha c_i, \forall i = 1, \dots, N$, then the optimal Lagrange multipliers satisfy:*

$$\begin{aligned}
\boldsymbol{\lambda}_{t+1} = \; & \boldsymbol{\lambda}_t + \frac{1}{\sum_{i=1}^{N}\frac{1}{c_i}}\Big(\boldsymbol{D}_{t+1}\left(1+\alpha\right) - \boldsymbol{D}_t - \alpha\mathbb{E}\left(\boldsymbol{D}_{t+1}\right) \\
& -\alpha\left(\boldsymbol{A}_{t+1} - \mathbb{E}\left(\boldsymbol{A}_{t+1}\right)\right)\Big), \\
\boldsymbol{\lambda}_0 = \; & \frac{1}{\sum_{i=1}^{N}\frac{1}{c_i}}\left(\boldsymbol{D}_0\left(1-\alpha\right) - \alpha\sum_{s=1}^{T}\mathbb{E}\left(\boldsymbol{A}_s\right) - \alpha\sum_{s=1}^{T-1}\mathbb{E}\left(\boldsymbol{D}_s\right)\right).
\end{aligned}$$

Using such a dynamics for the multipliers, it is straightforward to show that problem (5.2) splits into N independent optimization subproblems. Taking the state variable as $\left(\boldsymbol{X}_t^i, \boldsymbol{\lambda}_t, \boldsymbol{D}_t\right)$, the i-th subproblem can be solved using DP in dimension 3. In summary, we have replaced one N-dimensional problem by N 3-dimensional problems.

Note that the proportionality assumption on the cost coefficients in proposition 5.1 is rather unnatural. Nevertheless, it shows that, in some cases, there exist dynamics for the Lagrange multipliers that is independent of the decision variables.

To deal with more general cases, we propose to approximate the dual process $\boldsymbol{\lambda}$ by some parameterized short-memory process. That is, we try to identify the multipliers that are the closest to the optimal ones within a constrained subspace of stochastic processes. This approach is similar to that employed in the Approximate Dynamic Programming (ADP) method. Since it concerns dual variables rather than DP value functions, we refer to this approach as Dual Approximate Dynamic Programming (DADP).

The performance of such an approach highly depends on the choice of the subspace of stochastic processes in which we force the multipliers to lie. However, a major difference with ADP techniques is that approximating the dual variables may lead to violations of the coupling constraints. The larger the chosen subspace, the less the coupling constraints will be violated. Moreover, prior information on the problem may be useful to devise a suitable dynamics.

Let us now present the implementation of DADP. We constrain dual variables to satisfy:

$$\boldsymbol{\lambda}_0 = h_{\alpha_0}\left(\boldsymbol{\xi}_0\right), \tag{5.3a}$$

$$\boldsymbol{\lambda}_{t+1} = h_{\alpha_{t+1}}\left(\boldsymbol{\lambda}_t, \boldsymbol{\xi}_{t+1}\right), \qquad \forall t = 0, \dots, T-2. \tag{5.3b}$$

where h_{α_t} is an a priori chosen function parameterized by $\alpha_t \in \mathbb{R}^q$. We denote by $\mathcal{S}$ the set of all random processes that verify equation (5.3) for some real vector $\alpha = (\alpha_0, \dots, \alpha_{T-1})$. Given a vector α^k of coefficients, the first step of DADP is to solve the N subproblems (4.2) with the additional dynamics constraints (5.3). This is performed by DP using the augmented state $\left(\boldsymbol{X}_t^i, \boldsymbol{\lambda}_t\right)$. In order to update the Lagrange multipliers, we draw s trajectory samples of the noise $\boldsymbol{\xi}$ and integrate the dynamics (4.2b) and (5.3) using the optimal feedback laws, thus obtaining s trajectory samples of $\boldsymbol{X}^k$, $\boldsymbol{U}^k$ and $\boldsymbol{\lambda}^k$. We then perform a gradient step on $\boldsymbol{\lambda}$ sample by sample:

$$\boldsymbol{\lambda}_t^{k+\frac{1}{2},\sigma} = \boldsymbol{\lambda}_t^{k,\sigma} + \rho_t \sum_{i=1}^N g_t^i\left(\boldsymbol{X}_t^{i,k,\sigma}, \boldsymbol{U}_t^{i,k,\sigma}\right), \qquad \forall \sigma = 1, \dots, s,$$

with ρ_t obeying the rules of the step-size choice in Uzawa's algorithm [13]. Finally, in order to obtain coefficients α_t for the new iterate, we apply an operator $\mathcal{R}^s$ on the samples $\boldsymbol{\lambda}^{k+\frac{1}{2}}$:

$$\mathcal{R}^s\left(\boldsymbol{\lambda}^{k+\frac{1}{2}}\right) = \underset{\alpha_0,\dots,\alpha_{T-1}}{\arg\min} \sum_{\sigma=1}^s \left(\left\| h_{\alpha_0}\left(\boldsymbol{\xi}_0^\sigma\right) - \boldsymbol{\lambda}_0^{k+\frac{1}{2},\sigma} \right\|_{\mathbb{R}^d}^2 \right.$$
$$\left. + \sum_{t=0}^{T-2} \left\| h_{\alpha_{t+1}}\left(\boldsymbol{\lambda}_t^{k+\frac{1}{2},\sigma}, \boldsymbol{\xi}_{t+1}^\sigma\right) - \boldsymbol{\lambda}_{t+1}^{k+\frac{1}{2},\sigma} \right\|_{\mathbb{R}^d}^2 \right).$$

The last minimization produces coefficients $(\alpha_0^{k+1}, \ldots, \alpha_{T-1}^{k+1})$,[4] which define using equation (5.3) a new process $\boldsymbol{\lambda}^{k+1}$ that, by construction, lies in $\mathcal{S}$.

The procedure we describe is an heuristic application of price decomposition for two main reasons:

- we restrain the Lagrange multipliers to lie in a given set $\mathcal{S}$;
- we build at each iteration a price process that lies in $\mathcal{S}$ but has no reason to be the exact projection of the current iterate on $\mathcal{S}$.

It thus seems hard to give theoretical results about the properties of the solution given by this procedure. Such analysis should require some kind of measure of the last two approximations made on the price dynamics and its impact on the solution given by the algorithm (optimal value and feedbacks). This issue will rather be considered from a numerical point of view and illustrated on the example presented in Section 6.

The heuristic is outlined in Algorithm 1. Note that the termination criterion is rather elementary and inspired by a general one used in gradient methods.

Algorithm 1 Dual Approximate Dynamic Programming

```
Require: $\varepsilon > 0$, $\gamma > 0$, a shape $h_\alpha$ for the prices dynamics, $\alpha^0$.
  repeat
    $k \leftarrow k + 1$
    for $i = 1$ to $N$ do
      Solve $i$-th subproblem by DP using parameters $\alpha^k$ for the price dynamics,
      and obtain $\boldsymbol{X}^{i,k}$ and $\boldsymbol{U}^{i,k}$. Both implicitly depend on $\alpha^k$.
    end for
    Update parameters $\alpha^k$:
      $$\alpha^{k+1} = \mathcal{R}^s \left( \left( \boldsymbol{\lambda}_t^k + \rho_t \sum_{i=1}^{N} g_t^i \left( \boldsymbol{X}_t^{i,k}, \boldsymbol{U}_t^{i,k} \right) \right)_{t=0,\ldots,T-1} \right),$$
    where:
      $$\boldsymbol{\lambda}_{t+1}^k = h_{\alpha_t^k} \left( \boldsymbol{\lambda}_t^k, \boldsymbol{\xi}_{t+1} \right), \qquad \forall t = 0, \ldots, T-1.$$
  until $\left\| \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k \right\| < \varepsilon$
```

