Managing chance-constrained hydropower with reinforcement learning and backoffs

Florian Mitjana **- HEC Montréal**
Kenjy Demeester - Rio Tinto
Michel Denault - HEC Montréal

September 12th and 13th, 2022

Plan

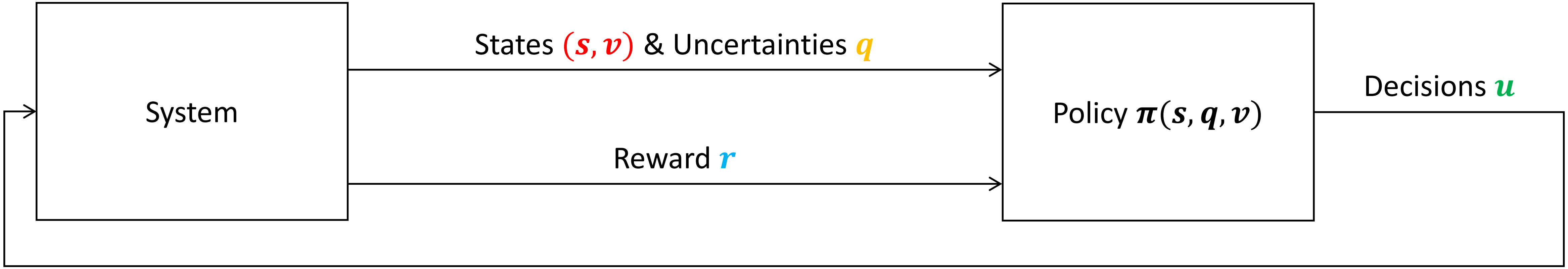
1. Introduction

2. Reinforcement learning for hydropower optimization

3. Numerical results

4. Conclusion and perspectives

Optimization of reservoir operations



Hydropower optimization :

Chance constraints on storage bounds

States:

- Endogenous: **Storage level s**
- Exogenous: **Hydrological variables v**

Decisions: **Releases $u \sim \pi(s, v)$**

Uncertainties: **Inflows q**

Reward: **Electricity production $r(s, u, q)$**

Hydropower optimization problem

$$\max_{\boldsymbol{u_0}, \dots, \boldsymbol{u_{T-1}}} \quad \mathbb{E}\left[\sum_{t=0}^{T-1} \boldsymbol{r_t}(\boldsymbol{s_t}, \boldsymbol{u_t}, \boldsymbol{q_t})\right]$$

$$s.t. \qquad \boldsymbol{s_{t+1}} = \boldsymbol{s_t} + \boldsymbol{q_t} - \boldsymbol{u_t} \quad \forall \boldsymbol{t} = \boldsymbol{0}, \dots, \boldsymbol{T-1}$$

Hard constraints $\boxed{\boldsymbol{u_t} \sim \boldsymbol{\pi}(\boldsymbol{s_t}, \boldsymbol{q_t}, \boldsymbol{v_t}) \in \mathbb{U}_t} \ \forall \boldsymbol{t} = \boldsymbol{0}, \dots, \boldsymbol{T-1}$

Release bounds:

$$\mathbb{U}_t = \left\{\boldsymbol{u_t} \mid \boldsymbol{u_t} \in \left[\underline{\boldsymbol{u_t}}; \overline{\boldsymbol{u_t}}\right]\right\}$$

Chance constraint $\boxed{\mathbb{P}\left(\bigcap_{t=1}^{T}\{\boldsymbol{s_t} \in \mathbb{S}_t\}\right) \geq 1 - \alpha}$

Storage bounds:

$$\mathbb{S}_t = \left\{\boldsymbol{s_t} \mid \boldsymbol{s_t} \in \left[\underline{\boldsymbol{s_t}}; \overline{\boldsymbol{s_t}}\right]\right\}$$

Stochastic Dynamic Programming (most common method)

1. Discretization of the endogenous $\boldsymbol{s_t}$ and exogenous $\boldsymbol{v_t}$ state spaces

2. **Policy building**: Solve the problem from $t = T-1$ to $t = 0$ (**backward**) for each point of the grid:

$$f_t(\boldsymbol{s_t}, \boldsymbol{v_t}) = \max_{\boldsymbol{u_t}} \ \mathbb{E}_{\boldsymbol{q_t}|\boldsymbol{v_t}} [\boldsymbol{r_t}(\boldsymbol{s_t}, \boldsymbol{u_t}, \boldsymbol{q_t}) + f_{t+1}(\boldsymbol{s_{t+1}}, \boldsymbol{v_{t+1}})]$$
$$s.t. \quad \boldsymbol{s_{t+1}} = \boldsymbol{s_t} - \boldsymbol{u_t} + \boldsymbol{q_t}$$
$$\boldsymbol{u_t} \in \mathbb{U}_{\boldsymbol{t}}$$
$$\boldsymbol{s_{t+1}} \in \mathbb{S}_{\boldsymbol{t+1}}$$

