POLITECNICO DI MILANO

SCHOOL OF **INDUSTRIAL AND INFORMATION ENGINEERING**

MASTER OF SCIENCE IN MECHANICAL ENGINEERING AND
MASTER OF SCIENCE IN MANAGEMENT ENGINEERING

LINEAR AND DYNAMIC OPTIMIZATION OF MULTI-RESERVOIRS HYDROELECTRIC SYSTEMS

Supervisors: Prof. Marco TARABINI
Prof. Simone FRANZÒ
Co-supervisor: Dott. Pietro MARZAROLI

Master thesis of:

Luca NOBILE Matr. 874414

Academic Year 2018 - 2019

Luca Nobile: *Linear And Dynamic Optimization Of Multi-Reservoirs Hydroelectric Systems* | Master's Degree Thesis in Mechanical Engineering, Politecnico di Milano.

Politecnico di Milano:
www.polimi.it

Scuola di Ingegneria Industriale e dell'Informazione:
www.ingindinf.polimi.it

Contents

List of Figures

List of Tables

Listings

Abstract

Water can be considered the scarce resource in the production of hydroelectric energy. With the liberalization of electricity market in Italy, the application of optimization techniques to reservoir operation has become a major focus in water resources planning and management.

The optimization problem is characterized by non-linear objective function and constraints, subject also to boolean conditions. In literature the studies refer majorly to simple water systems optimization or, to more complex problems but on a short time period, due to the considerable computational resources needed.

In this thesis is presented a method for complex systems optimization on a long time period, a "divide and conquer" optimization technique based on linear programming (LP) is implemented and tested. The adopted technique is demonstrated to be efficient and effective with respect to a comprehensive optimization, input data uncertainty has also been modelled to analyse to what extend the model can be segmented without significantly affecting the confidence interval of the optimal result.

The Italian electricity market modus operandi is analysed and a strategy to generate a bidding offer profile is proposed, thus demonstrating that the optimization model is both useful on the long term planning and on the short term operations.

A dynamic programming (DP) optimization approach is also explored, DP resulted to be more effective in the introduction of non-linear logics into the model with final higher optimization performances. The application of this model is demonstrated to be more efficient than LP in solving long term single reservoir problems but cumbersome in case of multiple-reservoir system.

Keywords: PoliMi, Optimal reservoir operation, Optimization, Linear Programming, Dynamic Programming, ARMA, Synthetic streamflows

Sommario

Nella produzione di energia idroelettrica l'acqua è considerata la risorsa scarsa. Con la liberalizzazione del mercato elettrico in Italia, l'applicazione di tecniche di ottimizzazione ha guadagnato sempre più importanza nella gestione delle aste idriche.

Il modello del problema è caratterizzato da una funzione obiettivo e da vincoli non lineari, spesso descritti da condizioni booleane. In letteratura, la soluzione di questo tipo di problemi si concentra soprattutto sul breve periodo, oppure sul lungo periodo ma solo caso di sistemi idrici relativamente semplici, a causa del grande costo computazionale richiesto.

In questa tesi viene presentato un metodo per l'ottimizzazione sul lungo periodo di aste idriche complesse, secondo un approccio al problema di tipo "Divide et impera" basato sulla programmazione lineare. La tecnica adottata si è dimostrata efficiente e non meno efficace rispetto ad una ottimizzazione globale del sistema, questa affermazione è basata anche sull'analisi quanto la soluzione ottima perda di significato in funzione dell'incertezza sui dati in ingresso.

Il modus operandi del mercato elettrico italiano è stato inoltre analizzato al fine di individuare una strategia per la composizione di un profilo di offerta che tenga conto della natura e dello stato attuale dell'asta idrica. Il modello sviluppato si dimostra quindi utile sia per la gestione sul lungo periodo che per la programmazione operativa nel breve.

Un ulteriore approccio basato sulla programmazione dinamica è stato implementato per verificarne le potenzialità. Il modello si è dimostrato più efficace da un punto di vista dell'ottimo, in quanto è stato possibile introdurre relazioni non lineari. Nella soluzione di un' asta idrica a singolo bacino sul lungo periodo, la programmazione dinamica ha costi computazionali inferiori a quella lineare, si è però dimostrata inefficiente nella risoluzione di aste idriche con più bacini.

Parole chiave: PoliMi, Gestione ottima, Ottimizzazione, Programmazione Lineare, Programmazione dinamica, ARMA, Afflussi sintetici

Chapter 1

Introduction

1.1 Description of the problem environment

In order to better understand the work accomplished, a general description of the context subject of study is given. In the following section some notions on how the Italian energy market works, the main characteristics of a multi-reservoir hydroelectric system, the case study taken into consideration and the differences between a long term and a short term optimization are described.

1.1.1 The Italian Power Exchange Market

The Power Exchange Market plays a fundamental role in the balancing and in the regulation of electric energy production. Since 1$^{\text{st}}$ April 2004, with the liberalization of the market in Italy, a system of interconnected power auctions determines the price of the energy. Understanding the dynamics behind the energy price is fundamental for a correct optimal allocation of power production.

