Approximate Stochastic Dynamic Programming for Hydroelectric Production Planning

Luckny Zéphyr[1,2,*], Pascal Lang[1,*], Bernard F. Lamond[1,*], Pascal Côté[1,3,*]

Abstract

This paper presents a novel approach for approximate stochastic dynamic programming (ASDP) over a continuous state space when the optimization phase has a near-convex structure. The approach entails a simplicial partitioning of the state space. Bounds on the true value function are used to refine the partition. We also provide analytic formulae for the computation of the expectation of the value function in the "uni-basin" case where natural inflows are strongly correlated. The approach is experimented on several configurations of hydro-energy systems. It is also tested against actual industrial data.

Keywords: `Dynamic programming`, curse of dimensionality, dynamic decision process, value function, simplicial state space partitioning.

1. Introduction

This paper presents an approximation scheme for stochastic dynamic programming (ASDP), particularly intended for contexts where the state space is continuous and stage optimization problems are nearly convex (e.g. involving a nearly concave period revenue function over a polyhedral feasible domain). An exemplary problem of this sort pertains to the planning of water releases for energy production in a multi-stage multi-echelon hydro-energy inventory system with random natural water inflows. We will consider a hydrologic network of reservoirs evolving over a finite set of time periods. Each period is characterized by initial states, control variables, possibly random perturbations, which together determine the period's final states. Solving a dynamic program entails computing an optimal "control policy" which at each stage maps initial states to control variables. Since the state space is continuous, the full map is not

*Corresponding author

Email address: `lz395@cornell.edu` (Luckny Zéphyr)

[1]Operations and Decision Systems, Université Laval, Palasis-Prince, 2325, Rue de la Terrasse, Québec, Qc, G1V 0A6, Canada

[2]Present address: Riley Robb Hall, Biological and Environmental Engineering, Cornell University, Ithaca, NY, USA,14853

[3]Rio Tinto Alcan, 1954 Davis, Saguenay (Québec) Canada, G7S 4R5

usually computable, requiring some finite discretization of the state space in each period.

Discretization of the state space is usually achieved via two complementary devices: (i) selecting a finite grid within the state space, and (ii) extending the value function assessed at grid points to the state space continuum. A traditional practice for (i) used to be coordinatewise independent (rectangular) grids, entailing an exponential complexity with respect to the number of state variables. An early proposal to mitigate this complexity consisted in Monte-Carlo sampling [1], enjoying reasonable asymptotic properties. However, the distribution of points in a Monte-Carlo sample usually entails some irregularities (voids and near-redundancies). Hence a recent strand of study has dealt with regularized or quasi-random sampling, seeking a more uniform covering of the state space. This involves techniques such as orthogonal arrays [2] or low-discrepancy sampling [3] among others.

From a finite grid evaluation, one seeks to extrapolate a full continuous value function. This "training" step resorts to an extrapolation model, i.e. a class of "fitting" functions. A member of this class is selected so as to minimize the "gap" between the model's evaluation and the presumed true value of the return function at grid points. Fitting models include Chebyshev polynomials [4], neural networks [5–7], splines function[6–8], kernels [9, 10] among others. Local (rather than global) estimation [3, 9–11] yet adds to the variety of available methods. The choice of a fitting model is of course crucial for the quality of the approximation. As often in statistical learning, a trade-off should be struck between empirical error at grid points and generalization error (risk of overfitting) over the value function's domain.

Thus various combinations of sampling-extrapolation strategies have been proposed in the "Value Function Approximation" literature [e.g. 2, 3, 6, 7, 9–15]. However, this literature is mainly concerned with fast computation of approximate value functions, sometimes giving short shrift to the optimization step of dynamic programming. The choice of a computationally convenient (e.g. smooth) fitting model is a potential source of error. So are overly simplistic representations of the optimization step. Such errors are cumulative over time periods, making it difficult to assess the real accuracy of a proposed approximation. The question of assessing the quality of a solution remains largely open [16, 17].

The purpose of our method is to permit a user-chosen balance between computational burden and accuracy. The proposed ASDP scheme has the following features: (i) the sample state grid is not given in advance but constructed on-line on the basis of optimality gap estimation; (ii) in the case of revenue maximization, the extension model is a minimal concavification operator, providing the best piecewise linear concave fit to grid point evaluations; differentiability is not assumed nor necessary; (iii) for each grid point, the optimization problem is cast in the form of an easily computable generalized linear program; and (iv) under progressive error–reducing densifications of grids, the value function converges to its concave envelope.

The proposed method is suited for problems with nearly concave determinis-

tic period revenue functions, a polyhedral optimization domain, and stochasticity in the linear constraints' right-hand-sides. This class includes multi-echelon problems with stocks and flows embedded in a network structure, with possibly stochastic inflows and outflows. We are more specifically concerned with the particular case of mid-term operations planning in an arborescent hydraulic energy production system. Section 2 reviews the current literature in reservoir management. Section 3 presents a theoretical optimization model in continuous state space. Section 4 presents the foundations of our method. Results of numerical experiments are reported in section 5. Section 6 presents the application of the method in the industrial context of a reservoir system at Rio Tinto (RT). A conclusion follows in section 7.

2. Models and methods for tactical reservoir management

The management of reservoir systems involves decisions on water utilization, i.e. the volume of water to be released for immediate production of energy and the volume to be stored for future usage. Such decisions are made under stochastic inflows and possibly stochastic energy prices, in order to maximize an expected revenue or minimize expected costs over some longer time horizon. A major difficulty stems from uncertainty about the future natural inflows process. Empirical studies often show wide disparities between inflow time-profiles over different years. A wide variety of stochastic optimization methods has been used in order to cope with such uncertainty. A major distinguishing feature among such methods is the number of stages, a "stage" being a decision epoch based on new information, leading to decision rules (a stage may encompass several time periods). Chance constrained programs [e.g. 18–21] are examples of one-stage models. Two-stage stochastic programming [e.g. 22–24] has long been a customary format.

Two and multi-stage stochastic programs are inherently complex due to the size of the inflows process. The uncertain parameters' distribution is usually approximated by a finite discrete distribution; the corresponding multi-period sample space is a "scenario tree" where each scenario is a time trajectory of realizations. Under this representation, a two- or multi-stage problem can be formulated as a "deterministic equivalent" mathematical program (with due non-anticipativity contraints). Thus [25, 26] formulate deterministic scenario-tree-based linear programs to be used on a rolling horizon basis. The very large size of such deterministic programs usually calls for some form of decomposition. Several decomposition strategies are available, such as cutting plane methods (e.g. nested decomposition [27], and operator splitting [28–31]). Stochastic dynamic programming (SDP) is a particular decomposition technique, where the multi-period problem is temporally decomposed into smaller subproblems in a coordinated way.

Scenario sampling is often resorted to in order to alleviate the computational burden [32–36]. Scenarios are usually based on historical records. In Ensemble Streamflow Prediction (ESP), the sample consists of representative

realized multi-period trajectories (scenarios), to serve for recourses in a two-stage program. Other sampling methods seek to replicate a scenario tree. The construction of a proper sample scenario tree is in itself a daunting question [37]. Sample sufficiency, confidence bands on the optimal value, and stopping criteria are also important concerns [16, 17, 38, 39].

One notable specificity of reservoir management is in the modeling of stochastic inflows. These may be subject to seasonality, persistence, and yearly intensity (e.g. due to variable snow covers). Studies [26, 40] indicate that in some (but not all) contexts the addition of hydrological variables improves the overall performance of optimization methods. Hydrographic dynamics may be modeled by time-series analysis of historical data. However available data often does not statistically support more than very elementary time-series models. In a more informal way, ESP models may help capture important hydrodynamics.

A second specificity pertains to the productivity of hydro-electric power plants. As a function of water flows, electricity generation may vary nonlinearly with the number of power generating groups in action, and with the height of the water head. These factors depend on the current state of the energy system and may have a significant influence on the power generation function [e.g. 41]. In theory, SDP provides a flexible, and accurate representation of multi-stage decision. However its application is plagued by the curse of dimensionality. Ad hoc state aggregation schemes have been proposed to overcome this difficulty [42]. More promising avenues may reside in the construction of grid points in state space. For instance, in the stochastic dual dynamic programming method (akin to Benders decomposition, see [32] and [43]), the state space is not discretized, but rather sampled. In order to motivate our approach, we now present a theoretical continuous state-space prototype SDP model.

3. Structure and control of a reservoir system

A hydroelectric system is generally composed of several reservoirs and power plants, each of which may be associated with a reservoir or be run-of-the river. The reservoirs' connections usually form an arborescence.

The management of reservoir systems is challenging, in large part, because of uncertain natural inflows. Operational decisions are concerned with water releases for energy production and spillage in case of excess water. We may assume that planned release decisions are made before observing natural inflows. The available information for these decisions then are the reservoirs' initial levels and possibly an observed previous hydrological variable that may be used to forecast future natural inflows. By contrast, spillage occurs after the realization of the natural inflows process. Therefore, spilled water is conditioned on both releases, and natural inflows.

To produce energy, water is sent through turbines that have finite mechanical capacities. Thus, bounds are imposed on the release of water. Reservoir systems also have finite capacities and are often used for other purposes such as irrigation, flood control, and recreational activities. As a consequence, admissible water stocks are restricted to some lower and upper bounds.

A multi-reservoir system may be modeled by a graph where each node represents a reservoir alone, or a run-of-the-river power plant, or a reservoir-power plant group. The nodes may be numbered from 1 to p in a topological order. Figure 1 shows an example with 6 nodes. Triangles represent reservoirs, rectangles represent power plants, full arcs represent planned releases or spillage, and dashed arcs represent natural inflows.

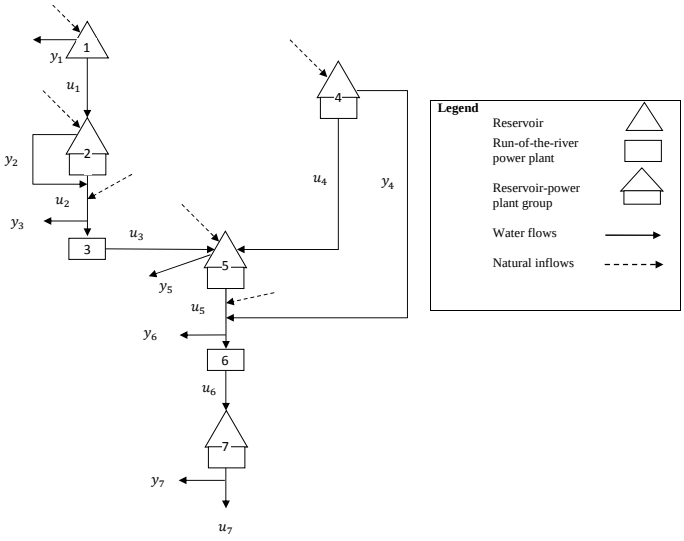


Figure 1: Example of a multi-reservoir system

We distinguish node types via the following sets of indices: (i) I_F: the set of nodes with power plant alone (run-of-river nodes); (ii) I_R: the set of nodes with a reservoir (alone or reservoir-power plant group), with $|I_R| = n$; (iii) I_C: the set of nodes with power plant, with $|I_C| = m$ and (iv) $I_{RC} = I_R \cap I_C$: the set of nodes with reservoir and power plant. A notation nomenclature is provided in Appendix B.

This graph comprises two networks corresponding to planned releases and spillage. The topology of these networks is described via the following $p \times p$ incidence matrices:

$$B_{ij} = \begin{cases} 1 & \text{if } i = j \\ -1 & \text{if the planned releases leaving node } j \text{ arrive at node } i \\ 0 & \text{otherwise} \end{cases}$$

and

$$C_{ij} = \begin{cases} 1 & \text{if } i = j \\ -1 & \text{if the water spilled at node } j \text{ arrives at node } i \\ 0 & \text{otherwise} \end{cases}$$

Optimization models are often used to determine releases and spillage at each node over a planning horizon of T periods, typically a year with weekly

intervals or a month with daily time steps. For each period t, let $u_{it},\ i \in I_C$, denote releases (in hm^3) and $y_{it},\ i = 1, \ldots, p$, spillage (in hm^3).

The state of the physical system is described by the reservoir levels at the end of period t, noted s_t. We note respectively $\underline{s}_t$ and $\bar{s}_t$ lower and upper bounds on such levels in period t. Similarly, $\underline{u}_t$ and $\bar{u}_t$ denote lower and upper bounds on the planned releases in each period t.

The natural inflows to the network's nodes in period t, noted Q_t, are modeled by a non-stationary process that depends stochastically on a scalar hydrological variable $\tilde{\epsilon}_t$, which forms an autoregressive stochastic process of order 1. We then have:

$$\tilde{\epsilon}_t = \alpha \tilde{\epsilon}_{t-1} + a_t, \tag{1}$$

where a_t is a white noise process (a sequence of unbiased independent and identically distributed centered random variables). Such hydrological variables are often used to model persistence phenomena in the natural inflows process [e.g. 35, 44, 45].

Thus, in each period t, the state of the system evolves according to the standard water balance equation:

$$s_t = s_{t-1} - B_{I_R} u_t - C_{I_R} y_t + Q_{I_R t}, \tag{2}$$

and the water conservation equation:

$$Q_{I_F t} - B_{I_F} u_t - C_{I_F} y_t = 0, \tag{3}$$

where, for any matrix $A \in \mathbf{R}^{p \times r}$, any vector $a \in \mathbf{R}^p$ and any subset $I \subset \{1, \ldots, p\}$, A_I (a_I) denotes the submatrix (the subvector) obtained by selecting the lines of A (the elements of a) indexed in I.