3. For the current states $(\boldsymbol{s_t}, \boldsymbol{v_t})$ and inflows $\boldsymbol{q_t}$, the release $\boldsymbol{u_t}$ is given by:

$$\operatorname*{argmax}_{\boldsymbol{u_t}} \ \boldsymbol{r_t}(\boldsymbol{s_t}, \boldsymbol{u_t}, \boldsymbol{q_t}) + f_{t+1}(\boldsymbol{s_{t+1}}, \boldsymbol{v_{t+1}})$$
$$s.t. \quad \boldsymbol{s_{t+1}} = \boldsymbol{s_t} - \boldsymbol{u_t} + \boldsymbol{q_t}$$
$$\boldsymbol{u_t} \in \mathbb{U}_{\boldsymbol{t}}$$
$$\boldsymbol{s_{t+1}} \in \mathbb{S}_{\boldsymbol{t+1}}$$

1

Computational effort increases exponentially with the precision of the grid

⇩

Curse of dimensionality

2

If $\boldsymbol{s_{t+1}} \notin \mathbb{S}_{\boldsymbol{t+1}}$, penalization of the objective function through a hyperparameter η

$$-\eta \sum_{t=0}^{T-1} \max(0, s_{t+1} \in \mathbb{S}_{\boldsymbol{t+1}})$$

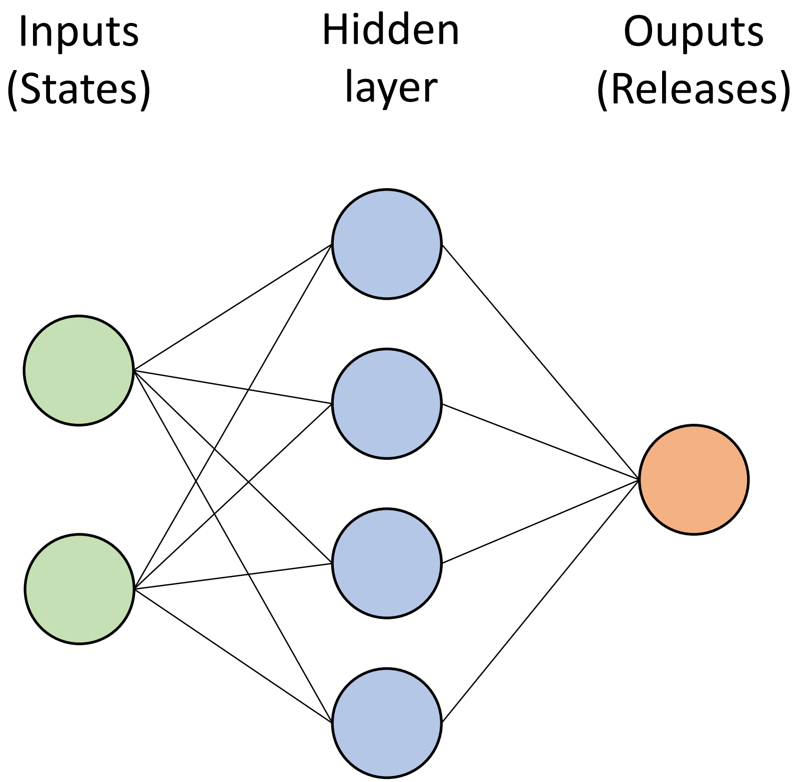
⇩

- *Tuning of the hyperparameter η*
- *Global penalization: decrease performances*

Reinforcement learning - Policy gradient method

Policy π_θ defined by a neural network

θ: parameters of neural network



REINFORCE method:

Generate a set of Monte Carlo simulations (**forward procedure**)

Compute the objective function:

$$J(\pi_\theta) = \mathbb{E}\left[\sum_{t=0}^{T-1} \boldsymbol{r}_t(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{q}_t)\right]$$

Compute the associated gradients: $\nabla_\theta J(\pi_\theta)$

Update the policy π_θ

Dealing with the chance constraint

Estimation

Chance constraint:

$$F = \mathbb{P}\left(\bigcap_{t=1}^{T}\{\boldsymbol{s_t} \in \mathbb{S}_t\}\right) \geq 1 - \alpha$$

With F cumulative distributive function (CDF)