The principles at the base of the Power Exchange Market are similar to the ones of a classic market, where the price is determined by the equilibrium between supply and demand, as shown in figure fig. 1.1.

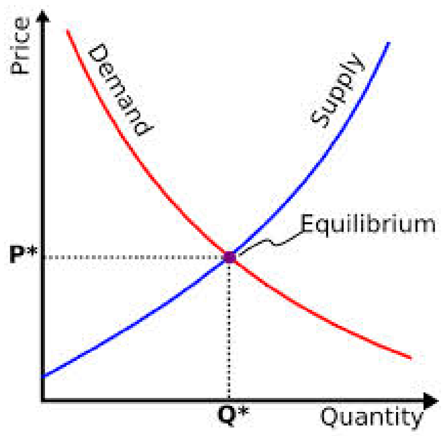


Figure 1.1: Equilibrium price

Due to the fact that electric energy cannot be stored, maintaining this equilibrium is of fundamental importance. Thanks to the implementation of a optimization algorithm described in *Manuale utente per operatori di Mercato* [19], to each producer is assigned a proportion of the energy that will have to provide at a determined period of time and the price associated.

As defined in *Guida al Mercato Elettrico* [18], Electricity market is composed by different sections divided on the base of the time-frame to which are referred. A synthetic representation is given in fig. 1.2 on the next page.

The focus will be on the MGP (Mercato del Giorno Prima) and MI (Mercato Infra-giornaliero) as they are the main markets for volume of energy traded. In particular the MGP is referred at the day after with

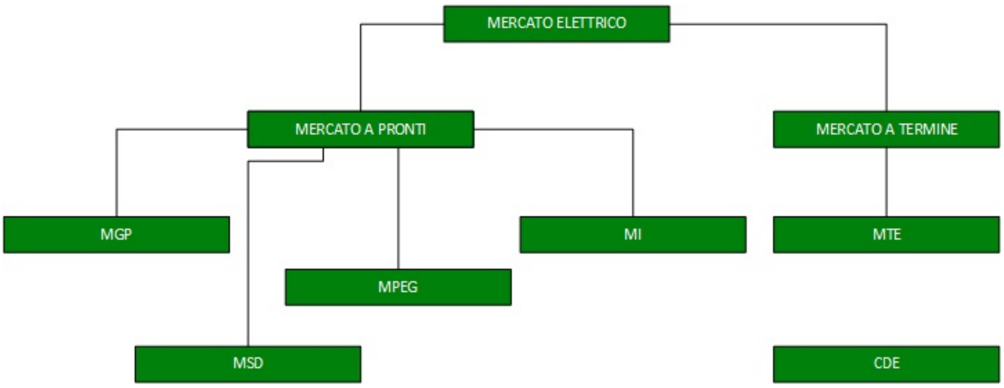


Figure 1.2: Electrical market structure

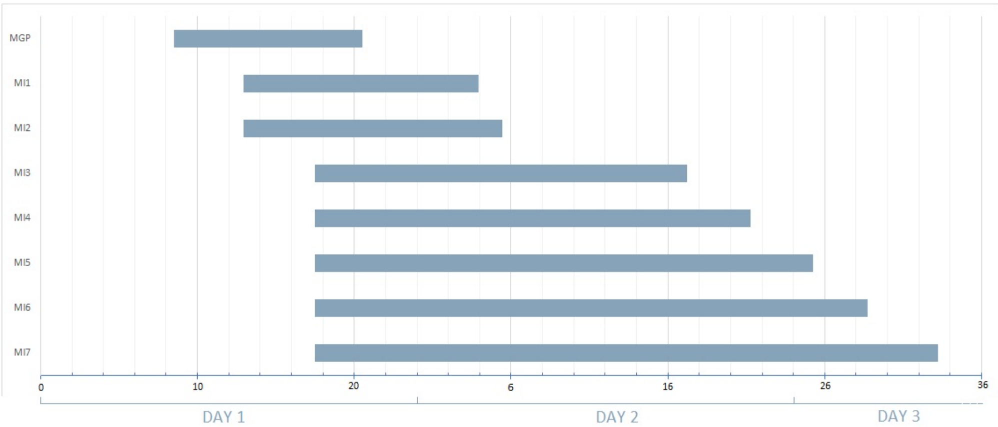


Figure 1.3: Electrical markets time slots

respect to the day in which the auction is held, meanwhile MI is articulated in 7 different decreasing-length time slots and the auction is held on the same day of the energy dispatch. As can be seen from figure 1.3 there is a partial overlap of the different auctions, this makes the optimal offer profile (in term of quantity of energy and price) very difficult to estimate, moreover there is a strong uncertainty on the success probability of the producer's offers.

Due to the complexity of the problem in short term analysis only the MGP will be considered, which is the main and most independent market, for each hour the producer can offer 4 different quantities of energy at 4 different prices. If the realized market equilibrium is above price set by the producer it wins the auction and must produce the amount of energy agreed at the final market equilibrium price. An example of an auction outcome is reported in table 1.1.

