POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA CIVILE,
AMBIENTALE E TERRITORIALE

Advancing water reservoir operations via multi-objective reinforcement learning

Tesi di Laurea Magistrale in
Environmental and Land Planning Engineering -
Ingegneria per l'Ambiente e il Territorio

Author: **Emiliano Longo**

Student ID: 996381
Advisor: Prof. Andrea Castelletti
Co-advisors:
Prof. Matteo Giuliani
Davide Spinelli
Academic Year: 2022-2023

Abstract

Water reservoir control presents numerous challenges, particularly with the rise of integrated management approaches that consider multiple aspects of the water system simultaneously. Additionally, in recent decades, optimizing adaptive control strategies has become a crucial topic due to rapidly changing hydrological drivers. The state of the art for reservoir operations is rapidly evolving, thanks to increasing computational power and the development of more efficient methods. Historically, problems have been simplified to fit the constraints of specific solution methods, often failing to address the complexities that real-world water systems exhibit.

Our work focuses on the implementation of the Multi-Objective Fitted Q Iteration (MOFQI) algorithm to find Pareto-optimal reservoir control policies. MOFQI is an offline, model-free, and multi-objective control algorithm. To do this we developed a framework that simulates a part of the water system to generate the dataset. We found that exploring the simplex surface randomly keeps the dataset size small and still allows the algorithm to converge in the objectives space. To estimate the Q-function we used an Extra-Trees regressor, the model trained in a single run potentially generates a continuous approximation of the Pareto front. We evaluated control performance using the hypervolume metric on training and validation trajectory to asses the hyperparameters of the algorithm. The framework is tested on the optimal operation of Lake Como, where three conflicting objectives drive lake regulation: flood and drought control and water supply. We benchmarked MOFQI against Stochastic Dynamic Programming (SDP). Empirical results demonstrate the competitiveness of the proposed framework with state-of-the-art methods in optimal control of the water reservoir. Most of the computational burdens that afflict SDP are overcome by MOFQI. Moreover, this method generates a widespread range of tradeoff Pareto optimal policies by nullifying the curse of multiple objectives. Our research demonstrates how the framework can be enhanced to potentially integrate with larger water systems, given its efficient computing ability that considers multiple states and available external information, such as the day of the year or hydrologic forecasts.

Keywords: Reservoir Operation; Reinforcement Learning; Multi-Objective optimization; Multi-Objective Fitted Q Iteration

Abstract in lingua italiana

Il controllo dei bacini idrici presenta numerose sfide, in particolare con l'aumento di approcci di gestione integrata che considerano simultaneamente molti aspetti del sistema idrico. Inoltre, negli ultimi decenni, l'ottimizzazione di strategie di controllo adattivo è diventato un'argomento cruciale per via dei rapidi cambiamenti dei fattori idrologici. Lo stato dell'arte per le operazioni dei bacini idrici è in rapida evoluzione grazie all'aumento della potenza computazionale e allo sviluppo di metodi più efficienti. Storicamente i problemi sono sempre stati semplificati per soddisfare i vincoli dei metodi di risoluzione, spesso fallendo nell'affrontare le complessità che caratterizzano i sistemi idrici reali.
Il nostro lavoro si concentra sull'implementazione dell'algoritmo Multi-Objective Fitted Q Iteration (MOFQI) per trovare politiche Pareto-ottimali per il controllo dei bacini idrici. Per fare ciò, abbiamo sviluppato una procedura che simula una parte del sistema idrico per generare il dataset. Abbiamo scoperto che l'esplorazione distribuita casualmente sulla superficie del simplesso mantiene ridotte le dimensioni del dataset e permette comunque all'algoritmo di convergere nello spazio degli obiettivi. Per stimare la funzione Q abbiamo usato come regressori gli Extra-Trees, il modello calibrato in un'unica esecuzione potenzialmente genera un'approssimazione continua della frontiera di Pareto. Abbiamo valutato la prestazione del controllo con la metrica dell'ipervolume sulle traiettorie di addestramento e validazione per determinare gli iperparametri dell'algoritmo. La procedura è testata per il controllo ottimale del Lago di Como in cui tre obiettivi in conflitto guidano la regolazione del lago: il controllo delle esondazioni e delle secche e la fornitura d'acqua. Abbiamo confrontato MOFQI con l'algoritmo Programmazione Dinamica Stocastica (PDS). I risultati empirici dimostrano la competitività della procedura proposta con i metodi allo stato dell'arte per il controllo ottimale dei bacini idrici. La maggior parte dei limiti computazionali che affliggono PDS vengono risolti da MOFQI. Inoltre questo metodo è in grado di generare un'ampia gamma di politiche di compromesso Pareto ottimali annullando il vincolo legato alla formulazione multi-obiettivo. Questo studio dimostra come questa procedura possa essere migliorata e potenzialmente implementata in sistemi idrici più grandi, grazie alla sua efficienza computazionale nel considerare più stati e le informazioni esterne disponibili, come ad esempio il giorno dell'anno o le previsioni

idrologiche.

Parole chiave: Gestione dei bacini idrici; Apprendimento per Rinforzo; Ottimizzazione Multi Obiettivo; Multi-Objective Fitted Q Iteration

Contents

1 | Introduction

In a changing climate, reservoir operations are becoming more and more relevant for regulating the water cycle and adapting to increasingly impactful floods and droughts as well as for achieving multi-sectoral agreements and supporting regional socio-economic growth (Yun et al., 2020; Ehsani et al., 2017; Padowski et al., 2015).

Access to freshwater is limited, and its regulation must account for population and economic growth. As a result, water availability constraints are increasingly seen as a limiting factor in meeting basic human needs for safe water and sanitation. The number of people adversely affected by droughts globally between 2000 and 2019 is estimated to be 1.43 billion; in the same period, 1.65 billion people were adversely affected by floods (Gleick and Palaniappan, 2010; Browder et al., 2021).
Already almost 50% of the existing rivers are controlled through a dam, still, construction and development of new dams are underway, mainly in countries with emerging economies, as a response to the need to close the electricity access gap to the population growth and economic development. New dam design and operation are currently challenging tasks to secure better water and energy supplies (Grill et al., 2019; Zarfl et al., 2015; Bertoni et al., 2019).

Water systems management usually serves specific purposes, which dictate how the reservoir is operated and when it stores or releases water. Often unplanned multiple water uses are caused by socio-economic driver change, such as environmental awareness, individual human responses in the uptake of measures, changes in water legislation, land use change, energy demand affecting hydropower value, and increasing pressure on water resources due to withdrawal from different growing sectors. Reservoir operations inevitably must deal with conflicting objectives, transboundary river management disputes, and multiple water uses. This context involves trade-offs in reservoir operation design, making it more complicated to find a solution to adopt within a set of options that aim to maximize their performance for all water uses (Benson, 2019; Hossen et al., 2023; Castelletti et al., 2008b; O'Connell, 2017).
Different approaches are constantly under development to face the full complexity of water

resources planning and management tasks, such as Participatory and Integrated Planning (Castelletti and Soncinisessa, 2006), an integrated approach to support decision-making accounting for all the stakeholders, physical and non-physical conditions and constraints. Innovative strategies and approaches are helping to consolidate a paradigm shift in water resources management, which is moving towards Integrated Water Resources Management, already adopted by many institutions, aiming at a more comprehensive understanding and balance between bottom-up and top-down processes (Soncini-Sessa et al., 2014; Pahl-Wostl et al., 2011).

Further complexity afflicting water reservoir operations is due to climate change that has affected planet Earth in recent decades with a steady increase in temperature on the land and ocean surface. Climate patterns are changing rapidly and these carry deep consequences for the environment. Hydrological systems are being affected in numerous ways: climate and weather extremes increase occurrences of floods in certain regions and harsher droughts for longer periods in other regions, glacier mass loss contributes to reducing freshwater availability, substantial damages affect freshwater ecosystems, water scarcity makes land cover becoming more inclined to desertification, and increases in frequency and intensity of extremes has reduced food and water security. The use and management of water afflict ecosystem and human vulnerability, and if vulnerability increases, it leads to competition for water resources (IPCC, 2022).

Policymakers need to face population growth, economic development, and water-related conflicts but also need to adapt reservoir control policies to the changing hydrological conditions. Long-term operating plans must be revisited and adapted to climate change to mitigate its effects. Among the most important objectives is the adaptation to the increased variability of hydro-meteorological quantities, which is to be pursued by building flexible infrastructures or adapting the management of existing ones (Damania et al., 2017; Benson, 2019; Garrote, 2017). In particular, it is demonstrated that adaptive management mitigates the effects due to changes in the hydrological scenario, compared to the use of heuristic regulation policies based on historical operation experience. The design of new policies that take into account the changing drivers of the system and that can operate the management of reservoir under a broader variation in hydrological drivers is fundamental for adaptive management (Pahl-Wostl, 2006; Georgakakos et al., 2012).

Mathematical methodologies allow us to describe all those processes quantitatively and define decision-making and management procedures that govern them. All the mentioned studies contribute to deepening our understanding of real-world system dynamics and advancing the performance of water reservoir control (Soncini-Sessa et al., 2014).

1.1. Thesis Contribution

Many works in the water reservoir operation domain aim to find an optimal operating policy. Advances in algorithms and computational efficiency allow us to address all facets of the problem more comprehensively. However, many aspects are still challenging tasks to be tackled, such as multi-objective purpose and uncertainty that affect environmental systems (Labadie, 2004; Giuliani et al., 2021; Hejazi et al., 2008).
A key feature of the approaches used to solve control problems is their scalability with respect to the number of objectives and states of the system. This characteristic makes it possible to generate tailor-made solutions that are best suited to each specific and complex case of the real-world system. Moreover, by being able to exploit the massive amount of available data from the widespread monitoring system and forecasts, which are becoming more skillful, control performances gain a substantial improvement in reservoir operations (Giuliani et al., 2021; Bauer et al., 2015; Giuliani et al., 2015).

Within this context, this thesis work contributes the implementation of a methodological approach for computing an estimate of the optimal control policies for a real case study, testing the feasibility of the method on a challenging real-world problem. The solving method is set up to be optimized with multiple objectives but also to be able to address multiple system states and many exogenous variables. The system dynamics is assumed to be a Markov Decision Process, and the optimal control problem is solved via reinforcement learning using the model-free, multi-objective, and offline algorithm Multi-Objective Fitted Q Iteration proposed by Castelletti et al. (2013), which partially mitigates the curses that afflict the original problem solution, alleviating the computational cost.
In particular, the main aspects that are investigated in the thesis are:

1. The training set generation. Different sampling techniques are tested aiming to reduce the computational cost.

2. The resulting performances with respect to the regression model involved in the algorithm and the discount rate parameter, also evaluating the algorithm convergence in a multi-objective framework.

3. The benchmarking of MOFQI to the state-of-the-art Stochastic Dynamic programming (SDP) algorithm, which contributes to assessing the feasibility and effectiveness of this method and also its competitiveness with the state-of-the-art in water reservoir management.

1.2. Thesis Outline

Chapter 1 Explains the context in which the current thesis work is developed by providing a more detailed and in-depth understanding of the motivations and needs that drive research on the topics covered, it also illustrates the contributions this work makes concerning the state-of-the-art, by describing the core of the work and the idea that is behind it.

Chapter 2 Introduces the state-of-the-art through a literature review on the water resources optimal control topic, giving a more detailed view of reinforcement learning with insights on its application in water resources management.

Chapter 3 Describes the methods and tools used, especially the detailed formulation of the problem, the Multi-Objective Fitted Q Iteration algorithm, the methods employed to generate the training set, and the approach used for evaluating the results in the context of multi-objective optimization, as well as how to assess the convergence of the algorithm.

Chapter 4 Describes the Lake Como case study and presents the characteristics of the system involved and its hydrological basin, together with the physical-based lake model used. This chapter presents stakeholder and regulation purposes and defines the objectives considered for this case study; in the end, presents the historical observations used in this study.

Chapter 5 Presents the numerical results of the control optimization procedures, introducing and commenting step-by-step the results. Provides insights on the optimized control laws and a detailed description of the sensitivity analysis. Finally, it shows the benchmarking of the Multi-Objective Fitted Q Iteration method with the state-of-the-art.

Chapter 6 Sums up the conclusions and suggests some new ideas and directions for further research about the topic.

2 | State of the art

2.1. Water Reservoir Optimal Control

The state-of-the-art for optimal water reservoir operations is rapidly evolving. Over the years, several approaches have been developed to address numerous challenges, in synergy with increasing computing power, more realistic and decision-relevant problem formulations could now be addressed, reducing simplifications and technical constraints (Labadie, 2004; Giuliani et al., 2021; Rupp, 2020).

Reservoir operation problems have traditionally been simplified to match the constraints of specific solution methods and computational capacities. These simplifications typically involve dividing the reservoir system into smaller, more manageable parts or treating it as a single equivalent unit. A variety of techniques, such as Linear Programming and Linear Quadratic Gaussian control, are then applied. However, these methods often fall short in addressing the complexities and non-linearity of real-world water systems (Giuliani et al., 2021; Barros et al., 2003).
Simplifications also extend to the optimal policy frame, either through rule curves that guide reservoir storage, assuming to be under normal conditions, or through an open-loop sequence of pre-planned release decisions, schematized as in fig. 2.1a. Although commonly used, rule curves have limitations, especially under changing hydroclimatic conditions, potentially leading to ineffective water management. On the other hand, open-loop decisions, based on deterministic assumptions, prove impractical in the real world due to their inability to adapt to unexpected changes in water inflow. Moreover, the lack of feedback information does not leverage the potential for adaptive and coordinated strategies, especially in multi-reservoir systems managing multiple conflicting objectives (Giuliani et al., 2021; Quinn et al., 2019).

Stochastic dynamic programming (SDP) has been studied in the water resources field since the '60s. SDP finds the optimal control for the problem formulated as a Markov decision process, which is considered one of the best formulations for capturing and representing all non-linearities of the system, as well as for closing the loop (fig. 2.1b) between

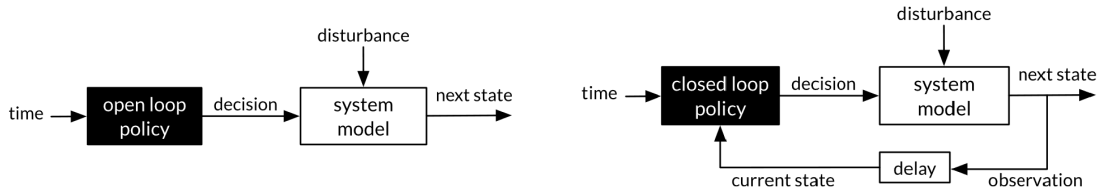


Figure 2.1: Scheme of open-loop policy (a) and closed-loop policy (b)

control decisions and the state of the system. Despite its wide use in literature and practical applications, SDP is limited by three main curses, raising a series of restrictions in addressing emerging challenges in water reservoir operation (Giuliani et al., 2021).

The three curses that afflict the SDP are:

1. The curse of dimensionality (Bellman, 1957) refers to the exponential growth in the computational cost with increasing state dimensionality in Stochastic Dynamic Programming. This issue becomes particularly evident when the system's dimensionality exceeds 2 or 3 reservoirs. It also occurs in a single reservoir when multiple states of the same system need to be included, for example, variables related to water quality or to other compartments influenced by the water dynamic (Giuliani et al., 2021; Powell, 2019; Kerachian and Karamouz, 2006).

2. The curse of modeling (Tsitsiklis and Van Roy, 1996), whereby any variable, even an exogenous one, must be modeled to be able to identify the state transition one step ahead, and thus calculate the value of the control action that minimizes the total cost. As a consequence, the inclusion of any variable amplifies the first curse (Powell, 2019).

3. The curse of multiple objectives (Cohon and Marks, 1975), involves the generation of a full set of Pareto optimal solutions for many-objective control problems, needed to support a posteriori decision-making. There are three specific limitations: the MDP formulation requirement for time-separable objective functions, the SDP requirement of repeated runs for each point of the Pareto front with different scalarization values due to the intrinsic single-objective nature of the algorithm, thereby its overall cost grows factorially with the number of objectives, and third, the exploration of diverse problem framings reflecting different risk attitudes, as the problem formulation allows only specific temporal aggregation and uncertainty

filtering criterion combinations (Giuliani et al., 2021).

Many attempts to go beyond some or all the curses aim to overcome SDP limitations, among these, two main approaches could be identified: Approximation in Value Space (AVS) and Approximation in Policy Space (APS). Both of these methods imply less restrictive problem formulation, and for that reason, their solutions are often considered suboptimal control. Despite suboptimality, approximate solutions that rely on the use of detailed simulations, broader sources of information, and a wider range of objectives may be closer to the real one, and thus more valuable for the real operators (Giuliani et al., 2021; Powell, 2019).
Based on the problem formulation adopted, fig. 2.2 summarizes the classification of more than 300 studies on reservoir operation published between 2005 and 2019.

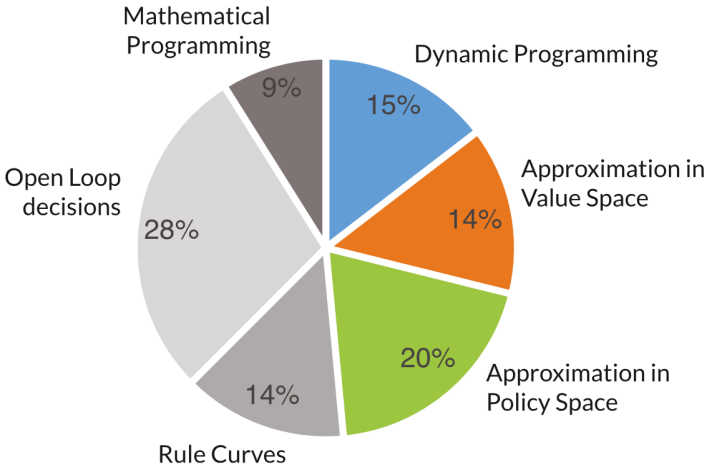


Figure 2.2: Classification depending on the adopted problem formulation of the considered studies analyzed in (Giuliani et al., 2021)

Among the methods defined as AVS, batch reinforcement learning is adopted in this thesis work. It is still not widely used in the field of water resources, but interest in this method is growing, mainly due to the RL algorithms advances and countless developments in method-related domains such as Artificial intelligence and Machine Learning (Pan, 2016; Legg and Hutter, 2007). In particular, recent advances in training deep neural networks lead to RL techniques with increasingly satisfying performances, able to solve problems with high-dimensional observation spaces and continuous actions domain, especially useful for physical control problems with a broader source of information (Bengio, 2009; Mnih et al., 2015; Lillicrap et al., 2019).

2.2. Reinforcement Learning

The idea behind reinforcement learning concerns goal-direct learning from the interaction between an agent and an environment. The agent receives feedback from the environment, gaining experience and improving control to achieve its goal.
RL problems are formalized as the optimal control for a Markov decision process; any method that is well suited to solving such problems via trial-and-error search is considered an RL method (Barto, 2018).

The experience upon which RL learns is given as a set of tuples, defined as $\langle s_t, a_t, s_{t+1}, R_{t+1} \rangle$, each one representing a single transition of the MDP. For a given state s_t at time t the agent acts a_t, this action has an effect on the environment state which, combined with the stochastic disturbance of the process, contributes to the state transitioning to a value s_{t+1}. The MDP formulation involves a reward value R_{t+1} concerning the agent behaviour, it is associated with each state transition from s_t to s_{t+1} used as a goodness measure of the action taken.

The underlying principle for learning the optimal control of an MDP is based on the so-called state-value function V (eq. 2.3a) defined as the expected value of the return G_t for a given state s, where the return G_t (eq. 2.2) is the discounted sum of the full trajectory of rewards experienced by the actor starting from the current state, with γ as discount factor.
The agent's goal is to find a policy that maximizes the V function, thus maximizing the sum of the immediate expected rewards $r(s, a)$ (eq. 2.1) and the discounted sum of the future expected return influenced by the agent's action.

$$r(s, a) = \mathbb{E}\left[R_t \,|\, s_{t-1} = s, a_{t-1} = a\right] \tag{2.1}$$

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.2}$$

$$V(s_t = s) = \mathbb{E}\left[G_t \,|\, s_t = s\right] \tag{2.3a}$$

$$V^{\pi}(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p(s' \,|\, s, \pi(s)) \cdot V^{\pi}(s') \tag{2.3b}$$

The value function is usually rewritten as eq. (2.3b), named Bellman equation (Bellman, 1957), where V^{π} depends on the policy π applied. This formulation makes explicit the

dependencies of the value function on the current expected reward and the discounted future rewards, doing so, it also explicit the transition model $p(s' \mid s, \pi(s))$ representing the transitioning probability from current state s to next state s' for the given policy $\pi(s)$ which determines the action a for the given state s.
SDP relies on this formulation to find the control that maximizes V but the transition probability model must be known.

Similarly, the action-value function Q (eq. 2.4) defines the value of taking an action a for a certain state s as the expected return starting from s, acting as a, and then following the policy π.

$$Q^{\pi}(s,a) = r(s,a) + \gamma \sum_{s' \in S} p(s' \mid s,a) \cdot V^{\pi}(s') \tag{2.4}$$

Solving an RL task could be trivially defined as finding a policy that maximizes the reward received by the agent. Given the Q function, an optimal policy formulation is the so-called greedy policy, defined by eq. (2.5). It identifies the selected action as equivalent to the one maximizing the state-action value computed in the state s, where the policy is applied.

$$\pi(s) = \arg\max_{a} Q(s,a) \tag{2.5}$$

Often, both the MDP transition model and the action-value function $Q(s,a)$ are unknown. Several methods have been developed to solve this particular case of the problem. Categorized as model-free, these approaches do not rely on the transition matrix, although, learning from the experience, they can estimate the value of Q. Once the action-value function is estimated, the optimal policy is defined as eq. (2.5).
Among the model-free methods, the most popular is Q-learning, introduced by (Watkins and Dayan, 1992). It approximates the action-value function through online learning, being represented in a tabular form the Q function is iteratively updated as the agent receives feedback from the environment. Furthermore, it is demonstrated that the Q-learning algorithm converges to the real Q function, making this method fundamental to the development of more efficient algorithms for solving more challenging problems (Barto, 2018).

Relying on Q-learning intuition, different methods overcome the tabular representation of the action-value function by estimating it through a parametric or non-parametric approximator. Taking advantage of the powerful capabilities of deep neural networks as

universal approximations, also thanks to the advances in training techniques, Mnih et al. (2015) introduces deep Q-learning (DQN) algorithm able to learn the Q function online by training the parameters of a deep neural network, achieving surprising performances. Based on this approach, many methods have been developed for addressing different challenges (Hessel et al., 2018).
A slightly different approach, which learns offline from a batch of experience tuples but is still based on the Q-learning intuition, is introduced by (Damien Ernst et al., 2005), called Fitted Q Iteration (FQI). It learns from a training set composed of several tuples as $\langle s_t, a_t, s_{t+1}, R_{t+1} \rangle$ sampled from the system assumed as an MDP and estimates the Q function using a dedicated non-parametric model with a regression tree structure. In doing so, two stages of the learning process could be identified: the dataset generation, which could be sampled from the historical trajectories or generated through a simulation model, and the batch learning process, which is still iterative learning. Also, variants that rely on the use of neural networks instead of decision trees have been developed for the batch learning approach, (Riedmiller, 2005).

A significant number of state-of-the-art RL techniques rely on deep learning models to estimate the Q function, losing the theoretical guarantees for converging to the optimal control. Nonetheless, these techniques have outperformed methods that are theoretically guaranteed to converge, additionally, their convergence and effectiveness are empirically demonstrated. Therefore, these methods are promising for their application in a wide range of different domains, including water reservoir control (Mnih et al., 2015; Xu et al., 2021).

2.3. Reinforcement Learning in Water Reservoir Operations

Reinforcement learning, especially batch and model-free RL, is still not widely adopted in the water reservoir operation domain and only some studies highlight its effectiveness in addressing the water reservoir control problem under different challenging conditions.
The experience dataset from which the algorithm learns can be a sampling of the historical trajectory, but this may fail to represent the MDP, and in addition, it forces the RL algorithm to learn from the historical control. A further step to generate a complete dataset is based on using a physical-based generative model to simulate the system component controlled by the agent. By doing so, the feedback from the environment could be explored throughout the whole state-action domain.
Using a reliable generative model allows the representation of complex physical dynamics

of the system and, thus, learning the action-value function based on a truthful representation of the real-world system, potentially leading to a comprehensive understanding of the control process and a better interpretability of the control results.
Particularly, in batch-mode RL, since the dataset generation and learning process are decoupled, adopting complex and reliable models affects the computational costs of only the dataset generation process, which could potentially be performed only once.

Different attempts were developed with both online and offline RL aiming to address three main challenges in finding optimal reservoir control: managing multiple endogenous states and actions, including multiple exogenous states, and searching optimal policies in multi-objective space.
Using the SDP solution as a benchmark, some studies present the potential of RL methods. DQN algorithm can effectively address the optimal reservoir control problem. Moreover, solutions' performance can be improved by including inflow forecast value as exogenous data (Xu et al., 2021). Multi-objective optimization with offline batch FQI algorithm can be tackled by including multiple objectives through a convex sum of them. Furthermore, multiple states of the system and also multiple controls can be addressed with the FQI algorithm (Castelletti et al., 2014, 2013, 2010).
Current state-of-the-art techniques implemented to solve the optimal control in water reservoir operations can be aligned with the overall state-of-the-art RL techniques, potentially leading, as seen for other control problems, to address many exogenous as well as endogenous states and multiple objectives with remarkable results.