Estimation via Empirical CDF:

$$F \approx \hat{F}_S = \frac{1}{S}\sum_{k=1}^{S}\mathbb{1}\left(\bigcap_{t=1}^{T}\{\boldsymbol{s_t^k} \in \mathbb{S}_t\}\right)$$

$$\hat{F}_S \sim \frac{1}{S}\mathcal{B}(S, F)$$

Policy building with **backoffs**

Tighten the feasible set $\mathbb{S}_t$ with backoffs $\boldsymbol{b}$:

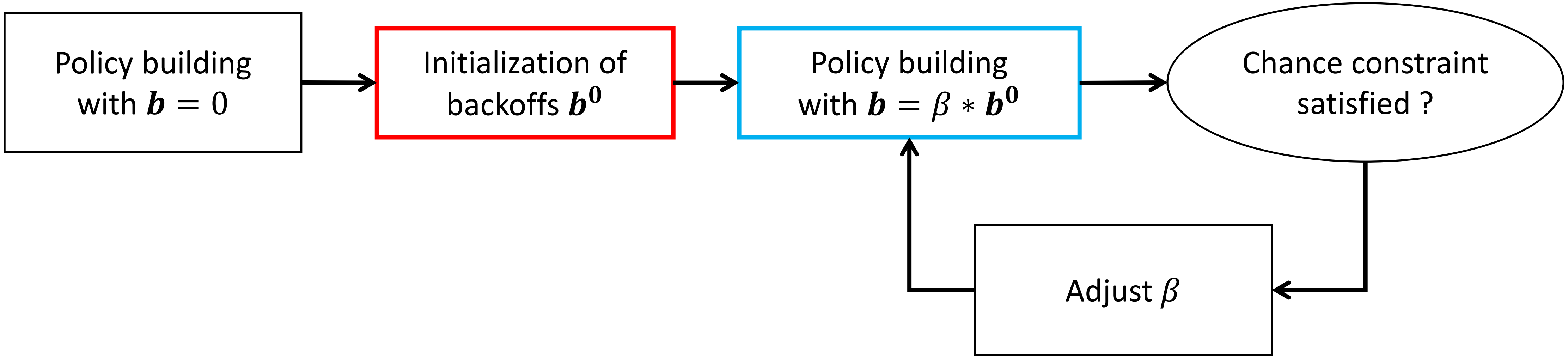
$$\overline{\mathbb{S}}_{\boldsymbol{t}} = \left\{\boldsymbol{s_t} \mid \boldsymbol{s_t} \in \left[\underline{\boldsymbol{s_t}} + \boldsymbol{b_{1,t}} \,;\, \overline{\boldsymbol{s_t}} - \boldsymbol{b_{2,t}}\right]\right\}$$

Adding control parameters

New objective function:

$$J(\pi_\theta) = \mathbb{E}\left[\sum_{t=0}^{T-1}\boldsymbol{r_t}(\boldsymbol{s_t}, \boldsymbol{u_t}, \boldsymbol{q_t}) - \kappa\sum_{t=0}^{T-1}\max(0, \boldsymbol{s_{t+1}} \in \overline{\mathbb{S}_{\boldsymbol{t+1}}})\right]$$

Policy building algorithm



How can we initialize backoffs values ?

Based on simulations, we can estimate the initial backoffs $\boldsymbol{b^0}$ to satisfy the chance constraint