Table 1.1: Example of electricity market auction outcome

Hour		1	2	3
Offer profile				
Offer 1	energy offered MW h	10	20	5
	at price €/MW h	55	60	45
Offer 2	energy offered MW h	20	30	15
	at price €/MW h	60	65	50
Offer 3	energy offered MW h	30	40	20
	at price €/MW h	65	70	55
Offer 4	energy offered MW h	60	50	25
	at price €/MW h	70	75	65
Outcome				
	Market equilibrium price	63	75	40
	Total energy sold MW h	**30**	**140**	**0**
	at price €/MW h	63	75	-

rejected: 5 (hour 3, Offer 1); 15 (hour 3, Offer 2); 30, 20 (hours 1 and 3, Offer 3); 60, 25 (hours 1 and 3, Offer 4)

accepted: 10, 20 (hours 1 and 2, Offer 1); 20, 30 (hours 1 and 2, Offer 2); 40 (hour 2, Offer 3); 50 (hour 2, Offer 4)

1.1.2 Generic system of Hydro-Electric Power Stations in cascade

In this paragraph a generic discussion of a multiple reservoir problem connected in cascade and the mathematical dynamics equations that govern the system are presented. The objective function and the main physical constraints are also modelled while particular considerations related to specificities of the case study taken into consideration are described in 1.1.3. Let's consider a cascade of hydro-electric power stations like the one shown in fig. 1.4 on the facing page.

The water volume in the considered reservoir i at time t is $Q_i(t)$, where $i = 1, 2, \ldots, I_{reservoirs}$ and the flow that pass through each control variable v at time t is $X_v(t)$, where $v = 1, 2, \ldots, N_{variable}$. Please note that each hydroelectric power station (designed in fig. 1.4 on the next page with Xv) can be composed by more than one variable, for example a power station can have two turbines and an emergency bypass for a total of 3 control variables. The emergency bypass is a variable allows the water to flow from the reservoir without producing power, and thus revenues. It is used to manage cases in which high streamflows in the reservoir cannot be fully

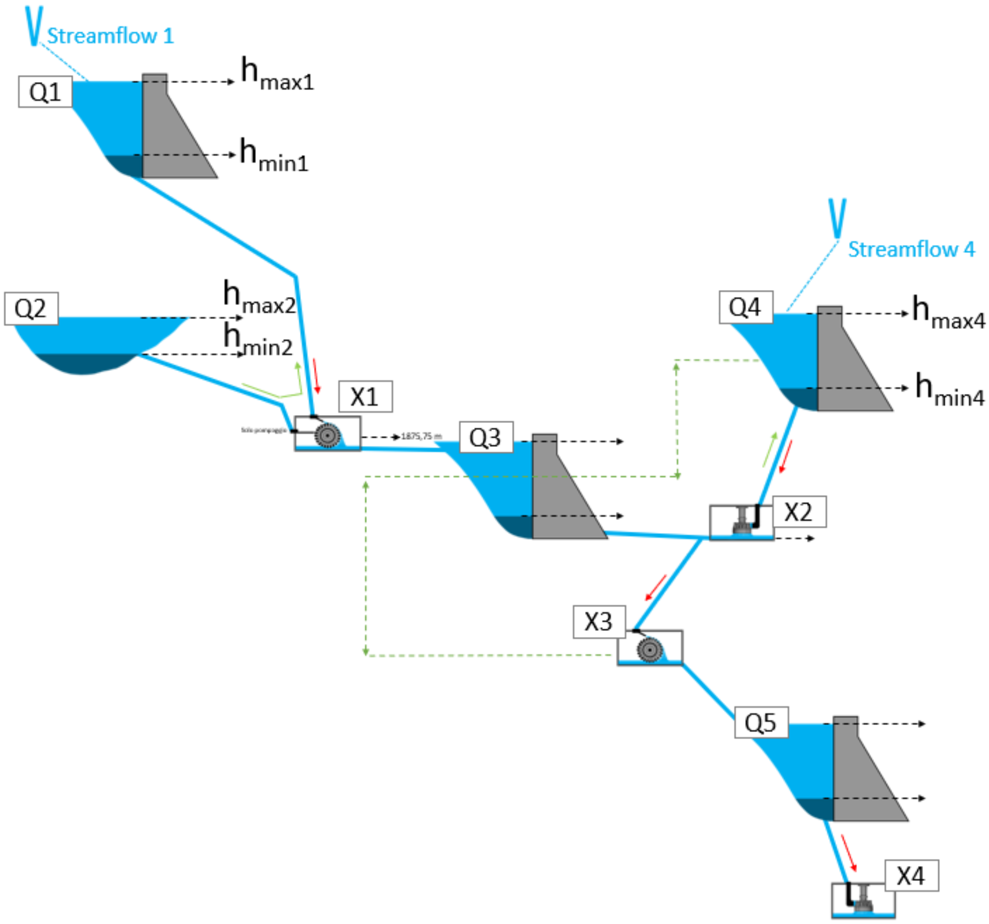


Figure 1.4: Example of a general cascade reservoir system

discharged by the turbines, they thus physically represents the locks of the dam or the river in case of on-the-river plants. Other data that influence the system are the price of the energy at hour t, $P(t)$ and the streamflow of water coming into each reservoir i at time t $A_i(t)$.