3 | Methods and Tools

3.1. Water Reservoir Problem Formulation

The optimal control problem is formulated as follows (Castelletti et al., 2008a).
The goal is to find the optimal policy that minimizes the q objectives of the problem, eq. (3.1). Each J^i objective measures the performances with respect to each one of the q goals, starting from a trajectory of step costs g_t^i. Assuming that each immediate and time-separable cost function $g_t^i(\cdot)$ well represents the relevant phenomenon, each objective J^i of the problem assumes a single value to represent the interest of the decision maker by aggregating over a finite time horizon h all the step costs, as reported in eq. (3.2a).
The uncertainty filtering criterion operator ψ must reflect the attitude of the water operator to deal with uncertainties, and the temporal aggregation operator Φ is a function of all the step costs over the time horizon. In the RL framework and for an infinite horizon, these operators result as in eq. (3.2b), where ψ is the expected value and Φ is the discounted sum. Both of these operators make it possible to obtain a single target value from the step cost trajectories, thus representing control performance over the time horizon.

$$\min_{\pi} \left| J^1(\pi), \ldots, J^q(\pi) \right| \tag{3.1}$$

$$J^i = \psi_{\varepsilon \sim \phi(\cdot)} \left[\Phi_{(t=0,\ldots,h)} \left(g_0^i(\mathbf{x}_0, a_0, \varepsilon_1), \ldots, g_h^i(\mathbf{x}_h) \right) \right] \tag{3.2a}$$

$$J^i = \lim_{h \to \infty} \mathbb{E}_{\varepsilon_1,\ldots,\varepsilon_h} \left[\sum_{t=0}^{h-1} \gamma^t g_t^i(\mathbf{x}_t, a_t, \varepsilon_{t+1}) \right] \tag{3.2b}$$

Objective functions are subject to the system dynamic, represented by eq. (3.3). The state vector $\mathbf{x}_t \in \mathcal{S}_{\mathbf{x}_t} \subset \mathbb{R}^{n_x}$ includes all the n_x state variables. The vector difference equation determines the state transitioning from t to $t+1$ (eq. 3.3a) where the states $\mathbf{x}_t$ transitions to $\mathbf{x}_{t+1}$ under the influence of the action a_t and also by the stochastic disturbance

$\varepsilon_{t+1} \in \mathcal{S}_{\varepsilon_{t+1}} \subset \mathbb{R}$, which could be described through its probability distribution function $\phi_t(\cdot)$ (eq. 3.3b). The release action $a_t \in \mathcal{A}_t(\mathbf{x}_t) \subseteq \mathcal{S}_{a_t} \subset \mathbb{R}$, follows the deterministic operating policy $\pi(\mathbf{x}_t)$ adopted in the simulation horizon h (eq. 3.3c).
The periodicity of the hydrologic system is captured through the current problem formulation, thus the stochastic disturbance that afflicts the system $\phi_t(\cdot)$, as well as the function f_t and the set of control laws $\mathcal{A}_t(\cdot)$, are periodic of period T.

$$\mathbf{x}_{t+1} = f_t\left(\mathbf{x}_t, \varepsilon_{t+1}, a_t\right) \tag{3.3a}$$

$$\varepsilon_{t+1} \sim \phi_t(\cdot) \tag{3.3b}$$

$$a_t = \pi(\mathbf{x}_t) \tag{3.3c}$$

Among the n_x system states that could be represented by $\mathbf{x}_t$ the reservoir storage x_t is usually the most relevant for the reservoir operators, its dynamic follows the mass balance equation eq. (3.4), where the next step storage x_{t+1} depends on the previous step storage x_t, on the reservoir net inflow ε_{t+1} between steps t and $t+1$, which includes net evaporation and other losses, and on the actual release $w_{t+1}(\cdot)$ that is determined by a nonlinear and stochastic function affected by the net inflow stochasticity between t and $t+1$ and, for that reason, may not coincide with the release action a_t.

$$x_{t+1} = f_t\left(x_t, \varepsilon_{t+1}, w_{t+1}(x_t, \varepsilon_{t+1}, a_t)\right) \tag{3.4}$$

In the current thesis work, the only endogenous state considered for optimal control is the storage of the system, thus $n_x = 1$ and $\mathbf{x}_t = x_t$.

3.2. Batch Reinforcement Learning Algorithms

Linking the problem formulation to the reinforcement learning method is fundamental for building an effective framework to solve the optimal control problem.
The system dynamic suits well the MDP nature of the system assumed by the RL methods, the endogenous states s_t of the MDP coincide with the hydrological states $\mathbf{x}_t$ of the system, and, as previously described in the current work, these are equal to the reservoir storage x_t (eq. 3.5).

$$\mathbf{x}_t = x_t = s_t \tag{3.5}$$

The problem formulation defines the immediate step cost functions $\mathbf{g}_t = [g_t^1(\cdot), \ldots, g_t^q(\cdot)]$, these make up the values of the respective objectives, which must be minimized, according to eq. (3.1). The key to ensuring that RL actually optimizes the desired objectives is a careful design of both the reward function and the objective functions. Similar to the limitation in the SDP curse of multiple objectives described in section 2.1, the objective functions could assume only specific ψ and Φ operators according to the reward function definition, e.g. for the infinite horizon formulation as eq. (3.2b).
Given the single objective nature of the reinforcement learning methods, only one reward function R_{t+1}^i could be considered from the vector of q reward functions $\mathbf{R}_{t+1} = \left[R_{t+1}^1(\cdot), \ldots, R_{t+1}^q(\cdot)\right]$, defined equal to minus the vector of cost function, i.e. $-\mathbf{g}_t = \mathbf{R}_{t+1}$. Therefore, since the RL algorithm learns a policy that maximizes all the immediate rewards R_{t+1}^i within a simulation horizon, so defined it also minimizes the immediate step cost g_t^i in the simulation horizon.

The reward function R_{t+1} adopted in the optimization frame could account for all the q single objective reward functions by a convex combination of these, i.e. the weighted sum of rewards (eq. 3.6). The weights vector $\boldsymbol{\lambda} = [\lambda^1, \ldots, \lambda^q] \in \Lambda^{q-1}$, where Λ^{q-1} is the unit $(q-1)$-dimensional simplex, must be selected to reflect the importance to be assigned to each objective. Doing so, by the linearity of the expected value and the weighted sum defined in eq. (2.3a) the value function related to the set of weights is V_λ^π defined as in eq. (3.7), where $\mathbf{V}^\pi = \left[V_1^\pi, \ldots, V_q^\pi\right]$ is the vector of the value functions related to each reward function.

$$R_{t+1} = \boldsymbol{\lambda}' \cdot \mathbf{R}_{t+1} \tag{3.6}$$

$$V_\lambda^\pi = \boldsymbol{\lambda}' \cdot \mathbf{V}^\pi \tag{3.7}$$

The resulting value function V_λ^π adopted in the RL framework leads to a policy π, defined as eq. (2.5), that maximizes all the reward R_{t+1}, received as feedback from the actor, with respect to the set of weight $\boldsymbol{\lambda}$ selected. Therefore, computing the value of the objectives $\mathbf{J}^\pi = \left[J_1^\pi, \ldots, J_q^\pi\right]$ within a time horizon h in which the policy is applied, these results in a single point in the q-dimensional space of the objectives, and thus if it is not Pareto-dominated by the other points generated with different policies it composes a single point of the Pareto frontier.
With this single objective approach, by reformulating the problem with different weight sets $\boldsymbol{\lambda}$ a finite subset of the Pareto front is obtained. Dealing with multiple objectives with

a single objective algorithm implies multiple runs of the same algorithm with different problem settings (Castelletti et al., 2010).

Batch Learning - FQI As other RL algorithms, Fitted Q Iteration (FQI) (Damien Ernst et al., 2005) learns from experience overcoming the curse of modeling that afflicts SDP. A continuous approximation of the action-value function on the entire state-action space overcomes the tabular representation used by SDP and also other RL algorithms, such as Q-learning. Therefore FQI does not need a dense state-action sampling to estimate the Q-function with a high resolution since it approximates the function relying on limited state-action samples. Thus a high-resolution policy could be derived with FQI based on a limited amount of samples and potentially achieve the same performances of SDP with a much denser sampling grid, thereby mitigating the effect of the curse of dimensionality.

The learning process is performed off-line, so from a batch of experience represented as a finite dataset $\mathcal{F}$ of tuples defined as $\langle s_t, a_t, s_{t+1}, R_{t+1}\rangle$, with cardinality $\#F$, as eq. (3.8). Regardless of how it is generated, e.g. from the sampling of historical trajectories or through the simulations of the system dynamics, this dataset is the only information needed to find the policy.

$$\mathcal{F} = \left\{ \langle s_t^l, a_t^l, s_{t+1}^l, R_{t+1}^l \rangle \mid l = 0, \ldots, \#F \right\} \tag{3.8}$$

Similarly to Q-learning, but acting offline, FQI estimates the action-value function (eq. 2.4) through regression on the training set $\mathcal{TS}$ (eq. 3.9) therefore it estimates the Q value for the whole state-action space $\mathcal{S}_s \times \mathcal{S}_a$, denoting the estimate with $\hat{Q}$. The action-value function is iteratively estimated with a regression algorithm $\mathcal{R}$ above the training set $\mathcal{TS}$. First, at iteration index $h = 0$, it initializes $\hat{Q}_0(s, a) = 0 \quad \forall s \in S \quad \forall a \in A$, then it updates the Q samples in the training set by bootstrapping on the value of the estimate at the previous iteration $h-1$, as shown by eq. (3.10). As an example, the first iteration, where $h = 1$, lead to $Q_t^l = R_{t+1}^l$ since $\hat{Q}_0(s, a) = 0$.

By doing so the algorithm iteratively extends the optimization horizon h and thus the estimated $\hat{Q}$ function converges to the real action-value function Q^*. Convergence is not analytically demonstrated for each regressor $\mathcal{R}$ but it is empirically proved for many of them, in particular, Extra-Trees regressor achieves good performances results in the FQI optimization frame (Damien Ernst et al., 2005).

$$\mathcal{TS} = \left\{ \left(s_t^l, a_t^l\right) \rightarrow Q_t^l \mid l = 0, \ldots, \#F \right\} \tag{3.9}$$

$$Q_t = R_{t+1} + \gamma \max_a \hat{Q}_{h-1}(\mathbf{s}_{t+1}, a) \tag{3.10}$$

The FQI algorithm proposed by Damien Ernst et al. (2005) generates a stationary policy π as eq. (2.5), however, natural systems are typically non-stationary therefore a periodic policy usually performs better, adapting the control to the seasonal variability.
Non-stationary policies could be generated by extending the state s_t to a vector $\mathbf{s}_t$ which includes the time among the states of the system, where the time component follows the straightforward deterministic transition function, i.e. $p(t' = \tau + 1 \,|\, t = \tau) = 1$ except for $p(t' = 1 \,|\, t = T) = 1$ with $\{\tau = 1, \ldots, T\}$. According to the new state vector defined as eq. (3.11), adopted with a periodic system by substituting s_t with $\mathbf{s}_t$ in the FQI algorithm section 3.2, the resulting policy generates T different operating rules $\{\pi_\tau(\cdot); \tau = 1, \ldots, T\}$.

$$\mathbf{s}_t = [s_t, \tau] \tag{3.11}$$

Extra-Trees Regressor Despite the algorithm's convergence is no longer guaranteed, Extra-Trees was demonstrated to achieve good performances as regressor $\mathcal{R}$ in the FQI framework (Damien Ernst et al., 2005).
Different methods to build tree ensembles have been developed, Extra-Trees works by building several M trees, and each tree is built from the complete training set $\mathcal{TS}$. To determine the node splitting condition, $\mathcal{K}$ cut-directions are randomly selected, and then the algorithm chooses among these $\mathcal{K}$ tests the one that maximizes the given score. The Extra-Trees algorithm stops splitting a node when the number of elements in this node is less than a parameter n_{min}. So three parameters are associated with the Extra-Trees algorithm: the number M of trees, the number $\mathcal{K}$ splitting tests, and the minimum leaf size n_{min}.

The score adopted for splitting nodes is the mean squared error (MSE) defined as in eq. (3.12), particularly suitable for regression problems, and in most cases equivalent to the variance reduction metric adopted by Geurts et al. (2006) and Damien Ernst et al. (2005). MSE score still guarantees the Q_t approximation as conditional expected values with respect to the input values in the right-hand side of eq. (3.10).
MSE measures the average distance between estimates due to splitting a node with a certain threshold and the Q^l values of the training set.

$$MSE = \frac{1}{\#F} \sum_{t=1}^{\#F} (\hat{Q} - Q^l)^2 \tag{3.12}$$

Previous studies about the sensitivity of Extra-Trees parameters in FQI application in the water reservoir optimal control problem highlight guideline ranges for Extra-Trees hyperparameters. The $\mathcal{K}$ selected variables at each node should be equal to the number of input variables. The minimum sample size n_{min} in a leaf should be at least equal to the number of disturbance samples for the current state. Finally, increasing the parameter M which determines the number of trees, reduces the variance of the Q function and the discrepancy between performance in training and validation (Castelletti et al., 2010). In the current work, $\mathcal{K}$ and n_{min} are assumed equal to the ones adopted in the literature, while the sensitivity to the number of trees M concerning the policy performance is explored.

3.2.1. Multi-Objective Fitted Q Iteration

As pointed out in par. *Batch Learning*, the FQI algorithm is designed for a single objective optimization, and in relation to the value function formulation it can incorporate a convex combination of multiple objectives, as eq. (3.7). A version of the algorithm proposed by Castelletti et al. (2013) and Pianosi et al. (2013), which is adopted in the current work, addresses and solves the multi-objective MDP problem formulation, making the algorithm computationally advantageous with respect to both the FQI and the SDP methods. In a single run, it can potentially generate a complete estimate of the Pareto front, mitigating the first limitation of the multiple objectives curse; while FQI and SDP require multiple runs for a finite approximation of the Pareto front, MOFQI only requires one.

The Multi-Objective Fitted Q Iteration (MOFQI) algorithm relies on the continuous approximation of the value function in the Λ^{q-1} unit $(q-1)$-dimensional simplex domain. The key idea is to enlarge the state vector adopted in the FQI, including the weight vector $\boldsymbol{\lambda} \in \Lambda^{q-1}$ among the system states variables as defined in eq. (3.13). These additional state variables follow the transition probability function $p(\boldsymbol{\lambda}^i \mid \boldsymbol{\lambda}^i) = 1$.

$$\mathbf{s}_t = [s_t, \tau, \boldsymbol{\lambda}] \tag{3.13}$$

The regressor $\mathcal{R}$, which estimates the action-value function, receives as input the weight vector $\boldsymbol{\lambda}$ and thus resulting in an action-value function $\hat{Q}_{\boldsymbol{\lambda}}$ parametrized by the weight vector; thus, the resulting policy $\pi_{\boldsymbol{\lambda}}$ is continuously approximated over the weight space, according to the greedy policy definition in eq. (2.5). The exploration of the simplex space through a representative sampling must be done to achieve a good approximation of the Q-function over the weights states, thus these additional samples must be included in the dataset $\mathcal{F}$ and provided as experience to the learning algorithm.

Algorithm 3.1 Pseudocode: Multi-Objective Fitted Q Iteration (MOFQI)

Input: $\mathcal{F} = \left\{ \langle \mathbf{s}_t^l, a_t^l, \mathbf{s}_{t+1}^l, R_{t+1}^l = \boldsymbol{\lambda}^l \cdot \mathbf{R}_{t+1}^l \rangle \mid l = 0, \dots, \#F \right\}$, regression algorithm $\mathcal{R}$, stop criterion
Initialization: $\hat{Q}_0(\mathbf{s}, a) = 0, \forall \mathbf{s} \in S, \forall a \in A$
for $h = 1$ to stopping condition **do**

Training set generation:
$\mathcal{TS} = \left\{ \left(\mathbf{s}_t^l, a_t^l\right) \rightarrow Q_t^l = R_{t+1}^l + \gamma \max_a \hat{Q}_{h-1}\left(\mathbf{s}_{t+1}^l, a\right) \mid l = 0, \dots, \#F \right\}$

Action-value function estimation:
$\hat{Q}_h = \mathcal{R}(\mathcal{TS})$

end for

3.3. Dataset Generation

The experience dataset used in offline learning could be composed of historical sampled trajectories, but in doing so the exploration of system dynamics is limited to the historical values. Another way to generate a dataset is to simulate the transition dynamic through a simulation model so that the whole state-action space can be explored; thus the agent learns the system transitions and the associated feedback in the whole domain of interests. Relying on a simulation model makes the current FQI framework a model-based method, but unlike model-based optimal control algorithms, such as SDP, which need to explicitly model the transition probabilities, in this case, only state transitions need to be simulated. Influenced by the action, the state transitions simulation generates the experience; the entire upstream part of the water system does not need to be simulated, it is considered through the historical trajectory of the stochastic disturbance that affects the system.

State-Action Sampling Although the FQI algorithm manages to mitigate the effects of the dimensionality curse by approximating the value function continuously, the accurate sampling of the entire state-action space still results in a major computational burden. For each sampled historical ε_t disturbance value a system transition associated with a state-action couple (s, a) could be simulated through the model and thus producing an experience tuple, as schematized in fig. 3.1.

The number of system's transitions $\mathcal{D}$ simulated by the model is defined by eq. (3.14). Assume to compute the cardinality of $\mathcal{D}$ for a case study with the following characteristics; the number of state variables n_s is equal to one as well as the number of control variables n_a and thus adopting a grid sampling of the state-action space the number of (s, a)

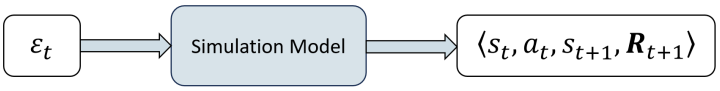


Figure 3.1: Scheme of transition sampling procedure, within the state-action space, via partial model-free simulation

couples is $(N_s \cdot N_a)$, where N_s is the number of state samples and N_a the number of action samples. Only one stochastic disturbance is considered, i.e. $n_\varepsilon = 1$, and it is sampled for k years of period T. As a result, the cardinality of $\mathcal{D}$ for this specific case is equivalent to $kT \cdot (N_s \cdot N_a)$.

$$\mathcal{D} = kT \cdot (N_s^{n_s} \cdot N_a^{n_a} \cdot N_\varepsilon^{n_\varepsilon}) \tag{3.14}$$

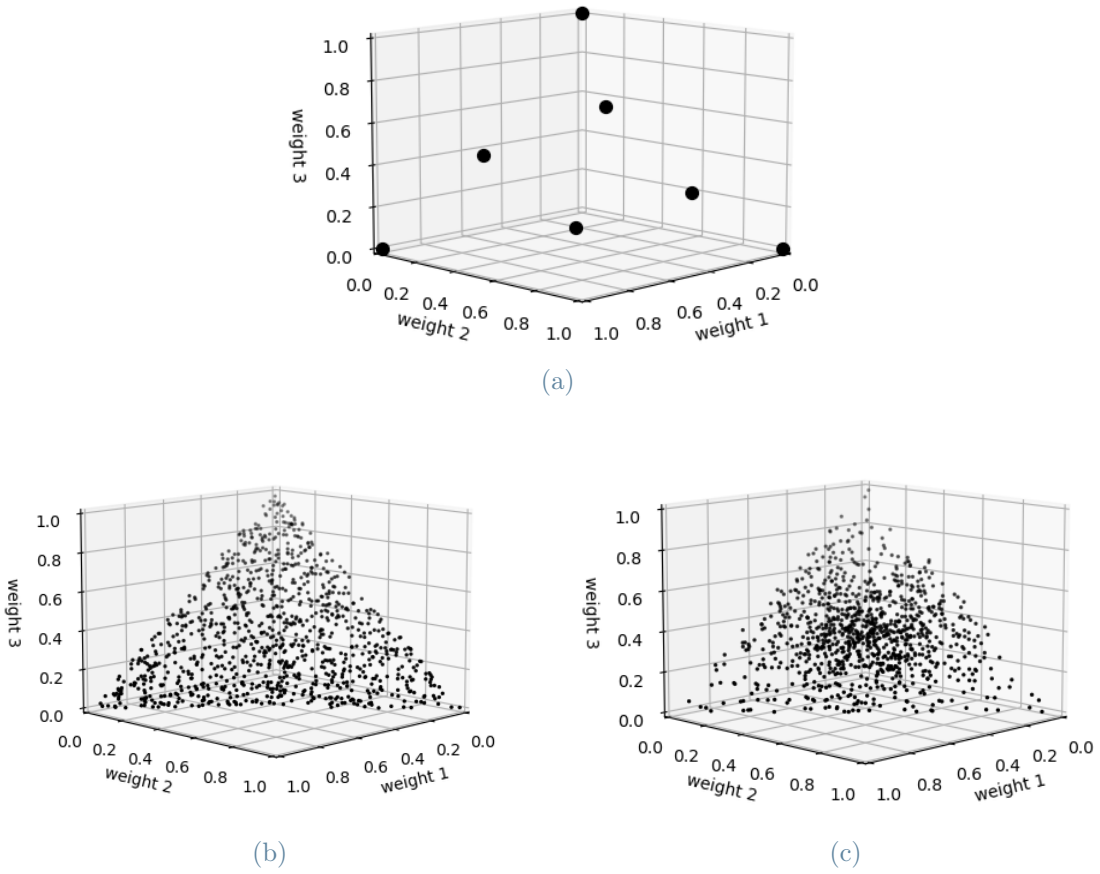
As discussed in section 3.2 and section 3.2.1, the weights and temporal states accounted in the MOFQI state vector defined by eq. (3.13) follow the plain transition model that does not need any simulation; these states variable could be directly added to the dataset coherently with their straightforward dynamics. The temporal variable representing the day of the year is bounded by the historical sampling of the disturbance, which means that the temporal state must be coherent with the related day of the measured variable. Each weight vector $\boldsymbol{\lambda}'^l$ that builds up the reward by aggregating each multi-objective MDP reward as $R^l_{t+1} = \boldsymbol{\lambda}'^l \cdot \mathbf{R}^l_{t+1}$ could be independently sampled since it is disjointed from the system transition.
The resulting structure of the dataset $\mathcal{F}$ provided as input to the MOFQI algorithm 3.1, is structured as table 3.1; where the cardinality $\#F$ depends on the number of sampled system transitions $\mathcal{D}$ and on the weight space sampling technique which determines the rewards convex sum R_{t+1}.

$\mathbf{s}_t$			a_t	$\mathbf{s}_{t+1}$			R_{t+1}
s_t^0	t^0	$\boldsymbol{\lambda}_t^0$	a_t^0	s_{t+1}^0	$(t+1)^0$	$\boldsymbol{\lambda}_{t+1}^0$	$\boldsymbol{\lambda}_t'^0 \cdot \mathbf{R}_{t+1}^0$
				...			
$s_t^{\#F}$	$t^{\#F}$	$\boldsymbol{\lambda}_t^{\#F}$	$a_t^{\#F}$	$s_{t+1}^{\#F}$	$(t+1)^{\#F}$	$\boldsymbol{\lambda}_{t+1}^{\#F}$	$\boldsymbol{\lambda}_t'^{\#F} \cdot \mathbf{R}_{t+1}^{\#F}$

Table 3.1: Structure of the dataset $\mathcal{F}$ for the MOFQI algorithm

Fixed Simplex Sampling In the current thesis work, two approaches for sampling the simplex surface are tested, the first, also adopted in Pianosi et al. (2013), fixes a number K of weights combinations (fig. 3.2a) and then evaluates all the $\mathcal{D}$ transitions for each weight vector, hence results in a dataset cardinality $\#F = K\mathcal{D}$. Applying this sampling

strategy expands the dataset dimension linearly with K, and the $K\mathcal{D}$ tuples amount easily grows to an unfeasible dataset dimension for an adequate weight space sampling; especially increasing the number of objectives accounted in the problem formulation which requires additional weights dimensions.



(a)

(b)

(c)

Figure 3.2: Example of 1000 samples from the unit $(q-1)$-dimensional simplex, where $q = 3$, an example of 7 samples for the fixed sampling technique (a); uniformly distributed (b) and sampled with a Gaussian distribution (c)

Random Simplex Sampling The second technique, proposed and tested in the current thesis work, aims to reduce the computational burden due to dataset dimension growth proportionally to K but still adequately representing the value function approximation over the weights space.

Since the weight values are independent of all the other state variables, each sample $\boldsymbol{\lambda}^i$ could be randomly sampled in the simplex domain according to a random distribution $\boldsymbol{\lambda}^i \sim \xi(\cdot)$. As an example in fig. 3.2b, there is the result of 1000 samples uniformly distributed over the simplex surface for $q = 3$ and in fig. 3.2c sampled based on a multivariate Gaussian distribution.

By doing so, the agent could potentially explore the full simplex surface without increasing the dataset dimension. The cardinality of dataset $\mathcal{F}$ remains equal to the number of the transition tuples simulated through the model, i.e. $\#F = \mathcal{D}$. In particular, the cardinality $\#F$ of the dataset $\mathcal{F}$ and the number of objectives considered in the problem formulation are independent; since the number of sampled points of the simplex is equal to the number $\mathcal{D}$ of tuples generated, this value does not depend on the number of objectives considered, i.e. on the number of dimensions of the simplex. Potentially this approach makes the dataset generation feasible even with a large number of objectives, as long as the collected samples represent the surface of the action-value function well enough to perform a good regression and converge the algorithm. Further, by enlarging the number of transition samples both state-action space and weight space are more densely sampled.
Within the current work, two random sampling distributions are tested and compared, a uniform one over the simplex surface and a second one based on a Gaussian distribution; the latter is adopted with the underlying idea to sample much more the tradeoff region of weight in the perspective of finding more accurate policies also in the tradeoff objectives space.