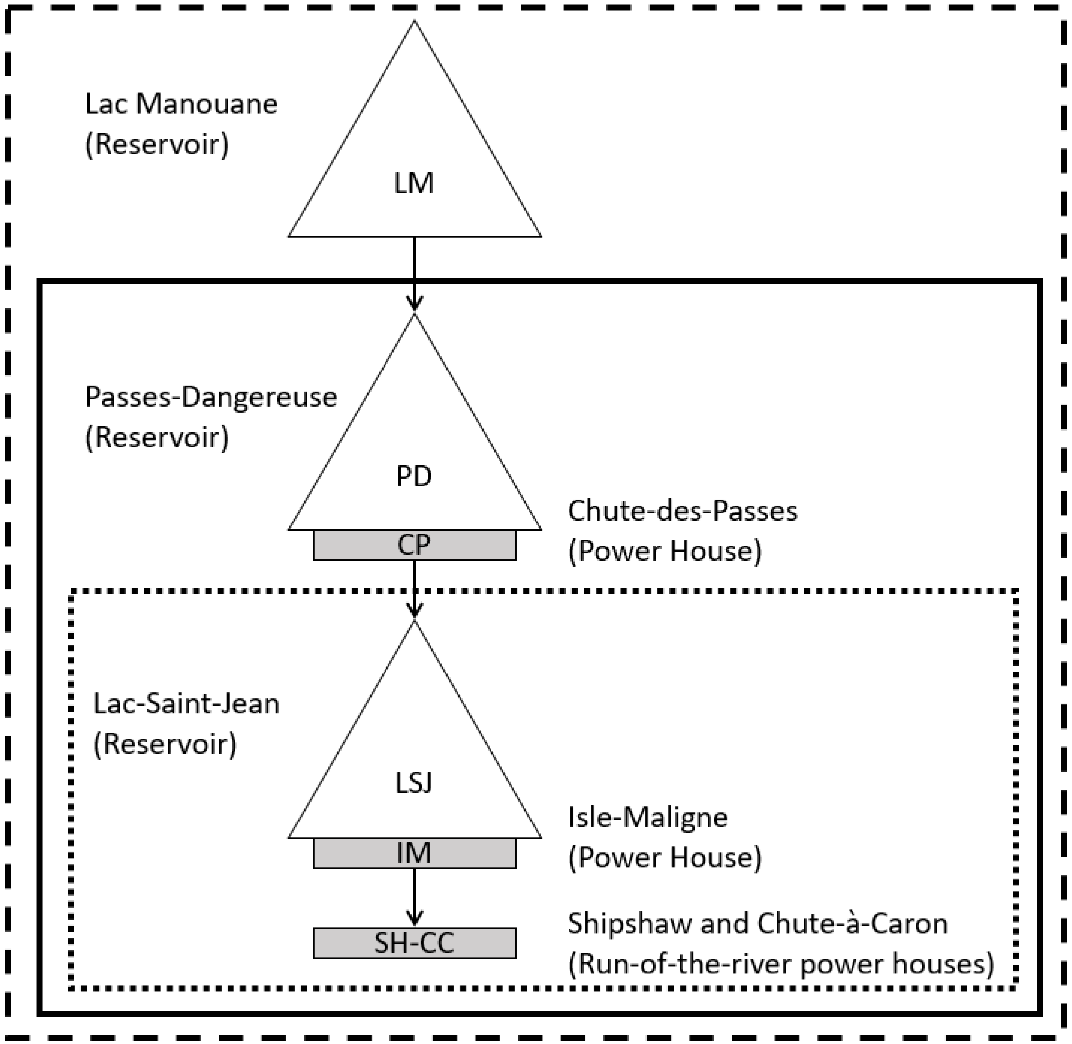
How can we adjust the backoffs values ?

Introduce a scalar parameter β and define:

$$\boldsymbol{b} = \beta * \boldsymbol{b^0}$$

Case study for Rio Tinto Quebec System

Saguenay-Lac-Saint-Jean basin (Québec – Canada)



Three configurations:

- ...... One-reservoir configuration
- ▬ Two-reservoir configuration
- – – Three-reservoir configuration

Benchmark method : SDP

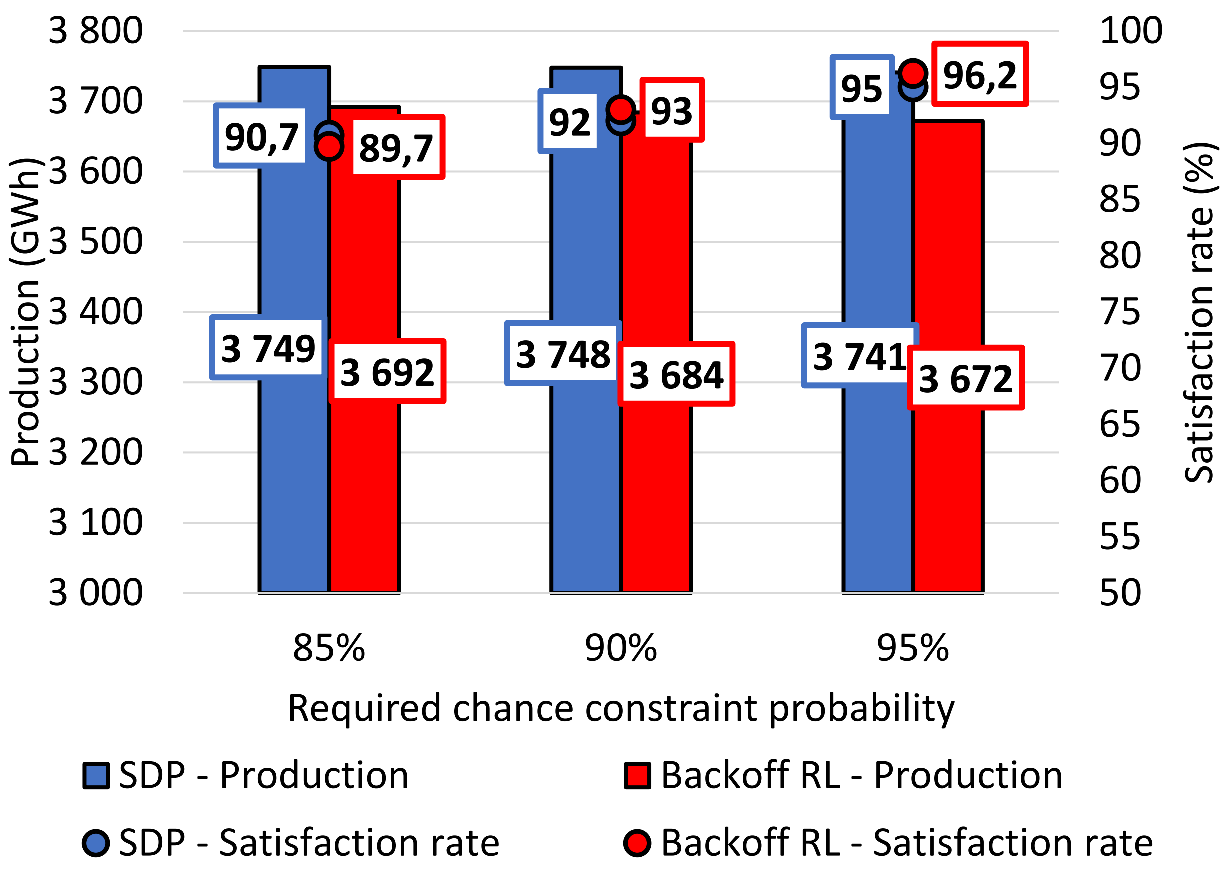
Set of 1000 scenarios:

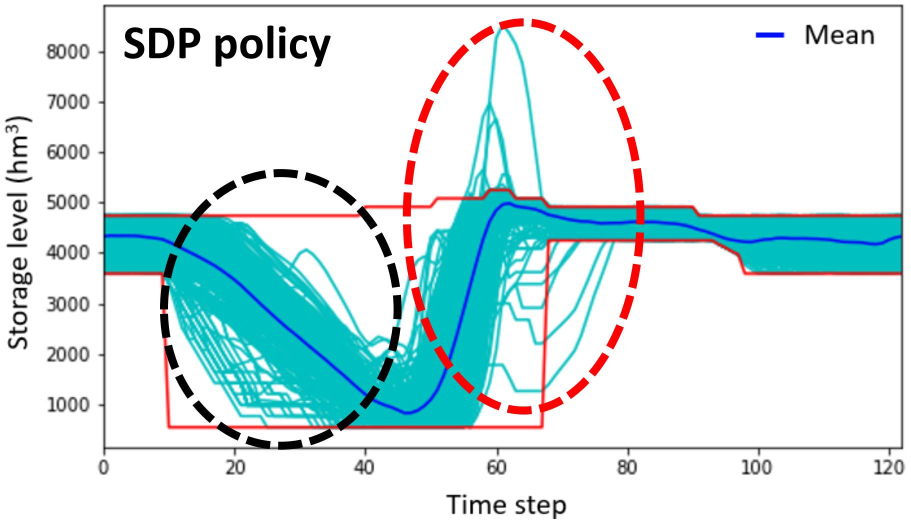
- 700 for the policy optimization
- 300 for the policy evaluation

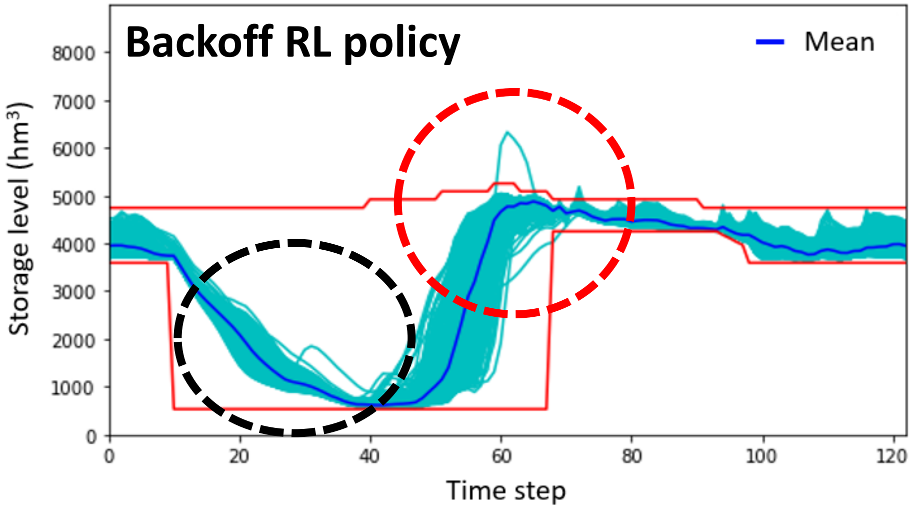
Satisfaction rate :