Eq.(3) reflects the fact that the water entering a run-of-the-river node in period t leaves in the same period since it is not stored.

Let $x_t = s_{t-1} - B_{I_R} u_t$ be the reservoir levels after the releases u_t and before the realization of the $\{Q_t\}$ process. For any node $i \in I_{RC}$, $\bar{f}_{it}(s_{i,t-1}, x_{it} + Q_t, \tilde{\epsilon}_{t-1})$ denotes its production function in period t. Head effects are taken into account through the reservoir levels at the beginning and the end of each period, and water release during the same period. For any node $i \in I_F$, energy production is a function of u_{it} alone since there is no stock variation; $f_{it}(u_{it})$ denotes such a function.

Optimization models generally seek to maximize the expected energy production or to minimize the expected production cost over the entire planning

horizon. This optimization can be stated as:

$$\underset{u_t,y_t}{\text{Max}} \sum_{t=1}^{T} E_{Q_t|\tilde{\epsilon}_{t-1}} \left[\sum_{i\in I_{RC}} \bar{f}_{it}(s_{i,t-1}, x_{it} + Q_t, \tilde{\epsilon}_{t-1}) + \sum_{i\in I_F} f_{it}(u_{it}) \right] \tag{4}$$

Subject to, for $1 \leq t \leq T$:

$$s_t = s_{t-1} - B_{I_R} u_t - C_{I_R} y_t + Q_{I_R t} \tag{5}$$

$$Q_{I_F t} - B_{I_F} u_t - C_{I_F} y_t = 0 \tag{6}$$

$$\underline{s}_t \leq s_t \leq \bar{s}_t \tag{7}$$

$$\underline{u}_t \leq u_t \leq \bar{u}_t \tag{8}$$

$$y_t \geq 0 \tag{9}$$

$$u_t \text{ conditioned on } s_{t-1} \text{ and } \epsilon_{t-1} \tag{10}$$

$$y_t \text{ and } s_t \text{ conditioned on } s_{t-1}, u_t \text{ and } Q_t \tag{11}$$

Let $f_{it}(u_{it}, s_{i,t-1}, \epsilon_{i,t-1}) = E_{Q_t|\tilde{\epsilon}_{t-1}} \left[\bar{f}_{it}(s_{i,t-1}, s_{i,t-1} - B_{IR} u_t + Q_t, \tilde{\epsilon}_{i,t-1}) \right]$, $\forall\, i \in I_{RC}$. Within the framework of SDP, we note $V_t(s_t, \epsilon_t)$ the value of available water at the end of period t under the information state (s_t, ϵ_t). Under the previous assumptions, for $t = T, T-1, \ldots, 1$, an SDP recursion reads:

$$\begin{aligned} V_{t-1}(s_{t-1}, \epsilon_{t-1}) = \underset{u_t}{\text{Max}} \Bigg\{ & \sum_{i\in I_{RC}} f_{it}(u_{it}, s_{i,t-1}, \epsilon_{i,t-1}) + \sum_{i\in I_F} f_{it}(u_{it}) + \\ & E_{Q_t, \tilde{\epsilon}_t|\epsilon_{t-1}} \left[\underset{s_t, y_t}{\text{Max}} V_t(s_t, \tilde{\epsilon}_t) \right] \Bigg\} \\ & \text{S.t. } (5-11) \end{aligned}$$

For a given period t, this optimization problem involves two decision stages. The "*ex post*" recourse stage aims at computing an optimal overflows spillage policy y_t^* as a function of planned stocks x_t and observed inflows Q_t. The "*ex ante*" stage seeks an optimal planned water release policy u_t^* as a function of initial stocks s_{t-1} and the observed hydrological variable ϵ_{t-1}.

In the ex post stage, given reservoir levels x_t and natural inflows Q_t, spillage decisions are made to maximize the value (in energy units) of remaining water. The ex post optimal decisions then are solutions to the following problem:

$$F_t(x_t, Q_t, \epsilon_t) = \underset{s_t, y_t}{\text{Max}} V_t(s_t, \tilde{\epsilon}_t) \tag{12}$$

$$\text{S.t. } s_t = x_t - C_{I_R} y_t + Q_{I_R t} \tag{13}$$

$$Q_{I_F t} - B_{I_F} u_t - C_{I_F} y_t = 0 \tag{14}$$

$$\underline{s}_t \leq s_t \leq \bar{s}_t \tag{15}$$

$$y_t \geq 0 \tag{16}$$

In the ex ante stage, reservoir levels s_{t-1} at the beginning of period t and the previous hydrological variable $\tilde{\epsilon}_{t-1}$ are observed. Release decisions are made

to maximize immediate and future energy production:

$$V_{t-1}(s_{t-1},\epsilon_{t-1}) = \underset{u_t,x_t}{\text{Max}}\Big\{ \sum_{i\in I_{RC}} f_{it}(u_{it}, s_{i,t-1}, \epsilon_{i,t-1}) + \sum_{i\in I_F} f_{it}(u_{it}) + \tag{17}$$

$$E_{Q_t,\tilde{\epsilon}_t|\epsilon_{t-1}}\left[F_t(x_t, Q_t, \tilde{\epsilon}_t)\right]\Big\} \tag{18}$$

$$\text{S.t. } x_t = s_{t-1} - B_{I_R} u_t \tag{19}$$

$$B_{I_F} u_t = 0 \tag{20}$$

$$\underline{s}_t \le s_t \le \bar{s}_t \tag{21}$$

$$\underline{u}_t \le u_t \le \bar{u}_t \tag{22}$$

The domain of the optimal policy of the ex post step is very large. It is nonetheless possible to formulate as a simple rule an optimal spillage policy consistent with the notion of increasing value of water.

Proposition 1. *If all nodes are equipped with an unlimited spillage system, function $V_{t-1}(s_{t-1},\epsilon_{t-1})$ is non-decreasing in s_{t-1}.*

Given that V_{t-1} is non-decreasing, the optimal spillage decisions for (12)-(16) can be found via the following simple rule [46]:

Proposition 2. *Let $Pred(i)$ be the set of predecessors of node i (the set of nodes j such that $C_{ij} = -1, j \in \{1,\dots,p\}$) in the spillage network. Under the assumptions of proposition 1, if $x_t \ge \underline{s}_t$, there exists a unique Pareto-minimal spillage policy given recursively by:*

$$y^*_{it}(x_t + Q_t) = \max\{0, x_{it} + Q_{it} + \sum_{j\in Pred(i)} y^*_{jt}(x_t + Q_t) - \bar{s}_{it}\}, \tag{23}$$

with the conventions $x_{it} = 0\ \forall i \in I_F$ and $\sum_{j\in Pred(i)} y^*_{jt}(x_t + Q_t) = 0$ if $Pred(i) = \emptyset$. This policy is optimal for the ex post spillage problem. The resulting final stock is:

$$s^*_{it}(x_t + Q_t) = \min\{\bar{s}_{it},\quad x_{it} + Q_{it} + \sum_{j\in Pred(i)} y^*_{jt}(x_t + Q_t)\} \tag{24}$$

Note that the optimal final stock satisfies the conditions $s^*_t(x_t + Q_t) \ge \underline{s}_t$. In case of water overflows (spillage), the upper bounds on the stock levels are saturated, i.e. $(\bar{s}_t - s^*_t)^T y^*_t = 0$. This simple spillage policy (Eq.(23)) states that starting with the upstream nodes, we only spill the excess water in each reservoir.

Under this spillage policy, the stochastic dynamic program simplifies to:

$$V_{t-1}(s_{t-1},\epsilon_{t-1}) = \underset{u_t,x_t}{\text{Max}}\Big\{ \sum_{i\in I_{RC}} f_{it}(u_{it}, s_{i,t-1}, \epsilon_{i,t-1}) + \sum_{i\in I_F} f_{it}(u_{it}) +$$

$$E_{Q_t,\tilde{\epsilon}_t|\epsilon_{t-1}}\left[V_t\left(s^*_t\left(x_t + Q_{I_R t}\right), \tilde{\epsilon}_t\right)\right]\Big\} \tag{25}$$

$$\text{S.t. } (19) - (22) \tag{26}$$

A distinguishing feature of hydro-energy systems is the possibly variable productivity of power plants. If the production functions are concave, the problem is convex and concavity of the value function propagates backwards:

Proposition 3. *If (i) $V_t(s_t, \epsilon_t)$ is concave in s_t, (ii) f_{it} is jointly concave in (s_{it}, u_{it}), $\forall i \in I_{RC}$, and (iii) f_{it} is concave in u_{it}, $\forall i \in I_F$, then V_{t-1} is concave in s_{t-1}.*

Nonconcavity of production functions may have several causes. On one hand, energy production is a function of water releases as well as of head effects, namely the difference between the upstream and downstream water levels, and of friction loss in penstock. Managers are often concerned with maintaining a high head in order to maximize efficiency, yet tailrace (channel that carries water away from the plant) effect may negatively affect power generation (net head reduction) as water release increases. However, head effects may be difficult to capture when forebay (the portion of a reservoir immediatly upstream the dam) and tailrace levels significantly depend on the rate of water inflow or release. The relationship between release rates and head size may then require careful attention [47].

A second reason is that the power delivered by a turbine varies nonlinearly with the flow rate. A minimum rate is needed for setting the turbine into motion. The turbine achieves its maximum efficiency at a given flow rate. Beyond an upper threshold, turbulence may actually decrease the turbine's efficiency. Of course, the total energy produced depends on the number of turbines in operation. Figure 2 depicts a hypothetical energy production function together with a concave piecewise affine approximation. Such an approximation is particularly acceptable if the operational policies derived from a mid term model serve as inputs to a short term model aimed at deriving daily or hourly operational policies. This is the case, for instance, at RT, our case study. It has been advocated, [e.g. 41], that the concave hull of the production function would be an adequate approximation.

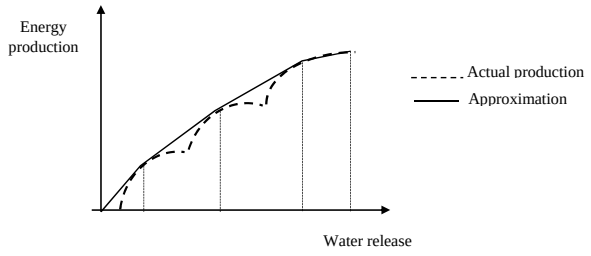


Figure 2: Example of a plant's energy production function and its approximation

4. Approximate stochastic dynamic programming

Our proposed approximation is in three parts. The optimization step is discussed in subsection 4.1. Subsection 4.2 deals with the approximation of the

value function by a finite grid. This leads in subsection 4.3 to a possible method for computing, without sampling, the expectation of the value function when inflows are strongly correlated.

4.1. Approximate optimization via generalized linear programming

We now turn to the solution of problem (25)-(26). Its feasible domain, a convex polyhedron, is partly generated by inequalities. The nonlinear functions in the objective are not in general differentiable. This context suggests that inner generalized linear programming (GLP, see [48]) is an appropriate method. Inner GLP performs interpolations of functions over a discrete sample of points. If the original problem is convex, well-conceived densification of the sample will ultimately converge to the original problem. In nonconvex cases (e.g. here if the production functions are not concave), GLP performs a convex approximation (a "convexification") of the original problem. See [49, pp. 694-699] for further details on sample points generation.

To implement the GLP, we shall need (i) a grid of states $\{\hat{x}_{jt}|j \in \Upsilon_t\}$ over which function $\hat{J}_t(x_t, \epsilon_{t-1}) =$
$E_{Q_t,\tilde{\epsilon}_t|\epsilon_{t-1}}\left[\hat{V}_t\left(s_t^*(x_t + Q_{IRt})\right)\right]$ has been evaluated and, (ii) for each power plant i, a sample of points $\{\hat{u}_{ijt}|j \in \Gamma_{it}\}$ where the production function is assessed. Furthermore, we distinguish power plants associated with a reservoir ($i \in I_{RC}$) from run-of-the river ones ($i \in I_F$). Given these fixed samples, the GLP seeks a "best" interpolation :

$$\hat{V}_{t-1}(s_{t-1},\epsilon_{t-1}) = \underset{u_t,x_t,\lambda,\mu}{\text{Max}} \Bigg\{ \sum_{i\in I_{RC}} \sum_{j\in\Gamma^1_{it}} f_{it}(\hat{u}^1_{ijt}, s_{i,t-1})\lambda^1_{ij} + \sum_{i\in I_F} \sum_{j\in\Gamma^2_{it}} f_{it}(\hat{u}^2_{ijt})\lambda^2_{ij} + \sum_{j\in\Upsilon_t} \hat{J}_t(\hat{x}_{jt},\epsilon_{t-1})\mu_j \Bigg\} \quad (27)$$

S.t: for all $1 \le t \le T$ (28)

$$x_t = s_{t-1} - B_{I_R} u_t \quad (29)$$

$$B_{I_F} u_t = 0 \quad (30)$$

$$x_t = \sum_{j\in\kappa_t} \hat{x}_{jt}\mu_j \quad (31)$$

$$\sum_{j\in\kappa_t} \mu_j = 1 \quad (32)$$

$$\underline{s}_t \le s_t \le \bar{s}_t \quad (33)$$

$$\underline{u}_t \le u_t \le \bar{u}_t \quad (34)$$

$$u_{it} = \sum_{j\in J^1_{it}} \hat{u}^1_{ijt}\lambda^1_{ij} \qquad i \in I_{RC} \quad (35)$$

$$\sum_{j\in J^1_{it}} \lambda^1_{ij} = 1 \qquad i \in I_{RC} \quad (36)$$

$$u_{it} = \sum_{j\in J^2_{it}} \hat{u}^2_{ijt}\lambda^2_{ij} \qquad i \in I_F \quad (37)$$

$$\sum_{j\in J^2_{it}} \lambda^2_{ij} = 1 \qquad i \in I_F \quad (38)$$

$$\lambda, \mu \ge 0 \quad (39)$$

where $u_t = (u^1_t, u^2_t)^T$, $\hat{u}_t = (\hat{u}^1_t, \hat{u}^2_t)^T$ and $\lambda = (\lambda^1, \lambda^2)^T$.