It should be underlined that, as already mentioned, this approach aims to represent the system on the entire surface of the simplex without increasing the cardinality of the dataset. As a consequence, compared to the fixed sampling in which all the system transitions are evaluated for each chosen $\boldsymbol{\lambda}$, this second method distributes randomly on the simplex surface the experience represented by the transition tuples. In this way the system states are widely explored, however, losing part of the experience for each combination of weights.

3.4. Evaluation Metrics

The workflow of the optimization frame is schematized in fig. 3.3, this could be subdivided into three main steps. First, given the disturbance trajectory and the simulation model the transition tuples could be generated, as described in section 3.3; then, as shown in section 3.2.1, given the dataset $\mathcal{F}$ the MOFQI algorithm learns the action-value function approximation from which the policy is generated. Finally, a portion of the available dataset, excluded a prior, is used to validate the result; therefore, the model after being trained using a certain trajectory is validated on a different one. The process of validating the results aims at assessing the ability of the model to generalize the performance; that is, assessing whether the model does not overfit its behaviour on the training trajectory.

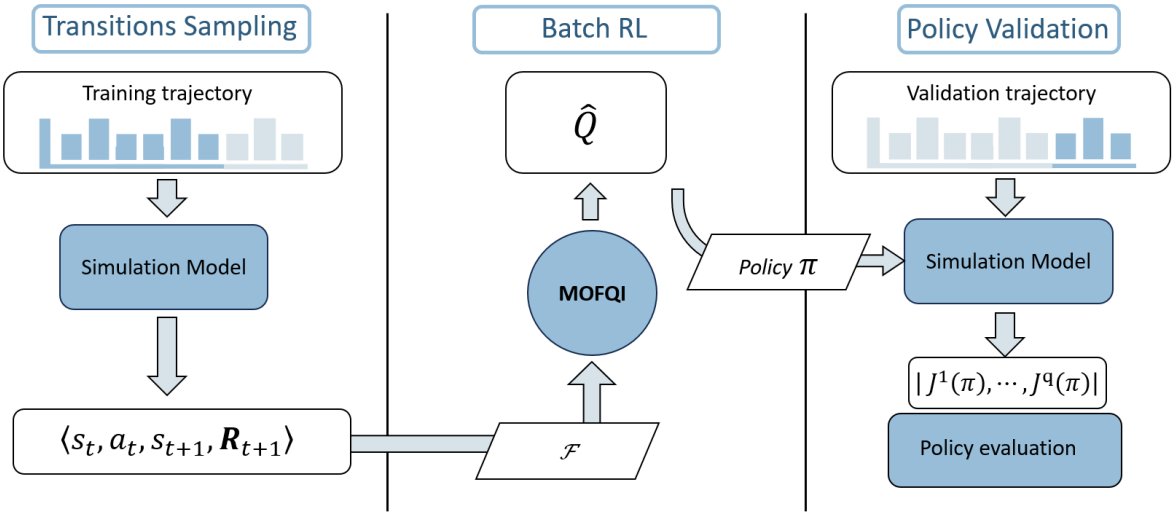


Figure 3.3: Work scheme adopted, divided into three main steps: dataset generation, batch learning, and policy validation

3.4.1. Pareto Front Evaluation

The policy learned by the MOFQI is simulated on the historical trajectories, the result of each simulation is a vector $\boldsymbol{J}$ of q objective values. Applying different weight combinations $\boldsymbol{\lambda}$ implies the adoption of the policy $\pi_{\boldsymbol{\lambda}}$ related to that specific weight state. Given an estimate $\hat{Q}$ of the action-value function, accounting for the weights state variable as described in section 3.2.1, different $\pi_{\boldsymbol{\lambda}}$ could be simulated over the same trajectory, leading to a set of different objectives value $\boldsymbol{J}^i \in S$. The objective values, obtained through the simulation, define the position of the policy performance in the objective space concerning the related weight state. Therefore for one estimated $\hat{Q}$ by MOFQI algorithm, potentially infinite policies $\pi_{\boldsymbol{\lambda}}$ could be simulated, generating the related Pareto front.

Perfect Operating Policy - POP To facilitate the interpretation of performance results of the generated MOFQI policies, these are compared to the solutions obtained with deterministic dynamic programming (DDP) (Bellman, 1957) by assuming perfect knowledge of the disturbance over the whole simulation horizon.
The DDP solution performance is the best that can be achieved for that time horizon, thus the resulting policies are the so-called Perfect Operating Policies (POP). For this reason, the set of solutions $S_{\text{POP}} \subset \mathbb{R}^q$ that approximates the POP Pareto front is used as a reference set for comparing the results.
It should be noted that the points $p_{\text{POP}} \in S_{\text{POP}}$ do not necessarily reach zero in the space of the objectives; these will reach the minimum achievable value in relation to the

disturbance trajectory, the system constraints and the weight combination assigned to the objectives.

Stochastic Dynamic Programming - SDP For a further and definitive comparison, the solutions generated via MOFQI are compared with the solution set $S_{\text{SDP}} \subset \mathbb{R}^q$ obtained via Stochastic Dynamic Programming (SDP); this method is considered a state-of-the-art technique regarding the water reservoir optimal control problem, as anticipated in section 2.3.
The SDP algorithm needs explicit modelling of the transition probability; the transition model is estimated from a dataset of historical inflows equivalent to the MOFQI training trajectory; then the SDP iteratively optimizes the solution relying on the Bellman equation (eq. 2.3b) for a discretized representation of the state-action space (Soncini-Sessa et al., 2014).

Hypervolume metric In Multi-Objective optimization, the hypervolume (HV) metric is widely adopted (Zitzler and Thiele, 1998; Guerreiro et al., 2022). For a given objective points set $S \subset \mathbb{R}^q$ the HV_S is the measure of the region weakly Pareto-dominated by S and bounded by a reference point p_{ref}.
The concept of Pareto-dominance can be defined as follows: considering a multi-objective maximization problem with q objectives, the region weakly dominated by a point $p \in S$ and bounded by a reference point $p_{\text{ref}} \in \mathbb{R}^q$ is defined as follow:

$$\{q \in \mathbb{R}^q \mid \exists p \in S : p \leq q \text{ and } q \leq p_{\text{ref}}\}$$

The HV_S metric is a measure that uniquely identifies the whole Pareto front for any points set S with any number of objectives q. this metric is adopted in multi-objective problems due to its capability to evaluate both convergence to the minimum values and diversity of the points set S.
To facilitate comparison between different solutions in the current framework the HV_S is normalized relatively to the HV_{pop} measure of the reference POP set of policies, as reported in eq. (3.15), therefore the closer it is to a unit value the more it reaches the best achievable performance.

$$HV = \frac{HV_S}{HV_{\text{pop}}} \tag{3.15}$$

3.4.2. Convergence Evaluation

The MOFQI algorithm convergence is no longer theoretically guaranteed since the regressor used is the Extra-Trees (Damien Ernst et al., 2005). The increase of the function $\hat{Q}$ does not guarantee that the estimate is converging to the optimal action-value function. Furthermore, due to the Extra-Trees structure reset at each iteration of the MOFQI, the differences between the action-value function approximation $\hat{Q}_h$ from one iteration to the next one never vanishes.
For these reasons, the definition of a suitable stopping condition for the algorithm is not straightforward and should be defined empirically. In order to evaluate the convergence in the current work two quantitative metrics are used.
The first metric evaluates the $\hat{Q}$ increase through algorithm iteration, computed as eq. (3.16). It quantifies the average absolute increase between two different iterations evaluated over the whole training set.

$$\Delta_{\hat{Q}} = \frac{1}{\#F} \sum_{i=0}^{\#F} \left(|\hat{Q}_h([\mathbf{s}, a]_i) - \hat{Q}_{h-1}([\mathbf{s}, a]_i)| \right) \tag{3.16}$$

On the other hand, the second metric, the normalized hypervolume HV previously defined, evaluates the resulting performances in terms of Pareto front convergence at each algorithm iteration. Thus, generating a set S_h from the estimated $\hat{Q}_h$ at iteration h allows to evaluate the performance of the policy by increasing algorithm iterations. This metric aims to capture the effectiveness in addressing all the q objectives given the estimated action-value function, i.e. it empirically evaluates the learned performances.
The stopping condition adopted is the maximum iteration number $\bar{h}$, empirically defined a posteriori based on both the mean absolute increase and the HV_{norm}, evaluated for the settings and hyperparameters of the current algorithm.

4 | Case study

4.1. Lake Como System

The optimal batch RL control problem is solved for the Lake Como case study, a large regulated lake in Northern Italy, just 50 km North of Milan (fig. 4.1).

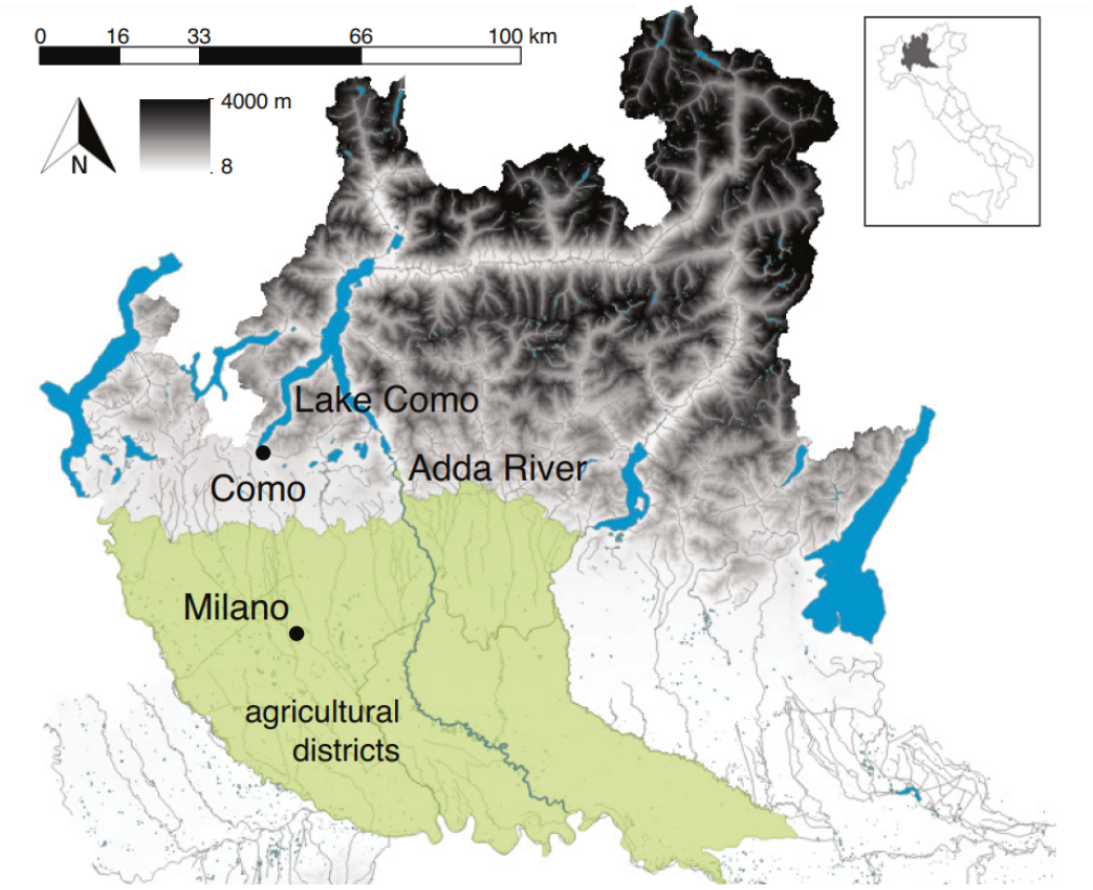


Figure 4.1: Map of Lake Como water system (Giuliani et al., 2020)

Lake Como, also known as Lario, has a surface of about 145 km^2, making it the third lake for dimensions in the country, and it's one of the deepest lakes in Europe, holds a total volume of 23.4 km^3, of which about 220 Mm^3 are the active storage capacity. Lake

Como, being of glacial origins, is characterized by the common Alpine topology and is surrounded by mountains. The lake is part of the Adda River Basin and has 37 tributary rivers in a catchment area of 4552 km^2. Among its tributaries, the main contribution is given by the Adda River, which is 323 km long, followed by the Mera, with a length of 50 km (mostly in Switzerland).
The lake is shaped like an upside-down "Y", and in the area denominated "Pian di Spagna" on the northern branch of Colico, the Adda and Mera Rivers enter the lake; while at the ends of the two southern branches are located the cities of Como (western branch) and Lecco (eastern branch).
The only emissary is the Adda River, which flows out of the lake in the South of the eastern branch. In its southwards course from Lake Como, the Adda River passes through two lakes in Garlate and Olginate, subsequently flow through a major irrigation district that spans about 1400 km^2 and extends across most of the provinces of the Lombardia region, until it reaches the Po River. Other than for irrigation purposes, the Adda is crucial for the supply of several hydropower plants in the area. There are 16 hydropower reservoirs upstream of Lake Como that account for a total of 545 Mm^3, while 17 run-of-the-river power plants are located downstream and depend on the lake releases.

Hydrological patterns

The annual water balance of the Lake Como basin, see hydrograph in fig. 4.2, is mainly driven by snowmelt during spring and precipitations in autumn. Snow melt during May-July is the largest contribution to the accumulation of the seasonal storage of the lake, which is used for irrigation supply in summer during the peak of water demand. The natural water availability in summer is often less than the water demand and makes the role of the lake operation essential for the system (Denaro et al., 2017; Giuliani and Castelletti, 2016).

This basin has a huge hydropower potential in the upper region, exploited thanks to a large storage capacity distributed in many small-to-medium reservoirs. Upstream reservoirs tend to accumulate the snowmelt during summer to use its hydropower potential in the following autumn and winter when the demand peaks making the energy production more valuable. On the other hand, downstream users need an adequate supply during the summer period and thus prefer to store the spring snow melt in the lake to use it during summer. Therefore, during drought periods, conflicts arise between upstream hydropower and downstream users (Anghileri et al., 2012).

In the future, a decrease in water availability is expected, a scenario that will expose more often agricultural districts to severe water stress (Iglesias and Garrote, 2015). Moreover,

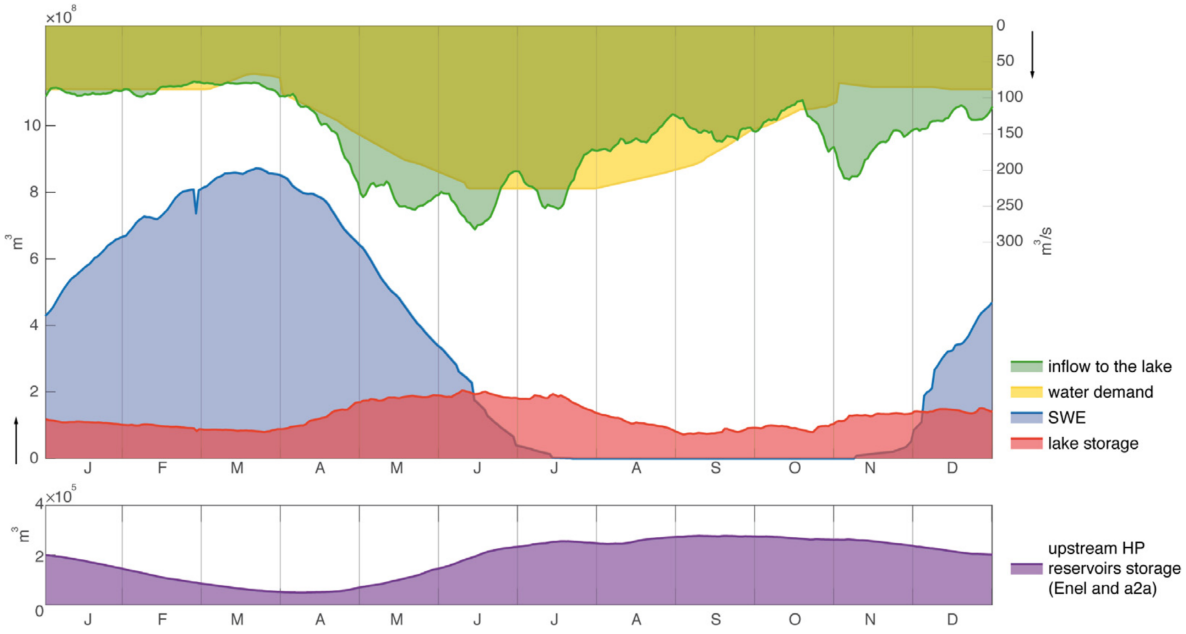


Figure 4.2: Main hydrological components of the inflow for the Lake Como basin, the patterns are obtained as moving averages of observed data over the period 2006 - 2013 (Denaro et al., 2017)

.

the projected increase in the temperature will affect particularly the snowpack dynamics, and as a consequence, will alterate the hydrologic regime (McDowell et al., 2014). Under a warmer climate, snow melt will start early, directly impacting flood control, but also high lake levels will last for longer periods to store additional volumes of water, a practice necessary to make up for shortages during summers (Forzieri et al., 2014). In addition, is expected that indirect negative effects hit the farmers as well, the projections suggest an increase in dry summers, mainly caused by a lack of precipitation and consequently decreased inflows to the lake, but for the irrigation supply, a key aspect of the drought will be the earthly distribution of rainfall over the cropping season and not only the total amount of seasonal rainfall, increasing the importance of proper lake regulation.

Decision-makers and stakeholders

The Lake Como system is characterized by the presence of different interests and several decision-makers and stakeholders, they are linked by sharing the water resource in its multiple uses and functions. The main stakeholders of the system include:

- Consorzio dell'Adda, it has responsibility for lake management and is the institutional operator of Lake Como. It is supervised by the Italian Ministry of the

Environment.

- Irrigation districts (farmers and their governance entities), which are served with water supply through a dense network of irrigation canals fed by Adda River flow, which is driven by the lake releases. There are four districts with a total surface of 1,400 km2, mostly cultivated with maize, rice and soy. Districts are managed by twelve consortia, these are public economic entities of the regional system of Lombardia, which address multiple purposes, from environmental protection to governing the artificial waters of the irrigated plain.
- Hydropower companies, the main ones are ENEL, A2A, and EDISON, these manage the large storage and hydropower production capacity of the Adda River Basin, both upstream and downstream of Lake Como.
- Regione Lombardia is the regional water authority.
- the water authority of the Po River Basin (ADBPO), which is the body that connects all regional and local bodies whose operations influence the dynamics of the Po River Basin.

4.2. Lake Como Regulation

In order to control the flow regime, a dam between Garlate and Olginate was built in 1946. Since its creation, the lake management, provided by the public authority "Consorzio dell'Adda", has had the purpose of flood protection along the lake shores and water supply to the downstream users. The conflict between the short-term flood protection goal and the long-term water supply objective hasn't been a limiting factor for economic development in the area, even if the water governance mainly focused on flood management (Denaro et al., 2017). Over the years, as the effects of climate change have intensified weather extremes, droughts in the region have increased in recent decades and inflows are projected to decrease further by mid-century (Giudici et al., 2021), making water supply a difficult task to achieve. In this changed context, nowadays severe droughts can fuel conflicts between stakeholders, but they also require new countermeasures, such as avoiding low lake levels, a new practice that aims to avoid environmental damage to the lake ecosystem and shores, as well as to permit navigation and recreational activities on the lake or on shore. Thus, currently, three main objectives are driving the lake management, which are: (i) flood control, (ii) water supply, and (iii) lake drought control. Moreover, the environmental legal constraint of the Minimum Environmental Flow also has a primary influence on the lake operation, as summarized below.

Flood control Historically, The flood control objective has always been related to the cities along the lake shores. In particular, the City Centre of Como (Piazza Cavour) is a very critical point. This city is the lowest on the lake shoreline (Giuliani et al., 2019), moreover, this location is suffering from a subsidence phenomenon which strongly affects the flood control problem. The leading causes of this phenomenon are the stratigraphic and geological setting in the Como basin and anthropogenic activities, like the exploitation of water resources from the deep main aquifer and land reclamation (Nappo et al., 2020). The problem has been studied for more than 40 years (Como, 1980) and the dam installation reduced the number of flood events, but it remains a crucial aspect of lake management.

Water supply Water demand from the downstream users is the sum of two main sector demands, irrigation and hydropower production. The cultivated area fed by the Adda River through seven main canals derived from the river after exiting Lake Como is about 1320 km^2, where maize is the most widely grown crop. For what concerns energy production, a system of small run-of-river power plants relies on the water stream, together with two thermoelectric powerplants (Consorzio dell'Adda, 2022). As a consequence of this mixture of activities, the total water demand follows a combination of the patterns of the different sectors: it peaks during summer, to respect the natural requirement of the crops in the irrigation period, while during winter it keeps steady values that are higher than the required amount for agriculture, but are necessary to keep the hydropower plants running.

Lake low-level control Lately, lake management has been concerned with a new aspect during the decision-making process, i.e. avoiding extremely low levels of the lake. Low lake levels are achieved frequently during dry seasons when water is released to satisfy the demand even when inflow in the long term is not sufficient, but also occur sporadically before extreme flood events when large amounts of water are released to create a buffer for the incoming water flows. Low levels are detrimental to all the stakeholders; for upstream hydropower companies because there is a direct loss in energy production (as they are required to release more water downstream at times of low levels), for the irrigation consortia as they are affected by disturbances to the crop cycle that require adaptation of the cropping practices and irrigation methods, and for the lake users because low levels affect navigation, as well as environmental and touristic aspects.

Minimum Environmental Flow - MEF Another constraint affecting the release of water is the Minimum Environmental Flow (MEF), i.e. the minimum flow released from

the lake to preserve the downstream environment and ecosystems. This amount of water is always required downstream whenever the natural inflow to the lake is sufficient to provide it and adds to the downstream stakeholders' demand. The MEF definition is regulated by the regional authority (Regione Lombardia) but is an argument of many debates and it changes over time as the right compromise between environmental and economic needs is difficult to find. For the release from Lake Como and the Adda catchment, the MEF is defined as the minimum between the legally defined value, which currently is 22 m^3/s, and the available inflow to the lake.

4.3. Lake Como Model

As discussed in section 3.1, in the current case study, the only endogenous state variable considered is the storage, according to the partially model-free framework, the system could be schematized as fig. 4.3. The water system is composed of a catchment that produces a net inflow ε that feeds the reservoir, this latter is operated on a daily basis releasing an outflow w according to the system dynamics, the release decision and the regulation objectives highlighted in section 4.2.

The mass balance equation enables the description of the dynamics of the reservoir storage, it is defined in the problem formulation through eq. (3.4) and explicitly formulated as follows:

$$s_{t+1} = s_t + \varepsilon_{t+1} - w_{t+1} \tag{4.1}$$

In this formula, s_t is the lake storage at time t, ε_{t+1} is the net inflow volume, that accounts not only for the total distributed inflows but also for evaporation loss and other factors, and w_{t+1} is the outflow volume. The state transition model (eq. 4.1) is updated at a daily time step, while the two quantities w_{t+1} and ε_{t+1} are defined as the integrated flow over the 24 hours interval $[t, t+1)$; thus, they are known only at time step $t+1$ and the integrated water balance is estimated with an hourly time step within the daily storage dynamic.
Assuming that the volume of the lake can be well approximated by the formula for calculating the volume of a cylinder, the reservoir s_t and the level h_t at the Malgrate hydrometer are related by the following relationship, where $S_{\text{lake}} = 145900000$ m^2:

$$h_t = \frac{s_t}{S_{\text{lake}}} - 0.4$$

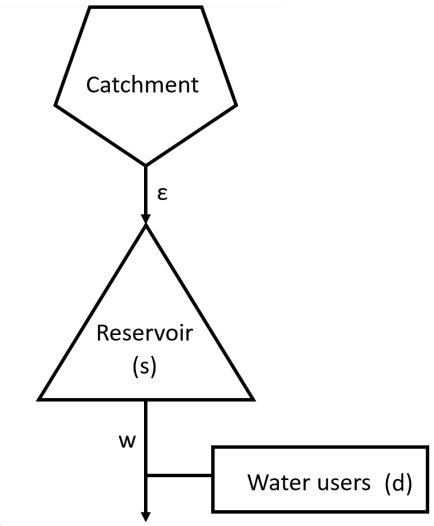


Figure 4.3: Schematic representation of the modeled water system of Lake Como

.