[4] Note that this optimization problem naturally splits with respect to the time steps and leads to T independent least-squares problems, the $T-1$ lasts being of the form:

$$\min_{\alpha_t} \sum_{\sigma=1}^{s} \left\| h_{\alpha_t} \left(\boldsymbol{\lambda}_t^{k+\frac{1}{2},\sigma}, \boldsymbol{\xi}_{t+1}^{\sigma} \right) - \boldsymbol{\lambda}_{t+1}^{k+\frac{1}{2},\sigma} \right\|_{\mathbb{R}^d}^2 .$$

Remark 5.1. While the operator $\mathcal{R}^s$ is meant to provide a process $\boldsymbol{\lambda}^{k+1}$ that lies in $\mathcal{S}$, it does not lead to some regression on the set $\mathcal{S}$. Indeed, since $\mathcal{S}$ can be non-convex (even when the operators $(h_{\alpha_t})_{t=0,\dots,T-1}$ are linear), solving a regression problem with respect to this set is in general numerically intractable.

Remark 5.2 (enhancement of $\mathcal{S}$)**.** It may be desirable to consider a larger set $\mathcal{S}$ in order to estimate more accurately the price process. For instance, one can extend relation (5.3) in order to include more memory in the process:

$$\boldsymbol{\lambda}_{t+1} = h_{\alpha_t}\left(\left(\boldsymbol{\lambda}_\tau\right)_{\tau \le t}, \left(\boldsymbol{\xi}_\tau\right)_{\tau \le t+1}\right).$$

However, this will in general increase the numerical complexity of DP in the solving of the subproblems.

Remark 5.3 (another formulation)**.** Alternatively, we could have considered a gradient algorithm that iterates directly on the parameters α of the dynamics (5.3). In this case, since we have no restrictions on α, the feasible set would have been convex. Unfortunately, because the dynamics (5.3) may be nonlinear with respect to α, the dual function ψ introduced in Section 4 might be non-concave with respect to α.

6. Numerical experiments

We tested this approach on a simple quadratic power management problem. On this small-scale example, we are able to compare DADP results to those obtained by DP. Consider a power producer who owns two types of power plants:

- Two hydraulic plants that are characterized at each time step t by their water stock $\boldsymbol{X}_t^i$ and power production $\boldsymbol{U}_t^i$, and receive water inflows $\boldsymbol{\xi}_{t+1}^i$, $i = 1, 2$. These two units are subject to dynamic constraints. Moreover, producing $u \in \mathbb{R}$ with unit i leads to a quadratic cost of $c_i u^2$.
- One thermal unit with a production cost that is quadratic with respect to its production $\boldsymbol{U}_t^3$. There are no dynamics associated with this unit.

Using these plants, the power producer must supply a power demand $\boldsymbol{D}_t$ at each time step t, over a discrete time horizon of $T = 25$ time steps. All noises, *i.e.* the demand $\boldsymbol{D}_t$ and the inflows $\boldsymbol{\xi}_t^1$ and $\boldsymbol{\xi}_t^2$ are chosen to be white noise processes.

In this model the quadratic costs on the hydraulic power productions ensure that, at least in the deterministic framework and without any approximation, the algorithm would build primal iterates that converge to the optimal solution of the original problem. For such power management problems, the production costs are often linear rather than quadratic. However, our aim through this example is to study the performance of the DADP algorithm, and we prefer focusing on a test problem which has adequate convexity properties.