The following simple property will be critical to refining the state space grid, as later presented.

Proposition 4. *$\hat{V}_{t-1}$ is concave in $s_{t-1}, \forall\ \epsilon_{t-1}$.*

In general, it cannot be predicted on which side the GLP approximation will err. However, as a basis for comparison, we have a definite answer in the convex case.

Proposition 5. *If the production functions f_i are concave in u_{it}, and if $\hat{V}_t \le V_t$, then GLP (27)-(39) underestimates the actual energy production in period t, and $\hat{V}_{t-1} \le V_{t-1}$.*

Finally, let us note that the optimal policy can be extended to a continuum by interpolation, since the feasible domain (28)-(39) is convex.

4.2. Approximation of the value function

The value function in period t, $\hat{V}_t$, will be computed by GLP over a finite grid of points in the state space. Elsewhere, it will be interpolated. The state grid will be progressively constructed as follows.

Note that instead of the hyperrectangle $[\underline{s}_t, \overline{s}_t]$, the stocks can be allowed to be confined to a more general polytope $\mathbb{P}_t$. Let Σ_t^o be an initial simplex containing $\mathbb{P}_t$. Such a simplex exists since $\mathbb{P}_t$ is bounded. For instance, in the case of a hyperrectangle $\mathbb{P}_t = [a_t, b_t]$, a suitable choice could be $\Sigma_t^o = \left\{s_t | s_t \geq a_t, e^T s_t \leq e^T b_t\right\}$, where $e^T = (1, 1, \ldots, 1)$. We will progressively construct finer partitions of Σ_t^o into simplices. The vertices of these simplices in $\mathbb{P}_t$ will be our grid points.

4.2.1. Operations on simplices

A simplex in $\mathbf{R}^n$ is the convex envelope of $n+1$ affinely independent points. Thus an $n \times (n+1)$ dimensional matrix S with affinely independent columns generates the simplex:

$$\Sigma(S) \triangleq \left\{ s \in \mathbf{R}^n \middle| \begin{pmatrix} s \\ 1 \end{pmatrix} = \begin{pmatrix} S \\ e^T \end{pmatrix} \lambda, \lambda \geq 0 \right\}.$$

Since S has affinely independent columns, $\begin{pmatrix} S \\ e^T \end{pmatrix}$ is of full rank. Matrix S will be called a simplex generator.

We present an overview of common operations on simplices, namely (i) testing whether a point belongs to a simplex, (ii) dividing a simplex, (iii) finding a smallest simplex containing a given point, and (iv) moving across simplices.

(i) *Membership test.* Given a matrix generator S and a vector $s \in \mathbf{R}^n$, check whether $s \in \Sigma(S)$. The answer is yes if and only if $\begin{pmatrix} S \\ e^T \end{pmatrix}^{-1} \begin{pmatrix} s \\ 1 \end{pmatrix} \geq 0$. This test requires a matrix inversion, hence at most $\mathcal{O}(n^3)$ elementary algebraic operations.

(ii) *Simplex division.* Grid points will be numbered in order of their creation. Assume a "parent" (n-dimensional) simplex $\Sigma(S)$ is to divided into k (n-dimensional) "children" simplices, where $2 \leq k \leq n+1$. This division will be organized around a *division point* $s^* \in \Sigma(S)$, which will eventually be incorporated into the grid. The number of children will depend on the location of the division point in the parent simplex (the choice of the division point will be discussed in subsection 4.2.2). We assume away the trivial case where the division point is at a vertex of the parent simplex. The division point is then located on the relative interior of some face F, which will be the minimal simplex within $\Sigma(S)$ containing s^*. This face can be identified by its vertices, which are the columns of S indexed in I, where $I = \left\{ i \middle| \left[\begin{pmatrix} S \\ e^T \end{pmatrix}^{-1} \right]_i \begin{pmatrix} s \\ 1 \end{pmatrix} > 0 \right\}$. It is readily verified that $dim\ F = |I| - 1$. The parent simplex will have $|I|$ children. Each

child will be generated by the matrix S_i' obtained by replacing column $i \in I$ by s^*. It can be verified that S_i' has affinely independent columns.

As an instance, if s^* is in the interior of $\Sigma(S)$, then $F = \Sigma(S)$, and the latter has $n+1$ children (e.g. $[s^*, v, w]$, $[u, s^*, w]$, and $[u, v, s^*]$ in Figure 3 (a)). If F is an edge, two children (e.g. $[u', s^*, v']$, and $[s^*, v', w']$ in Figure 3 (b)) are generated. In all cases, identifying the children of a simplex requires a matrix inversion entailing $\mathcal{O}(n^3)$ operations.

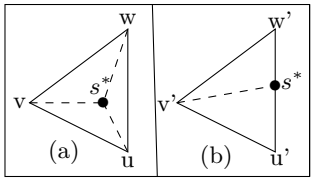


Figure 3: Illustration of the division of simplices

Since some simplices are divided into smaller simplices, the overall partition of the state space has a nested structure, which can be represented as an arborescence. The nodes of this arborescence are simplices. Its arcs represent division. A simplex which is not divided (a leaf in the arborescence) will be called *active*. The set of active simplices forms the finest available partition of the state space, the one of interest to us. We nonetheless keep track of the arborescence to be able to locate a point efficiently. To that end, with each simplex in the arborescence we associate the list of its $n+1$ vertices.

Each simplex division generates one additional grid point (i.e. the division point), and k children, where $2 \leq k \leq n+1$. The parent simplex is then removed from the active list. Thus if κ is the current number of grid points and N the current number of simplices, $\kappa \leq N \leq n\kappa$.

We call *neighbor* of a simplex $\Sigma(S)$ a simplex $\Sigma(S')$ sharing n vertices with $\Sigma(S)$. We keep track of the neighbors of each active simplex. The children of a simplex are neighbors to each other. Each child may also have one neighbor inherited from its parent. Identifying the latter involves, for each parent's neighbor, comparing two lists of n elements. In total, updating the lists of all children's neighbors involves at most $\mathcal{O}(n^4)$ operations.

(iii) *Simplex traversal.* Starting from a point $s \in \Sigma(S)$, our task is to move as far as possible in a given direction $\delta \in \mathbf{R}_+^n \backslash \{0\}$ without leaving the simplex. We say that simplex $\Sigma(S)$ *properly contains* the point s relatively to the direction δ, which we note $s \in_\delta \Sigma(S)$, if a positive step is possible, that is if for any sufficiently small $\varepsilon > 0, s + \varepsilon\delta \in \Sigma(S)$. It follows from the definition that $s \in_\delta \Sigma(S)$ if and only if $\forall\ 1 \leq j \leq n+1, \lambda_j = 0 \Rightarrow \sigma_j \geq 0$, where $s \in \Sigma(S), \lambda = \begin{pmatrix} S \\ e^T \end{pmatrix}^{-1} \begin{pmatrix} s \\ 1 \end{pmatrix}$ and $\sigma = \begin{pmatrix} S \\ e^T \end{pmatrix}^{-1} \begin{pmatrix} \delta \\ 0 \end{pmatrix}$. This property holds since by definition, $s \in_\delta \Sigma(S)$ if and only if $\forall \varepsilon > 0, \lambda + \varepsilon\sigma \geq 0$.

The maximal displacement will get us to a point $s' \neq s$ located on the

boundary of $\Sigma(S)$ and such that $s' \notin_\delta \Sigma(S)$. Formally, we seek the step length

$$\theta^* = \max\{\theta \in \mathbf{R}_+ | s + \theta^* \delta \in \Sigma(S)\}.$$

It is readily verified that $\theta^* = \min\left\{ -\frac{\lambda_j}{\sigma_j} | \sigma_j < 0 \right\}, 1 \leq j \leq n+1$. We also see that a simplex traversal requires a matrix inversion ($\mathcal{O}(n^3)$ operations) and $\mathcal{O}(n)$ comparisons.

4.2.2. Refinement of the state grid

Suppose we have an initial set of grid points over which function $\hat{V}_{t-1}$ has been evaluated. These points are the vertices of simplices. We may want to densify the grid to improve the approximation. Additional grid points may be obtained by iteratively dividing the existing simplices until meeting a desired level of precision.

The hydrological variable $\tilde{\epsilon}_{t-1}$ being held constant, it will temporarily be omitted for the sake of simplicity. We assume that for any grid point s_{t-1} we have an evaluation of the value function $\hat{V}_{t-1}(s_{t-1})$ and a subgradient $g_{t-1} \in \partial \hat{V}_{t-1}(s_{t-1})$. Suppose simplex $\Sigma(S^j_{t-1})$ is to be divided. The imprecision around the true value function over $\Sigma(S^j_{t-1})$ will be assessed via the maximum difference, over $\Sigma(S^j_{t-1})$, between an upper bound and a lower bound. Both bounds are based on the concavity of $\hat{V}_{t-1}(s_{t-1})$.

The lower bound simply consists in an interpolation from simplex vertices $s^i_{t-1}, 1 \leq i \leq n+1$. Letting $z^i_{t-1} = \hat{V}_{t-1}(s^i_{t-1}), 1 \leq i \leq n+1$, the lower bound is $\underline{V}_{t-1}(s_{t-1}) = z^T_{t-1} \begin{pmatrix} S^j_{t-1} \\ e^T \end{pmatrix}^{-1} \begin{pmatrix} s_{t-1} \\ 1 \end{pmatrix}, s_{t-1} \in \Sigma(S^j_{t-1})$, where $z^T_{t-1} = (z^1_{t-1}, z^2_{t-1}, \cdots, z^{n+1}_{t-1})$. The upper bound will result from tangency conditions. For any grid point ξ^0_{t-1}, with $z^0_{t-1} = \hat{V}_{t-1}(\xi^0_{t-1})$, and $g^0_{t-1} \in \partial \hat{V}_{t-1}(\xi^0_{t-1})$, the linearization $w_{t-1}(s_{t-1}; \xi^0_{t-1}) \overset{\Delta_{t-1}}{=} z^0_{t-1} + g^0_{t-1}(s_{t-1} - \xi^0_{t-1})$, with support $\{\xi^0_{t-1}\}$ satisfies:

$$w_{t-1}(s_{t-1}; \xi^0_{t-1}) \geq \hat{V}_{t-1}(s_{t-1}) \ \forall s_{t-1} \in \mathbb{P}_{t-1}.$$

This can be generalized to a multi-support $\Xi_{t-1} = \{\xi^0_{t-1}, \xi^1_{t-1}, \cdots, \xi^m_{t-1}\}$. The function $\overline{V}_{t-1}(s_{t-1}; \Xi_{t-1}) \overset{\Delta_{t-1}}{=} \mathrm{Min}_{1 \leq k \leq m} w_{t-1}(s_{t-1}; \xi^k_{t-1})$ (hereafter $\overline{V}_{t-1}(s_{t-1})$ for short) is indeed an upper bound on $\hat{V}_{t-1}$ over $\mathbb{P}_{t-1}$. The multi-support Ξ_{t-1} will include all vertices $s^i_{t-1}, 1 \leq i \leq n+1$, of $\Sigma(S^j_{t-1})$, but also possibly neighboring vertices. The resulting upper bound is local, hence not the tightest possible, in order to limit the computational burden.

In summary, the *evaluation* of simplex $\Sigma(S^j_{t-1})$ seeks an imprecision measure Δ^j_{t-1} and a division point s^*_{t-1} where this imprecision is realized:

$$\Delta^j_{t-1} = \mathrm{Max}_{s_{t-1}} \left\{ \overline{V}_{t-1}(s_{t-1}) - \underline{V}_{t-1}(s_{t-1}) | s_{t-1} \in \Sigma(S^j_{t-1}) \right\},$$
$$s^*_{t-1} \in \mathrm{Argmax}_{s_{t-1}} \left\{ \overline{V}_{t-1}(s_{t-1}) - \underline{V}_{t-1}(s_{t-1}) | s_{t-1} \in \Sigma(S^j_{t-1}) \right\}.$$

We note that $\hat{V}_{t-1}(s_{t-1})$ is affine over $\Sigma(S^j_{t-1})$ if and only if $\Delta^j_{t-1} = 0$, in which case it is of no interest to further divide $\Sigma(S^j_{t-1})$. For each period $t-1, t = 2, \ldots, T+1$, the foregoing optimization can be cast as a linear program:

$$\Delta^j_{t-1} = \underset{\mu_{t-1}, s_{t-1}, \lambda}{\text{Max}} \mu_{t-1} - z^T_{t-1}\lambda \tag{40}$$

$$\text{S.t. } \mu_{t-1} \leq z^i_{t-1} + g^i_{t-1}(s_{t-1} - \xi^i_{t-1}), \qquad 1 \leq i \leq m \tag{41}$$

$$s_{t-1} = S^j_{t-1}\lambda \tag{42}$$

$$e^T\lambda = 1 \tag{43}$$

$$s_{t-1} \in \mathbb{P}_{t-1} \tag{44}$$

$$\lambda \geq 0 \tag{45}$$

Assuming for instance that $|\Xi_{t-1}| \leq 2(n+1)$ and that $\mathbb{P}_{t-1}$ is defined by p inequalities, this program in standard equality form involves at most $3(n+1)+p+1$ equalities and $4(n+1)+p$ variables. The number of operations required to solve the linear program is polynomial in these two quantities.