The release follows the control action a_t, which is taken at time t, but the real outflow volume w_{t+1} is subject to some constraints. More precisely, the latest lake regulation boundaries are set with respect to the Malgrate's hydrometer, with the lower limit at -0.4 m and the upper one at 1.1 m. Below the minimum level, the release is stopped to avoid consequences such as impeding navigability, damaging the natural environment, and fulfilling some sanitary measures, like impeding the emersion of civil and industrial drains. The upper limit is instead the reference level to define floods in Piazza Cavour in Como; over this lake level, it is mandatory to open completely the bulkheads. Moreover, the real discharge can't exceed $w^{\text{nat}}(h_t)$, the function describing the relationship between discharge and level in natural flow conditions. The release is also subject to fulfil the Minimum Environmental Flow, if a sufficient inflow to the lake is available, i.e. $q_{t+1}^{MEF} = \min\left(q^{ef}, \varepsilon_{t+1}\right)$, where the minimum between the current MEF value ($q^{ef} = 22$ m^3/s) and the available inflow ε_{t+1} has to be released. The combination of legislative and natural constraints defines the relationship between a_t and w_{t+1}. Both variables must remain within the decision space, see fig. 4.4, bounded by the minimum and maximum release. Making use of the variables $w_{t+1}^{\max}$,$w_{t+1}^{\min}$ and w_{t+1}^{ar} for some intermediate steps, these first constraints can be expressed by the following set of equations:

$$w_{t+1}^{\max} = \begin{cases} 0 & h_t < -0.4 \\ w^{\text{nat}}(h_t) & h_t \geq -0.4 \end{cases}$$

$$w_{t+1}^{\min} = \begin{cases} 0 & h_t < -0.4 \\ \min(q_{t+1}^{\text{MEF}}, w^{\text{nat}}(h_t)) & h_t \geq -0.4 \end{cases}$$

$$w_{t+1}^{\text{ar}} = \min(w_{t+1}^{\max}, \max(w_{t+1}^{\min}, u_t))$$

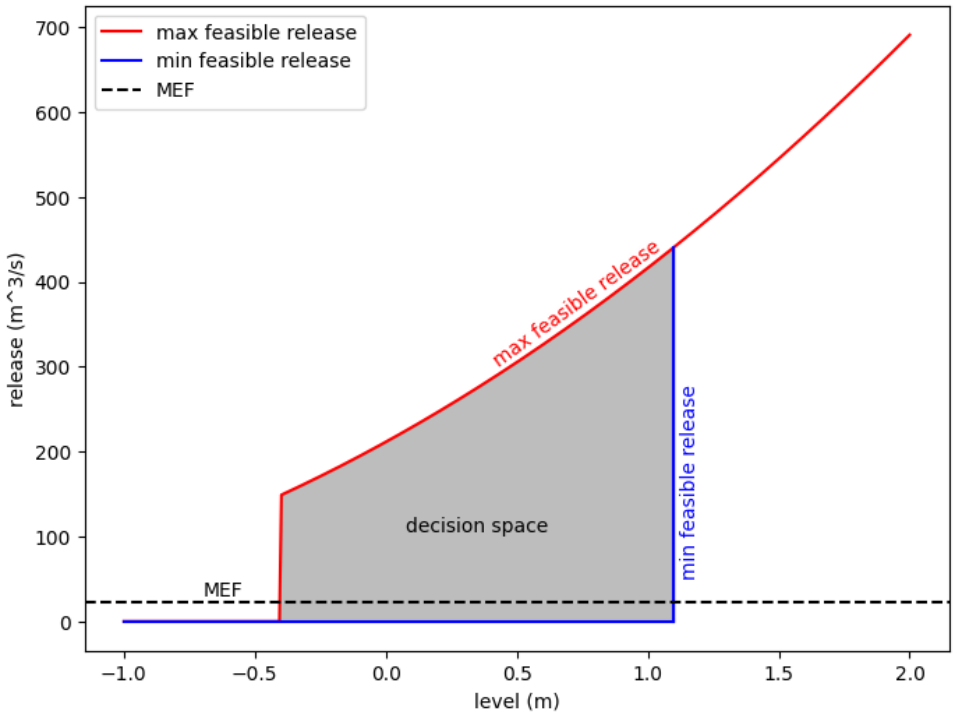


Figure 4.4: Operational discretion zone for lake operations, delimited by the maximum and minimum feasible release functions

.

The variable w_{t+1}^{ar} is not the real outflow volume because the dam opening operation also has some physical and operational constraints. The physical limitation is imposed by the bulkhead opening speed so that the water flow can increase up to only 30 m^3/s every 2 hours. This physical constraint is declined into two different operational constraints based on the lake level: when the lake is in flood risk conditions ($h_t \geq 0.8$ m), the dam operation functions the whole day, allowing for a maximum release increase of 360 $m^3/s/day$; instead, if the lake level is low (below 0.8 m), the dam operations are carried out only for 16 hours per day, reducing the maximum daily increase to 240 $m^3/s/day$. These constraints don't apply during dam closing and outside the regulation boundaries.

The following equation formalizes the constraints described above:

$$w_{t+1} = \begin{cases} \min(w_{t+1}^{\text{ar}}, w_t + 240) & h_t < 0.8 \\ \min(w_{t+1}^{\text{ar}}, w_t + 360) & 0.8 \leq h_t \leq 1.1 \\ w_{t+1}^{\text{ar}} & h_t > 1.1 \end{cases}$$

To summarize, $w_{t+1} = f(s_t, a_t, \varepsilon_{t+1}, w_t)$ is the equation describing the actual release dynamic, with f a nonlinear relation describing all these constraints.

4.4. Objectives

The regulation objectives described in section 4.2 are evaluated through indicators related to flood control, water supply, and low levels. In the following equations, the simulated time horizon is defined by H, and the period T is the hydrological year. The index t indicates the time step, i.e. the day in the daily-step simulation, ranging from 0 to H, while τ refers to the day of the year, thus limited between 1 and 365.
These objective indicators are used to evaluate the control performance in the objective space and thus must be coherent to what the optimal control algorithm learns to minimize, see section 3.1.

Flood Control The first objective is related to flood frequency control and the indicator is defined as the annual average number of days [day/year] when a flood occurs. The flood threshold has been defined as $h^{flo} = 1.1$ m.
The objective is computed as:

$$J^{flo} = \frac{1}{H/T} \sum_{t=0}^{H-1} g_{t+1}^{flo} \tag{4.2}$$

where:

$$g_{t+1}^{flo} = \begin{cases} 1 & \text{if } h_{t+1} > h^{flo} \\ 0 & \text{otherwise} \end{cases}$$

Water Supply The second objective is related to the deficit of the water demand d_τ, assumed to be deterministic and cyclostationary, and the indicator is defined as the daily

average of water supply deficit equivalent $[(\mathrm{m}^3/\mathrm{s})^{\mathrm{eq}}/\mathrm{day}]$, computed as:

$$J^{def} = \frac{1}{H}\sum_{t=0}^{H-1} g_{t+1}^{def} \tag{4.3}$$

where:

$$g_{t+1}^{def} = \left(\max\left(d_\tau - \left(r_{t+1} - q_{t+1}^{MEF}\right), 0\right)\right)^{\beta_\tau}$$

where β_τ is a time-varying exponent that penalizes with different importance the deficit during summer and winter. This latter parameter is selected to emulate the decision-making process of the operator so that the deficit is squared during summer (from 1$^{\mathrm{st}}$ of April to 10$^{\mathrm{th}}$ of October), while $\beta_\tau = 1$ during winter. Common practice is to use a squared deficit for supplies of the cultivated area because the crop is more sensitive to higher water lack rather than more frequent but smaller ones. Water demand is mainly driven by agriculture during summer and by hydropower production during winter, this combination has led to this time-varying exponent that shifts the deficit from summer to winter, where it is less expensive for agriculture.

Low-level Control The third objective is related to the avoidance of low levels, and the indicator is the annual average number of days [day/year] when the lake level goes below a static low-value threshold. The low-level threshold has been defined as $h^{low} = -0.2$ m. The objective function is computed as:

$$J^{low} = \frac{1}{H/T}\sum_{t=0}^{H-1} g_{t+1}^{low} \tag{4.4}$$

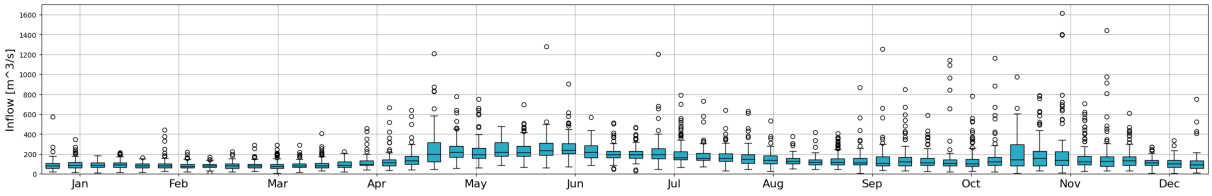
where:

$$g_{t+1}^{low} = \begin{cases} 1 & \text{if } h_{t+1} < h^{low} \\ 0 & \text{otherwise} \end{cases}$$

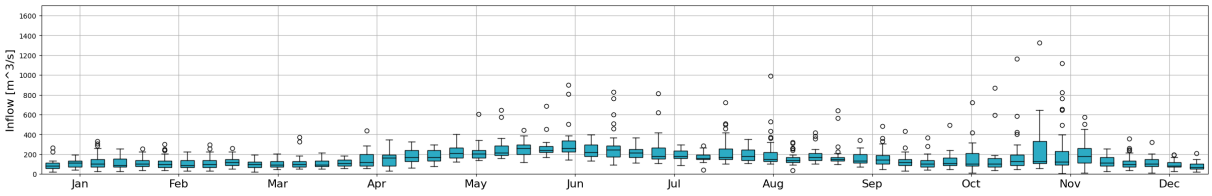
4.5. Inflow Data

The available measurements, provided by Consorzio dell'Adda, are the time series of the lake level at the hydrometer of Malgrate and the release of the regulated lake. The net inflow is calculated at a daily time step by inverting the mass balance equation (eq. 4.1). As schematized in fig. 3.3, the available dataset (1999-2018) is split into training and validation sets, respectively 75% for the training set, from 1999 to 2013 (fig. 4.5a), and 25% for the validation set, from 2014 to 2018 (fig. 4.5b).

It should be noted that in the training set, more than in the validation set, there are numerous inflow values due to extreme events that go beyond the lake's physical regulation capabilities, causing inevitable flooding events, in the case of heavy inflows. Among the most important recorded in the dataset is the inflow of November 27, 2002, which caused one of the highest historical floods recorded in Piazza Cavour.
Some of these extreme inflow values have a supply of water that cannot in any way be released quickly enough to avoid flooding in Como, it is clear comparing maximum release in the operating space in fig. 4.4 with the highest inflow peaks in fig. 4.5a. Therefore neither the perfect policy, section 3.4.1, nor the past control of the lake operator was able to avoid these floods.



(a)



(b)

Figure 4.5: Historical net inflow data represented with weekly boxplots, training set trajectory from 1999 to 2013 (a), and validation set trajectory from 2014 to 2018 (b)

5 | MOFQI - Case study results

This chapter reports the results obtained applying the MOFQI algorithm, as presented in **Methods and Tools**, to the optimal control problem for the Lake Como system, described in **Case of study**.
The presentation of the results follows the workflow diagram shown in fig. 3.3. Then the sensitivity analysis on the hyperparameters is presented before the benchmarking on the SDP results.

5.1. Optimization and Control Results

5.1.1. Transitions Sampling

The Lake Como model, which simulates the dynamic of the system, is used to create the experience dataset containing the set of four-tuples $\langle \mathbf{s}_t, a_t, \mathbf{s}_{t+1}, \mathbf{R}_{t+1} \rangle$.
To do this, the entire state-action space is sampled to cover the part of the decision space more densely, see fig. 4.4. The extremes of the sampled action space are the minimum release recorded in the historical time series i.e. 15 m^3/s and the upper bound is the maximum release value in the decision space i.e. $w^{\text{nat}}(1.1 \text{ m}) = 440.865 \text{ m}^3/\text{s}$, according to the maximum release curve. The maximum release is equal to the natural flow when the level is above -0.4 m, thus it is proportional to the hydraulic head and represented through the following equation:

$$w^{\text{nat}}(h) = 33.37(h + 2.5)^{2.015}$$

The action space is then sampled with 10 values uniformly selected between 15 and 440.865.
On the other hand, the state sampling interval is wider than the operational zone limits, i.e. $[-0.4; 1.1]$. The extremes of the sampling interval are -0.4 and 3.5 to have an approximation of the action-value function on all states reachable by the system. Sampling the state space more densely inside the operational area and coarsely outside of it gives more

relevance to the approximation of the action-value function within the area of interest. Therefore the state space is sampled with 15 values, 13 of these samples are inside the operational interval and the other 2 samples are between 1.1 m and 3.5 m.

The state-action space is sampled with a grid of 150 points, resulting from the combination of the 10 action samples and 15 state samples, as shown in fig. 5.1. For each one of these state-action couples, the system transition is simulated for all the daily inflow recorded in the 15 years of historical trajectory, as schematized in fig. 3.1.
Considering the leap days in the training dataset trajectories, i.e. from 1999 to 2013, the period is $T = 365.2\bar{6}$ and the total amount of four-tuples sampled through the system simulation is equal to $821850 = 15 \times 365.2\bar{6} \times (10 \times 15)$, computed by eq. (3.14).

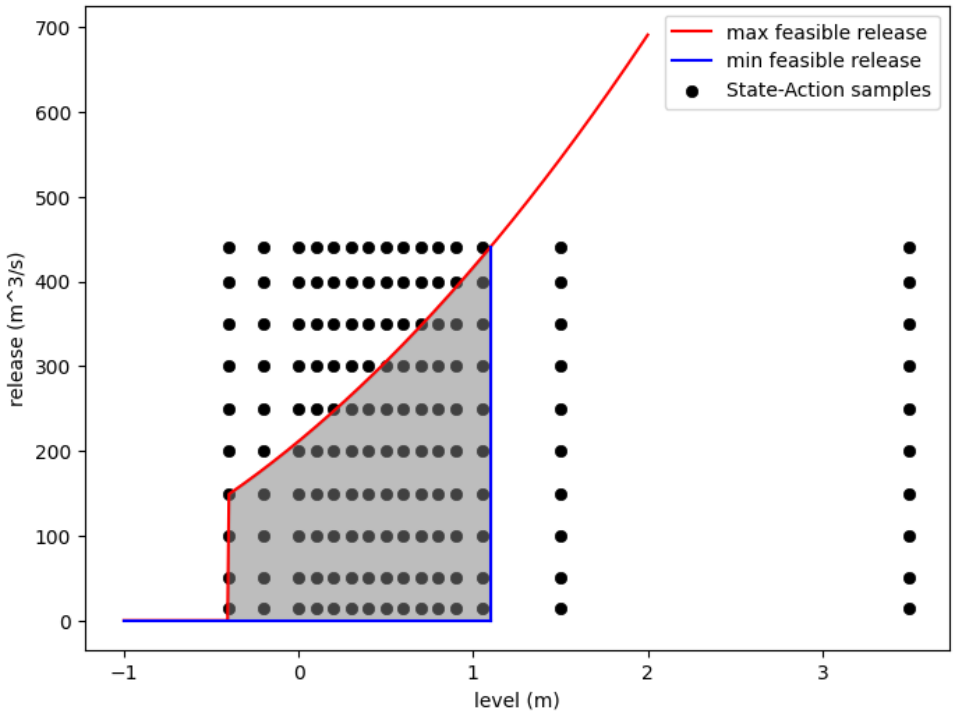
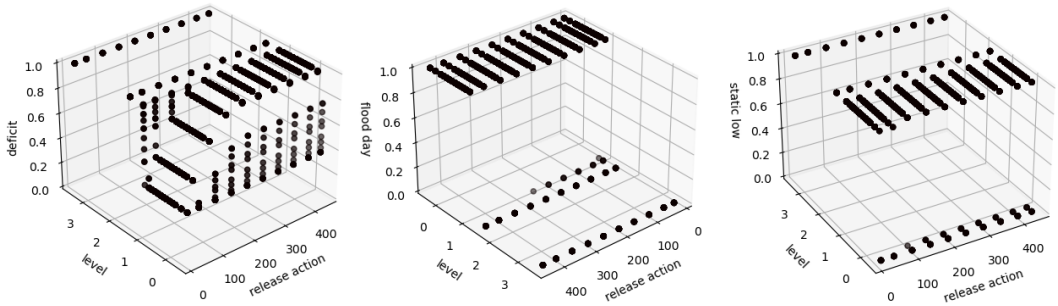


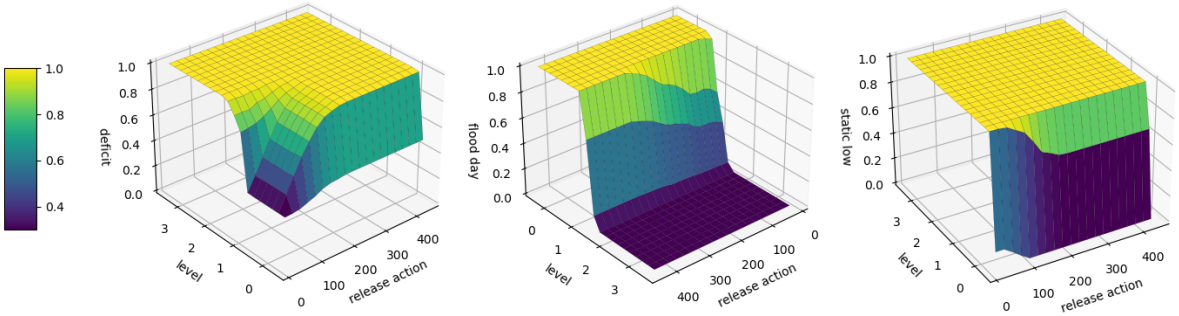
Figure 5.1: State-Action space grid samples

Simulating the system transition implies also computing the step costs; so each transition tuple include the reward vector $\mathbf{R}_{t+1} = \left[R^{flo}_{t+1}(\cdot), R^{def}_{t+1}(\cdot), R^{low}_{t+1}(\cdot)\right]$.
Since the optimal control solution concerns the convex sum of rewards of eq. (3.6). R^{flo}_{t+1}, R^{def}_{t+1}, R^{low}_{t+1} are normalized in interval $[0; 1]$ to rescale the different magnitude that characterizes each step-cost function. As an example, in fig. 5.2a are reported the normalized reward samples for the 135^{th} day of the year for the three different rewards, i.e. each plot represents $15 \times (10 \times 15)$ samples; while in fig. 5.2b is represented the estimated single-objective reward using Extra-Trees computed through a regression on samples reported

in fig. 5.2a. This surface could be also interpreted as the first FQI iteration, i.e. the expected reward $r(s, a)$, for the three objectives separately.
For the multi-objective algorithm MOFQI the surface estimated by regression has six dimensions, in addition to the state, the action and the day of the year the surface has three additional dimensions, one for each objective weight.



(a)



(b)

Figure 5.2: Single-objective normalized reward samples in state-action space, for the three case study step costs, for $\tau = 135$ (a). Single-objective normalized reward approximation through Extra-Trees in state-action space for $\tau = 135$ (b)

To build the dataset $\mathcal{F}$ for the multi-objective algorithm, defined in eq. (3.8) and represented in table 3.1, the reward values must be aggregated according to the weight sampling technique. As described in section 3.3 two techniques are tested.
The first method, chosen $K = 7$, involves sampling 7 different points of the 2-dimensional simplex surface for each transition tuple sampled, thus the dataset cardinality is $\#F = 7 \times 821850$.
The second method allows the entire simplex area to be sampled while keeping the size of the dataset equal to the number of transition tuples in the system, i.e. $\#F = 821850$. Both the Gaussian-based and the uniform distribution were tested and compared to each other, as well as compared to the fixed simplex sampling technique. In the next section, these methods and distributions and their effects on convergence to the optimal control

solution will be explained in detail.
Once the reward is aggregated through the convex sum according to the weights sampled, the dataset $\mathcal{F}$ is provided as input to the MOFQI algorithm, see algorithm 3.1.

5.1.2. Multi-Objective Fitted Q Iteration

The MOFQI algorithm iteratively updates the action-value function approximation $\hat{Q}_h$ as a result of regression $\mathcal{R}$ on the training set $\mathcal{TS}$. The training set is simultaneously updated according to the off-policy bootstrap operator defined in eq. (3.10), which updates the Q^l samples relying on the previous estimate $\hat{Q}_{h-1}$.
Initializing $\hat{Q}_0$ as zero, the first iteration returns the approximation of the expected reward $\hat{Q}_1 = r(\mathbf{s}, a)$. Then, successive iterations, according to the discount rate γ estimate the expected discounted sum of future rewards by iteratively extending the time horizon of the temporal aggregation.
The maximum number of iterations $\bar{h}$ is based on the metrics defined in section 3.4.2, the normalized HV and the mean absolute increase of the $\hat{Q}_h$ function evaluated through algorithm iteration. Results show that the number of iterations strictly depends on the discount rate, if $\gamma < 1$ the discounted value rapidly goes to zero by increasing the iteration since it depends on γ^h. Lower gamma values lead to lower iteration needed to estimate most of the magnitude of the action-value function, thus converging more rapidly to the real Q function.
On the other hand, the HV metric evaluates the performances of the set of policies $\pi_h^{\boldsymbol{\lambda}}(\mathbf{s}) = \arg\max_a \hat{Q}_h^{\boldsymbol{\lambda}}(\mathbf{s}, a)$ with $\mathbf{s} = [s, \tau, \boldsymbol{\lambda}]$ generated by sampling an adequate amount of weight vectors $\boldsymbol{\lambda}$ for estimated model $\hat{Q}_h$. The number of 2-dimensional simplex samples to adequately evaluate the Pareto front for the estimated model results to be 500, more in detail analyzed and described in the next subsection.
The HV metric evaluates the convergence in terms of performance thus assessing whether the learning process actually learns better control; this metric also defines the best model in terms of performance, which corresponds to the one that produces the maximum value of HV at the end of iteration $\bar{h}$. HV metric depends on many hyperparameters and settings of the algorithm, the most relevant ones are the discount rate γ, the weight space sampling technique and the number of Trees M. In the next section, an analysis of these hyperparameters assesses the best combination of these; summarized in table 5.1, to reach good performances both on the training and validation set.

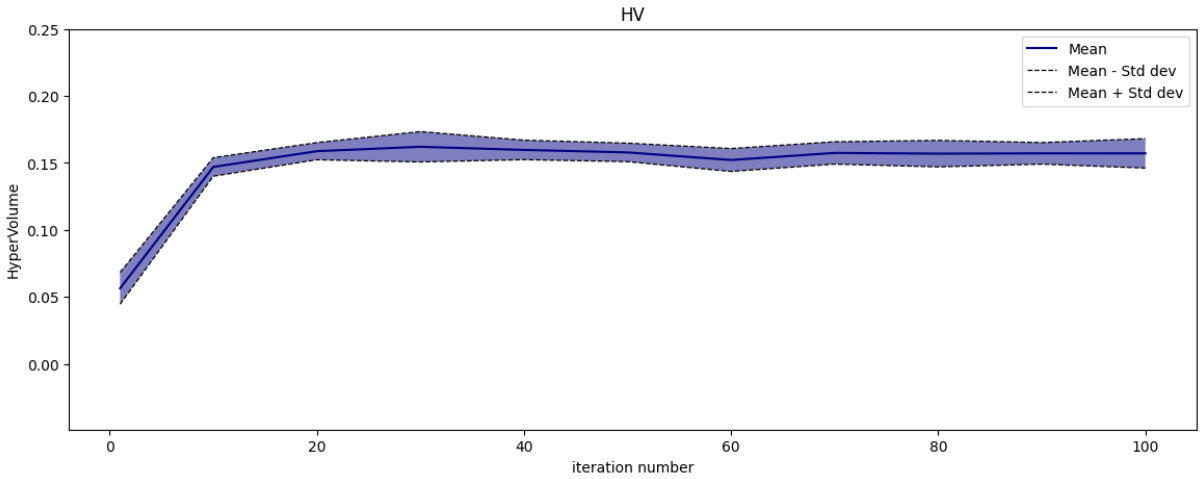
The results of HV and $\Delta_{\hat{Q}}$ metrics computed for the best parametrization are shown in fig. 5.3, calculated for 100 iterations. Since the random simplex sampling depends on the random extraction of values, 10 different models with different random seed initializations

discount rate γ	0.85
Trees number M	50
Simplex sampling technique	Random sampling

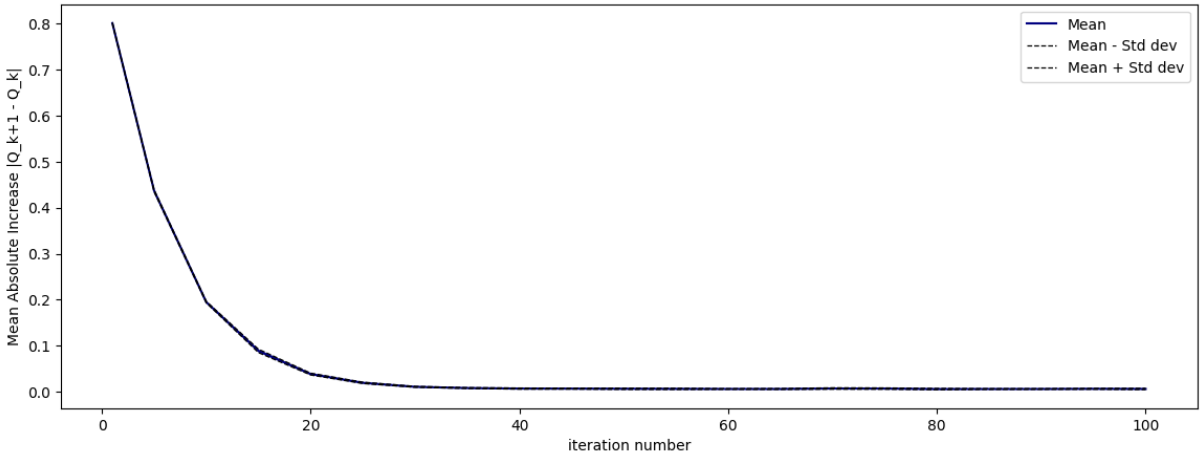
Table 5.1: Hyperparameters and weight sampling techniques that lead to the highest performance in terms of normalized hypervolume metric.

are trained and evaluated as sensitivity analysis with respect to the random sample generation.
The fig. 5.3a show HV metric average value and standard deviation; the hypervolume effectively increases until the 30^{th} iteration, with significant growth in the first 10 iterations, then up to the 30th it grows slowly and settles to an asymptotic value. The different seeds do not affect the maximum iteration $\bar{h}$, but slightly affect the HV metric value at the convergence, generating a variance in the final result. Reaching the 30^{th} iteration could also be interpreted as the number of days taken into account to estimate the action-value function and thus to make the decision since the control has a daily cadence. As anticipated, the HV grows and the slope is mainly determined by the discount rate γ.
The fig. 5.3b show the average absolute increase $\Delta_{\hat{Q}}$ defined in eq. (3.16) of the estimated action-value function between two successive iterations, computed for 10 different models. The slope decreases exponentially and proportionally to 0.85^h; coherently with the HV increases the $\hat{Q}$ function grows ends around the 30^{th} iteration. It must be noticed that the standard deviation of this last metric is around zero, thus the random seed does not affect the $\hat{Q}$ increases. The $\Delta_{\hat{Q}}$ metric decreases exponentially but never vanishes, this is due to the Extra-Trees regressor used to estimate $\hat{Q}$ since at each iteration the algorithm regenerates non-parametric regressors from scratch, thus the approximated function fluctuates around an average surface value due to the randomness of Extra-Trees generation.