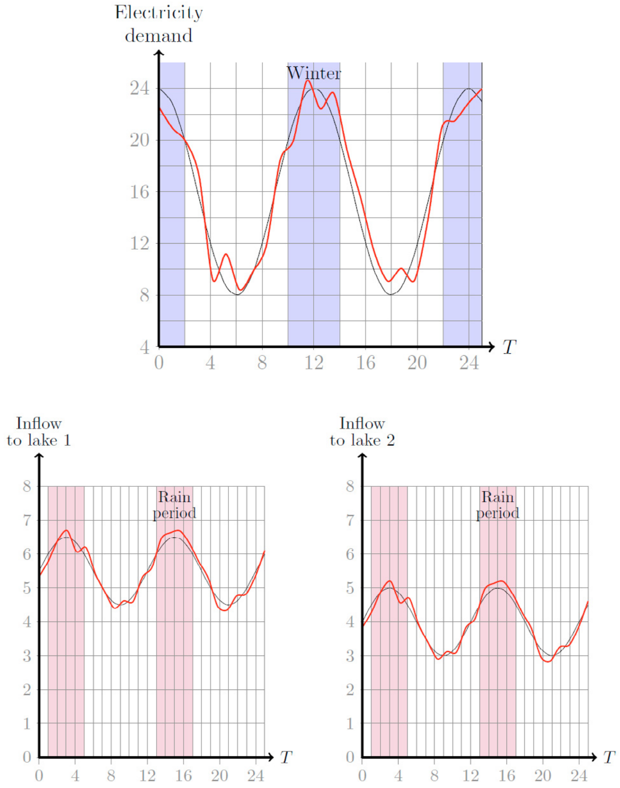

FIGURE 1. Mean of $\boldsymbol{D}_t$, $\boldsymbol{\xi}_t^1$ and $\boldsymbol{\xi}_t^2$ over time (in red/grey is one sample trajectory).(A color version of this figure is available at www.rairo-ro.org.)

The problem reads[5]:

$$\min_{\boldsymbol{X},\boldsymbol{U}} \quad \mathbb{E}\left(\sum_{t=0}^{T-1}\left(c_1\left(\boldsymbol{U}_t^1\right)^2 + c_2\left(\boldsymbol{U}_t^2\right)^2 + L_t\left(\boldsymbol{U}_t^3\right)\right) + K^1\left(\boldsymbol{X}_T^1\right) + K^2\left(\boldsymbol{X}_T^2\right)\right) \tag{6.1a}$$

$$\text{s.t.} \quad \boldsymbol{X}_{t+1}^i = \boldsymbol{X}_t^i - \boldsymbol{U}_t^i + \boldsymbol{\xi}_{t+1}^i, \qquad \forall i = 1,2, \quad \forall t = 0,\dots,T-1, \tag{6.1b}$$

$$\boldsymbol{U}_t^1 + \boldsymbol{U}_t^2 + \boldsymbol{U}_t^3 = \boldsymbol{D}_t, \qquad \forall t = 0,\dots,T-1, \tag{6.1c}$$

$$\underline{x}^i \leq \boldsymbol{X}_t^i \leq \overline{x}^i, \qquad \forall i = 1,2, \quad \forall t = 1,\dots,T, \tag{6.1d}$$

$$0 \leq \boldsymbol{U}_t^i \leq \overline{u}^i, \qquad \forall i = 1,2, \quad \forall t = 0,\dots,T-1, \tag{6.1e}$$

$$0 \leq \boldsymbol{U}_t^3, \qquad \forall t = 0,\dots,T-1, \tag{6.1f}$$

$$\boldsymbol{U}_t^i \text{ is } \sigma\left\{\boldsymbol{D}_0, \boldsymbol{\xi}_0^1, \boldsymbol{\xi}_0^2, \dots, \boldsymbol{D}_t, \boldsymbol{\xi}_t^1, \boldsymbol{\xi}_t^2\right\}\text{-measurable}, \quad \forall i = 1,2,3. \tag{6.1g}$$

[5] In this example, we consider two hydraulic plants with characteristics:

$$\underline{x}^1 = 0, \qquad \overline{x}^1 = 50, \qquad \overline{u}^1 = 6, \qquad K^1(x) = -7x,$$
$$\underline{x}^2 = 0, \qquad \overline{x}^2 = 40, \qquad \overline{u}^2 = 6, \qquad K^2(x) = -12x, \qquad \epsilon = 0.1$$

where $\underline{y}$ (resp. $\overline{y}$) denotes a lower (resp. upper) bound for variable y. Moreover, producing u with the thermal plant costs $L_t(u) = u + u^2$.