Percentage of the 300 episodes for which the constraints are satisfied

One-reservoir configuration: Production

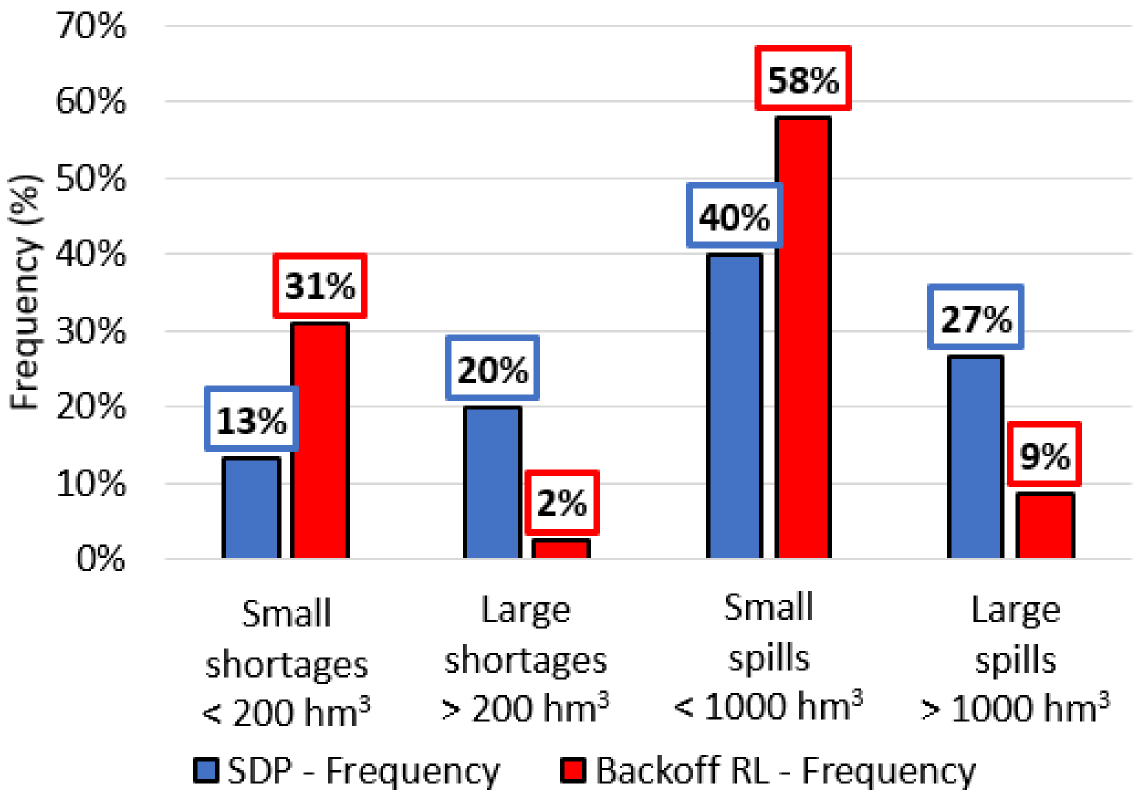






Decrease of the production of ~2%

One-reservoir configuration: shortages and spills



In average	SDP	RL
Small shortages $(\mathbf{hm^3})$	25	41
Large shortages $(\mathbf{hm^3})$	13 574	329
Small spills $(\mathbf{hm^3})$	106	103
Large spills $(\mathbf{hm^3})$	5 158	4 329

One-reservoir configuration conclusions:

− Backoff RL decreases of the production of ~2%

+ Backoff RL reduces substantially water shortages and spills

Two- and three-reservoir configurations: Production

Two-reservoir configuration

	Production (GWh)	Satisfaction rate
SDP Discretization: 5	5 398	71.3 %
SDP Discretization: 6	5 522	67.3 %
SDP Discretization: 7	5 574	71.0 %
SDP Discretization: 8	5 641	78.7 %
Backoff RL	5 674	96.5 %