Empirical results show that the MOFQI algorithm for the formulated optimal control problem effectively converges. As previously highlighted, even if the random sampling associates each transition tuple to a different weight vector $\boldsymbol{\lambda}$, depending on the random seed, this does not affect the $\hat{Q}_h$ grow. On the contrary, the random simplex sampling slightly affects the performance of the resulting policy, it generates a variance in the hypervolume metric. Therefore, among the 10 different $\hat{Q}_{\bar{h}}$ models the best performing one is determined by evaluating which one has good performances on both the training and validation set.



(a)



(b)

Figure 5.3: Training convergence evaluation metrics for ten different random seeds, see section 3.4.2, hypervolume metric (a) and action-value function improvement (b)

5.1.3. Policy Evaluation and Validation

Once the action-value function model $\hat{Q}_{\bar{h}}$ is defined, the policies $\pi_{\bar{h}}^{\boldsymbol{\lambda}}$ could be simulated for any weight vector $\boldsymbol{\lambda}$ and for the given net inflow trajectory which could be the training or validation one.

Assigning a weight value to each objective and simulating the calibrated model's policy allows for calculating the associated step costs. Following this, by aggregating step costs, it is possible to compute the objectives indicator $[J_{\boldsymbol{\lambda}}^{flo}, J_{\boldsymbol{\lambda}}^{def}, J_{\boldsymbol{\lambda}}^{low}]$, thereby quantitatively assessing the simulated policy's performance with respect to all three objectives.

Scenarios with various combinations of vectors $\boldsymbol{\lambda}$ can be simulated for a single model of

the action-value function, leading to different corresponding values for the three objective indicators. A collection of coordinate vectors $[J^{flo}, J^{def}, J^{low}]$ is the starting point from which the Pareto-dominant points that constitute the Pareto frontier are then obtained for a single model $\hat{Q}_{\bar{h}}$.

To find the set of points $S_{\hat{Q}_{\bar{h}}}$ which defines the Pareto front approximation for the $\hat{Q}_{\bar{h}}$ model, only points $p_{\boldsymbol{\lambda}} = [J^{flo}_{\boldsymbol{\lambda}}, J^{def}_{\boldsymbol{\lambda}}, J^{low}_{\boldsymbol{\lambda}}]$ which are not Pareto dominated among all the resulting coordinate vectors need to be considered. Therefore, to obtain the Pareto front approximation, those points that are not Pareto dominated must be selected among a set of points which cover different areas in the objective space; this set is generated by sampling a number N_{ws} of weights and simulating the control $\pi^{\boldsymbol{\lambda}^i}_{\bar{h}}$ for each sampled weight vector $\boldsymbol{\lambda}^i$ with $i = 1, \ldots, N_{ws}$.

To increase results robustness the HV metric is computed on a validation set, to observe the behaviour of the control performance on a different net inflow trajectory which is not used to estimate the action-value function model. As schematized in the diagram reported in fig. 5.4 the HV metric in validation is computed on a set of points $S^{\text{val}}_{\hat{Q}_{\bar{h}}}$. To generate the validation set, first, N_{ws} policies must be simulated on the training trajectory, then in the objective space must be selected the Pareto-dominant points $p^i \in S^{\text{train}}_{\hat{Q}_{\bar{h}}}$ where $\{p^i | i = 1, \ldots, \#S^{\text{train}}_{\hat{Q}_{\bar{h}}}\}$ and $\#S^{\text{train}}_{\hat{Q}_{\bar{h}}}$ is the cardinality of $S^{\text{train}}_{\hat{Q}_{\bar{h}}}$. Subsequently, the policies composing the set $S^{\text{train}}_{\hat{Q}_{\bar{h}}}$ must be simulated on the validation scenario, then among the resulting set of points only the Pareto-dominant ones compose the validation set $S^{\text{val}}_{\hat{Q}_{\bar{h}}}$.

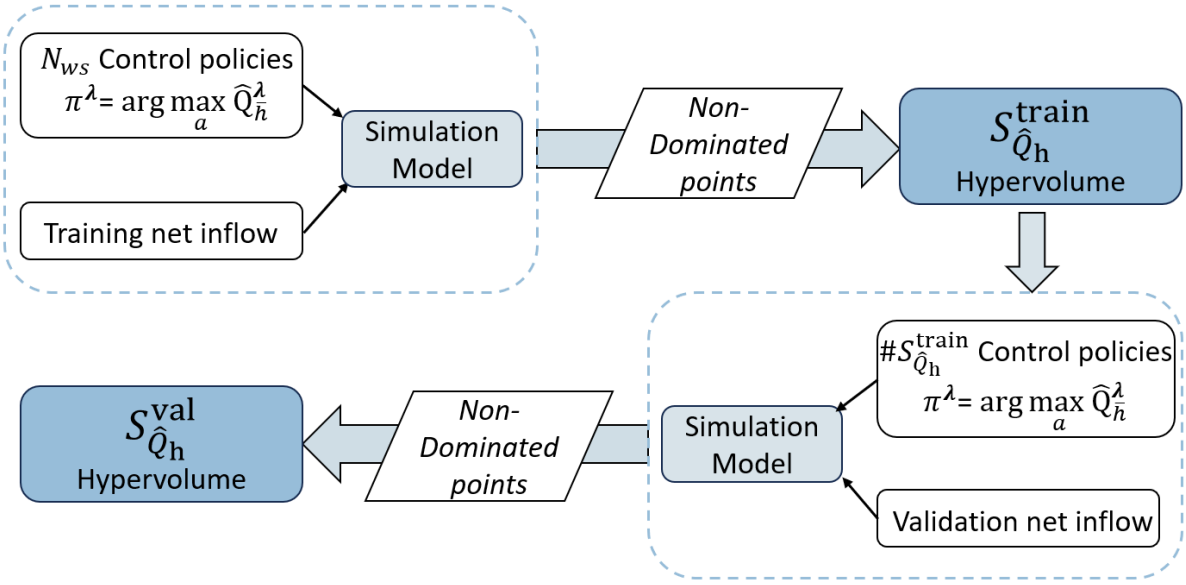


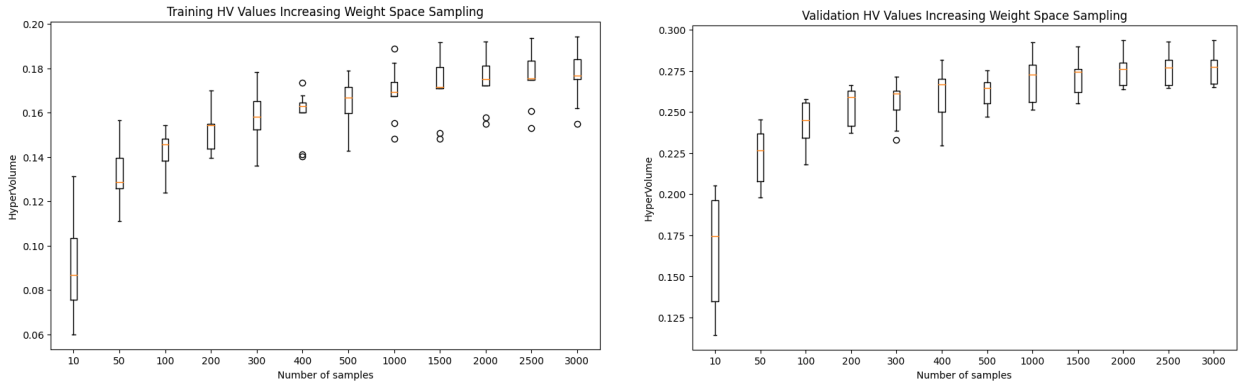
Figure 5.4: Diagram of procedure for generating the set of points which approximate the Pareto front for the training net inflow and the validation net inflow

Selecting a higher number of initial weights samples N_{ws} leads to a more numerous set

of points $S_{\hat{Q}_{\bar{h}}}$ which allows exploring Pareto front tradeoffs widely and thus enlarging the score of HV metric. The fig. 5.5a reports the boxplots of HV measures by increasing the number N_{ws} of samples of the 2-dimensional simplex surface for the ten models trained with hyperparameters reported in table 5.1. Results show that the number of samples $N_{ws} = 500$ is enough to represent the Pareto front for the estimated model; this value is selected as a compromise between computational demand and the amount of Pareto front tradeoff points is the set $S_{\hat{Q}_{\bar{h}}}$.

The dependency of the validation hypervolume metric on the number N_{ws} of samples selected to determine the HV metric for the training set of points confirms the results obtained for the training HV metric; as shown in fig. 5.5b the number $N_{ws} = 500$ is still enough for the Pareto front representation.

It should be noted that the HV value between training and validation is not comparable. As anticipated in section 3.4.1, the value of hypervolume depends on a reference point in the objective space, moreover, the objectives indicators depend on the step costs generated with respect to the trajectory of net inflow which affects the system transitions. Thus the HV comparison can be performed only under the same disturbance trajectories, i.e. either training or validation. To strengthen the results, the comparison between training and validation scores is done on the trajectory of boxplots of this metric within the training and validation scenarios, with respect to the number of simplex samples.



(a) (b)

Figure 5.5: Hypervolume values at the 30$^{\text{th}}$ iteration, for ten different models, increasing the amount of simulated simplex samples, training dataset (a) and validation dataset (b)

Once the number of samples to be recorded has been defined to adequately represent the Pareto front generated by each model, from the resulting HV scores only one model is selected among the 10 trained ones, looking for the one that has a good rank both in validation and training, i.e. that performs well on both the training and validation

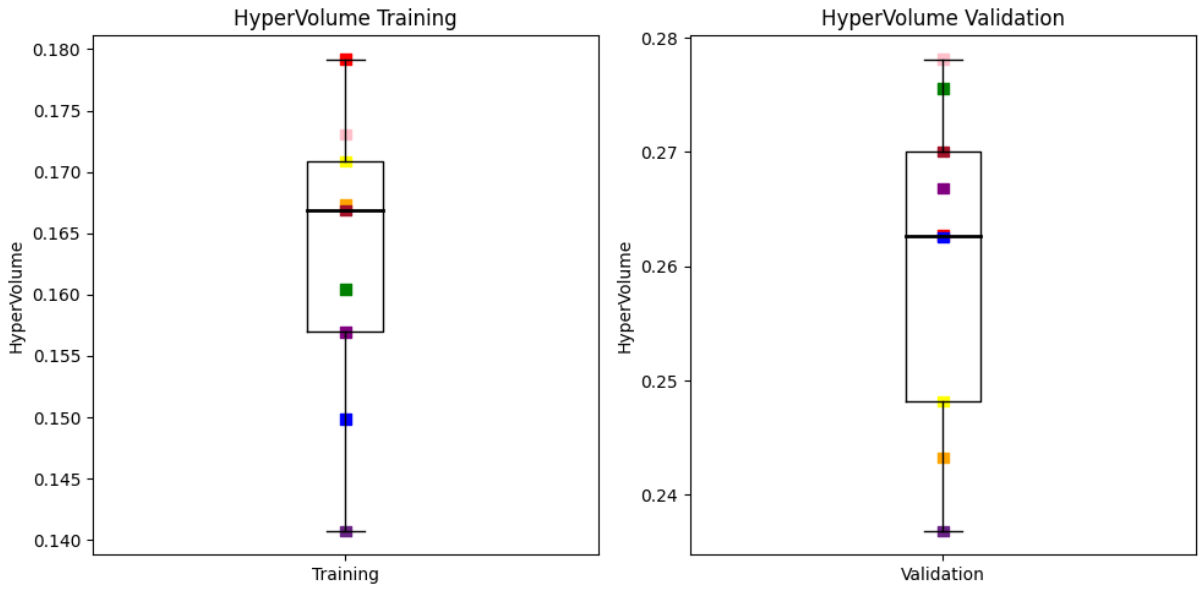
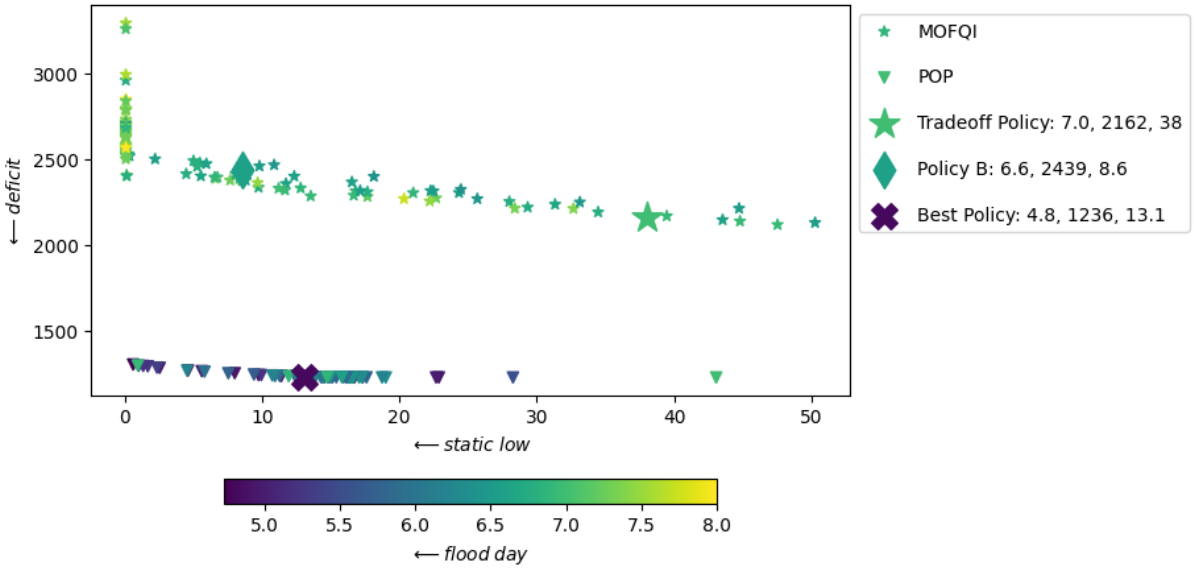


Figure 5.6: Hypervolume values at the 30$^{\text{th}}$ iteration, for ten different models, training dataset (a) and validation dataset (b)

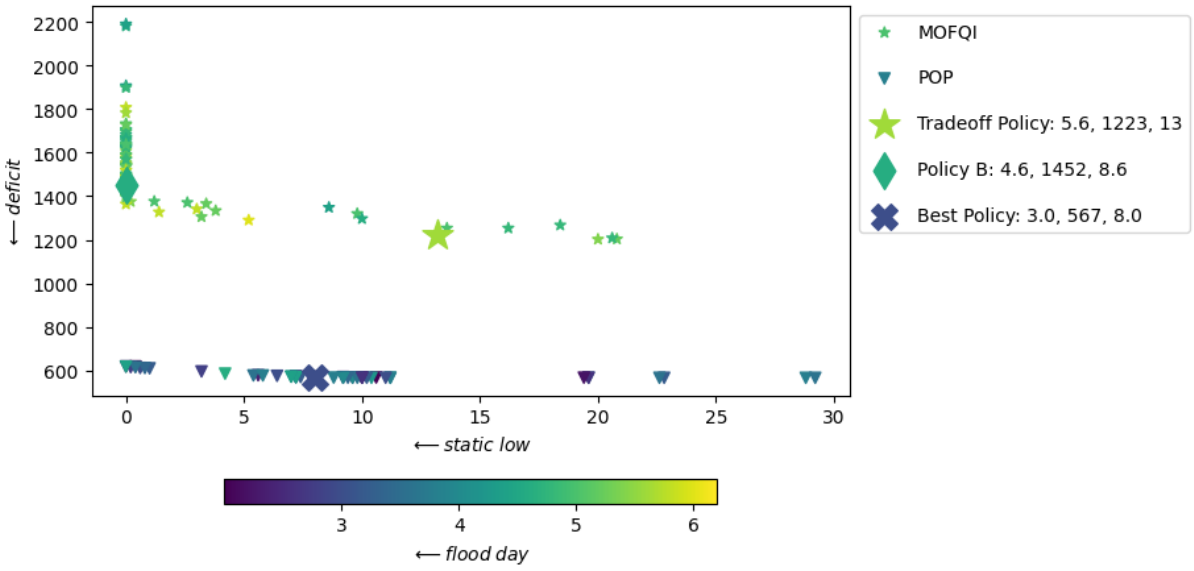
datasets. The results of the comparison are reported in fig. 5.6 which represent the HV score of every single model for $N_{\text{ws}} = 500$, in both the training and the validation boxplots. The model chosen from the 10 trained with the hyperparameters shown in the table 5.1 is ranked second in training and first in validation, where the ranking is done according to the HV metric that must be maximized. For the selected model the set of points in the objective space could be represented for training trajectory in fig. 5.7a, i.e. $S^{\text{train}}_{\hat{Q}_{\bar{h}}}$, and for the validation one in fig. 5.7b, i.e. $S^{\text{val}}_{\hat{Q}_{\bar{h}}}$.

Pareto front approximation is shown in fig. 5.7, it is represented together with the DDP Pareto front approximation both in training and validation.
According to the performance concerning the objective indicators, the reservoir operator could select the control policy to employ from the Pareto front set. As an example, a tradeoff policy is selected by choosing the closest to the best policy, i.e. the policy selected as the best among the DDP solutions. The Euclidean distance is calculated by rescaling the axes of the indicators in order to uniform the orders of magnitude, by dividing the value of J^{flo} by 10, the value of J^{def} by 500, and the value of J^{low} by 40. In addition, another policy far from the tradeoff one is selected among the same set of points, to compare the control laws and the dynamics of the system under the different types of policies and provide details on the interpretation of the control laws, this results are presented in the next section.

As previously described, training and validation HV results are not comparable since



(a)



(b)

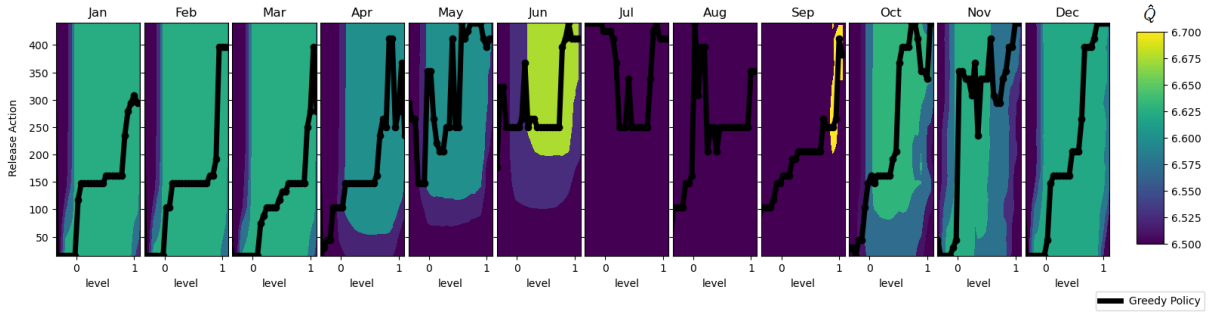
Figure 5.7: Pareto front approximation of MOFQI results at the 30$^{\text{th}}$ iteration, training $S^{\text{train}}_{\hat{Q}_{30}}$ (a) and validation $S^{\text{val}}_{\hat{Q}_{30}}$ (b)

these values depend on the stochastic disturbance that affects the system, hence different scenarios of net inflow lead to different distributions of the indicators, see fig. 5.7. Despite these differences, the shape of the Pareto front results similar for both the two series of net inflows which affect the system dynamics, hence the policies behave similarly regardless of the input trajectory with respect to the three objectives.
In both fig. 5.7a and fig. 5.7b the policies performance is widely spread on the deficit and static low indicators while are not expanded in the flood direction. Minimizing the flooding indicator is therefore consistent with minimizing the other two objectives. On the other hand, minimizing the deficit leads to higher average low-level days per year, highlighting the conflict between deficit reduction and the low level of the lake.
Furthermore, through the deficit indicator axes, there is a larger loss in performance with respect to the DDP solutions, since MOFQI and DDP Pareto fronts partially overlap on both the low-level and flood indicator axes but not on the deficit one; therefore this arrangement of the Pareto front suggests that the target of the deficit is the most difficult to minimise with the learning process adopted, compared to the potential improvement that is physically achievable and represented through the DDP set of points.

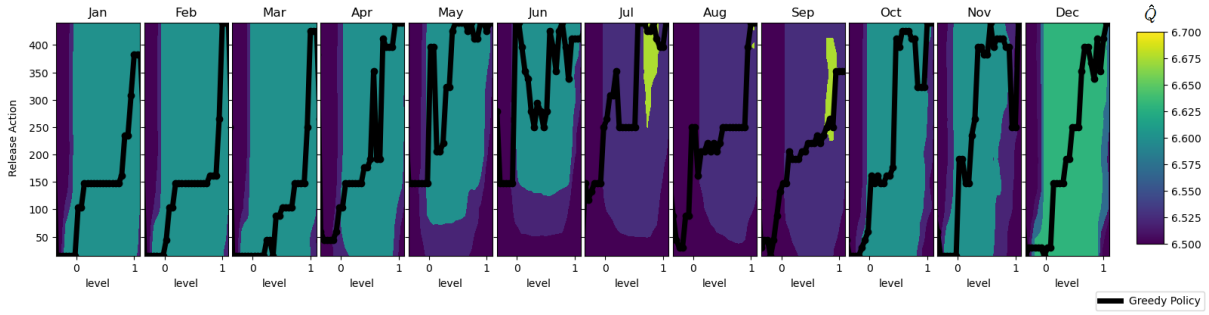
5.1.4. Control Results and Interpretability

Selecting a policy in the Pareto front set allows choosing the desired performances for each objective, this implies fixing a set of weights as input to the $\hat{Q}$ function. As an example, two policies $p^{\text{tradeoff}}, p^{\text{B}} \in S^{\text{train}}_{\hat{Q}_{\bar{h}}}$ are selected by fixing $\boldsymbol{\lambda}^{\text{tradeoff}}$ and $\boldsymbol{\lambda}^{\text{B}}$ to provide insights on the control results. Once the weight vector $\boldsymbol{\lambda}$ is defined, the resulting policy is a cyclostationary operating rule $\{\pi^{\boldsymbol{\lambda}}_{\tau}(\cdot); \tau = 1, \dots, T\}$ represented in fig. 5.8 for the two selected weight vectors.

In fig. 5.8a is represented the monthly average $\hat{Q}$ function which is estimated for the weight vector of the selected tradeoff policy, i.e. $\boldsymbol{\lambda}^{\text{tradeoff}} = [\lambda^{\text{flo}} = 0.463; \lambda^{\text{def}} = 0.428; \lambda^{\text{low}} = 0.109]$. In the same plot, through a black line, the greedy policy is also represented, i.e. $\pi^{\boldsymbol{\lambda}}(\mathbf{s}) = \arg\max_a \hat{Q}^{\boldsymbol{\lambda}}(\mathbf{s}, a)$, which defines the release action as a function of the level of the lake; thus the control is periodically defined. In the specific case scenario of $\boldsymbol{\lambda}^{\text{tradeoff}}$, the policy is the periodic operating rule $\pi^{\boldsymbol{\lambda}^{\text{tradeoff}}}$ which produces low deficit indicator value and high low-level indicator value, coherently with the position of the tradeoff policy in fig. 5.7.
Given the set of operating rules, the control could be simulated for a given net inflow trajectory. The tradeoff policy control results for the training trajectory of inflow, i.e. from 1999 to 2013, are represented in fig. 5.9; this result shows how the storage effectively stays within low-level and flood thresholds, except for particularly dry days mainly during



(a)



(b)

Figure 5.8: Set of monthly average $\hat{Q}^{\boldsymbol{\lambda}}(s,a)$ function values and the corresponding control law $\{\pi_\tau^{\boldsymbol{\lambda}}(\cdot);\tau=1,\ldots,T\}$ determined as greedy polici: $\pi^{\boldsymbol{\lambda}}(s)=\arg\max_a \hat{Q}^{\boldsymbol{\lambda}}(s,a)$. Tradeoff policy whith $\boldsymbol{\lambda}^{\text{tradeoff}}$ (a) policy B with $\boldsymbol{\lambda}^{\text{B}}$ (b)

the summer period. The low-level period coincides with the penalty period for the deficit since the tradeoff policy is selected among the ones that perform better for the deficit; therefore the deficit is the priority target at the expense of low-level days. On the other hand, flooding days occur in correspondence to the extreme events of net inflows. It is also shown that the control defined by the greedy policy has a higher variance compared to the current release which is the result of physical constraints on the release decision.