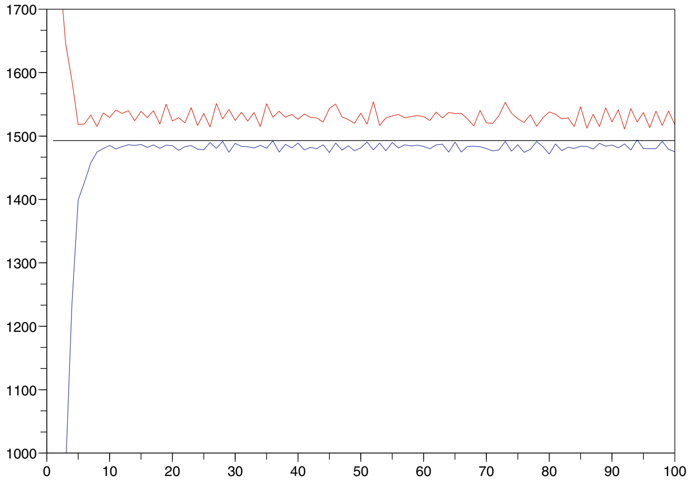

FIGURE 2. Value of the dual function (lower curve), of the primal function (upper curve) and optimum (middle curve) along with iterations. (A color version of this figure is available at www.rairo-ro.org.)

In this problem, the state $\boldsymbol{X}_t$ is two-dimensional, hence DP remains numerically tractable and we can use the DP solution as a reference. In order to use DADP, we choose an auto-regressive process for the Lagrange multipliers:

$$\boldsymbol{\lambda}_0 = \beta_0 \boldsymbol{D}_0 + \gamma_0, \tag{6.2a}$$

$$\boldsymbol{\lambda}_{t+1} = \alpha_t \boldsymbol{\lambda}_t + \beta_t \boldsymbol{D}_{t+1} + \gamma_t, \tag{6.2b}$$

where $(\beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1, \ldots)$ are the design parameters of the price dynamics shape.

We then perform the algorithm and depict its convergence in Figure 2. We first draw the values of the dual function ψ introduced in Section 4 along with iterations (lower curve) and observe that it converges to the optimal value of the original problem computed by DP. Note that each value of ψ is computed by Monte Carlo simulation over 10^3 scenarios. Each value on this curve is a lower bound for the optimal value of the original problem.

We also draw the cost of the problem with all constraints satisfied (primal cost) at each iteration (upper curve). As explained in Section 5, DADP does not ensure that the coupling constraint (6.1c) is satisfied. Fortunately, we here have an easy way to design a feasible strategy out of the strategies obtained by Algorithm 1. This is achieved by choosing the thermal unit strategy so as to ensure feasibility of the coupling constraint:

$$\boldsymbol{U}_t^3 = \boldsymbol{D}_t - \left(\boldsymbol{U}_t^1 + \boldsymbol{U}_t^2\right). \tag{6.3}$$