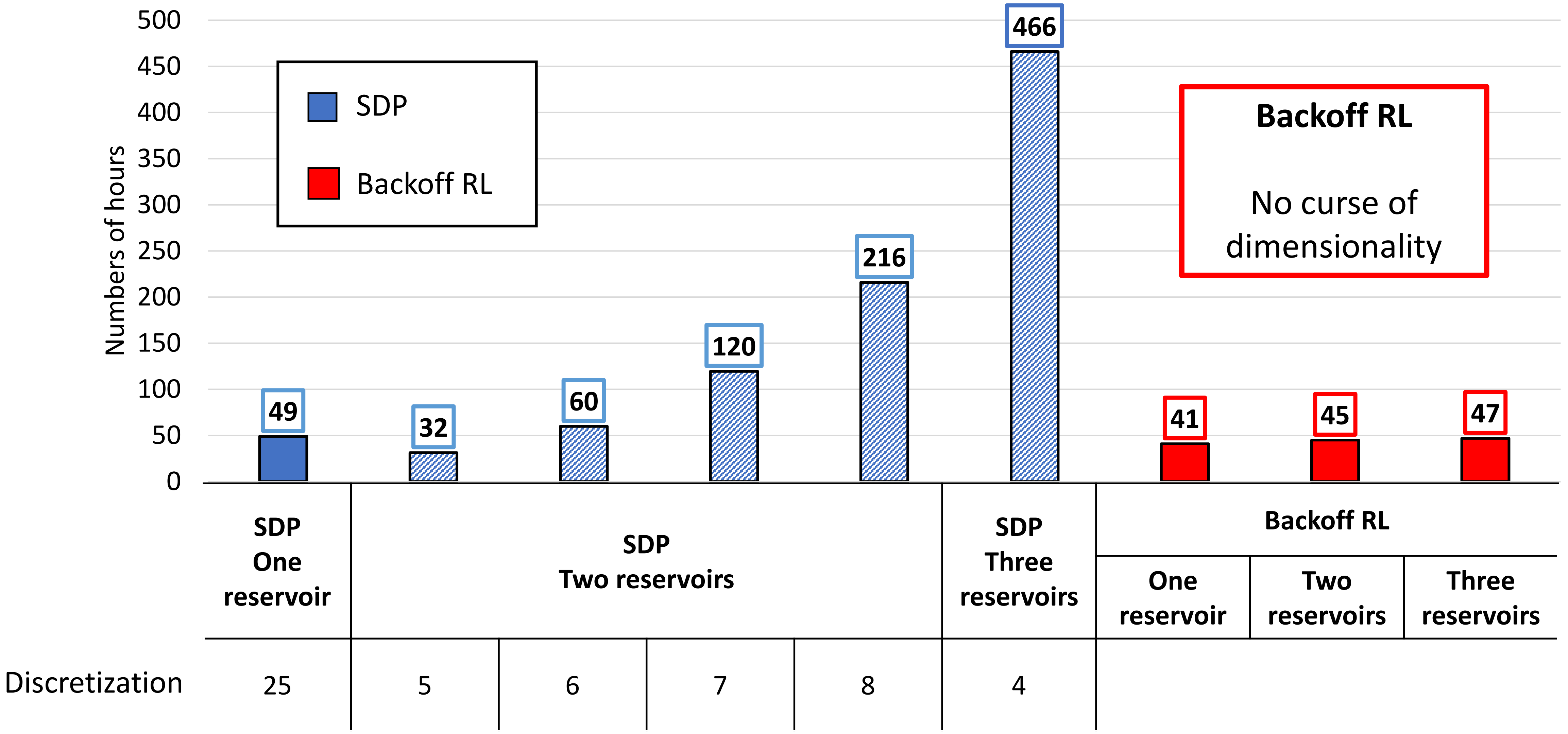
Three-reservoir configuration

	Production (GWh)	Satisfaction rate
SDP Discretization: 4	5 075	54.5 %
Backoff RL	5 705	96.9 %