The current outflow could effectively fill the downstream demand curves also during the summer period, except for July, August, and September where the deficit is almost always above zero. This low water supply during these three months is partially due to adverse hydrological conditions since the net inflow is reduced even if the demand is still high and the deficit penalized. Moreover in October and November hydrological conditions due to rainfall lead to severe net inflow extremes usually leading to flooding conditions; thus emptying the reservoir before inflow extremes prevents the occurrence of flooding.
Coherently with the unfulfilled water demand, the low-level values, and the subsequent flooding periods, the estimate $\hat{Q}$ function highlights this period as the one with the lower value in the whole state-action space. As represented in fig. 5.8a, July, August, and

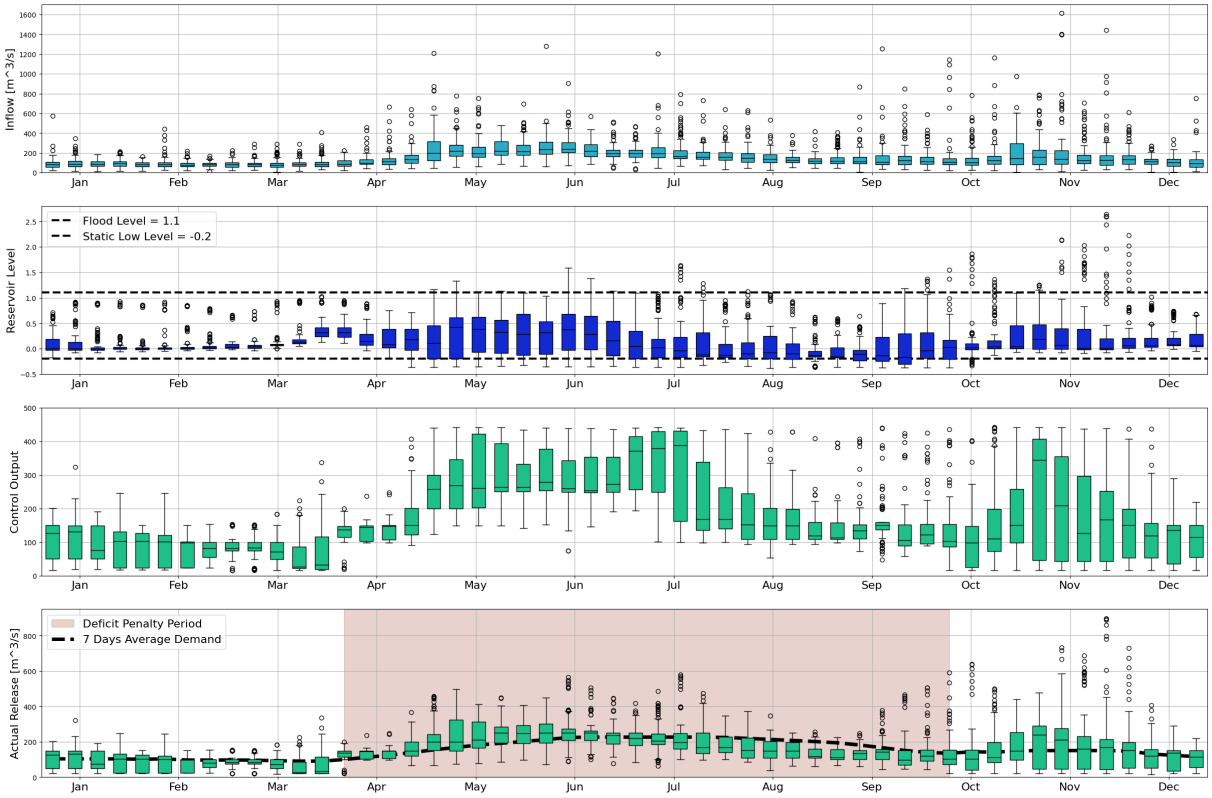


Figure 5.9: **Trade-off policy:** Simulation results, represented with weekly boxplots, using the training set trajectory from 1999 to 2014. The policy, selected as a trade-off within the finite approximation of the Pareto front, is generated through MOFQI with convergence reached at the 30$^{\text{th}}$ iteration. The resulting objectives values are $p^{\text{tradeoff}} = [J^{flo} = 7; J^{def} = 2161.59; J^{low} = 38]$

September are the months with the lower average value of the $\hat{Q}$ functions; thus indicating that for that period, states and actions carry a lower expected value of discounted rewards than for other periods.
On the contrary, in June, due to the coexistence of available water inflow due to snow melt and high water demand, this scenario led to the highest peak of $\hat{Q}$ values in the state-action space.

As a comparison, a second policy is selected among the Pareto front set, see fig. 5.7; policy B is selected to represent the part of the frontier that is furthest from the tradeoff policy, i.e. is a policy that produces lower low-level indicator value and higher deficit indicator value than the tradeoff policy. Selecting the corresponding $\boldsymbol{\lambda}^{\text{B}} = [\lambda^{\text{flo}} = 0.642; \lambda^{\text{def}} = 0.195; \lambda^{\text{low}} = 0.163]$ yields the set of periodic control laws $\{\pi_\tau^{\boldsymbol{\lambda}}(\cdot); \tau = 1, \dots, T\}$ reported in fig. 5.8b.
Weighting less the deficit has a direct impact on the action-value function surface. Dif-

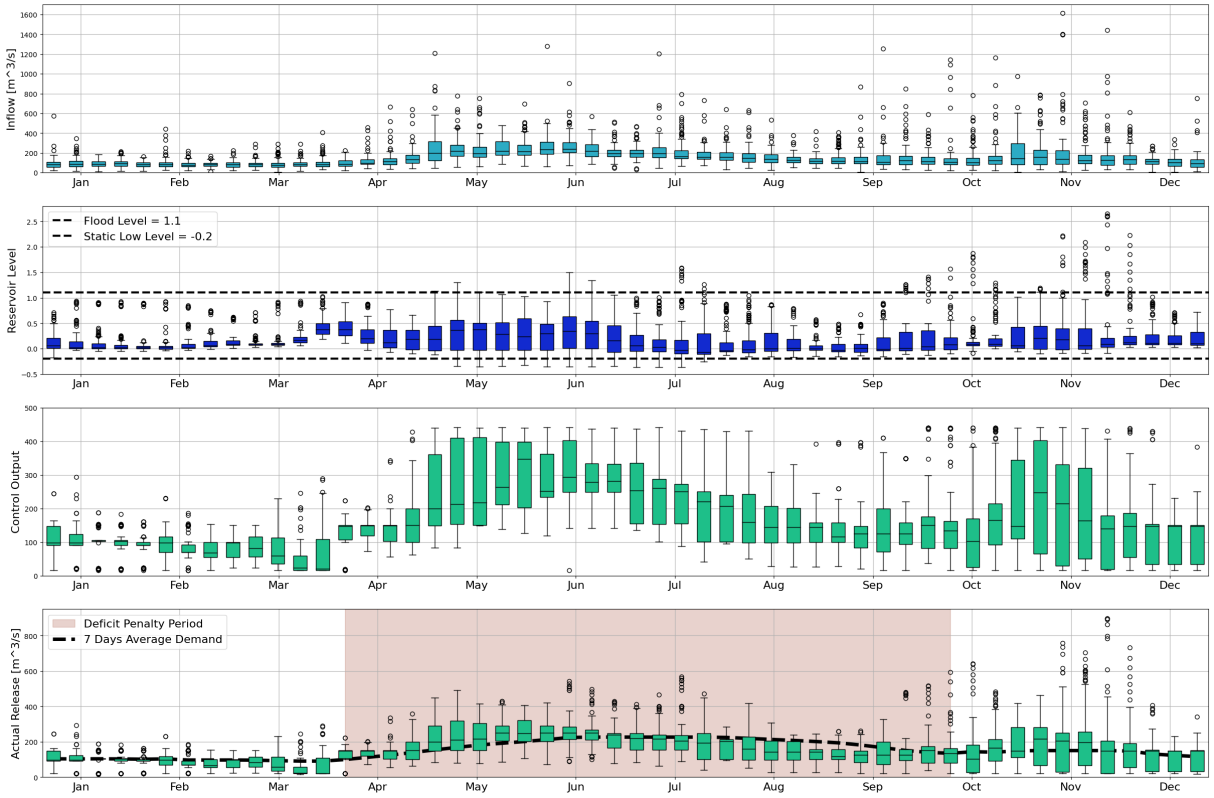


Figure 5.10: **Policy B:** Simulation results, represented with weekly boxplots, using the training set trajectory from 1999 to 2014. The policy, selected among the farthest from the trade-off one within the finite approximation of the Pareto front, is generated through MOFQI with convergence reached at the 30$^{\text{th}}$ iteration. The resulting objectives values are $p^{\text{B}} = [J^{flo} = 6.6; J^{def} = 2438.8; J^{low} = 8.6]$

ferences between fig. 5.8a and fig. 5.8b surfaces lead to a different greedy policy and a different control law. Policy B provides a slightly higher release decision for high lake level and lower release for low lake level, this became particularly evident comparing the two greedy policies of fig. 5.8 in March, July, and November months.

Policy B is employed to control Lake Como with the net inflow training trajectory, and results are reported in fig. 5.10. The impact of the policy on low-level days is evident, particularly during the summer period when there are significantly fewer such days compared to the tradeoff policy control shown in fig. 5.9. The average number of flooding days remains relatively consistent with the other control and the entire generated frontier ranges over a limited flooding indicator interval. However, a significant difference in deficits may not be immediately apparent from the graph. It can be seen that the release box plots are below the demand curve, even in July as well as August and September; as a result of a lower weight value on the deficit objective.

5.2. Algorithm Convergence and Hyperparameters Sensitivity

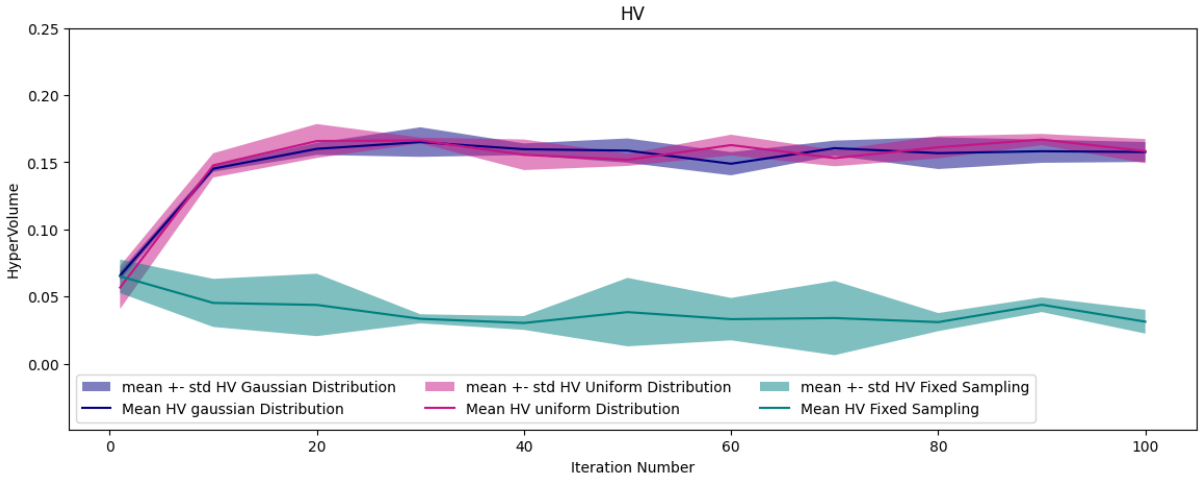
The current section provides results of the sensitivity analysis of hypervolume value in relation to the weights space sampling technique, the number M of Extra-Trees employed, and the discount rate γ.
Particularly, the two random sampling techniques proposed in the current thesis and described in section 3.3 are compared against the fixed weight space sampling in terms of convergence and hypervolume value. In addition to the Extra-Trees hyperparameters defined as stand out in the literature and described more in detail in section 3.2, this section defines the remaining one, i.e. the number M of Extra-Trees; therefore, we illustrate the dependency of hypervolume on the number M of trees. Finally, the sensitivity of the HV and $\Delta_{\hat{Q}}$ by changing the discount rate γ conclude the sensitivity analysis which carries out the best performing model.

5.2.1. Simplex Sampling and Convergence

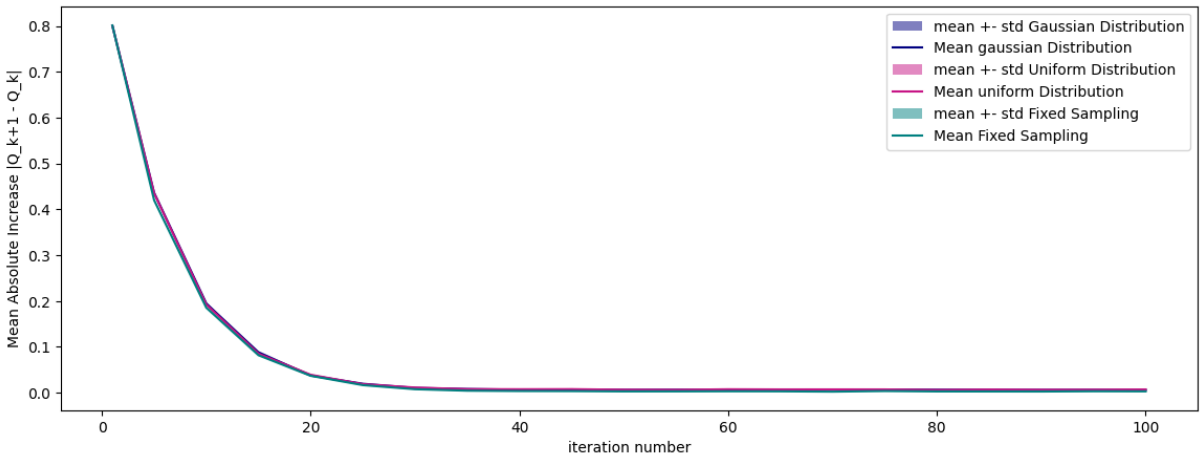
As shown in section 3.3, two different 2-dimensional simplex sampling methods were tested. The first consists of sampling a number K of values, in particular 7 for the current case study, and for each of them evaluating all the transition tuples generated with the Lake model. In this way, the MOFQI method counteracts the curse of multiple objectives but still significantly increases the size $\#F = K\mathcal{D}$ of the dataset linearly with the number K of weights samples.
In the current work, only 7 weights vectors are sampled, since accounting for more weights exacerbates the already time-demanding computation, thus with this technique the dataset dimension is $K\mathcal{D} = 7 \times 821850$.

The second technique, as illustrated in section 3.3, consists of exploring the whole simplex surface by sampling for each transition tuples a unique weight vector according to a random distribution. By doing so, the curse of multiple objectives vanishes since the dataset dimension $\#F = \mathcal{D}$ depends only on the state-action space sampling, thus the dataset dimension is $\mathcal{D} = 821850$.
In particular, for this second technique, two random distributions are tested as reported in fig. 3.2; one uniformly distributed over the 2-dimensional simplex, the other based on a multivariate Gaussian distribution centered in the middle of the simplex. This second distribution focuses on the simplex's central area, which corresponds to balanced tradeoff values of the three objectives.

To compare different approaches as reported in fig. 5.11, five models are generated for each of these, using different random seeds for the weights sampling. Then, different results are compared in terms of convergence evaluated with the previously defined metrics, HV and $\Delta_{\hat{Q}}$, and subsequently evaluated on the validation trajectory.
Surprisingly, the first sampling method does not converge in terms of hypervolume, i.e. it does not increase the performance on the three objectives. Probably sampling 7 weights combinations results insufficient for the 2-dimensional simplex, even if drastically increasing the size of the training set.
On the other hand, $\Delta_{\hat{Q}}$ shows an increase in the Q function, due to the discounted sum that is performed updating the training set, thus proportional to γ^h, i.e. 0.85^h.



(a)



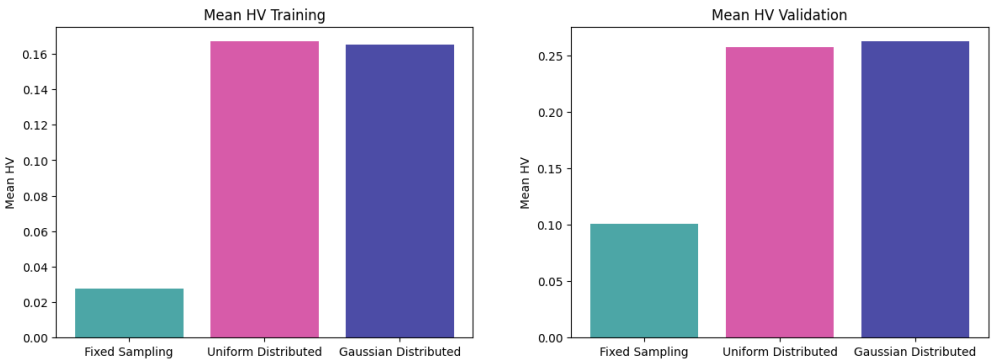
(b)

Figure 5.11: Training convergence evaluation metrics for five different random seeds, see section 3.4.2, for different simplex sampling techniques, hypervolume metric (a) and action-value function improvement (b)

Simplex random sampling, on the contrary, in addition to having seven times fewer tuples in the dataset, actually manages to converge in the objectives space, qualitatively following the trend of $\Delta_{\hat{Q}}$; thus MOFQI iteratively learns better control policies accounting for the future expected rewards.
In contrast, there appears to be no substantial difference between the two random sampling methods, they both converge equally to the same HV value. Since quantitative metrics do not show any difference, sampling based on Gaussian distribution is employed, following the theoretical idea behind the approach.

As expected and empirically shown by these results, the number of iterations required to achieve convergence of the MOFQI algorithm depends solely on the discount factor γ applied and shows no dependence on the type of dataset used in this comparison. The results therefore show convergence in both cases, but depending on the dataset, there is a convergence towards an action-value surface that actually generates better control, and thus the Pareto front also converges.

To strengthen the results obtained in training, the models at convergence are also evaluated on a validation dataset; fig. 5.12 show the consistency between the performance results on the training and validation datasets by comparing the average value of the HV metric of the five calibrated models for each sampling type. The results due to the random sampling control are confirmed to be better also in validation. The difference between the HV values of fixed and both the two random samplings is smaller in validation than in training. This difference in performance between the two scenarios may be due to the overfitting of the control on the training dataset.

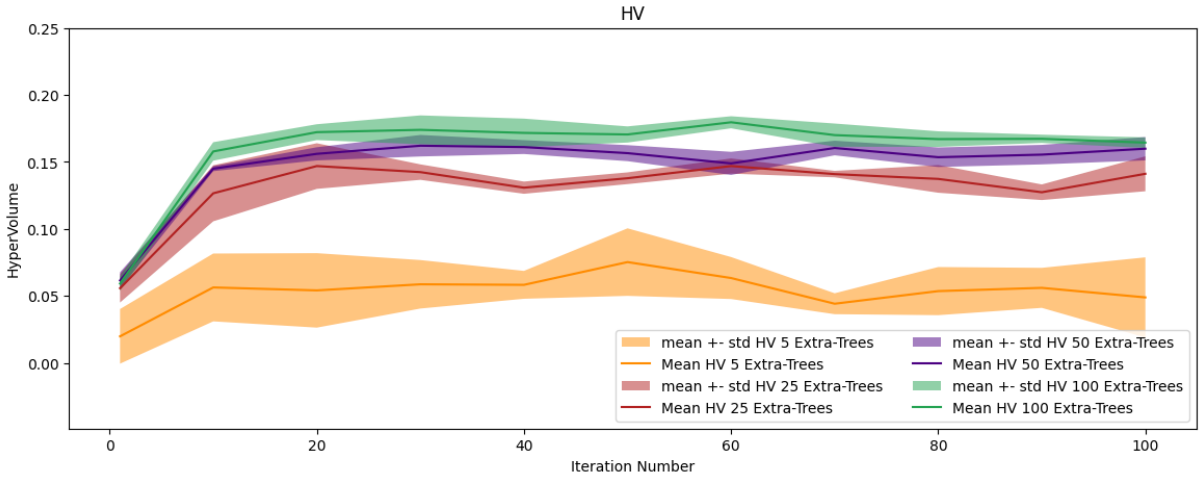


(a) (b)

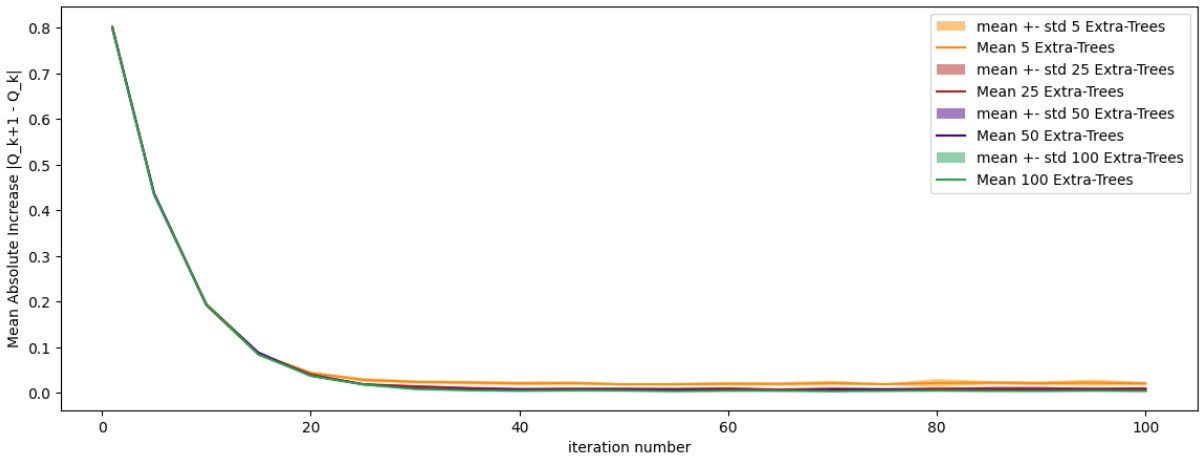
Figure 5.12: Mean hypervolume value at the 30^{th} iteration for different simplex sampling techniques evaluated for five different random seeds, training dataset (a) and validation dataset (b)

5.2.2. Regression Model and Convergence

As anticipated in section 3.2 and illustrated in published studies (Castelletti et al., 2010), the number of trees M affects convergence and performance, particularly by reducing the variance and discrepancy between training and validation.
These results are also empirically confirmed for this case study. In fig. 5.13 are reported the results using random simplex sampling based on the Gaussian distribution, the technique and distribution chosen, with four different values of M, and for each one 5 different models are trained, based on a different random seed.



(a)



(b)

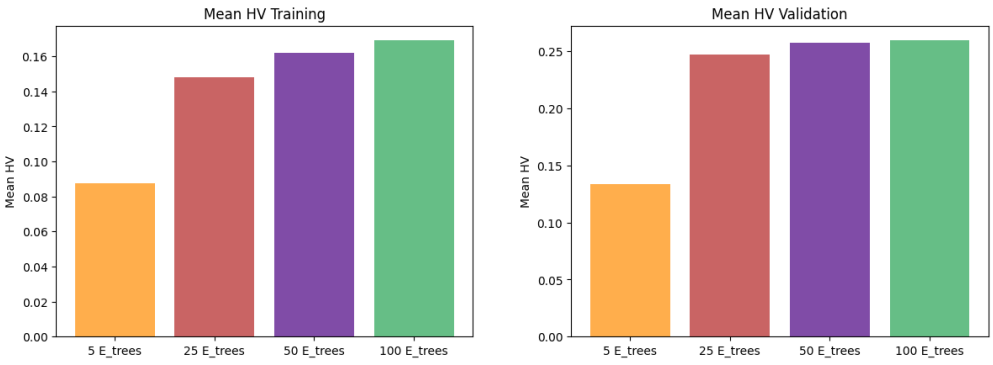
Figure 5.13: Training convergence evaluation metrics for five different random seeds, see section 3.4.2, for different Extra-Trees structures, hypervolume metric (a) and action-value function improvement (b)

First, the results in fig. 5.13b show there is no dependency between the trajectory of

$\Delta_{\hat{Q}}$ and the model used for the regression. Furthermore, it stands out that the mean absolute increase at convergence is higher for lower M, e.g. the values evaluated with 5 Extra-trees have the higher value of $\Delta_{\hat{Q}}$. This is due to the higher variance between the Q surface from one iteration to another, as described in section 3.4.2. This phenomenon is not evident before convergence is reached, as the increase due to the actual learning of the Q-function predominates the fluctuations due to the randomness of the regressor.

On the other hand, as shown in fig. 5.13a, with regard to HV with respect to iterations, it can be seen that regardless of the number of trees, the model iteratively increases the value of the metric. As far as the goodness of the model is concerned, it can be determined that increasing M increases the value of the HV at convergence, moreover increasing M decreases the variance of the value of the hypervolume. Therefore the solution set is highly variable in the objective space using a low number of extra trees. The results also show an increase in the HV value at convergence which asymptotically converges as the Extra-Trees forest dimension grows.

To increase results' robustness, the policies of the models at convergence, are also evaluated on the validation set and compared with the training results. As shown in fig. 5.14 by increasing the value M, the trend of the HV is similar in training and validation. With the exception of the smaller increase in validation by increasing M in fig. 5.14b, it can be seen that for 50 and 100 Extra-Trees there is no substantial difference in validation. It can therefore be said that $M = 50$ is adequate to generalize the solution of the problem also for disturbance trajectories other than the training ones.