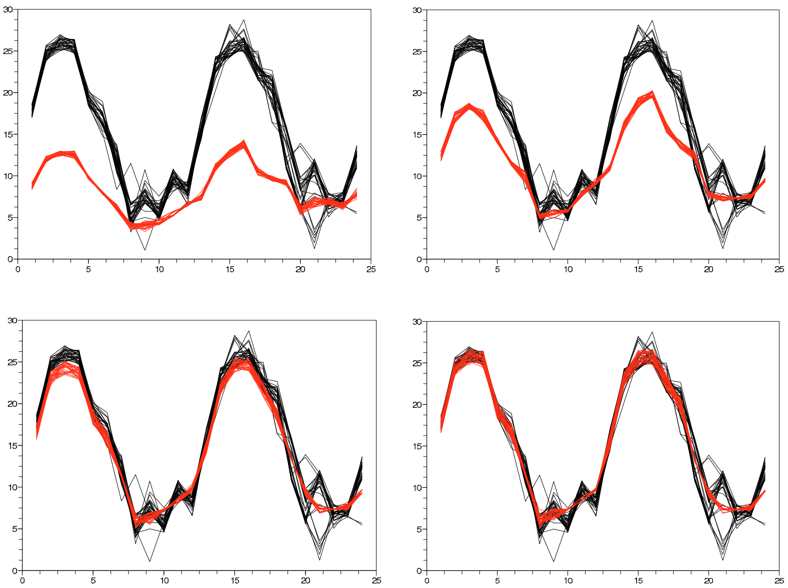

Figure 3. Comparison, using 100 samples, of the approximate prices trajectories (red/grey) with the optimal ones (black), after 10, 20, 50, and 90 iterations of the algorithm (from left to right and top to bottom). (A color version of this figure is available at www.rairo-ro.org.)

That is, whereas DADP returns three strategies (for each of the hydraulic units and for the thermal unit), we only use the first two strategies associated with the hydraulic units and use relation (6.3) for computing the thermal unit strategy during simulations in order to ensure demand satisfaction.

Figure 2 shows that the algorithm behaves well, in the sense that the value of the objective function converges quite quickly to a neighborhood of the optimal value. However, even after 100 iterations the curve is still a bit noisy. This may be due to the employed price dynamics, which generates a non-convex set $\mathcal{S}$ of stochastic processes. Consequently, there is no reason for the procedure that maps $\boldsymbol{\lambda}^{k+\frac{1}{2}}$ to $\boldsymbol{\lambda}^{k+1} \in \mathcal{S}$ to have Lipschitz properties. In other words, small variations in the actual gradient of the dual function and hence on the variable $\boldsymbol{\lambda}^{k+\frac{1}{2}}$ can result in large changes in $\boldsymbol{\lambda}^{k+1}$.

The key to convergence in DADP is to obtain a dynamics for the Lagrange multipliers that accurately matches the optimal one. We have represented in Figure 3 the dynamics of the multipliers computed by DADP after 10, 20, 50 and 90 iterations, and those derived from DP. We observe that the approximate price dynamics issued from DADP satisfactorily converges to the optimal one. This indicates that:

(1) the dual process converged, although the set of stochastic processes defined by (6.2) is non-convex;

(2) there is no need to enhance the chosen dynamics (6.2) for the multipliers in this particular problem.

Note that the optimal prices derived from DP are obtained by numerically differentiating the Bellman functions, hence the numerical instabilities we observe in the lower parts of the DP prices curves.

Remark 6.1 (numerical complexity)**.** We chose to validate the method on a two-dimensional power management problem, so we could compute a reference solution using DP. Note that there would be no additional difficulty in implementing the same algorithm with a larger number of hydraulic units, *i.e.* with a larger-dimensional state. The complexity of DADP grows linearly with the number of subsystems. The most time consuming part of the algorithm is solving the subproblems. However, this calculations can be easily parallelized on a computer, so that the time needed at each DADP iteration remains constant with respect to the number of subsystems.

7. Conclusion

In this paper we present an approach, called Dual Approximate Dynamic Programming (DADP), to solve large-scale stochastic optimal control problems in a price decomposition framework, without discretizing randomness. On the one hand, it provides a lower bound for the value of the original problem. On the other hand, we are able to design feasible strategies for the application we are concerned with. In order to be able to solve subproblems using DP, we suppose that the Lagrange multipliers obey some parameterized dynamics. The DADP algorithm then iterates on the parameters of these dynamics. What is original in this approach is the use of a dual variable in the optimal local feedback functions as an auxiliary variable that sums up the remaining part of the system.

On an example, we show that this approach is very attractive from a numerical point of view. Using rather simple dynamics for the multipliers, we obtained surprisingly good results with a small number of iterations. The main advantage of the method is that the complexity of the algorithm grows linearly with respect to the number of subsystems so that the curse of dimensionality is circumvented for the considered class of problems.