In summary, each iteration of the refinement phase consists in the following steps:

(i) find an active simplex with maximal imprecision Δ^j_{t-1} (requiring $\mathcal{O}(N)$ operations);

(ii) update the list of active simplices ($\mathcal{O}(n+2)$ operations) and the list of grid points (1 operation);

(iii) for each child i:

- update its vertices (1 operation) and its neighbors ($\mathcal{O}((n+1)n^2)$ operations);
- solve (40)-(45) (polynomial in n and p), and note the optimal Δ^i_{t-1} and s^{i*}_{t-1}; and
- update the imprecision of any neighbor which shares the child's division point by solving (40)-(45) (polynomial in n and p).

4.3. Computing the expectation of the value function

The SDP recursion (25)-(26) has two major sources of complexity, namely (i) taking an expectation and (ii) optimizing. The object of this section is the computation of the expectation $E_{Q_t, \tilde{\epsilon}_t | \epsilon_{t-1}} \left[V_t \left(s^*_t \left(x_t + Q_{I_R t} \right), \tilde{\epsilon}_t \right) \right]$ for all discrete states (x_t, ϵ_{t-1}). We assume that the hydrological variable evolves according to a known transition probability distribution $P(\tilde{\epsilon}_t = \epsilon_t | \tilde{\epsilon}_{t-1} = \epsilon_{t-1})$. This is akin to considering that the hydrological variable is Markovian, which is a common

assumption in reservoir management (see [40]; [34]; [41]). We then have:

$$\begin{aligned} E_{Q_t,\tilde{\epsilon}_t|\epsilon_{t-1}}\left[V_t\left(x_t+Q_{I_R t}\right)\right] &= E_{\tilde{\epsilon}_t|\epsilon_{t-1}} E_{Q_t|\epsilon_{t-1}}\left[V_t\left(s_t^*(x_t+Q_{I_R t})\right)\right] \\ &= \sum_{\epsilon_t} P(\tilde{\epsilon}_t=\epsilon_t|\tilde{\epsilon}_{t-1}=\epsilon_{t-1}) \\ &\quad E_{Q_t|\epsilon_{t-1}}\left[V_t\left(s_t^*(x_t+Q_{I_R t})\right)\right]. \end{aligned}$$

Taking the expectation then amounts to computing:

$$J_t(x_t,\epsilon_{t-1}) = E_{Q_t|\epsilon_{t-1}}\left[V_t\left(s_t^*\left(x_t+Q_{I_R t}\right),\epsilon_t\right)\right]$$

for a discrete set of values (x_t, ϵ_{t-1}). This may again be simplified if the natural inflows to the reservoirs are spatially perfectly correlated. This "uni-basin" hypothesis is approximately verified in many real situations, where the reservoirs are fed by a common hydrological basin. From a practical point of view, the dimension of the natural inflows' support decreases from n to 1. This assumption is often used by hydrologists in case of planning horizon with monthly or weekly time step (see [42]). For instance this hypothesis will be used later in the case study.

In the case of perfect correlation, we may consider a scalar variate $\tilde{\tau}_t \geq 0$ such that

$$Q_t = q_t \tilde{\tau}_t,$$

where $q_t \in \mathbf{R}^n$ represents the relative contributions of natural inflows to the reservoirs, and $\tilde{\tau}_t \geq 0$.

Numerical integration might be used to compute function $J_t(x_t, \epsilon_{t-1})$ whenever the probability distribution of $\tilde{\tau}_t$ is known. However, the final stocks space being partitioned into simplices, we are able to obtain analytical forms for the expectation. We will first analyze the trajectory of the final stocks in the partitioned space. Second, we will address the evaluation of function $\hat{V}_t$ over this trajectory.

The vertices of the simplices form an irregular grid over which $\hat{V}_t$ has been evaluated. For any simplex $\Sigma(S_t^j)$, let $z_i^t = \hat{V}_t(s_t^i, \epsilon_t), 1 \leq i \leq n+1$. For any other point $s_t \in \Sigma(S_t^j)$, $\hat{V}_t$ is interpolated over $\Sigma(S_t^j)$. The interpolation is uniquely determined as the affine form:

$\hat{V}_t(s_t,\epsilon_t) = z_t^T \begin{pmatrix} S_t^j \\ e^T \end{pmatrix}^{-1} \begin{pmatrix} s_t \\ 1 \end{pmatrix}$, where $z_t^T = (z_t^1, \ldots, z_t^{n+1})$.

Under the uni-basin hypothesis, the optimal spillage and final stocks trajectories are unidimensional paths in $\mathbf{R}^n$. From (24) we see that the final stocks' trajectory $\hat{s}_t(\tau) \stackrel{\Delta}{=} s_t^*(x_t + q_t\tau)$ is linear until attaining a state space boundary. We call *corner* a point at which a space state boundary becomes saturated. At each new corner, the trajectory direction changes, the new boundary remaining saturated until the end of the trajectory, $\bar{s}_t$. The trajectory $\hat{s}_t(\tau)$ is illustrated in Figure 4.

In addition to corners, the trajectory $\hat{s}_t(\tau)$ crosses the simplices at intermediate *traversal points* (see Figure 4). We may define a set of *nodes* $\Theta = \{\theta_i | i = 1, \ldots, w\}$ of the parameter $\tilde{\tau}_t$ such that $\hat{s}_t(\theta_i)$ is a corner or a traversal point.

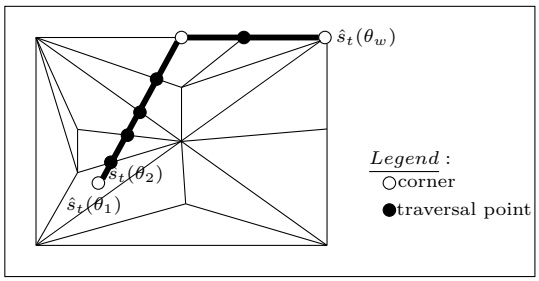


Figure 4: Example of final stocks' trajectory

Computing the trajectory entails a sequence of simplex traversals. Each traversal is followed by the determination of a neighboring simplex via a pivot operation ($\mathcal{O}(n^2)$ operations). The number of simplices to be traversed over the whole trajectory is at worst $\mathcal{O}(N)$.

Function $\hat{V}_t$ is evaluated at each node $\theta_i, i = 1, \cdots, w$. Let $h_t(\theta_i, \epsilon_t) = \hat{V}_t(\hat{s}_t(\theta_i), \epsilon_t)$. Function h_t's slopes change at the nodes in Θ. Thus, h_t is piecewise affine. For each segment (θ_{i-1}, θ_i), the slope is constant and is given by:

$$\alpha_i = \frac{\omega_i - \omega_{i-1}}{\theta_i - \theta_{i-1}}, \quad i = 2, \ldots, w+1,$$

where $\omega_i = \hat{V}_t(\hat{s}_t(\theta_i), \epsilon_t)$. In particular, $\alpha_{w+1} = 0$. Function h_t is constant for $\tau \geq \theta_w$, the stocks of water attaining their maximal levels. We have

$$h_t(\theta_i, \epsilon_t) = \begin{cases} \omega_i & i = 2, \ldots, w \\ \omega_w & i = w+1. \end{cases}$$

Then, function h_t is of the form:

$$h_t(\tau, \epsilon_t) = \begin{cases} \omega_{i-1} + \alpha_i(\tau - \theta_{i-1}) & \theta_{i-1} \leq \tau \leq \theta_i, \quad i = 2, \ldots, w, \\ \omega_w & \tau \geq \theta_w. \end{cases}$$

4.3.1. Analytical form

Function $J_t(x_t, \epsilon_{t-1})$ is calculated over each segment $(\theta_{i-1}, \theta_i), i = 2, \cdots, w+1$. Let $F_t(\tau)$ be the cumulative distribution of $\tilde{\tau}_t$ conditioned on the value ϵ_{t-1}. Over each interval $[\theta_{i-1}, \theta_i]$, the conditional expectation is computed as $\int_{\theta_{i-1}}^{\theta_i} h_t(\tau, \epsilon_t) dF_t(\tau)$. Using the linearity property of the expectation, we then get

$$J_t(x_t, \epsilon_{t-1}) = \sum_{i=2}^{w+1} \int_{\theta_{i-1}}^{\theta_i} h_t(\tau, \epsilon_t) dF_t(\tau),$$

where

$$\begin{aligned}\sum_{i=2}^{w+1}\int_{\theta_{i-1}}^{\theta_i} h_t(\tau,\epsilon_t)dF_t(\tau) &= \sum_{i=2}^{w+1}\int_{\theta_{i-1}}^{\theta_i}(\omega_{i-1}+\alpha_i(\tau-\theta_{i-1}))dF_t(\tau)\\ &= \sum_{i=2}^{w+1}\alpha_i E[\tilde{\tau}_t|\epsilon_{t-1}]_{\theta_{i-1}}^{\theta_i} + \sum_{i=2}^{w+1}\gamma_{i-1}\left[F_t(\theta_i)-F_t(\theta_{i-1})\right]\\ &= \sum_{i=2}^{w+1}\alpha_i E[\tilde{\tau}_t|\epsilon_{t-1}]_{\theta_{i-1}}^{\theta_i} + \sum_{i=2}^{w}\gamma_{i-1}\left[F_t(\theta_i)-F_t(\theta_{i-1})\right] + \\ &\quad \omega_w[1-F_t(\theta_w)],\end{aligned}$$

with $\gamma_i = \omega_i - \alpha_{i+1}\theta_i$ and $\theta_{w+1} = +\infty$.

We provide more details for the specific cases of normal and log-normal distributions.

4.3.2. Case of the truncated normal distribution

Assume $\tilde{\tau}_t$ is a normal random variable with mean $\mu(\epsilon_{t-1})$ and standard deviation $\sigma(\epsilon_{t-1})$, we have:

$$\begin{aligned}F_t(\tau) = P(\tilde{\tau}_t \le \tau) &= \Phi\left(\frac{\tau-\mu(\epsilon_{t-1})}{\sigma(\epsilon_{t-1})}\right)\\ f_t(\tau) = F_t'(\tau) &= \frac{1}{\sigma(\epsilon_{t-1})}\phi\left(\frac{\tau-\mu(\epsilon_{t-1})}{\sigma(\epsilon_{t-1})}\right),\end{aligned}$$

where $\phi(v) = \frac{1}{\sqrt{2\pi}}e^{\frac{-v^2}{2}}$ and $\Phi(v) = \int_{-\infty}^{v}\phi(t)dt$ are respectively the density function and the cumulative distribution of the standard normal variable. Let $G(\tau) = \mu(\epsilon_{t-1})F_t(\tau) - (\sigma(\epsilon_{t-1}))^2 f_t(\tau)$. The standard normal distribution has the property: $\phi'(v) = -v\phi(v)$. It follows that

$$\int \tau f_t(\tau)d\tau = G(\tau).$$

Therefore

$$E[\tilde{\tau}_t|\epsilon_{t-1}]_{\theta_{i-1}}^{\theta_i} = G(\theta_i) - G(\theta_{i-1})$$

and

$$J_t(x_t,\epsilon_{t-1}) = \frac{1}{1-F_t(0)}\sum_{i=2}^{w+1}\left\{\alpha_i\left[G(\theta_i)-G(\theta_{i-1})\right] + \gamma_{i-1}\left[F_t(\theta_i)-F_t(\theta_{i-1})\right]\right\}.$$

4.3.3. Case of the log-normal distribution

$\tilde{\tau}_t$ is a log-normal variable with parameters $\mu(\epsilon_{t-1})$ and $\sigma(\epsilon_{t-1})$ if $\ln\tilde{\tau}_t$ is a normal variable with mean $\mu(\epsilon_{t-1})$ and variance $(\sigma(\epsilon_{t-1}))^2$. The density function of $\tilde{\tau}_t$ is: $f_t(\tau;\mu(\epsilon_{t-1}),\sigma(\epsilon_{t-1})) = \frac{1}{\sqrt{2\pi}\sigma(\epsilon_{t-1})\tau}e^{-\frac{(\ln\tau-\mu(\epsilon_{t-1}))^2}{2(\sigma(\epsilon_{t-1}))^2}}$; its mean

is $\nu = E[\tilde{\tau}_t] = e^{\mu(\epsilon_{t-1})+\frac{(\sigma(\epsilon_{t-1}))^2}{2}}$, and its k$^{\text{th}}$ moment about the origin $\nu_k = E[\tilde{\tau}_t^k] = e^{k\mu(\epsilon_{t-1})+\frac{k^2(\sigma(\epsilon_{t-1}))^2}{2}}$.