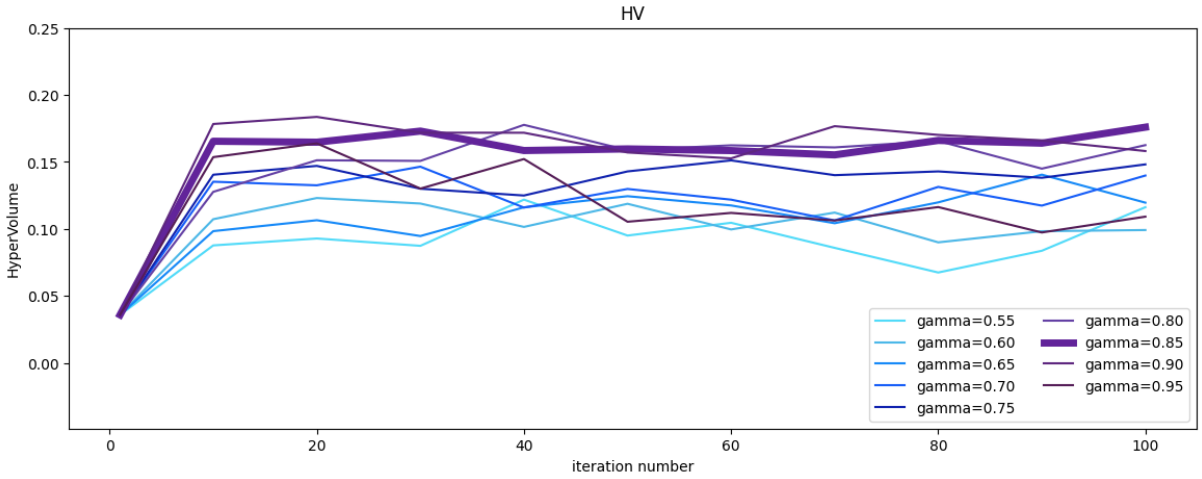


(a) (b)

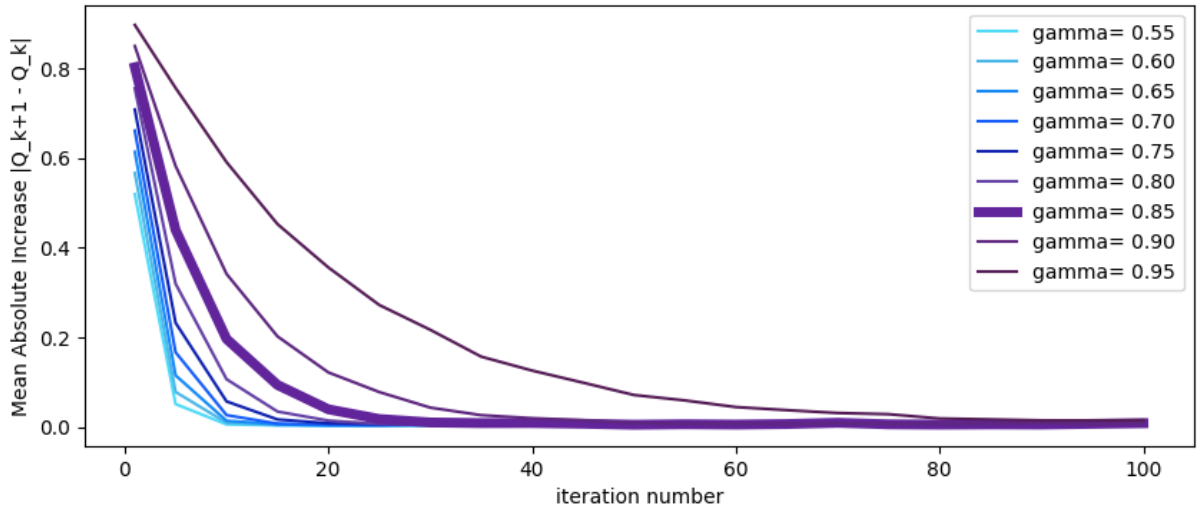
Figure 5.14: Mean hypervolume value at the 30^{th} iteration for different models structures evaluated for five different random seeds, training dataset (a) and validation dataset (b)

5.2.3. Discount Rate Gamma and Convergence

The discount rate γ adopted in the MOFQI framework affects the iterative update of the training set $\mathcal{TS}$ and thus the estimated action-value function $\hat{Q}$. This subsection illustrates the sensitivity analysis of the HV metric with respect to the discount rate.



(a)



(b)

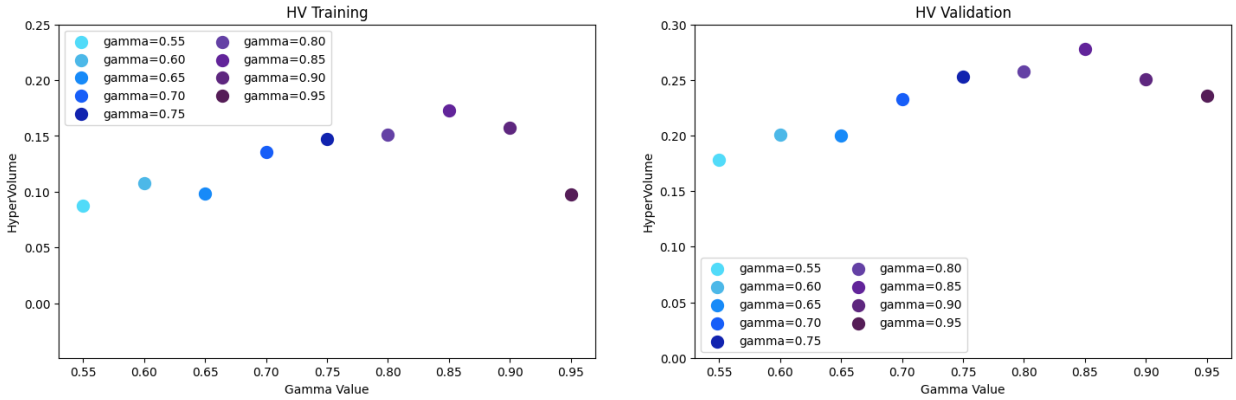
Figure 5.15: Training convergence evaluation metrics for different discount rate values, hypervolume metric (a) and action-value function improvement (b)

The HV metric assesses the convergence and diversity of the set of points that make up the Pareto front; thus, this metric based on weakly Pareto-dominated volume simultaneously evaluates the minimization of the three objective indicators. These indicators are defined in section 4.4 where the discount rate is set equal to one; while, using nine different

discount rate values to compute eq. (3.10), the mean absolute increase $\Delta_{\hat{Q}_h}$ and HV are compared and computed on the training trajectory, these results are reported in fig. 5.15.

The discount rate γ defines the number of iterations useful to learn the most amount of the Q value, it also affects the resulting performances in terms of HV. This result confirms the aforementioned conclusions in previous paragraphs about discount rate influence.
Theoretically, higher γ values allow the model to perform actions that account for longer horizon reward values, and the optimal value of the time horizon to consider depends on the reward dynamics. In multi-objective problems, reward values change in relation to the weights used to aggregate different rewards. The horizon considered to better behave in terms of HV metric takes into account all the possible single-objective reward combinations; hence the discount rate value which maximizes the HV is a tradeoff that expands the Pareto front in all the objectives space directions.

The choice of γ, which affects the HV values, is based only on this latter metric computed on both training and validation trajectories. In order to compare the model performance with different discount rates, each model at convergence is evaluated for the two different trajectories, as reported in fig. 5.16. Following these results, the value of $\gamma = 0.85$ is the one for which the HV at convergence is the highest for both the trajectories and also the one with the least fluctuations through the training iterations (fig. 5.15a).
On the other hand, lower γ results in a shorter iterative process to reach the convergence and leads to low hypervolume value with high variance; while increasing the value of gamma achieves high HV, at least equal to the one defined as the best, but the variability of the HV values is higher.



(a) (b)

Figure 5.16: Hypervolume value at convergence for different discount rate values, training (a) and validation (b)

5.3. Benchmarking MOFQI to state-of-the-art

The current section concludes the presentation of results with the benchmark between MOFQI and SDP, empirically demonstrating that the MOFQI algorithm in the current multi-objective case study is competitive with state-of-the-art methods.
As pointed out in section 2.1, the SDP algorithm suffers from the three curses that considerably limit its practical application. To compare SDP and MOFQI results, the same problem formulation and the same data are used, but two differences are present concerning the system representation in the optimization process. Another difference relates to the weight space sampling since SDP is a single objective method.

State-Action Space Discretization The first difference is related to the first of the three curses. Having a tabular and not a continuous representation of the state-action space, SDP needs a denser sampling grid to represent the state-action space, as described in detail in section 3.2. In the current case study, the number of sampled state-action couples for SDP is 257277 ($N_s = 191; N_a = 1347$), while for MOFQI the samples are only 150 ($N_s = 15; N_a = 10$). The SDP results can be considered state-of-the-art solutions for optimal reservoir control. Since in SDP the storage classes are equivalent to 1cm of variation of the level, and the release classes are equivalent to 0.1 m^3/s of flow variation, the grid can be considered very close to a continuum approximation. By coarsely discretizing it, the performance of the SDP algorithm degenerates. As an example, relying on the same 150 points used in the MOFQI algorithm, the SDP front is entirely outside the hypervolume defined by the reference point used in the benchmarking, i.e. $HV = 0$.

Simulation Model The second difference involves the simulation of the system since MOFQI generates the dataset only once and as a separate step from optimization, as described in section 3.3. SDP, on the other hand, needs to simulate the transition during the optimization process according to the probability model, this operation is done for each state-action pair of the dense grid that samples these two dimensions. For this reason, the simulation of the system using a more realistic, and so more time-demanding model, is limited by computational time. In the case study analyzed, in order to represent the real system, the simulation of the mass balance is done with hourly integration. In the generation of MOFQI's dataset, the model adopted to simulate the system transitions performs an hourly integration, even if the difference in terms of HV is minimal between the MOFQI results using the hourly rather than the daily integration step, see the results in table 5.2. On the contrary, in the SDP frame, the hourly integration is prohibitive in terms of computational time due to the excessive number of simulations that must be

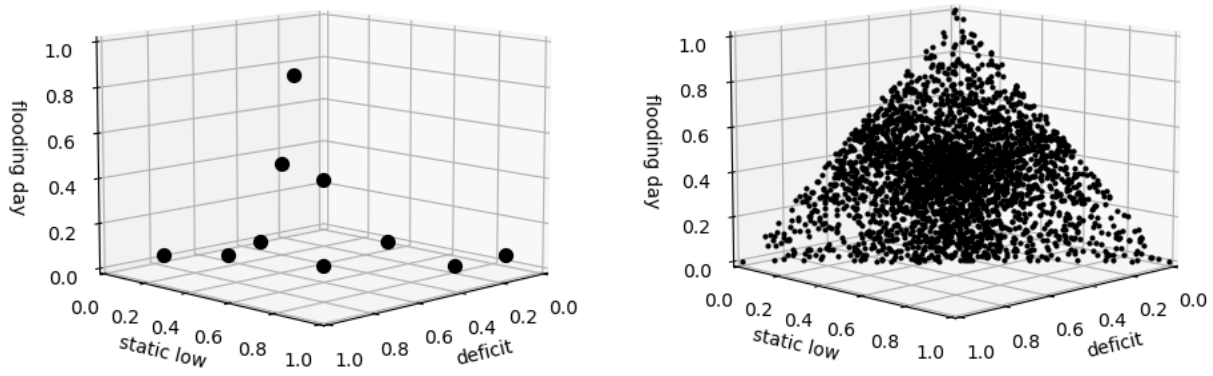
performed for the numerous state-action couples.

	Training	Validation
Daily integration	0.164	0.268
Hourly integration	0.173	0.278

Table 5.2: HV metric for the models trained with datasets generated with daily integration model and hourly integration model

Simplex Sampling SDP is a single-objective algorithm; thus, due to the curse of multiple-objectives, it requires running the optimization for each combination of weights assigned to the objectives. This presents two disadvantages, both related to the limited computational time available. First, since it can only explore the objectives space through trial and error, SDP requires multiple runs to obtain the desired points to create its Pareto front. In the current benchmarking, 10 samples distributed across the 2-dimensional simplex surface are used to generate the SDP's Pareto front (fig. 5.17a). Second, due to this computationally limited trial and error approach, solutions in the objective space are not equally spread, thus leading to many detrimental concavity zones on the Pareto front. These disadvantages are not present when using the MOFQI algorithm, as it can approximate the solution continuously over the entire weight space and in a single run. Potentially, after optimization, an unlimited number of policies can be simulated, as described in section 3.2.1 and shown in the results of section 5.1.3. For this reason, in the following comparison, 3000 different policies (fig. 5.17b) are sampled from the model to generate an extended and well-spread Pareto front in the objective space, thus reducing the concavities.

Benchmarking Pareto Front The benchmarking results are reported for both training and validation, the fig. 5.18 represents the Pareto fronts projected on the deficit and static low axes; while fig. 5.19 show the projections on the deficit and flood day axes.
The HV metric is computed using a reference point with coordinates equal to the 99^{th} percentile of the MOFQI indicators distribution. This is done firstly not to consider potential outliers values and secondly to evaluate the Pareto front without favoring the improvement of any of the objectives in the HV calculation.
The most evident result is MOFQI's ability to generate a large and continuous Pareto front; a fundamental characteristic for the practical application in water reservoir control. The broad exploration of the objectives space leads to a wide range of alternatives, among these, the most suitable can be selected to match the interests of the decision maker.
About the quantitative evaluation of performance, it can be observed that the HV is

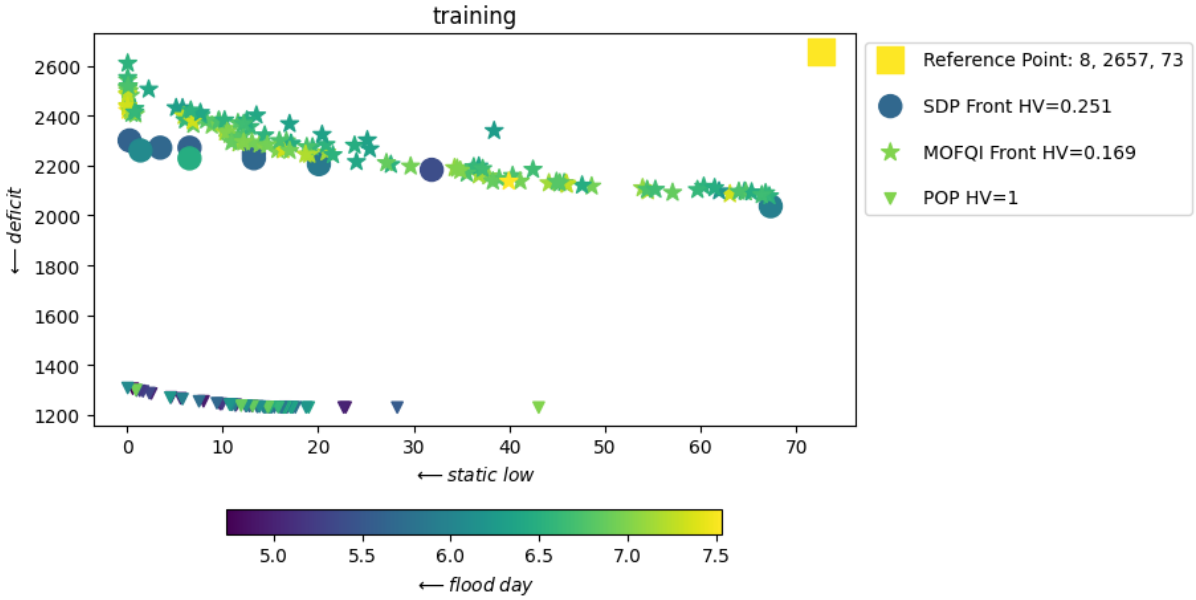


(a) (b)

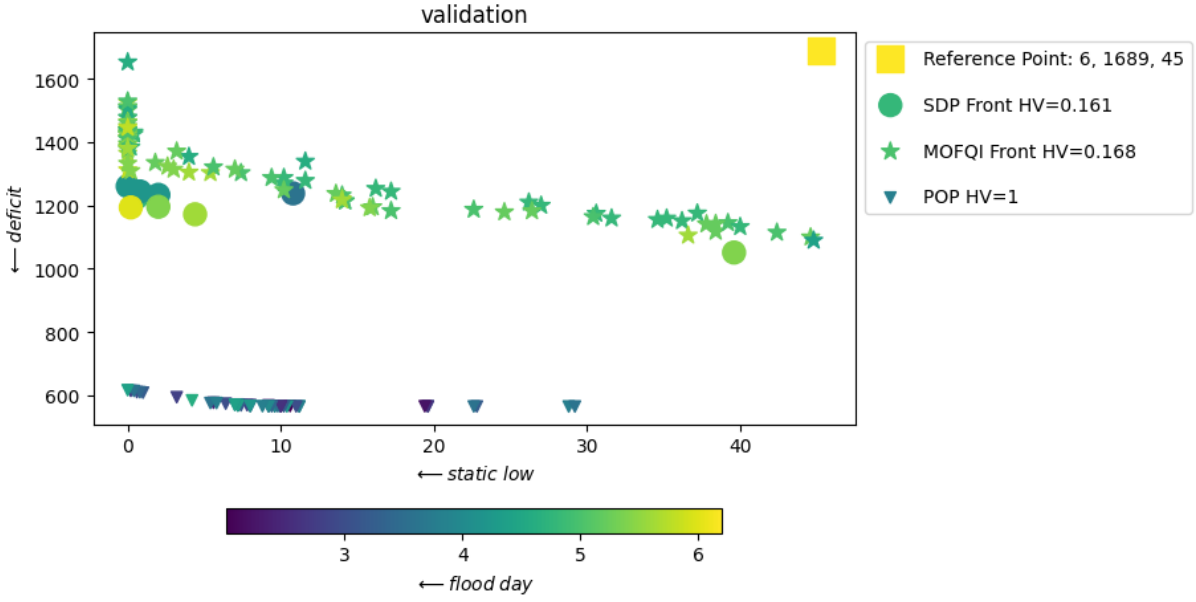
Figure 5.17: Simplex samples to generate frontiers for benchmarking, SDP weights samples (a) and MOFQI weights samples (b)

higher for SDP in training, this is due to higher values in particular for the flooding days objective, even if the difference between the minimum values of the two methods is 0.8 [day/year]. As regards the other two objectives, there is a minimal difference, not negligible only in the central part of the Pareto front, where the minimum value of the deficit indicator of SDP is 40 $[(m^3/s)^{eq}/day]$ lower than MOFQI's minimum and, on the contrary, the minimum value of the low-level indicator is 0.2 [day/year] lower for MOFQI. However, the situation changes drastically when considering the results of validation. The difference with respect to the flooding objective, which is present in training, is almost completely nullified. Furthermore, the diversity of the points in the frontier is reduced, thus increasing the concavity. These two factors act simultaneously, reducing the SDP's HV value and leading to a value slightly lower than the HV of MOFQI. It can therefore be deduced that the SDP has a more pronounced bias than MOFQI, induced by the statistical differences of the training data, on which the transition model is estimated, compared to the validation data. As a result, MOFQI solutions seem to be more robust, varying less with respect to the distribution of net inflow data to which the system is subjected.

Another remarkable result concerns the computation time. As described in section 3.2.1, MOFQI mitigates the curse of dimensions and is also unaffected by the other two curses; these characteristics lead to a significant computational advantage. For the current case study, set up as described above, there are 119 points in the MOFQI's set that compose the Pareto front; assuming to compute the same amount of frontier points with SDP, MOFQI takes about 0.2% of the time spent. Moreover, this advantage increases exponentially with the number of states and exogenous variables considered due to the SDP curses.

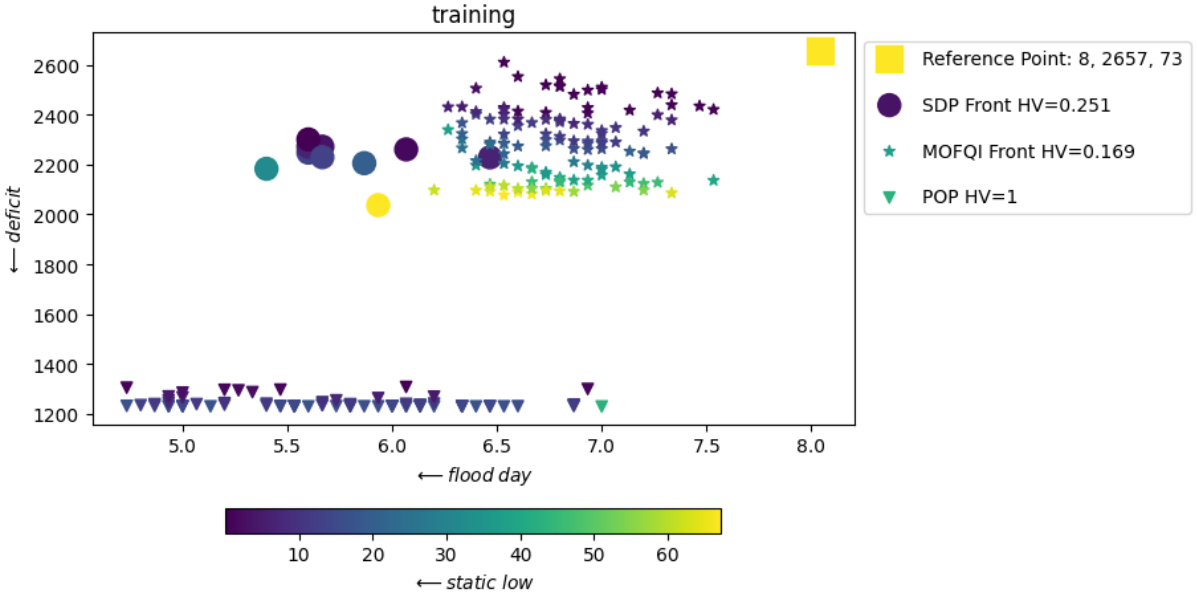


(a)

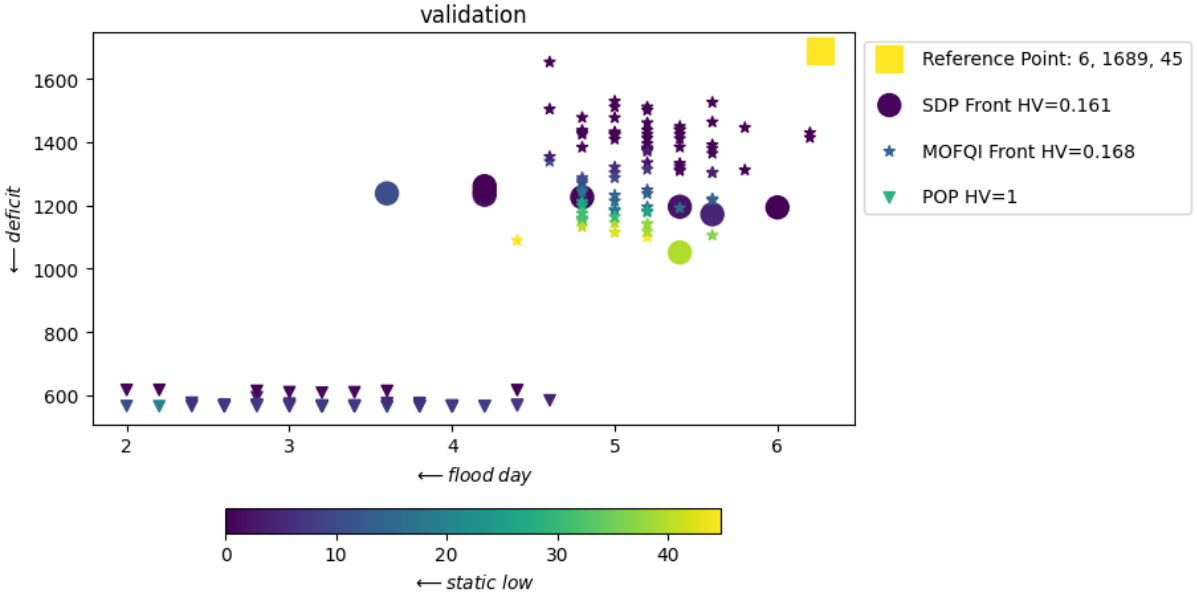


(b)

Figure 5.18: Pareto front projection on the deficit and static low axes; 3000 policies are sampled for generating the MOFQI Front, training (a) and validation (b)



(a)



(b)

Figure 5.19: Pareto front projection on the deficit and flood day axes; 3000 policies are sampled for generating the MOFQI Front, training (a) and validation (b)

6 | Conclusions and future research

This thesis investigates the feasibility and effectiveness of the MOFQI algorithm in addressing the challenges characterizing water system management, first studied theoretically and then demonstrated empirically in a real-world case study. The thesis presents a framework that relies on partial model-free simulation to explore the system's transitions, these constitute the experience dataset provided to the control optimization algorithm. The framework is scalable with respect to the number of system states, accounting for additional state variables inevitably increases the number of tuples in the dataset, however, thanks to the mitigation of the curse of dimensionality the computational costs are alleviated. Furthermore, nullifying the curse of modeling, this framework is also easily scalable with respect to the number of exogenous variables. The key findings are as follows.

(1) MOFQI, being an offline batch RL algorithm, has the characteristic of decoupling the generation of the experience, i.e. the dataset, and the optimization process. This characteristic leads to a computational advantage, particularly for the dataset generation in the partial model-free approach, as used in the proposed case study. The generation of the dataset allows the adoption of different techniques to explore the state-action space, this step is crucial to determine a learning process that effectively converges to the optimal Q-function. As shown empirically, not all the generated datasets lead to a simultaneous increase of the Q-function value and the performance value in the objective space. In particular, the simplex random sampling method proved to be effective in representing the system in the multi-objective frame, and leading to an effective learning process, giving also a further advantage to the overall computational cost of the method.