There are still several important theoretical questions. Even in the situation where the knowledge of optimal dual variables gives in a straightforward manner an optimal primal solution (*e.g.* under Lagrangian stability assumption), an additional difficulty arises with our methodology. Since we constrain the dual variables to lie in some a priori chosen subset, we cannot state that the coupling constraints will be satisfied. Hence it would be useful to be able to evaluate the distance between the solution given by the heuristic and the feasible set; this would also give clues on how to choose well-suited dynamics for the dual variables on particular problems. Furthermore, the stochastic process subset on which we constrain the dual variables is possibly non-convex. In this context, it might be valuable to use more enhanced numerical methods for the update of the Lagrange multipliers. Further studies will be concerned with these issues.

References

[1] K.J. Arrow, L. Hurwicz and H. Uzawa, *Studies in linear and nonlinear programming.* Stanford University Press (1958).

[2] Z. Artstein, Sensitivity to σ-fields of information in stochastic allocation. *Stoch. Stoch. Rep.* **36** (1991) 41–63.

[3] R. Bellman and S.E. Dreyfus, Functional approximations and dynamic programming. *Math. Tables Other Aides comput.* **13** (1959) 247–251.

[4] R. Bellman, *Dynamic programming*, Princeton University Press. New Jersey (1957).

[5] D.P. Bertsekas, *Dynamic programming and optimal control*, 2nd edition, Vol. 1 & 2, Athena Scientific (2000).

[6] K. Barty, J.-S. Roy and C. Strugarek, A stochastic gradient type algorithm for closed loop problems. *Math. Program.* (2007).

[7] J. Blomvall and A. Shapiro, Solving multistage asset investment problems by the sample average approximation method. *Math. Program.* **108** (2006) 571–595.

[8] D.P. Bertsekas and J.N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific (1996).

[9] G. Cohen and J.-C. Culioli, Decomposition Coordination Algorithms for Stochastic Optimization. *SIAM J. Control Optim.* **28** (1990) 1372–1403.

[10] G. Cohen, Auxiliary Problem Principle and decomposition of optimization problems. *J. Optim. Theory Appl.* (1980) 277–305.

[11] J.M. Danskin, *The theory of max-min.* Springer, Berlin (1967).

[12] D.P. de Farias and B. Van Roy, The Linear Programming Approach to Approximate Dynamic Programming. *Oper. Res.* **51** (2003) 850–856.

[13] I. Ekeland and R. Temam, *Convex analysis and variational problems.* SIAM, Philadelphia (1999).

[14] P. Girardeau, *A comparison of sample-based Stochastic Optimal Control methods.* E-print available at: `arXiv:1002.1812v1, 2010`.

[15] H. Heitsch, W. Römisch and C. Strugarek, Stability of multistage stochastic programs. *SIAM J. Optim.* **17** (2006) 511–525.

[16] J.L. Higle and S. Sen, *Stochastic decomposition.* Kluwer, Dordrecht (1996).

[17] T. Pennanen, Epi-convergent discretizations of multistage stochastic programs. *Math. Oper. Res.* **30** (2005) 245–256.

[18] A. Prékopa, *Stochastic programming*, Kluwer, Dordrecht (1995).

[19] A. Shapiro, On complexity of multistage stochastic programs. *Oper. Res. Lett.* **34** (2006) 1–8.

[20] A. Shapiro and A. Ruszczynski (Eds.), *Stochastic Programming*, Elsevier, Amsterdam (2003).

[21] C. Strugarek, *Approches variationnelles et autres contributions en optimisation stochastique*, Thèse de doctorat, École Nationale des Ponts et Chaussées, 5 (2006).

[22] A. Turgeon, Optimal operation of multi-reservoir power systems with stochastic inflows. *Water Resour. Res.* **16** (1980) 275–283.

[23] J.N. Tstsiklis and B. Van Roy, Feature-based methods for large scale dynamic programming. *Mach. Lear.* **22** (1996) 59–94.