As previously, let Φ be the cumulative distribution of the standard normal variable. Let $\rho(v) = \frac{\ln v - \mu(\epsilon_{t-1}) - (\sigma(\epsilon_{t-1}))^2}{\sigma(\epsilon_{t-1})}$. We conclude with the following result:

Proposition 6. *For $0 \leq a < b$, $E[\tilde{\tau}_t|\epsilon_{t-1}]_a^b = \nu\left[\Phi\left(\rho(b)\right) - \Phi\left(\rho(a)\right)\right]$.*

Let us observe that the cumulative distribution of $\tilde{\tau}_t$ is : $F_t(\tau) = \Phi(\rho(\tau) + \sigma(\epsilon_{t-1}))$. Then

$$J_t(x_t, \epsilon_{t-1}) = \sum_{i=2}^{w+1} \{\nu\alpha_i \left[\Phi(\rho(\theta_i)) - \Phi(\rho(\theta_{i-1}))\right] + \gamma_{i-1}\left[\Phi(\rho(\theta_i) + \sigma(\epsilon_{t-1})) - \Phi(\rho(\theta_{i-1}) + \sigma(\epsilon_{t-1}))\right]\}.$$

5. Numerical experimentation

The goal of this section is to empirically examine trade-offs between complexity and accuracy. We define complexity in terms of two dimensions: number of reservoirs n and number of grid points κ. For fixed n, relative imprecision is measured by the ratio $\rho_\kappa = \frac{\Delta_\kappa}{\Delta_1}$, where Δ_κ is the imprecision (40) corresponding to the κ^{th} grid point. Two lines of enquiry are conducted. In subsection 5.1, controlling for the number of reservoirs n, we examine the relationship between relative accuracy and number of grid points. In subsection 5.2, controlling for the number of grid points κ, we ask to what extent relative imprecision depends on the state space dimension n, and thus withstands the curse of dimensionality. The results are briefly discussed in subsection 5.3.

5.1. Accuracy vs. state grid density

We consider network configurations with $n = 3$, 4, 5, 6 and 10 reservoirs that form an arborescence. For each n, a fixed configuration is considered and 40 test problems are generated. The time span of each problem is 10 periods. The parameters of each test-problem are generated randomly and independently in each period (see Table 1), introducing non-stationarity as well as a high degree of variation among test problems. Production functions for run-of-the-river power plants $i \in I_F$ are of the form

$$f_i(u_{it}) = \beta_i[(u_{it} + \gamma_i)^{\alpha_i} - \gamma_i^{\alpha_i}],\ 0 \leq \alpha_i \leq 1,\ \beta_i > 0,\ \gamma_i \geq 0.$$

These functions are concave and non-decreasing, reflecting the fact that head effects often can be neglected. Production functions for plants downstream of a reservoir $i \in I_{RC}$ are of the form

$$f_i(u_{it}, s_{it}) = \beta_i[(u_{it} + \gamma_i)^{\alpha_i} - \gamma_i^{\alpha_i} + (u_{it}s_{it})^{\theta_i}],\ 0 \leq \alpha_i \leq 1,\ \beta_i > 0,\ \gamma_i \geq 0,\ 0 \leq \theta_i \leq 1.$$

The interaction term $u_{it}s_{it}$, denoting a positive effect of head height on production efficiency, destroys concavity to an extent depending on the parameter θ_i.

In each period t, the value function is approximated over grid points generated by the simplicial partitioning method described in subsection 4.2.2, by solving problem (27)-(39) for each such grid point. The expectation is computed as described in subsection 4.3. For each test problem, the following parameters are randomly and independently generated from a uniform distribution on a support of the form $[a, b]$ as depicted in Table 1. In addition, in each period, a vector q_t of relative contributions to the reservoirs, with $\sum_{i=1}^{n} q_{it} = 1$, is randomly generated.

Table 1: Bounds on the model parameters

Parameter	a	b
$\underline{s}_{it}, i \in I_R$	150	600
$\bar{s}_{it}, i \in I_R$	800	7000
$\underline{u}_{it}, i \in I_R$	0	0
$\bar{u}_{it}, i \in I_R$	$0.05\bar{s}_{it}$	$1.5\bar{s}_{it}$
$\bar{u}_{it}, i \in I_F$	950	3500
β	0.9	1.5
α	0.7	0.9
γ	0.25u	0.7u
θ	0.05	0.1

For each of the test problems, the division algorithm is performed recursively on each of the 10 value functions. The division stops when the number κ of grid points reaches a prescribed maximum $\bar{\kappa}$. The relative imprecision is then measured on the initial value function. Since larger grid sizes are expected to be required with larger space dimension, we set $\bar{\kappa} = \mathcal{O}(n(n+1))$. For a fixed number n of reservoirs, 40 test problems are run. For each κ, relative imprecisions are then sorted from smallest to largest, and we note the 10^{th}, 50^{th} and 90^{th} percentiles.

In Figure 5, the trajectories of $ln\ \rho_\kappa$ against κ are mapped for these three percentiles. They seem to indicate a deteriorating rate of convergence. To ascertain this effect, we run two regression models for each percentile trajectory. Model 1 (Log-Log) is of the form: $ln\ \rho_\kappa = \beta + \alpha ln\kappa + \text{error}$. It implies a sublinear convergence with varying rate $(1 + \frac{1}{\kappa})^\alpha (\alpha < 0)$. The geometric average of this rate of convergence over a span $\{1, \cdots, \bar{\kappa}\}$ is $(1 + \bar{\kappa})^{(\alpha/\bar{\kappa})}$. Model 2 (Log-Lin) is of the form: $ln\ \rho_\kappa = \beta + \alpha\kappa + \text{error}$. It implies a linear convergence with constant rate $e^\alpha (\alpha < 0)$.

Tables 2 report regression results for 3 percentiles of 5 reservoir configurations. With a high degree of significance they concur to suggest a sublinear convergence. Indeed, regardless of the number of reservoirs, both regressions (Log-log, and Log-lin) exhibit high statistical significance, as illustrated by the values of R^2 (> 0.70 in all, but one case), however with a clear edge to the

Log-log models (R^2 in general greater than 0.90). These results also confirm those depicted in Figure 5; the rate of convergence (RoC) slightly deteriorates as the number of reservoirs increases.

Table 2: Imprecision vs. grid size
(n = # of reservoirs, RoC = estimated rate of convergence)

		Percentile	10%	50%	90%
n=3	Log-log	Estimated model	$ln\ \rho_\kappa =$ - 0.3014 - 0.9452 ln κ	$ln\ \rho_\kappa =$ 0.1332 - 0.8486 lnκ	$ln\ \rho_\kappa =$ 0.2557 - 0.6931 lnκ
		RoC	0.9727	0.9629	0.9667
		R^2	0.9803	0.9982	0.9943
	Log-lin	Estimated model	$ln\ \rho_\kappa =$ -2.5976 - 0.02165 κ	$ln\ \rho_\kappa =$ -1.8679 - 0.0204 κ	$ln\ \rho_\kappa =$ -1.3574 - 0.0170 κ
		RoC	0.9786	0.9798	0.9831
		R^2	0.7104	0.7994	0.8301
n=4	Log-log	Estimated model	$ln\ \rho_\kappa =$ 0.105 - 0.5751ln κ	$ln\ \rho_\kappa =$ 0.1675 - 0.4935 lnκ	$ln\ \rho_\kappa =$ 0.1227- 0.3750 lnκ
		RoC	0.9849	0.9870	0.9901
		R^2	0.9836	0.9870	0.8523
	Log-lin	Estimated model	$ln\ \rho_\kappa =$ -1.4861 - 0.0089 κ	$ln\ \rho_\kappa =$ -1.2037 - 0.0076 κ	$ln\ \rho_\kappa =$ - 0.9659- 0.0052 κ
		RoC	0.9912	0.9925	0.9947
		R^2	0.8598	0.8523	0.7131
n=5	Log-log	Estimated model	$ln\ \rho_\kappa =$ - 0.0822 - 0.3462 lnκ	$ln\ \rho_\kappa =$ - 0.0340 - 0.2774 lnκ	$ln\ \rho_\kappa =$ 0.1820 - 0.2707 lnκ
		RoC	0.9933	0.9946	0.9947
		R^2	0.9895	0.9946	0.9842
	Log-lin	Estimated model	$ln\ \rho_\kappa =$ -1.1819 - 0.0036 κ	$ln\ \rho_\kappa =$ - 0.9229 - 0.0028 κ	$ln\ \rho_\kappa =$ - 0.6725 - 0.0028 κ
		RoC	0.9964	0.9972	0.9972
		R^2	0.8083	0.7827	0.8259
n=10	Log-log	Estimated model	$ln\ \rho_\kappa =$ - 0.6811- 0.1457 ln κ	$ln\ \rho_\kappa =$ - 0.1183 - 0.1731 lnκ	$ln\ \rho_\kappa =$ 0.08844 - 0.1182 lnκ
		RoC	0.9975	0.9971	0.9980
		R^2	0.9009	0.9595	0.9521
	Log-lin	Estimated model	$ln\ \rho_\kappa =$ -1.1659 - 0.0013 κ	$ln\ \rho_\kappa =$ - 0.6829 - 0.0016 κ	$ln\ \rho_\kappa =$ - 0.2917- 0.0011 κ
		RoC	0.9987	0.9984	0.9989
		R^2	0.7474	0.8643	0.9083
n=12	Log-log	Estimated model	$ln\ \rho_\kappa =$ - 0.2261 - 0.1223 ln κ	$ln\ \rho_\kappa =$ - 0.0912- 0.0856 lnκ	$ln\ \rho_\kappa =$ 0.0590 - 0.0428 lnκ
		RoC	0.9989	0.9992	0.9996
		R^2	0.9484	0.9995	0.9607
	Log-lin	Estimated model	$ln\ \rho_\kappa =$ -0.7061 - 0.0006 κ	$ln\ \rho_\kappa =$ - 0.4474 - 0.0003 κ	$ln\ \rho_\kappa =$ - 0.1132 - 0.0002 κ
		RoC	0.9994	0.9997	0.9998
		R^2	0.8723	0.6455	0.7857

5.2. Accuracy vs. state space dimension

We consider fixed network configurations with $n = 4$, 6, 8, 10 and 12 reservoirs. For each configuration, we consider a time span of a single period and as-

sume that the terminal value function is of the form $V_1(s) = \frac{1}{2}s'^T s' - \frac{1}{2}||s - s'||^2$, where $s' > Q$ is a target reservoir level.

We use the same production functions as in subsection 5.1, and problem parameters are similarly generated. The maximum grid size is set at $\bar{\kappa} = 100n$, then at $\bar{\kappa} = 200n$. For each $(n, \bar{\kappa})$ pair, we randomly generate 20 test problems. We note the 10^{th}, 50^{th} and 90^{th} percentiles of the attained relative imprecisions, as reported in Table 3. Increasing the number of grid points has little effect on the attained imprecision as the number of reservoirs increases.

Table 3: Attained relative imprecision

Grid size	Percentile	n = 4	n = 6	n = 8	n = 10	n = 12
$100n$	10%	0.030	0.192	0.542	0.666	0.747
	50%	0.036	0.232	0.573	0.721	0.788
	90%	0.043	0.253	0.644	0.787	0.803
$200n$	10%	0.021	0.089	0.546	0.631	0.737
	50%	0.024	0.107	0.498	0.666	0.770
	90%	0.028	0.136	0.557	0.760	0.825

5.3. Discussion of the results

Our method rests on two principles : (i) simplicial partitioning of the state space, and (ii) bounding the approximation error. The fundamental underlying goal is to balance accuracy and computational efficiency, as a joint trade-off. The method is adaptive, as error bounding is performed online. No prior state grid is required. The bounding mechanism is strict, requiring no ex-post validation. It requires some assumptions on problem structure, which are often deemed reasonable in multi-echelon multi-stage logistics.