(2) Discount rate affects the tradeoff between immediate rewards from the current action and future expected gain, this parameter influences both the number of iterations and the value of the HV. For each problem formulation, the discount rate must be carefully analyzed since it depends on the system dynamics and the objectives considered. Furthermore, future works deserve a more in-depth analysis with regard to the convergence in the direction of each objective, given that the optimal discount rate evaluated separately

on each of the individual objectives could be different from the multi-objective discount rate defined through the HV metric.

(3) The estimated Q-function and the related control law rely on the physical representation of the system and thus can lead to a consistent and strong explainability of the results. The Q-function provides information regarding the values of each action for each state of the system; as shown in the results, this value could be representative of the characteristics of the step-costs trajectories obtained by simulating the control policy. However, this work leaves a large space for research in the explainability and interpretability of the MOFQI policies.

(4) The MOFQI algorithm is proven to be competitive with state-of-the-art methods. Particularly compared with SDP, it carries many advantages, not only related to the computational requirement but also to the quality of the generated Pareto front; indeed, it is able to generate a Pareto front that is diversified and well distributed along tradeoffs. The main advantages are summarized as follows.

- The explicit probability density function model is not needed. Instead, SDP requires a model to be defined. This characteristic allows MOFQI to be directly applied having only historical observations, while for SDP a further step is needed to estimate the statistical model.
- In the partial model-free approach, MOFQI can rely on a more complex and realistic simulation model since the system's transitions are simulated only once during the dataset generation process and potentially for fewer state-action couples than SDP. In this way, MOFQI overcomes the limitations on the correct representability of the real system.
- Continuous approximation of the action-value function mitigates effects of the curse of dimensionality using a coarser state-action grid for the dataset, while SDP still relies on a tabular representation. Therefore, it reduces the computational cost, even when multiple system states are included.
- Including objectives weights as a system state nullifies the computational burdens of the curse of multiple objectives. Contrary to what happens for SDP, MOFQI continuously approximates the Pareto front in a single run with a single architecture. Having a wide range of choices is a relevant advantage from the perspective of finding a suitable tradeoff policy to provide to the decision-maker.
- The curse of modeling is no longer present, and any exogenous information could be included with a slight increase in the computational burden, potentially improving control performances. As widely reported in the literature, this is a characteristic

that advantages the use of exogenous variables since relying on this information can bring a significant increase in control results (Giuliani et al., 2015).

In summary, this thesis work contributes to consolidating MOFQI applications in water reservoir management, in particular, it empirically defines the main characteristics and settings of the method in a multi-objective context, verifying its competitiveness compared to the state-of-the-art. The results of this work can also be a starting point to deepen and broaden the analysis of the application of this method in the water system management domain.
Numerous aspects deserve further investigation and leave space for improvement in the performance of the results; starting from the in-depth analysis of the discretization of the state-action space and the subsequent weighted aggregation through simplex sampling. As previously explained, this step is fundamental in order to generate an effective learning process, as different approaches can lead to more performing policies.
The sensitivity analysis of all the hyperparameters involved in the algorithm, including those not investigated in the present work, can highlight configurations with which the Q-function leads to better control. Based on the results of this thesis, future research could expand this framework by including exogenous variables such as forecasts of future inflows or meteorological variables, relying on the same algorithm formulation, as generalized by Liu et al. (2021); in this way, the related learning process could exploit the additional information potentially leading to an overall improvement in control performance. By modifying some aspects of the iterative process, neural network architectures can be used as regressors instead of Extra-Trees. These can bring some advantages, mainly related to the use of numerous input variables and the speed of parameter calibration techniques, such as demonstrated by Mnih et al. (2015) and Zhao et al. (2016) in fields different than water management.

All the considerations and results outline the characteristics and potential improvements of the framework described in this thesis. Due to the highlighted advantages, this method can be applied to different case studies of water systems management, especially for problems with an integrated representation of all the aspects and compartments related to the hydrological system. Future works can expand the analysis on larger systems, testing the scalability and computational efficiency of the method, even including hydrologic exogenous information.

Bibliography

D. Anghileri, A. Castelletti, F. Pianosi, R. Soncini-Sessa, and E. Weber. Optimizing watershed management by coordinated operation of storing facilities. *Journal of Water Resources Planning and Management*, 139:492–500, 2012. doi: 10.1061/(ASCE)WR.1943-5452.0000313.

M. T. L. Barros, F. T.-C. Tsai, S.-l. Yang, J. E. G. Lopes, and W. W.-G. Yeh. Optimization of Large-Scale Hydropower System Operations. *Journal of Water Resources Planning and Management*, 129(3):178–188, May 2003. ISSN 0733-9496, 1943-5452. doi: 10.1061/(ASCE)0733-9496(2003)129:3(178). URL `https://ascelibrary.org/doi/10.1061/%28ASCE%290733-9496%282003%29129%3A3%28178%29`.

R. S. S. a. A. G. Barto. *Reinforcement learning: An introduction*. MIT Press, Cambridge, Massachusetts, 2 edition, 2018.

P. Bauer, A. Thorpe, and G. Brunet. The quiet revolution of numerical weather prediction. *Nature*, 525(7567):47–55, Sept. 2015. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature14956. URL `https://www.nature.com/articles/nature14956`.

R. Bellman. *Dynamic Programming*. Princeton University Press, 1957. URL `https://www.bibsonomy.org/bibtex/29cdd821222218ded252c8ba5cd712666/m-toman`.

Y. Bengio. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, 2(1):1–127, 2009. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000006. URL `http://www.nowpublishers.com/article/Details/MAL-006`.

R. D. Benson. Reviewing Reservoir Operations: Can Federal Water Projects Adapt to Change? *Columbia Journal of Environmental Law*, page Vol. 42 No. 2 (2017): Volume 42.2, Nov. 2019. doi: 10.7916/CJEL.V42I2.3739. URL `https://journals.library.columbia.edu/index.php/cjel/article/view/3739`. Publisher: Columbia Journal of Environmental Law.

F. Bertoni, A. Castelletti, M. Giuliani, and P. M. Reed. Discovering Dependencies, Trade-Offs, and Robustness in Joint Dam Design and Operation: An Ex-Post Assessment of the Kariba Dam. *Earth's Future*, 7(12):1367–1390, Dec. 2019. ISSN 2328-4277, 2328-

4277. doi: 10.1029/2019EF001235. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2019EF001235.

Browder, Greg, Ana Nunez Sanchez, Brenden Jongman, Nathan Engle, Eelco Van Beek, Melissa Castera Errea, and Stephen Hodgson. An EPIC Response: Innovative Governance for Flood and Drought Risk Management—Executive Summary., 2021. URL http://documents.worldbank.org/curated/en/419061623699802863/Main-Report.

A. Castelletti and R. Soncinisessa. A procedural approach to strengthening integration and participation in water resource planning. *Environmental Modelling & Software*, 21 (10):1455–1470, Oct. 2006. ISSN 13648152. doi: 10.1016/j.envsoft.2005.07.013. URL https://linkinghub.elsevier.com/retrieve/pii/S1364815205001520.

A. Castelletti, F. Pianosi, and R. Soncini-Sessa. Integration, participation and optimal control in water resources planning and management. *Applied Mathematics and Computation*, 206(1):21–33, Dec. 2008a. ISSN 00963003. doi: 10.1016/j.amc.2007.09.069. URL https://linkinghub.elsevier.com/retrieve/pii/S009630030700999X.

A. Castelletti, F. Pianosi, and R. Soncini-Sessa. Water reservoir control under economic, social and environmental constraints. *Automatica*, 44(6):1595–1607, June 2008b. ISSN 00051098. doi: 10.1016/j.automatica.2008.03.003. URL https://linkinghub.elsevier.com/retrieve/pii/S0005109808001271.

A. Castelletti, S. Galelli, M. Restelli, and R. Soncini-Sessa. Tree-based reinforcement learning for optimal water reservoir operation: FITTED Q ITERATION FOR WATER RESERVOIRS. *Water Resources Research*, 46(9), Sept. 2010. ISSN 00431397. doi: 10.1029/2009WR008898. URL http://doi.wiley.com/10.1029/2009WR008898.

A. Castelletti, F. Pianosi, and M. Restelli. A multiobjective reinforcement learning approach to water resources systems operation: Pareto frontier approximation in a single run: MOFQI for Large-Scale Water Resources Systems Operation. *Water Resources Research*, 49(6):3476–3486, June 2013. ISSN 00431397. doi: 10.1002/wrcr.20295. URL http://doi.wiley.com/10.1002/wrcr.20295.

A. Castelletti, H. Yajima, M. Giuliani, R. Soncini-Sessa, and E. Weber. Planning the Optimal Operation of a Multioutlet Water Reservoir with Water Quality and Quantity Targets. *Journal of Water Resources Planning and Management*, 140(4):496–510, Apr. 2014. ISSN 0733-9496, 1943-5452. doi: 10.1061/(ASCE)WR.1943-5452.0000348. URL https://ascelibrary.org/doi/10.1061/%28ASCE%29WR.1943-5452.0000348.

J. L. Cohon and D. H. Marks. A review and evaluation of multiobjective programing

techniques. *Water Resources Research*, 11(2):208–220, Apr. 1975. ISSN 0043-1397, 1944-7973. doi: 10.1029/WR011i002p00208. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1029/WR011i002p00208.

C. Como. Relazione di sintesi della commissione per lo studio dei fenomeni di subsidenza. *Documenti e Ricerche*, 34, 1980.

Consorzio dell'Adda. 2022. URL https://www.addaconsorzio.it.

R. Damania, S. Desbureaux, M. Hyland, A. Islam, S. Moore, A.-S. Rodella, J. Russ, and E. Zaveri. *Uncharted Waters: The New Economics of Water Scarcity and Variability*. The World Bank, 2017. doi: 10.1596/978-1-4648-1179-1.

Damien Ernst, Pierre Geurts, and Louis Wehenkel. Tree-Based Batch Mode Reinforcement Learning. *Journal of Machine Learning Research*, 2005.

S. Denaro, D. Anghileri, M. Giuliani, and A. Castelletti. Informing the operations of water reservoirs over multiple temporal scales by direct use of hydro-meteorological data. *Advances in Water Resources*, 103:51–63, 2017. ISSN 0309-1708. doi: https://doi.org/10.1016/j.advwatres.2017.02.012. URL https://www.sciencedirect.com/science/article/pii/S0309170816304651.

N. Ehsani, C. J. Vörösmarty, B. M. Fekete, and E. Z. Stakhiv. Reservoir operations under climate change: Storage capacity options to mitigate risk. *Journal of Hydrology*, 555:435–446, Dec. 2017. ISSN 00221694. doi: 10.1016/j.jhydrol.2017.09.008. URL https://linkinghub.elsevier.com/retrieve/pii/S0022169417305991.

G. Forzieri, L. Feyen, R. Rojas, M. Flörke, F. Wimmer, and A. Bianchi. Ensemble projections of future streamflow droughts in europe. *Hydrology and Earth System Sciences*, 18(1):85–108, 2014. doi: 10.5194/hess-18-85-2014. URL https://hess.copernicus.org/articles/18/85/2014/.

L. Garrote. Managing water resources to adapt to climate change: Facing uncertainty and scarcity in a changing context. *Water Resources Management*, 31:2951–2963, 2017. doi: https://doi.org/10.1007/s11269-017-1714-6.

A. Georgakakos, H. Yao, M. Kistenmacher, K. Georgakakos, N. Graham, F.-Y. Cheng, C. Spencer, and E. Shamir. Value of adaptive water resources management in Northern California under climatic variability and change: Reservoir management. *Journal of Hydrology*, 412-413:34–46, Jan. 2012. ISSN 00221694. doi: 10.1016/j.jhydrol.2011.04.038. URL https://linkinghub.elsevier.com/retrieve/pii/S0022169411003015.

P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*,

63(1):3–42, Apr. 2006. ISSN 0885-6125, 1573-0565. doi: 10.1007/s10994-006-6226-1. URL http://link.springer.com/10.1007/s10994-006-6226-1.

F. Giudici, D. Anghileri, A. Castelletti, and P. Burlando. Descriptive or normative: How does reservoir operations modeling influence hydrological simulations under climate change? *Journal of Hydrology*, 595:125996, 2021. ISSN 0022-1694. doi: https://doi.org/10.1016/j.jhydrol.2021.125996. URL https://www.sciencedirect.com/science/article/pii/S0022169421000433.

M. Giuliani and A. Castelletti. Is robustness really robust? how different definitions of robustness impact decision-making under climate change. *Climatic Change*, 135: 409–424, 2016. doi: https://doi.org/10.1007/s10584-015-1586-9.

M. Giuliani, F. Pianosi, and A. Castelletti. Making the most of data: An information selection and assessment framework to improve water systems operations. *Water Resources Research*, 51(11):9073–9093, Nov. 2015. ISSN 0043-1397, 1944-7973. doi: 10.1002/2015WR017044. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1002/2015WR017044.

M. Giuliani, M. Zaniolo, A. Castelletti, G. Davoli, and P. Block. Detecting the state of the climate system via artificial intelligence to improve seasonal forecasts and inform reservoir operations. *Water Resources Research*, 55:9133–9147, 2019.

M. Giuliani, L. Crochemore, I. Pechlivanidis, and A. Castelletti. From skill to value: isolating the influence of end user behavior on seasonal forecast assessment. *Hydrology and Earth System Sciences*, 24(12):5891–5902, Dec. 2020. ISSN 1607-7938. doi: 10.5194/hess-24-5891-2020. URL https://hess.copernicus.org/articles/24/5891/2020/.

M. Giuliani, J. R. Lamontagne, P. M. Reed, and A. Castelletti. A State-of-the-Art Review of Optimal Reservoir Control for Managing Conflicting Demands in a Changing World. *Water Resources Research*, 57(12):e2021WR029927, Dec. 2021. ISSN 0043-1397, 1944-7973. doi: 10.1029/2021WR029927. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2021WR029927.

P. H. Gleick and M. Palaniappan. Peak water limits to freshwater withdrawal and use. *Proceedings of the National Academy of Sciences*, 107(25):11155–11162, June 2010. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1004812107. URL https://pnas.org/doi/full/10.1073/pnas.1004812107.

G. Grill, B. Lehner, M. Thieme, B. Geenen, D. Tickner, F. Antonelli, S. Babu, P. Borrelli, L. Cheng, H. Crochetiere, H. Ehalt Macedo, R. Filgueiras, M. Goichot, J. Higgins,

Z. Hogan, B. Lip, M. E. McClain, J. Meng, M. Mulligan, C. Nilsson, J. D. Olden, J. J. Opperman, P. Petry, C. Reidy Liermann, L. Sáenz, S. Salinas-Rodríguez, P. Schelle, R. J. P. Schmitt, J. Snider, F. Tan, K. Tockner, P. H. Valdujo, A. Van Soesbergen, and C. Zarfl. Mapping the world's free-flowing rivers. *Nature*, 569(7755):215–221, May 2019. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-019-1111-9. URL https://www.nature.com/articles/s41586-019-1111-9.

A. P. Guerreiro, C. M. Fonseca, and L. Paquete. The Hypervolume Indicator: Problems and Algorithms. *ACM Computing Surveys*, 54(6):1–42, July 2022. ISSN 0360-0300, 1557-7341. doi: 10.1145/3453474. URL http://arxiv.org/abs/2005.00515. arXiv:2005.00515 [cs].

M. I. Hejazi, X. Cai, and B. L. Ruddell. The role of hydrologic information in reservoir operation – Learning from historical releases. *Advances in Water Resources*, 31(12): 1636–1650, Dec. 2008. ISSN 03091708. doi: 10.1016/j.advwatres.2008.07.013. URL https://linkinghub.elsevier.com/retrieve/pii/S0309170808001309.

M. Hessel, J. Modayil, H. Van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver. Rainbow: Combining Improvements in Deep Reinforcement Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. ISSN 2374-3468, 2159-5399. doi: 10.1609/aaai.v32i1.11796. URL https://ojs.aaai.org/index.php/AAAI/article/view/11796.

M. Hossen, J. Connor, and F. Ahammed. How to Resolve Transboundary River Water Sharing Disputes. *Water*, 15(14):2630, July 2023. ISSN 2073-4441. doi: 10.3390/w15142630. URL https://www.mdpi.com/2073-4441/15/14/2630.

A. Iglesias and L. Garrote. Adaptation strategies for agricultural water management under climate change in europe. *Agricultural Water Management*, 155:113–124, 2015. ISSN 0378-3774. doi: https://doi.org/10.1016/j.agwat.2015.03.014.

IPCC. Impacts, adaption and vulnerability. summary for policymakers. *Climate Change 2022*, 2022. URL https://www.ipcc.ch/report/ar6/wg2/downloads/report/IPCC_AR6_WGII_SummaryForPolicymakers.pdf.

R. Kerachian and M. Karamouz. Optimal reservoir operation considering the water quality issues: A stochastic conflict resolution approach. *Water Resources Research*, 42(12): 2005WR004575, Dec. 2006. ISSN 0043-1397, 1944-7973. doi: 10.1029/2005WR004575. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2005WR004575.

J. W. Labadie. Optimal Operation of Multireservoir Systems: State-of-the-Art Review. *Journal of Water Resources Planning and Management*, 130(2):93–111,

Mar. 2004. ISSN 0733-9496, 1943-5452. doi: 10.1061/(ASCE)0733-9496(2004)130:2(93). URL https://ascelibrary.org/doi/10.1061/%28ASCE%290733-9496%282004%29130%3A2%2893%29.

S. Legg and M. Hutter. Universal Intelligence: A Definition of Machine Intelligence. 2007. doi: 10.48550/ARXIV.0712.3329. URL https://arxiv.org/abs/0712.3329. Publisher: arXiv Version Number: 1.

T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning, July 2019. URL http://arxiv.org/abs/1509.02971. arXiv:1509.02971 [cs, stat].

V. Liu, J. R. Wright, and M. White. Exploiting Action Impact Regularity and Exogenous State Variables for Offline Reinforcement Learning. 2021. doi: 10.48550/ARXIV.2111.08066. URL https://arxiv.org/abs/2111.08066. Publisher: arXiv Version Number: 5.

G. McDowell, E. Stephenson, and J. Ford. Adaptation to climate change in glaciated mountain regions. *Climatic Change*, 126:77–91, 2014. doi: https://doi.org/10.1007/s10584-014-1215-z.

V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, Feb. 2015. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature14236. URL https://www.nature.com/articles/nature14236.

N. Nappo, M. F. Ferrario, F. Livio, and A. M. Michetti. Regression analysis of subsidence in the como basin (northern italy): New insights on natural and anthropic drivers from insar data. *Remote Sensing*, 12, 2020. doi: doi:10.3390/rs12182931.

E. O'Connell. Towards Adaptation of Water Resource Systems to Climatic and Socio-Economic Change. *Water Resources Management*, 31(10):2965–2984, Aug. 2017. ISSN 0920-4741, 1573-1650. doi: 10.1007/s11269-017-1734-2. URL http://link.springer.com/10.1007/s11269-017-1734-2.

J. C. Padowski, S. M. Gorelick, B. H. Thompson, S. Rozelle, and S. Fendorf. Assessment of human–natural system characteristics influencing global freshwater supply vulnerability. *Environmental Research Letters*, 10(10), Oct. 2015. ISSN 1748-9326. doi: 10.1088/1748-9326/10/10/104014. URL https://iopscience.iop.org/article/10.1088/1748-9326/10/10/104014.

C. Pahl-Wostl. Transitions towards adaptive management of water facing climate and global change. *Water Resources Management*, 21(1):49–62, Dec. 2006. ISSN 0920-4741, 1573-1650. doi: 10.1007/s11269-006-9040-4. URL http://link.springer.com/10.1007/s11269-006-9040-4.

C. Pahl-Wostl, P. Jeffrey, N. Isendahl, and M. Brugnach. Maturing the New Water Management Paradigm: Progressing from Aspiration to Practice. *Water Resources Management*, 25(3):837–856, Feb. 2011. ISSN 0920-4741, 1573-1650. doi: 10.1007/s11269-010-9729-2. URL http://link.springer.com/10.1007/s11269-010-9729-2.

Y. Pan. Heading toward Artificial Intelligence 2.0. *Engineering*, 2(4):409–413, Dec. 2016. ISSN 20958099. doi: 10.1016/J.ENG.2016.04.018. URL https://linkinghub.elsevier.com/retrieve/pii/S2095809917300772.

F. Pianosi, A. Castelletti, and M. Restelli. Tree-based fitted Q-iteration for multi-objective Markov decision processes in water resource management. *Journal of Hydroinformatics*, 15(2):258–270, Apr. 2013. ISSN 1464-7141, 1465-1734. doi: 10.2166/hydro.2013.169. URL https://iwaponline.com/jh/article/15/2/258/3425/Treebased-fitted-Qiteration-for-multiobjective.

W. B. Powell. A unified framework for stochastic optimization. *European Journal of Operational Research*, 275(3):795–821, June 2019. ISSN 03772217. doi: 10.1016/j.ejor.2018.07.014. URL https://linkinghub.elsevier.com/retrieve/pii/S0377221718306192.

J. D. Quinn, P. M. Reed, M. Giuliani, and A. Castelletti. What Is Controlling Our Control Rules? Opening the Black Box of Multireservoir Operating Policies Using Time-Varying Sensitivity Analysis. *Water Resources Research*, 55(7):5962–5984, July 2019. ISSN 0043-1397, 1944-7973. doi: 10.1029/2018WR024177. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2018WR024177.

M. Riedmiller. Neural Fitted Q Iteration – First Experiences with a Data Efficient Neural Reinforcement Learning Method. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, J. Gama, R. Camacho, P. B. Brazdil, A. M. Jorge, and L. Torgo, editors, *Machine Learning: ECML 2005*, volume 3720, pages 317–328. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005. ISBN 978-3-540-29243-2 978-3-540-31692-3. doi: 10.1007/11564096_32. URL http://link.springer.com/10.1007/11564096_32. Series Title: Lecture Notes in Computer Science.

K. Rupp. 48 Years of Microprocessor Trend Data, July 2020. URL https://zenodo.org/record/3947823. Publisher: Zenodo Version Number: 2020.0.

R. Soncini-Sessa, E. Weber, and A. Castelletti. *Integrated and Participatory Water Resources Management - Theory: Theory*. Elsevier Science, Amsterdam, 2014. ISBN 978-0-08-055141-8. OCLC: 1048571339.

J. N. Tsitsiklis and B. Van Roy. Feature-Based Methods for Large Scale Dynamic Programming. In L. P. Kaelbling, editor, *Recent Advances in Reinforcement Learning*, pages 59–94. Springer US, Boston, MA, 1996. ISBN 978-0-7923-9705-2 978-0-585-33656-5. doi: 10.1007/978-0-585-33656-5_5. URL http://link.springer.com/10.1007/978-0-585-33656-5_5.

C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, May 1992. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00992698. URL http://link.springer.com/10.1007/BF00992698.

W. Xu, F. Meng, W. Guo, X. Li, and G. Fu. Deep Reinforcement Learning for Optimal Hydropower Reservoir Operation. *Journal of Water Resources Planning and Management*, 147(8):04021045, Aug. 2021. ISSN 0733-9496, 1943-5452. doi: 10.1061/(ASCE)WR.1943-5452.0001409. URL https://ascelibrary.org/doi/10.1061/%28ASCE%29WR.1943-5452.0001409.

X. Yun, Q. Tang, J. Wang, X. Liu, Y. Zhang, H. Lu, Y. Wang, L. Zhang, and D. Chen. Impacts of climate change and reservoir operation on streamflow and flood characteristics in the Lancang-Mekong River Basin. *Journal of Hydrology*, 590:125472, Nov. 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2020.125472. URL https://linkinghub.elsevier.com/retrieve/pii/S002216942030932X.

C. Zarfl, A. E. Lumsdon, J. Berlekamp, L. Tydecks, and K. Tockner. A global boom in hydropower dam construction. *Aquatic Sciences*, 77(1):161–170, Jan. 2015. ISSN 1015-1621, 1420-9055. doi: 10.1007/s00027-014-0377-0. URL http://link.springer.com/10.1007/s00027-014-0377-0.

D. Zhao, Y. Zhu, L. Lv, Y. Chen, and Q. Zhang. Convolutional fitted Q iteration for vision-based control problems. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 4539–4544, Vancouver, BC, Canada, July 2016. IEEE. ISBN 978-1-5090-0620-5. doi: 10.1109/IJCNN.2016.7727794. URL http://ieeexplore.ieee.org/document/7727794/.

E. Zitzler and L. Thiele. Multiobjective optimization using evolutionary algorithms — A comparative case study. In G. Goos, J. Hartmanis, J. Van Leeuwen, A. E. Eiben,

T. Bäck, M. Schoenauer, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature — PPSN V*, volume 1498, pages 292–301. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998. ISBN 978-3-540-65078-2 978-3-540-49672-4. doi: 10.1007/BFb0056872. URL `http://link.springer.com/10.1007/BFb0056872`. Series Title: Lecture Notes in Computer Science.

List of Figures

List of Tables

List of Algorithms

Acknowledgements

First and foremost, I would like to thank Prof. Andrea Francesco Castelletti and Prof. Matteo Giuliani for granting me the opportunity to develop this thesis. Your knowledge and teachings have guided me along a fascinating path that end with this project. My sincere thanks also extend to my co-advisor, Davide Spinelli, whose dedicated support and assistance were invaluable to the development of my thesis.

Special thanks go to all the people who have been part of this formative journey, to the students with whom I have had the opportunity to bond, and to the friends who have made my university journey somewhat light. In particular to my thesis fellows Bruno, Lorenzo, Carola, Francesco and Valentina with whom I pleasantly shared this final phase.

I take this opportunity to express my deep gratitude to my parents, without whom none of this would have been possible.

Last but not least, I am sincerely thankful to all my close friends whose support has been nothing short of fundamental to reaching the end of this path.