The equations that govern the volume of water inside a reservoir is described by the following discrete-time control system:

$$Q_i(t+1) = Q_i(t) + A_i(t) + \sum_{v \in In_i} X_v(t) - \sum_{v \in Out_i} X_v(t) \tag{1.1}$$

$$t = 1, 2, ..., T$$

$$i = 1, 2, ..., I$$

$$V_i(0) = V_0$$

Where $V_i(0)$ is the vector of initial stored water volumes in the reservoir i, In_i represents the set of variables that are tributaries of the reservoir i while Out_i is the set of emissary variables of the reservoir. Please note that the time has been discretized according to time segments of 1 hour. There are two main reasons for the discretization, the first is that the control of power load on turbines is set hourly and also the price of energy is given hourly, the second motivation is that more efficient optimization approaches can be applied for discrete systems, as explained in chapter 2.2.

Considering fig. 1.4 on the preceding page, the water balance for the reservoir Q5 is written as:

$$Q_5(t+1) = Q_5(t) + A_5(t) + X_3(t) - X_4(t)$$

A control variable $X_v(t)$ can used to represent both a turbine or a pump, this is due to the fact that in some plants the water can be pumped in reverse direction overnight(or when the price of energy is considerably low) to be accumulated and then used to produce electric energy when it's price is high. Spillways are not explicitly considered in this model but are introduced into the streamflow, that are net of the spillway for each reservoir.

The control variables and the reservoirs are subject to the following constraints, that can be function of time:

$$H_{min}(t) \leq H_i(t) \leq H_{max}(t) \qquad \text{max and min height} \tag{1.2}$$

$$\Delta H_{min}(t) \leq \Delta H_i(t) \leq \Delta H_{max}(t) \qquad \text{max and min rate of change} \tag{1.3}$$

$$X_{V,min}(t) \leq V_v(t) \leq X_{V,max}(t) \qquad \text{max and min flows} \tag{1.4}$$

$$Pw_{v,min}(t) \leq Pw_v(t) \leq Pw_{v,max}(t) \qquad \text{max and min power} \tag{1.5}$$

$$Xc_{min}(t) \leq \sum_{v \in C} X_v(t) \leq Xc_{max}(t) \qquad \text{combined max and min flow} \tag{1.6}$$

$$Pc_{min}(t) \leq \sum_{v \in C} P_v(t) \leq Pc_{max}(t) \qquad \text{combined max and min power} \tag{1.7}$$

and to the following boundary conditions:

$$Q_i(t_{start}) = \overline{Q}_{i,start} \qquad \text{intitial reservoir volume} \tag{1.8}$$

$$Q_i(t_{end}) = \overline{Q}_{i,end} \qquad \text{final reservoir volume} \tag{1.9}$$

Constraint 1.2 limits the maximum and minimum altitude that a reservoir can have, this constraint is used to respect the physics of the system but also the legislations about flood and draught, thus can be time dependant on a seasonal basis. The height of a reservoir is strictly related to its volume, depending on the shape of the basin considered, the function can be expressed with the following relation: $H_i(t) = f_i(Q_i(t))$. Constraint 1.3 refers instead to the maximum rate of change in height, this limit is present on certain cases where problems related to the basin structure may arise, the constraint is often a step function, based on the volume of water stored into the reservoir. Constraints 1.4 and 1.5 refers to the upper and lower limits of the control variables in term of water flow and power generated. If the variable is a turbine this limit is discontinuous, the turbine can be turned off or it can control its output between a minimum flow and a maximum one, so the range of controllability is: $X_v(t) \in \{0; X_{min}(t) \div X_{max}(t)\}$. In the case of a pump, a similar discussion can be done, but now maximum and minimum values are taken negative. The combined flow 1.6 and power limits 1.7 are used when a single power plant has more than one variable, in this case a minimum vital outflow or a maximum admissible flow for the river downstream must be defined, the constraint is expressed as the sum of all the variables of that plant. Please note that constraints 1.4 and 1.6 can be imposed also at the same time.

Considering a discrete-time optimal control problem, the functional, representative of the profit, is written as:

$$L(X(t), Q(t), P(t)) = \sum_{t=1}^{T} Price(t) * \sum_{v=1}^{N_v} [X_v(t) * ken(H_v(t)) * \eta_v(X_v(t))] \tag{1.10}$$

Where $ken(H_v(t))$ is the energetic coefficient of the water indicative of its potential energy, it tells how much energy is stored in a cubic meter of water: $\mathrm{kWh/m^3}$, it is function of the pressure head of water which is proportional to the height of the reservoir $H_v(t)$ from which the turbine takes water from. The efficiency of the turbine is expressed with $\eta_v(t)$ and depends from the load curve of the turbine, usually it reaches maximum at nominal capacity, while it decreases at lower flow rates the efficiency degrades.

The final objective of the optimization is to find a set of $\overline{X}(t)$ that maximize 1.10 subject to the dynamics described in 1.1 and to constraints eqs. (1.2) to (1.7).