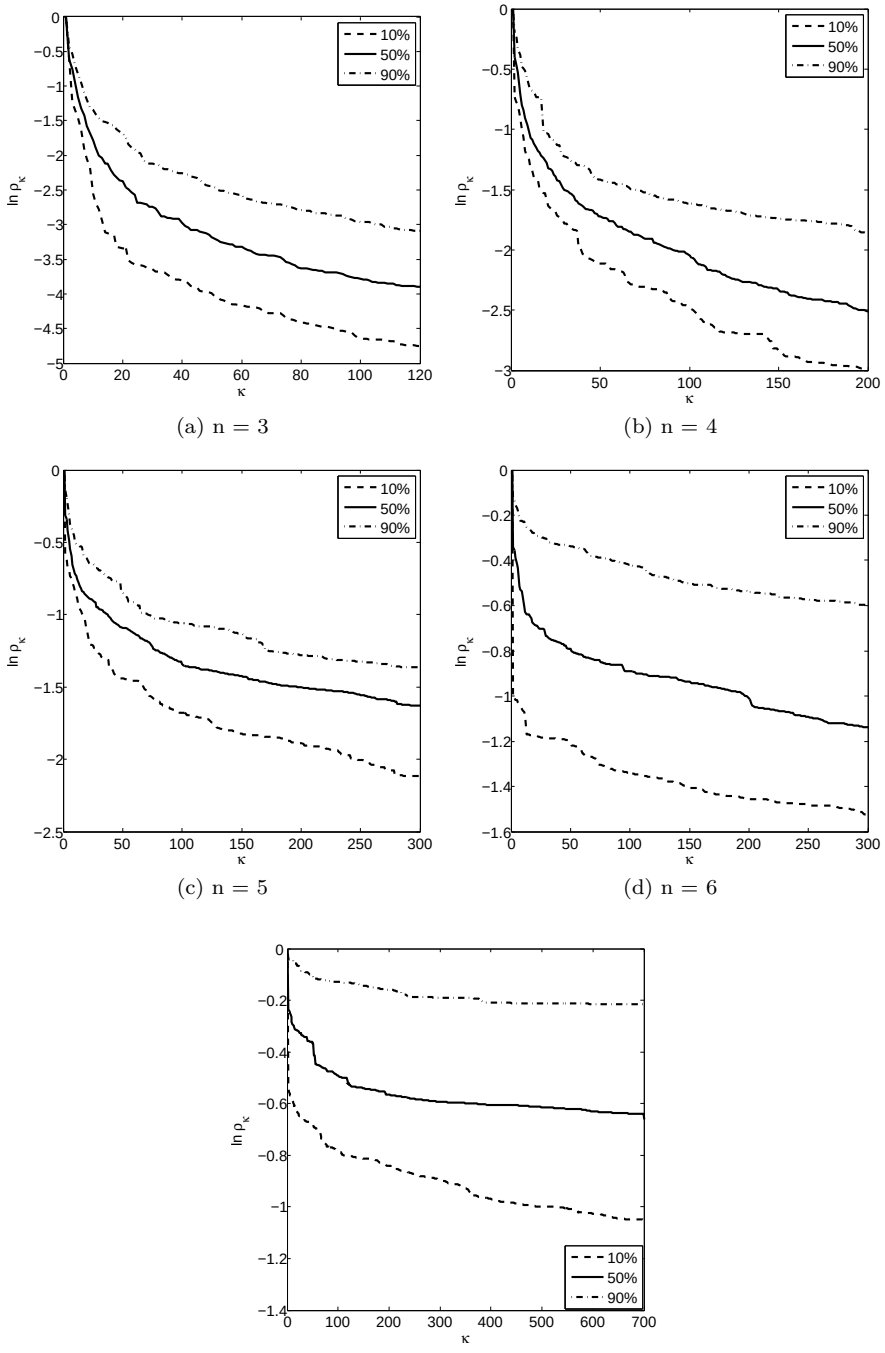
The preceding experimentation reflects a particular implementation of our method, suggesting the following remarks.

1. The generic method is highly parallelizable.

2. The value function used in the experiments has continuously varying curvature, resulting in uniformly-distributed grid points. By contrast, in hydropower management, the value function is almost linear on a large part of the state domain, nonlinearities occurring in the vicinity of reservoir capacities.

3. The present simplex division method turns out to yield very elongated simplices, thus overstating upper bounds on the value function (hence overstating errors). It could easily be revised to produce more regular partitions.

4. The present division criterion does not take into account the likelihood that a simplex will be visited. Visit frequencies could be assessed by simulation. A division criterion based on expected error (weighted by visit frequencies) would concentrate divisions where they matter most.



(a) n = 3

(b) n = 4

(c) n = 5

(d) n = 6

(e) n = 10

Figure 5: Evolution of the relative imprecision as a function of the number of grid points κ for 5 reservoir configurations

6. Case study

The aluminum division of Rio Tinto (RT) is an international company that produces and sells aluminum on the international market. RT has power plants located in Saguenay, a city of the province of Québec (Canada). The hydro-electric system is composed of six power plants, of which three are run-of-the river. The installed capacity is 3100 MW. However, because of insufficient natural inflows at some times during the year, the average production is about 2100 MW. The system feeds four aluminum plants for an overall load of about 2300 MW. Hence all internal electricity production is absorbed by aluminium smelting.

Section 6.1 describes the system under study. Sections 6.2 and 6.3 report the experimental framework adopted to solve RT's problem as well as the results of these experiments, respectively.

6.1. Description of the system under study

For operational reasons, RT is interested in studies on the south portion of the system as illustrated in Figure 6. The full arcs represent turbined flows, the dash line are spillage and the dotted lines illustrate natural inflows.This subsystem is undersized compared to the volume of natural inflows, and is more difficult to manage compared with the upstream portion of the system. Furthermore, the upstream turbines have low capacity. The downstream subsystem, of interest to us, comprises two reservoirs: Chute-Du-Diable and Lac Saint-Jean associated with the power plants Chute-Du-Diable and Isle-Maligne, respectively, and two run-of-the-river power plants: Chute-Savanne and Shipshaw. RT's research department assesses the value function of this subsystem through an SDP model. The model is parallelized, which allows the use of a sufficiently fine-grained grid. Hence, this model provides an optimal solution and is used as a standard by RT to evaluate the performance of other methods on this subsystem. RT's ultimate goal is the complete basin system, for which RT has no benchmark. Our mandate consisted in replicating the value functions for the south subsystem, with a view to reducing the size of the grid as compared to the standard SDP benchmark.

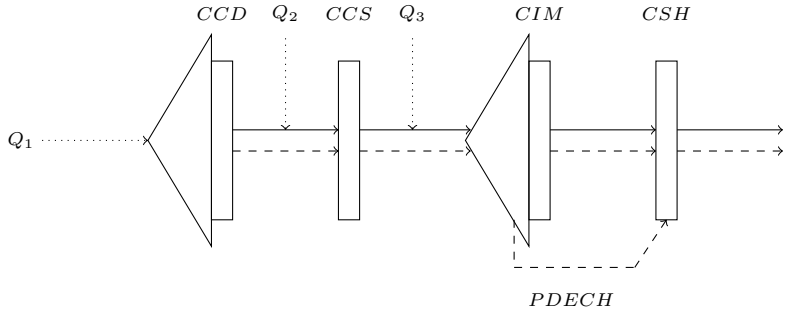


Figure 6: Representation of the RT's downstream subsystem

In the following, we will use the same notation as earlier.

Power generation is a function of the turbine's feeds and the level of the reservoirs for the power plants Chute-Du-Diable (CCD) and Isle-Maligne (CIM) (associated with the similarly labeled reservoirs). Though Chute-Savanne (CCS) is a run-of-the power plant, besides the turbine's feeds, its power generation also is a function of the level of the reservoir Lac Saint-Jean (LSJ), since the latter affects the level of its tailbay. Lastly, as concerned the power plant Shipshaw (CSH), power generation only depends on water release rate.

The maximal water release rate in the power plants Chute-Du-Diable and Isle-Maligne depends on the level of reservoir in the beginning of each period, whereas it is a constant for the power plants Chute-Savanne and Shipshaw. $\tilde{B}_i(s_{it}), i = 1, 3$, denote these maximal release rate functions. Furthermore, in the four power plants, any excess of water is spilled and causes loss of power by increasing the level of the tailbay. $\tilde{A}_i(y_{it}), i = 1, \cdots, 4$ denotes linear approximations of the loss of power functions. The reservoir LSJ is equipped with another spillway (PDECH). The water flowing through this spillway does not influence the power generation, however this spillway has a limited capacity that depends on the storage of water, which may influence the spillage in power plant CIM.

In period t, $y_{3,1t}$ denotes spillage through the power plant CIM, and $y_{3,2t}$ spillage through the outlet PDECH. $\tilde{D}(s_{3t})$ denotes the maximal water rate through the spillway PDECH.

During the summer, the bounds on the reservoir LSJ are very tight in order to allow navigation and recreational activities. If at the end of the spring the natural inflows are low, it may not be possible to release water from this reservoir during the summer. An optimization problem may then be infeasible. Therefore, following practices at RT's research group, we penalize bounds violation in the objective function. We will use penalty terms of the form $-\phi \max\{0, \underline{s}_{t+1} - s_{t+1}\}$ and $-\phi \max\{0, s_{t+1} - \bar{s}_{t+1}\}$; where each hm^3 of water under or over the regulatory limit is penalized by the amount ϕ supplied by RT. The planning horizon of RT's problem is the year with weekly time steps. In week t, we use the previous natural inflows Q_{t-1} as hydrological variable. Thus, for $t = 52, 51, \cdots, 1$, the SDP recursion is $V_t(s_t, Q_{t-1}) = E_{Q_t|Q_{t-1}}[G_t(s_t, Q_t)]$, with

$$G_t(s_t, Q_t) = \underset{u_t, y_t}{\text{Max}}\Big\{ \sum_{i=1,3} f_i(s_{it}, s_{i,t+1}, u_{it}) + f_2(s_{3t}, s_{3,t+1}, u_{2t}) + f_4(u_{4t}) - \sum_{j=1,2,4} \tilde{A}_j(y_j) - \tilde{A}_3(y_{3,1t}) - \phi\tilde{d}_{1t} - \phi\tilde{d}_{2t} + V_{t+1}(s_{t+1}, Q_t)\Big\} \tag{46}$$

$$\text{S.t. } s_{t+1} = s_t + Q_{IR_t} - B_{IR}u_t - C_{IR}y_t \tag{47}$$

$$\tilde{d}_{1t} \geq \underline{s}_{t+1} - s_{t+1} \tag{48}$$

$$\tilde{d}_{2t} \geq s_{t+1} - \bar{s}_{t+1} \tag{49}$$

$$0 \leq u_{it} \leq \tilde{B}_i(s_{it}), i = 1, 3 \tag{50}$$

$$u_{1t} + y_{1t} + Q_{2t} - u_{2t} - y_{2t} = 0 \tag{51}$$

$$u_{3t} + y_{3,1t} + y_{3,2t} - u_{4t} - y_{4t} = 0 \tag{52}$$

$$0 \leq u_{it} \leq \bar{u_i}, i = 2, 4 \tag{53}$$

$$y_{3,2t} \leq \tilde{D}(s_{3t}) \tag{54}$$

$$y_t \geq 0 \tag{55}$$

$$\tilde{d}_{1t}, \tilde{d}_{2t} \geq 0 \tag{56}$$

6.2. Experimental framework

We test the simplicial methodology on RT's subsystem through a two-phase approach: estimation of the value functions and simulation of the optimal operational policy. In each week, we use the simplicial decomposition scheme to construct the state space grids. As discussed in Section 4.1, in each week, a linear approximation of problem (46)–(56) ($\hat{G}_t$) is solved for each point of the grid. Recall that to implement GLP, moreover state space grid points, we also need evaluations of the production functions over a sample of points. These evaluations were provided by RT, but we use interpolation error analysis to reduce the size of the original samples.

RT's model assumes that release decisions are made after observation of the natural inflows. These latter are modeled as Markovian processes and, based on recommendations from RT, are assumed to obey log-normal distributions. In period t, assume that $\{q_t^i | 1 \leq i \leq l_t\}$ are the possible realizations of the process Q_t, and for each value of Q_{t-1}, let $P(Q_t = q_t^i | Q_{t-1})$ be the associated conditional transition probabilities. We then have $\hat{V}_t(s_t, Q_{t-1}) = \sum_{i=1}^{l_t} P(Q_t = q_t^i | Q_{t-1}) \hat{G}_t(s_t, Q_t)$.

We construct a multiple regression model with seasonal variables to estimate the conditional parameters of the log-normal distributions, which we use to estimate the conditional probabilities. The regression parameters are estimated through a 33-year sample of historical inflows. We adopt a similar scheme as in [50] to discretize the inflow processes. In week t, suppose the possible realizations of the inflows process are divided into l_t classes for which $q_t^1, \cdots, q_t^{l_t}$ are the respective center. Let $F_t(Q_t = q_t^i | Q_{t-1})$ be the cumulative distribution of the

log-normal distribution of $(Q_t = q_t^i|Q_{t-1})$ of parameters $\mu_t(Q_{t-1})$ and $\sigma^2(Q_{t-1})$. For each $Q_t = q_t$, we take $P(Q_t = q_t^1|Q_{t-1}) = F_t(q_t^1), P(Q_t = q_t^i|Q_{t-1}) = F_t(q_t^i) - F(q_t^{i-1}), i = 2, \ldots, l_t - 1$, and $P(Q_t = q_t^{lt}|Q_{t-1}) = 1 - F_t(q_t{}^{l_t-1})$. The conditional probability of each value of Q_t then is estimated as the conditional probability of its class.

The terminal value of water is first considered to be null. To deal with en-of-horizon effects, 208 weeks are being considered, the last 52 serve to eliminate such effects for the first 52, taking into account e.g. seasonal and persistence phenomena.

In the second phase, the estimated value functions are used to simulate the operational policy for another 25-year sample of historical inflows. Given an observed state (s_t, Q_t), the optimal policy is $[u_t, s_{t+1}, y_t] \in \text{Argmax}\{\hat{G}_t(s_t, Q_t)\}$, which is obtained by solving the linear approximation of (46)–(56). An initial storage s_1 is chosen corresponding to the maximal level of the first reservoir and about 98% of the second one. In the first period the linear approximation of (46)–(56) is solved for the state (s_1, Q_1). Afterwards, the approximation is solved for (s_2, Q_2), where s_2 is the optimal final stock at the end the first period. The process is repeated until the last week of the 25-year simulation horizon. For comparison purposes, in the simulation phase we use the same functions as in RT's model to evaluate the production corresponding to the optimal releases.

The results of the simulations are compared to those obtained with a classical SDP scheme constructed by RT and used as standard by RT's operations research group. To construct the value functions, in each week, RT's approach uses a fixed $10 \times 30 \times 7$ grid points (these numbers are the number of discretization points of the first and second reservoirs' levels, and the number of discretization values of the inflows, respectively). We similarly discretize the inflow processes.

6.3. Results and analysis

The two approaches (RT and the simplicial scheme) are compared based on the average effectiveness of the simulated policy defined as the ratio of the average generated power to the total outflows (water release and spillage). We do not report the raw results for confidentiality reasons.