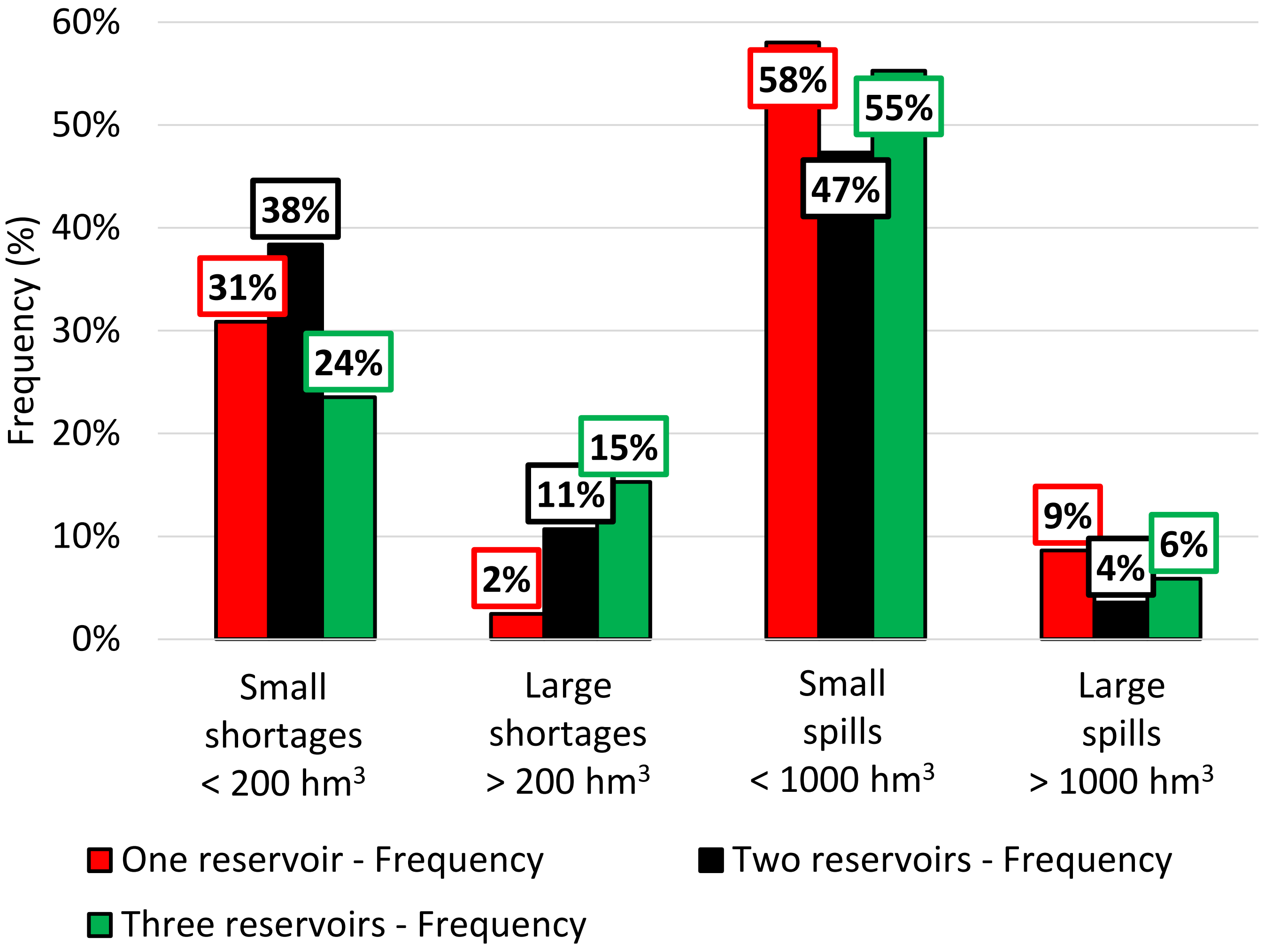
Backoff RL

Parameters needed no adjustment to perform adequately

Computational time in hours



Backoff RL: Shortages and spills for all configurations



Average $(\mathbf{hm^3})$	**One Reservoir**	**Two Reservoir**	**Three Reservoir**
Small shortages	41	41	55
Large shortages	329	1 142	731
Small spills	103	98	93
Large spills	4 329	2 098	1 548

Conclusion on hydropower optimization and RL

- **No curse of dimensionality**

  Computational time remains stable while the number of reservoirs increases

- **Electricity production equivalent** with a benchmark method

- **Substantial reduction of water shortages and spills**

Accepted article in Advances in Water Ressources (manuscript in production)

Perspectives

- **Add exogenous variables**

  Soil moisture levels or weather forecasts

- **Introduce another chance constraints**

  Minimum power production at each time step

- **Improve computational time**:

  Parallelization

Thanks for your attention

Precision on the chance constraint

Chance constraint:

$$F = \mathbb{P}\left(\bigcap_{t=1}^{T}\{\boldsymbol{s_t} \in \mathbb{S}_t\}\right) \geq 1-\alpha$$

With F cumulative distributive function (CDF)

Estimation via Empirical CDF:

$$F \approx \hat{F}_S = \frac{1}{S}\sum_{k=1}^{S}\mathbb{1}\left(\bigcap_{t=1}^{T}\{\boldsymbol{s_t^k} \in \mathbb{S}_t\}\right)$$

$$\hat{F}_S \sim \frac{1}{S}\mathcal{B}(S, F)$$

Confidence interval on F:

$$\mathbb{P}\left(F \geq \hat{F}_{lb}\right) > 1-\epsilon$$

where $\hat{F}_{lb} = betainv\left(\epsilon; S\hat{F}_S, S - S\hat{F}_S + 1\right)$

With $\hat{F}_{lb} \geq 1-\alpha$:

$$\mathbb{P}(F \geq 1-\alpha) > 1-\epsilon$$