1.1.3 Case study presentation

In the previous chapter a general problem for multiple Hydro-Electric Power Stations in cascade has been defined, now a detailed description of the real system taken as case study is provided.

The studies are focused on the existent hydroelectric concession, for disclosure reasons name and location of the water system along with the specific data regarding the system cannot be provided, although the principle scheme is described in figure 1.5; reservoirs are marked in blue, while power plants are in red. Informations about the main parameters of the concession are reported in table 1.2 for what concerns the variables, and in table 1.3 for informations about the reservoirs[1]. The method is valid for a wide range of systems, absolute value are not descriptive of the system physics; what defines the behaviour of a reservoir/power plant system is the ratio between the reservoir capacity and the nominal flow of the related turbine. This ratio represents the maximum hours of stock, a

[1] For disclosure reasons the displayed data are normalized, no real data regarding the case study are provided

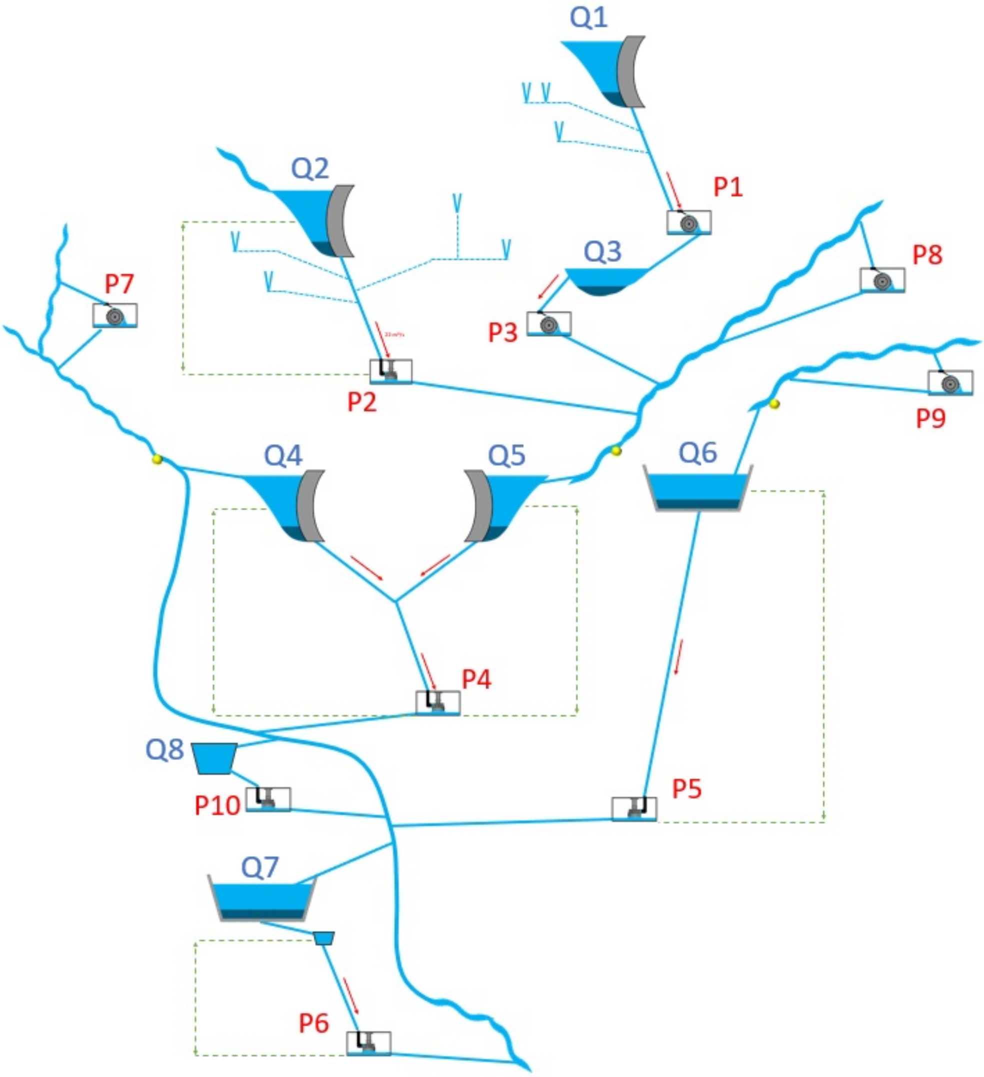


Figure 1.5: Case study principle scheme

high ratio corresponds to a big reservoir while a low ratio indicates a small reservoir with respect to the turbine capacity.