Table 4 reports the relative average effectiveness (RT/simplicial) for each power plant as well as statistics pertaining to the number of grid points per week to construct the value functions. We use three different values $\rho_\kappa = \frac{\Delta_\kappa}{\Delta_1}$ as thresholds on the attained relative imprecision. The solution time (to estimate the 52 value functions) is also reported for each such threshold.

Overall, on average, the effectiveness of the policy is very similar in both cases, with a slight edge in favor of our method. Also observe that the effectiveness of release policy of CCD is slightly better with RT's approach than with ours. Since these results are very similar, we conjecture that the slight differences might be due to the fact that the production functions are approximated differently in the two models. However, the computational burden is significantly reduced with our approach. In each week, our method spares evaluations

of the value function. In each week, RT's scheme performs 2100 evaluations of the value function (300 stock grid points $\times$ 7 discrete values of inflows) for a total of 109 200 evaluations over the 52 weeks. By contrast, with an imprecision $\rho = 0.8$, in each week, on average, our method performs 413 evaluations of the value functions (59 stock grid points $\times$ 7 discrete values of inflows), for a total of 21 476 evaluations over the entire year, which roughly represents 20% of the total evaluations performed by the SDP used by RT.

Table 4: Comparison between the two models: relative effectiveness (RT/Simplicial) of the simulated policy and number of grid points per week

	Relative effectiveness				
	CCD	CCS	CIM	CSH	Overall average
Simp. ($\rho = 0.8$)	1.037	0.958	0.988	1.000	0.997
Simp. ($\rho = 0.5$)	1.037	0.958	0.992	1.000	0.997
Simp. ($\rho = 0.1$)	1.037	0.958	0.992	1.000	0.997
Approach	Number of stock grid points per week				
	Min	Max	Average	–	–
RT	300.000	300.000	300.000	–	–
Simp. ($\rho = 0.8$)	57.000	65.000	59.000	–	–
Simp. ($\rho = 0.5$)	62.000	78.000	68.327	–	–
Simp. ($\rho = 0.1$)	117.000	553.000	188.238	–	–
	Solution time for the simplicial model in seconds				
	$\rho = 0.8$	$\rho = 0.5$	$\rho = 0.1$	–	–
	175.032	200.144	662.166	–	–

Let us also observe that, in our model, while the solution time increases significantly with the precision $1-\rho$, the effectiveness of the policy remains quite similar. These results reveal that densifying the stock grids, i.e. performing significant number of evaluations of the value function in each week, does not necessarily improve the operational policy of the system.

In addition, Figure 7 shows the average (over 1300 weeks) of the weekly simulated trajectory of the reservoirs as prescribed by both approaches (RT and simplicial with $\rho = 0.8$). In both cases, at the beginning of the year, the level of the reservoirs are high. Then the reservoirs are progressively emptied in anticipation of the spring run off. Then refill occurs. Except for the summer, RT's model operates the reservoir CD with higher reservoir levels than does our model. However, in general this is the contrary for LSJ. Since RT's software provides an optimal solution for the maximization of the expected energy produced, and since the simplicial algorithm yields a very close value, we are in the presence of multiple near-optimal solutions.

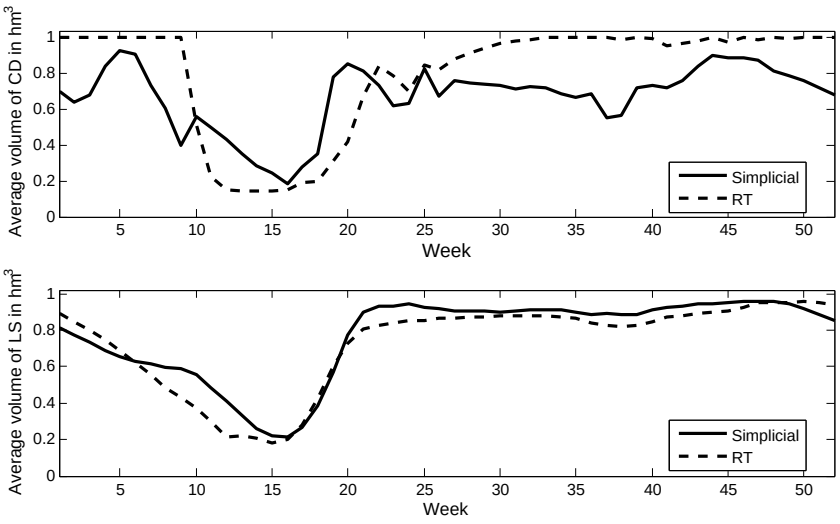


Figure 7: Comparison of the average optimal trajectory of the reservoirs

7. Conclusion

We presented an adaptive methodology for approximate stochastic dynamic programming, based on simplicial partitioning of the state space. The vertices of the simplices form an irregular grid where the value function is assessed. The grid is adaptively constructed in the course of problem solution and will be denser where the curvature of the true value function is higher.

The optimization step used here is a Generalized Linear Programming format, compatible with many mathematical programming structures, including combinatorial aspects. It automatically yields a concavification of the (approximate) value function. It is therefore best suited for problems which are nearly convex, e.g. with a convex feasible domain and a close to concave objective to be maximized. Concavity of the approximate value function allows us to assess its "imprecision". This measure guides where refinements of the grid should be conducted.

The simplex partitioning methodology yet simplifies another task, namely computing the expectation of the value function. It is possible to exploit the simplicial partition on a basin, i.e. a subset of nodes in the reservoir network where natural inflows are strongly correlated. Perfect correlation allows us to consider the inflow process as unidimensional. This facilitates analytical or numerical integration, sparing costly sampling. Although we illustrated a uni-basin case, the method can easily be extended to a hydrographic region partitioned into several disjoint basins.

We experimented our method on 400 test problems. Not unexpectedly, imprecision increased with the number of reservoirs and with restrictions on the

grid size. We think however, that this first version of our method could be significantly perfected.

Our scheme was also applied in an industrial context, using a large data base. Our method was compared with the fine regular grid SDP algorithm used as a standard at the company's research laboratories. We were able to replicate the optimal value of water while the computational burden significantly decreased, as in each week our model spared evaluations of the value function. Thus, the problem could be solved with an acceptable precision for an appropriately selected sample of grid points complemented by interpolation. For these reasons, we believe this paper makes a useful scientific contribution to the field of SDP.

Acknowledgments

The authors gratefully acknowledge funding from the Natural Sciences and Engineering Research Council of Canada and Rio Tinto Alcan, under Collaborative Research and Development Grant number 451382. The authors also acknowledge valuable comments from anonymous referees that helped improve the quality of the paper.

Appendix A: Proofs of propositions

Proof of Proposition 1. The spillage network is arborescent. Without loss of generality, assume the nodes are labeled in a topological order (as in Figure 1). The incidence matrix C of this network is then lower triangular. Since each node is equipped with a spillage system, the diagonal elements of this matrix are all equal to 1. It follows that C is non-singular and $C^{-1} \in \{0,1\}^{p\times p}$ [51].

Now, let us show that for all $ds \geq 0, V_{t-1}(s_{t-1}+ds, \epsilon_{t-1}) \geq V_{t-1}(s_{t-1}, \epsilon_{t-1})$. To simplify the notation, we omit the hydrological variable ϵ_{t-1}.

Define

$$\phi_t(s_{t-1}, u_t, Q_t, y_t) = \sum_{i\in I_{RC}} f_{it}(u_{it}, s_{i,t-1}, \epsilon_{t-1}) + \sum_{i\in I_F} f_{it}(u_{it}) + V_t(s_{t-1} - Bu_t - Cy_t + Q_t, \epsilon_t).$$

Function ϕ_t measures the value of water for given reservoirs' levels s_{t-1}, release decisions u_t, observed inflows Q_t and spillage decisions y_t. Given an initial stock s_{t-1}, in period t, the optimal policy consists in a rule $u_t^*(s_{t-1})$ followed by a rule $y_t^*(s_{t-1} - Bu_t + Q_t)$. It follows that

$$V_{t-1}(s_{t-1}) = E_{Q_t}\left[\phi_t(s_{t-1}, u_t^*(s_{t-1}), Q_t, y_t^*(s_{t-1} - Bu_t^*(s_{t-1}) + Q_t))\right].$$

Suppose the initial stock increases from s_{t-1} to $\hat{s}_{t-1} = s_{t-1} + ds$ $(ds \geq 0)$. Consider a new policy where such stock increases are spilled:

$$\hat{u}_t(\hat{s}_{t-1}) = u_t^*(s_{t-1}),$$

$$\hat{y}_t(\hat{s}_{t-1} - B\hat{u}_t(\hat{s}_{t-1}) + Q_t) = y_t^*(s_{t-1} - Bu_t^*(s_{t-1}) + Q_t) + dy,$$

with $Cdy = ds$.

This policy is feasible. Indeed (i) the planned releases do not change and satisfy their bounds, (ii) the final stock does not change, (iii) the spillage increases by $dy = C^{-1}ds \geq 0$. In addition, one verifies that

$$\phi_t(s_{t-1}, \hat{u}_t(\hat{s}_{t-1}), Q_t, \hat{y}_t(\hat{s}_{t-1} - B\hat{u}_t(\hat{s}_{t-1}) + Q_t)) = \\ \phi_t(s_{t-1}, u_t^*(s_{t-1}), Q_t, y_t^*(s_{t-1} - Bu_t^*(s_{t-1}) + Q_t)).$$

Since the new policy is feasible but not necessarily optimal for the initial stock level $s_{t-1} + ds$, we get:

$$\begin{aligned} V_{t-1}(s_{t-1} + ds) &\geq E_{Q_t}\left[\phi_t(s_{t-1}, \hat{u}_t(\hat{s}_{t-1}), Q_t, \hat{y}_t\hat{s}_{t-1} - B\hat{u}_t(\hat{s}_{t-1}) + Q_t))\right] \\ &= E_{Q_t}\left[\phi_t(s_{t-1}, u_t^*(s_{t-1}), Q_t, y_t^*(s_{t-1} - Bu_t^*(s_{t-1}) + Q_t))\right] \\ &= V_{t-1}(s_{t-1}). \end{aligned}$$

□

Proof of Proposition 3. If $V_t(s_t, \epsilon_t)$ is concave in s_t, so is the expectation by the linearity property of the exception. Also if the production functions are concave, the objective is also concave (sum of concave functions). The claim follows since the feasible domain is a convex polyhedron. □

Proof of Proposition 4. The objective is linear and is maximized. The set $D_t = \{u_t, x_t, \lambda, \mu | (28) - (39)\}$ is convex. It follows that $\hat{V}_{t-1}$ is concave in $s_{t-1}\ \forall\ \epsilon_{t-1}$. □

Proof of Proposition 5. Since the production functions f_t are concave in u_t, for the same releases, the production value estimated as the best interpolation (convex combination) over samples points $\{\hat{u}_{jt} | j \in \Gamma_t\}$ is a lower bound on the actual production. In addition, since $\hat{V}_t \leq V_t$, we have $\hat{J}_t(x_t, \epsilon_{t-1}) \leq E_{Q_t, \tilde{\epsilon}_t | \epsilon_{t-1}}\left[V_t\left(s_t^*\left(x_t + Q_{I_R t}\right), \tilde{\epsilon}_t\right)\right]\ \forall x_t$ (by linearity of the expectation). Therefore, for any post-release reservoir levels x_t, the expectation evaluated as the best interpolation over sample points $\{\hat{x}_{jt} | j \in \Upsilon_t\}$ is underestimated. Lastly, let us observe that the polyhedron (28)-(39) (feasible domain of $\hat{V}_{t-1}$) is included in the polyhedron (26) (feasible domain of V_{t-1}). Since the objective is maximized, these prove our claim. □

Proof of Proposition 6. We have

$$\begin{aligned} E[\tilde{\tau}_t]_a^b &= E[e^Z]_{\ln a}^{\ln b}, \text{ where } Z \sim N(\mu(\epsilon_t), \sigma(\epsilon_t)^2) \\ &= \frac{1}{\sqrt{2\pi}\sigma(\epsilon_t)} \int_{\ln a}^{\ln b} e^{-\frac{1}{2}\left(\frac{z-\mu(\epsilon_t)}{\sigma(\epsilon_t)}\right)^2 + z} dz. \end{aligned}$$

We also have

$$\begin{aligned}
-\frac{1}{2}\left(\frac{z-\mu(\epsilon_t)}{\sigma(\epsilon_t)}\right)^2 + z &= -\frac{1}{2\sigma(\epsilon_t)^2}\left(z^2 - 2\mu(\epsilon_t)z + \mu(\epsilon_t)^2 - 2\sigma(\epsilon_t)^2 z\right) \\
&= -\frac{1}{2\sigma(\epsilon_t)^2}\left(z^2 - 2\left(\mu(\epsilon_t) + \sigma(\epsilon_t)^2\right)z + \mu(\epsilon_t)^2\right) \\
&= -\frac{1}{2\sigma(\epsilon_t)^2}\left[\left(z - \left(\mu(\epsilon_t) + \sigma(\epsilon_t)^2\right)\right)^2 - \left(\mu(\epsilon_t) + \sigma(\epsilon_t)^2\right)^2 + \right. \\
&\quad \left. \mu(\epsilon_t)^2\right] \\
&= -\frac{\left(z-\left(\mu(\epsilon_t)+\sigma(\epsilon_t)^2\right)\right)^2}{2\sigma(\epsilon_t)^2} + \mu(\epsilon_t) + \frac{\sigma(\epsilon_t)^2}{2},
\end{aligned}$$

Hence

$$E[\tilde{\tau}_t|\epsilon_t]_a^b = \frac{\nu}{\sqrt{2\pi}\sigma(\epsilon_t)} \int_{\ln a}^{\ln b} e^{-\frac{1}{2}\left(\frac{z-\mu(\epsilon_t)-\sigma(\epsilon_t)^2}{\sigma(\epsilon_t)}\right)^2} dz.$$

By the change of variable $t = \frac{z-\mu(\epsilon_t)-\sigma(\epsilon_t)^2}{\sigma(\epsilon_t)}$, we get

$$E[\tilde{\tau}_t|\epsilon_t]_a^b = \nu \int_{\rho(a)}^{\rho(b)} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} dt.$$

The integrand is the density function of the standard normal variable, which proves the proposition. □

Appendix B: Notation nomenclature

Sets

I_R	Network nodes with a reservoir
I_C	Network nodes with a power plant
I_F	Network nodes with power plant alone
I_{RC}	Network nodes with a reservoir and a power plant
$\mathbf{R}$	Real numbers
$\mathbf{R^n}$	n-dimensional vectors
$\mathbb{P}_t$	A general polytope

Parameters

n	Number of reservoirs
B	Connectivity matrix
$B_{I_F}(B_{I_R})$	Submatrix obtained by selecting the lines of B indexed in the set $I_R(I_F)$
C	Spillage routing matrix
$C_{I_F}(C_{I_R})$	Submatrix obtained by selecting the lines of C indexed in the set $I_R(I_F)$
T	Planning horizon
$\underline{s}_t$ (resp. $\overline{s}_t$)	lower (resp. upper) bound on reservoir levels
$\underline{u}_t$ (resp. $\overline{u}_t$)	lower (resp. upper) bound on turbined flows
Q_t	Natural inflows to the reservoirs in period t
$Q_{I_F t}(Q_{I_R t})$	Vector obtained by selecting the elements of Q_t indexed in the set $I_R(I_F)$
$\tilde{\epsilon_t}$	Hydrological variable in period t
e	Vector filled with ones
$\Sigma(S)$	Simplex generated by $n+1$ columns of matrix S
δ	Direction vector in $\mathbf{R}_+^n \backslash \{0\}$
Δ	Upper bound on the imprecision over a given simplex
κ	Number of grid points
ρ_κ	Relative imprecision at the κ^{th} grid point
N	Number of simplices
q_t	Relative contributions of natural inflows to the reservoirs in period t
$\tilde{\tau}_t$	Scalar variate representing the total inflows to the network in period t

Decision variables

u_t	Volume of released water in period t
x_t	Reservoir levels in period t after release decisions and before the realization of natural inflows
y_t	Spilled water after the realization of natural inflows
s_t	Reservoir levels at the end of period t
s^*	Division point of $\Sigma(S)$

Functions

$f_i(\cdot)$	Production function of power plant $i \in I_C$
$V_t(\cdot)$	Future value of water in energy units
$J_t(\cdot)$	Expected value of water in energy units
$\hat{V}_t(\cdot)$	Approximated future value of water in energy units
$\hat{J}_t(\cdot)$	Approximated expected value of water in energy units

References

[1] J. Rust, Using randomization to break the curse of dimensionality, Econometrica: Journal of the Econometric Society (1997) 487–516.

[2] V. C. Chen, Measuring the goodness of orthogonal array discretizations for stochastic programming and stochastic dynamic programming, SIAM Journal on Optimization 12 (2) (2002) 322–344.

[3] C. Cervellera, M. Gaggero, D. Macciò, Low-discrepancy sampling for approximate dynamic programming with local approximators, Computers & Operations Research 43 (2014) 108–115.

[4] J. Rust, Numerical dynamic programming in economics, Handbook of computational economics 1 (1996) 619–729.

[5] D. P. Bertsekas, J. N. Tsitsiklis, Neuro-dynamic programming: an overview, in: Decision and Control, 1995., Proceedings of the 34th IEEE Conference on, Vol. 1, IEEE, 1995, pp. 560–564.

[6] C. Cervellera, A. Wen, V. C. Chen, Neural network and regression spline value function approximations for stochastic dynamic programming, Computers & Operations Research 34 (1) (2007) 70–90.

[7] H. Fan, P. K. Tarun, V. C. Chen, Adaptive value function approximation for continuous-state stochastic dynamic programming, Computers & Operations Research 40 (4) (2013) 1076–1084.

[8] S. A. Johnson, J. R. Stedinger, C. A. Shoemaker, Y. Li, J. A. Tejada-Guibert, Numerical solution of continuous-state dynamic programs using linear and spline interpolation, Operations Research 41 (3) (1993) 484–500.

[9] D. L. Martinez, D. T. Shih, V. C. Chen, S. B. Kim, A convex version of multivariate adaptive regression splines, Computational Statistics & Data Analysis 81 (2015) 89–106. doi:10.1016/j.csda.2014.07.015.

[10] C. Cervellera, D. Macciò, R. Marcialis, Function learning with local linear regression models: an analysis based on discrepancy, in: The 2013 International Joint Conference on Neural Networks (IJCNN), 2013, pp. 1–8. doi:10.1109/IJCNN.2013.6706802.

[11] C. Cervellera, D. Macciò, A comparison of global and semi-local approximation in t-stage stochastic optimization, European Journal of Operational Research 208 (2) (2011) 109–118.

[12] C. Cervellera, D. Macciò, M. Muselli, Functional optimization through semilocal approximate minimization, Operations research 58 (5) (2010) 1491–1504.

[13] C. Cervellera, V. C. Chen, A. Wen, Optimization of a large-scale water reservoir network by stochastic dynamic programming with efficient state space discretization, European Journal of Operational Research 171 (3) (2006) 1139–1151.

[14] V. L. Pilla, J. M. Rosenberger, V. Chen, N. Engsuwan, S. Siddappa, A multivariate adaptive regression splines cutting plane approach for solving a two-stage stochastic programming fleet assignment model, European Journal of Operational Research 216 (1) (2012) 162–171.

[15] R. Chen, T. Deng, S. Huang, R. Qin, Optimal crude oil procurement under fluctuating price in an oil refinery, European Journal of Operational Research 245 (2) (2015) 438–445.

[16] G. Bayraksan, D. P. Morton, Assessing solution quality in stochastic programs, Mathematical Programming 108 (2-3) (2006) 495–514.

[17] J. Linderoth, A. Shapiro, S. Wright, The empirical behavior of sampling methods for stochastic programming, Annals of Operations Research 142 (1) (2006) 215–241.

[18] T. Ouarda, J. Labadie, Chance-constrained optimal control for multireservoir system optimization and risk analysis, Stochastic Environmental Research and Risk Assessment 15 (3) (2001) 185–204.

[19] W. Jiekang, Z. Jianquan, C. Guotong, Z. Hongliang, A hybrid method for optimal scheduling of short-term electric power generation of cascaded hydroelectric plants based on particle swarm optimization and chance-constrained programming, IEEE Transactions on Power Systems 23 (4) (2008) 1570–1579.

[20] Y. Zeng, X. Wu, C. Cheng, Y. Wang, Chance-constrained optimal hedging rules for cascaded hydropower reservoirs, Journal of Water Resources Planning and Management 140 (7) (2013) 04014010.

[21] W. Van Ackooij, R. Henrion, A. Möller, R. Zorgati, Joint chance constrained programming for hydro reservoir management, Optimization and Engineering 15 (2) (2014) 509–531.

[22] A. Seifi, K. W. Hipel, Interior-point method for reservoir operation with stochastic inflows, Journal of Water Resources Planning and Management 127 (1) (2001) 48–57.

[23] Y. Lee, S.-K. Kim, I. H. Ko, Two-stage stochastic linear programming model for coordinated multi-reservoir operation, in: Operating Reservoirs in Changing Conditions, ASCE, 2006, pp. 400–410.

[24] W. Xu, C. Zhang, Y. Peng, G. Fu, H. Zhou, A two stage bayesian stochastic optimization model for cascaded hydropower systems considering varying uncertainty of flow forecasts, Water Resources Research 50 (12) (2014) 9267–9286. doi:10.1002/2013WR015181.

[25] D. Etkin, P. Kirshen, D. Watkins, C. Roncoli, M. Sanon, L. Some, Y. Dembele, J. Sanfo, J. Zoungrana, G. Hoogenboom, Stochastic programming for improved multiuse reservoir operation in burkina faso, west africa, Journal

of Water Resources Planning and Management 141 (3) (2013) 04014056. doi:10.1061/(ASCE)WR.1943-5452.0000396,04014056.

[26] D. R. Kracman, D. C. McKinney, D. Watkins, L. Lasdon, Stochastic optimization of the Highland Lakes system in Texas, Journal of Water Resources Planning and Management 132 (2) (2006) 62–70.

[27] J. R. Birge, F. Louveaux, Introduction to stochastic programming, Springer Science & Business Media, 2011.

[28] D. Salinger, T. Rockafellar, Dynamic splitting: an algorithm for deterministic and stochastic multiperiod optimization, in: Stochastic Programming E-Print Series, Vol. 6, Springer Verlag, 2003.

[29] T. Pennanen, M. Kallio, A splitting method for stochastic programs, Annals of Operations Research 142 (1) (2006) 259–268.

[30] M. L. dos Santos, E. L. da Silva, E. C. Finardi, R. E. Gonçalves, Practical aspects in solving the medium-term operation planning problem of hydrothermal power systems by using the progressive hedging method, International Journal of Electrical Power & Energy Systems 31 (9) (2009) 546–552.

[31] L. Zéphyr, P. Lang, B. F. Lamond, Adaptive monitoring of the progressive hedging penalty for reservoir systems management, Energy Systems 5 (2) (2014) 307–322.

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

[33] C. J. Donohue, J. R. Birge, The abridged nested decomposition method for multistage stochastic linear programs with relatively complete recourse, Algorithmic Operations Research 1 (1) (2006) 20–30.

[34] B. Faber, J. Stedinger, Reservoir optimization using sampling sdp with ensemble streamflow prediction (ESP) forecasts, Journal of Hydrology 249 (1) (2001) 113–133.

[35] Y.-O. Kim, H.-I. Eum, E.-G. Lee, I. H. Ko, Optimizing operational policies of a Korean multireservoir system using sampling stochastic dynamic programming with ensemble streamflow prediction, Journal of Water Resources Planning and Management 133 (1) (2007) 4–14.

[36] Y. Lee, S. Kim, I. Ko, Multistage stochastic linear programming model for daily coordinated multi-reservoir operation, Journal of Hydroinformatics 10 (1) (2008) 23–41.

[37] H. Heitsch, W. Römisch, Scenario tree modeling for multistage stochastic programs, Mathematical Programming 118 (2) (2009) 371–406.

[38] T. Homem-de Mello, V. L. De Matos, E. C. Finardi, Sampling strategies and stopping criteria for stochastic dual dynamic programming: a case study in long-term hydrothermal scheduling, Energy Systems 2 (1) (2011) 1–31.

[39] A. Shapiro, Analysis of stochastic dual dynamic programming method, European Journal of Operational Research 209 (1) (2011) 63–72.

[40] J. A. Tejada-Guibert, S. A. Johnson, J. R. Stedinger, The value of hydrologic information in stochastic dynamic programming models of a multireservoir system, Water Resources Research 31 (10) (1995) 2571–2579.

[41] Q. Goor, R. Kelman, A. Tilmant, Optimal multipurpose-multireservoir operation model with variable productivity of hydropower plants, Journal of Water Resources Planning and Management 137 (3) (2010) 258–267.

[42] A. Turgeon, Stochastic optimization of multireservoir operation: The optimal reservoir trajectory approach, Water Resources Research 43 (5) (2007) W05420.

[43] A. Tilmant, D. Pinte, Q. Goor, Assessing marginal water values in multipurpose multireservoir systems via stochastic programming, Water Resources Research 44 (12) (2008) W12431.

[44] S. J. Mousavi, M. Karamouz, M. B. Menhadj, Fuzzy-state stochastic dynamic programming for reservoir operation, Journal of Water Resources Planning and Management.

[45] T. Zhao, X. Cai, X. Lei, H. Wang, Improved dynamic programming for reservoir operation optimization with a concave objective function, Journal of Water Resources Planning and Management 138 (6) (2011) 590–596.

[46] B. F. Lamond, P. Lang, Stochastic optimization of multi-reservoir systems with power plants and spillways, WIT Transactions on Ecology and the Environment 104 (2007) 31–40. doi:10.2495/RM070041.

[47] J. I. Pérez-Díaz, J. R. Wilhelmi, L. A. Arévalo, Optimal short-term operation schedule of a hydropower plant in a competitive electricity market, Energy Conversion and Management 51 (12) (2010) 2955–2966.

[48] J. F. Shapiro, *Mathematical Programming: Structures and Algorithms*, Wiley New York, 1979.

[49] M. S. Bazaraa, H. D. Sherali, C. M. Shetty, Nonlinear programming: theory and algorithms, John Wiley & Sons, 2013.

[50] M. Karamouz, H. V. Vasiliadis, Bayesian stochastic optimization of reservoir operation using uncertain forecasts, Water Resources Research 28 (5) (1992) 1221–1232.

[51] M. Bazaraa, J. Jarvis, H. Sherali, *Linear programming and network flows*, Wiley, New York, 1990.