Table 1.2: General information about variables in the system considered as case study, normalized with respect to plant $P10$

variable	**plant / reservoir**	**typology**	**nominal flow norm.**	**nominal power** MW
X1	P6	turbine	32	30
X2	P6	turbine	32	30
X3	P6	turbine	32	30
X4	P6	turbine	32	30
X5	P6	turbine	32	30
X6	P10	turbine	100	50
X7	P4	turbine	96	90
X8	P2	turbine	31	20
X9	P1	turbine	8	30
X10	P3	turbine	10	15
X11	P5	turbine	8	15
X12	Q8	bypass	400	0
X13	Q7	bypass	400	0
X14	Q4	bypass	400	0
X15	Q2	bypass	400	0
X16	Q1	bypass	400	0
X17	Q6	bypass	400	0
X18	Q5	bypass	400	0
X19	Q3	bypass	400	0

Table 1.3: General information about reservoirs in the system considered as case study

reservoir	hours of stock[2]
Q1	750
Q2	38
Q3	4
Q4	6
Q5	6
Q6	2
Q7	0.2
Q8	1

[2] Hours of stock is the amount of water stored in the reservoir expressed in hours, calculated as the ratio between reservoir capacity and turbine nominal flow

The dynamics that governs the system, the constraints imposed and the input data coming from real operational data and are only partially described by the eqs. (1.1) and (1.3) to (1.7); with respect to the general case, particular constraints are added. From figure 1.5 can be noticed how the structure of interconnected reservoirs is not a pure cascade one, there are run-of-the-river generation plants P_7, P_8, P_9, P_{10}, moreover P_4 power plant has an inflow coming both from Q_4 and Q_5 Basins. The case of run-of-the river plants is trivial to optimize, in the turbine should flow the most water possible respecting the constraints since no storage capacity is available, thus for the purpose of this thesis they are not relevant variables and have been neglected. Q_4 and Q_5 reservoirs presents instead an added limit, since the pressure pipes that bring water to P_4 are interconnected, there is a backflow from one basin to the other in case their height is not equal, moreover the portion of water collected from the two reservoirs is not constant but is function of their pressure head.

Another important constraint regards the operation of turbines, if a turbine is turned on several times in a day there may be the risk of premature wear and failures. This can be considered a soft limitation since it's only a preference expressed by the operator of the plant, in fact, if the price of energy is particularly high in a determined time span the optimal solution is to power the turbine disregarding the number of activations.

The last important variable to consider is the concentration time; if the upstream reservoir is far from the downstream one the transfer of water is not immediate but may take a while, so the balance of water in the reservoir is shifted in time, and equation 1.1 becomes:

$$Q_i(t+1) = Q_i(t) + A_i(t) + \sum_{v \in In_i} X_v(t - Tc_v) - \sum_{v \in Out_i} X_v(t) \qquad (1.11)$$

Where Tc_v is the concentration time of the variable X_v and referred to the downstream reservoir.

The problem presents also uncertainty in input parameters, in particular for what concerns streamflows to reservoirs and forecasted price. Variance in this data causes a variation in the optimum solution. In order to analyse the system behaviour, a synthetic generation of input time series that reflect the real ones is generated as described in chapter 3.

1.1.4 Long term and short term optimizations

Optimization of electric energy production is performed on different time horizons. Long-term objectives focus on calculating the best strategy to generate as much revenues as possible, keeping into account that there is a yearly cyclicity, and so the state of the system at the end of the year must be coherent with the initial conditions (see eqs. (1.8) and (1.9)). However, in reality, it is often the short-term optimization that dictates the operational strategy, in the short term there is lower uncertainty on the input variables and constraints, for example the number of turbine activations becomes an important decisional factor. Short-term optimization is run more than one time per day, according to the market outcomes, and the optimization time-span is usually 3 to 7 days length; thus the algorithm is required to solve the problem in a short computational time.

Also the objective changes: on the long-term the optimal strategy is based on the forecasted energy price while on the short-term also the electrical market dynamics described in 1.1.1 play a fundamental role. Therefore, incorporating long-term goals into short-term objectives is crucial. It's not only important to follow the guidelines dictated by the long-term optimization, but It's also needed to identify a suitable offer profile for the energy auction in the different hours (see 1.1).

1.1.5 Considerations about problem complexity

As can be seen from sections 1.1.1 to 1.1.3 the problem presents an high degree of complexity. Although its dynamic(1.11) has an intrinsically linear structure, the majority of constraints and the functional 1.10 are expressed through non-linear functions. Linearisation of some functions is possible, in particular if a linearisation is applied to $H_i(Q_i(t))$, $ken(H_v(t))$ and $\eta_v(X_v(t))$ the objective function ends up to be cubic-dependent from the control variables:

$$L(X(t), P(t)) = \sum_{t=1}^{T} Price(t) * \sum_{v=1}^{N_v} [X_v(t) * a * Q_i(t, X_v)(t) * b * X_v(t)]$$

where a and b are the two linearisation coefficients (for the sake of simplicity the linearisation line intercept coefficients are neglected). Please note that the linearisation of ken function is split in 3 steps: firstly the function is linearised around H_i, then the height of reservoir is linearised according to Q_i which is considered to be in linear dependency with X_v.

Step and boolean constraints such as the minimum load for the turbines, maximum height rate of change and the maximum number of turbine activation cannot be linearised or even, in the latter case, expressed by an algebraic function. This impose serious limits to the algorithms that can be used for the optimization. Moreover, for what concerns the long-term, the problem dimension is huge: it requires to find the optimal value for each control value in each hour on a one year period (8760 hours). This further limits the types of optimization techniques that can be used since computational time and memory are scarce resources. Another important thing to consider is the uncertainty in input parameters, this causes a variance in the possible outcome of the optimization problem, thus diminishing the importance of getting the optimal solution and shifts the focus to the calculation of a more robust solution.

It clearly appears that in order to deal with the problem it is needed to find a trade-off between approximation of the model (and so the choice of algorithm used), computational resources required and optimal result.

1.2 Outline

The thesis is structured as follows:

In this chapter A general description of the context subject of study is given. Some notions on how the Italian energy market works, the main characteristics of a multi-reservoir hydroelectric system, the case study taken into consideration and the differences between a long term and a short term optimization are described.

In the second chapter a state of the art in hydro-economics optimization of multiple reservoirs systems operation is provided. The different models are classified, both implementation examples and results are reported from literature and an overview of the main optimization techniques for reservoir management is reported. In this chapter also methodologies for daily streamflow identification and synthetic time-series generation are described. The latter section is dedicated instead to the introduction of the currently adopted base optimization model on which this work of Thesis is based.

In the third chapter the analysis of historical time series identification and the methods for the generation of synthetic data-set are described. Water streamflows and electricity price are the two time series analysed in this chapter. Assumptions made, techniques adopted, considerations and results are reported in this chapter.

In the fourth chapter the optimization technique adopted for the long term optimization is described. The long term optimization pf problem with objectives and assumptions are reported. Details about the working principles of the "divide and conquer" strategy and a base line on the algorithm implementations are given. Performances are reported and compared with the base optimization approach, optimal model parameters identification and optimum variance analysis is also performed.

In the fifth chapter the short term optimization is introduced. The focus is shifted on the generation strategy of a possible bid profile to be offered on the market. The assumptions made, the analysis method utilized, comments on results and a final practical example of strategy evaluation is reported.

In the sixth chapter the optimization technique of dynamic programming (DP) is introduced. Objective and assumptions made are reported and a different model of the problem is implemented for the solution with DP. Dynamic programming algorithm implementation is reported with particular attention to underline potentialities and weaknesses. Methods of state space discretization are described and implemented, basics on a possible optimal allocation logic are given. Finally performances of the new approach are compared to the existing base model, pointing out strength points and limitations.

Chapter 2

State of Art

2.1 Hydro-economic models of multi-reservoir systems

The management and operation of a system responsible for the storage of a valuable commodity will always be a topic open for investigation and potential improvement, attention focuses on improving the operational effectiveness and efficiency of existing reservoir systems for maximizing the beneficial uses. This dissertation provides a state of the art in hydro-economics optimization of multiple reservoirs operation, the aim of these models is to provide support to decision making in water resources management.

As described in [30] the different hydro-economics models can be classified according to the following three main characteristics:

1. Simulation vs Optimization: Simulation models uses rule-based algorithms to describe the system, Optimization models maximize an objective function in respect to a set of constraints. Simulation models are better suited in the case a system is governed by complex non-linear system dynamics but their application is limited to small scale problem since they are computationally more expensive than optimization techniques.

2. Representation in time:

   (a) Deterministic models, that uses an historical or synthetically generated time-series as input. A general dissertation of these techniques can be found in [13].

(b) Stochastic models, that include the statistical profiles characteristics in the model input and provide the results in form of probability distributions. Labadie in [15], defines these techniques as Explicit Stochastic Optimization.

(c) Dynamic optimization models, is both a mathematical optimization method and a computer programming method. The sub-problems are nested recursively in time inside larger problems so that dynamic programming methods are applicable. This technique is applied in chapter 6.

3. Model integration

   (a) Modular approach, where the components (water reservoirs in the case of this work) are optimized independently.

   (b) Holistic approach, where all the components are integrated in one single problem. The benefits of this latter technique is that can be easier to find a global optimum, on the other hand, the problem formulation is much more complex.

Table 2.1, reported by Vat in [30], presents an overview of publications on the application of hydro-economic models for optimal reservoirs operation, describing informations about the location and type of technique used for optimization. Most of the researches listed in the table optimize the reservoir use not only from a power generation point of view but consider also other uses such as irrigation or flood control.

Please note that all the listed models presented follow an holistic approach based mainly on optimization techniques such as mathematical programming. The majority of the models inspected presents strong simplifications in the description of the water system, this is mainly due to cope with the constraints given by the different optimization techniques.

The holistic based approach has been applied in this thesis since these techniques are the most widespread in literature for the economic optimization of reservoirs operation. Moreover, thanks to the general diffusion of these approaches also in other fields they have solid mathematical theoretical basis and a robust demonstration of their effectiveness. In the following table 2.2, reported from [30], are listed the (dis-)advantages of holistic programming versus modular heuristic programming.

Table 2.1: Overview of studies using hydro-economic models for reservoir optimization

References	Location	Functions	Model type
Lund and Ferreira, 1996	Missouri River	USA Hydropower, flood control, recreation, water supply and navigation	Holistic, ISO, deterministic, linear programming
Rosegrant et al, 2000 Cai et al, 2003	Maipo River	Chili Irrigation, hydropower and environmental flow	Holistic, non-stochastic, deterministic, non-linear programming
Bielsa and Duarte, 2001	Vadiello Reservoir, Spain	Irrigation and hydropower	Holistic, non-stochastic, deterministic
Babel et al, 2005	Nong Pla Lai Reservoir	Thailand Irrigation, domestic, industry, hydropower	Holistic, deterministic, linear programming
Whittington et al, 2005	Nile	Ethiopia, Egypt and Sudan Irrigation and hydropower	Holistic, non-stochastic, deterministic, non-linear programming
Ringler et al, 2006	Dong Nai River	Vietnam Irrigation, hydropower, domestic and Industry	Holistic, non-stochastic, deterministic, non-linear programming
Pulido-Velázquez et al, 2006	Adra River	Spain Irrigation and domestic	Holistic, non-stochastic, deterministic, non-linear programming
Schoups et al, 2006a and 2006b	Yaqui River	Mexico Irrigation	Holistic, ISO,deterministic, non-linear programming
Watkins and Moser, 2006	Panama	Canal System Water supply, hydropower, navigation and flood control	Holistic, ISO, deterministic, linear programming
Consoli et al, 2007	Pozzillo Reservoir, Italy	Irrigation	Holistic, non-stochastic, deterministic, non-linear programming
Tilmant et al, 2008	Euphrates	Turkey and Syria Irrigation and hydropower	Holistic, ESO, Stochastic Dual Dynamic Programming
Goor et al, 2010	Nile	Ethiopia, Egypt and Sudan Irrigation and hydropower	Holistic, ESO, Stochastic Dual Dynamic Programming
Tilmant et al, 2010	Tilmant and Kinzelbach	Mozambique Environmental flows, irrigation and hydropower	Holistic, ESO, Stochastic Dual Dynamic Programming
Bartlett et al, 2012	Nam Ngum River	Laos Irrigation and hydropower	Holistic, non-stochastic, deterministic, non-linear programming
Xu et al, 2014	Nanpan River	China Hydropower, irrigation and water quality	Holistic, non-stochastic, deterministic, dynamic programming
Mirchi et al, 2015	Florida	USA Irrigation, public water supply, flood protection and environmental flows	Holistic
Rougé and Tilmant, 2015	Tigris and Euphrates	Turkey and Syria Irrigation and hydropower	Holistic, ESO, Dynamic Programming
Satti et al, 2015	Blue Nile	South Sudan Irrigation and hydropower	Holistic, non-stochastic, non-linear optimization

Table 2.2: (Dis-)advantages of holistic approach and modular approach in hydro-economic reservoir optimization

Holistic programming	Modular heuristic
Widespread and efficient algorithms can be applied (+)	Usually they are expensive from a computational point of view (-)
Able to identify unique global optimal solution (+)	Approximation of the optimal solution, risk of identifying a local optimum (-)
Optimization of release flow only, operation rules have to be derived (-)	Optimization of any simulation model input, also possible directly on operation rules (+)
Requires recoding and simplification to model the water resources and economic systems (-)	Can be constructed around existing simulation models (+)

2.2 Optimization techniques for optimal multi-reservoir operations

Optimizing the benefits in a water resource system is a non-trivial task. The solution to the problem is difficult since the system dynamics are often non-linear, a large number of variables is involved and boundary conditions have often a stochastic nature. Nevertheless, different mathematical programming techniques have been developed for the computation of optimal operating strategies in multiple water reservoirs systems([10]). The majority of these techniques produce satisfactory results for the tasks they are developed for but a generic methodology that can handle the most complex problems in a generic form has yet to be identified. The rapid development in computer technologies has made the adoption of more sophisticated mathematical methods possible, the nature of reservoir systems problems usually require the optimization of an objective function in a large decision space of optimal parameters set. This problems has motivated the development of various optimization techniques.

As seen in the table 2.1 and in literature [13, 15, 35], the use of mathematics, in particular Operational Research and Optimal control techniques, is a widely and well established approach to develop optimal management strategies for the operation of water reservoir systems.

According to [13] the most commonly used mathematical optimization techniques can be classified as follows:

1. Linear programming(LP)
2. Non-linear programming(NLP)
3. Augmented Lagrangian Optimization
4. Dynamic programming(DP)
5. Computational intelligence
6. Simulations

The first three methods require an holistic formulation of the problem while the latter two can be implemented through a modular approach.

In the state-of-the-art review by Labadie in [15], where the different techniques for optimal reservoir operation have been investigated, Linear Programming(LP) is praised for its efficiency and capability to reach a global optimal solution. However also Mixed Integer Linear Programming and Dynamic Programming models have been considered, although less computationally efficient, these models provide an alternative solution for representing non-linear constraints. In extension to Yeh work in [35], Labadie explores also heuristic methods, simulation methods and neural network models; this is due to improvements in computer technologies since the latter techniques are computationally expensive.

2.2.1 Linear Programming

In a problem where the objective function is linear and all the constraints (equalities and inequalities) can be expressed in a linear way too, the LP technique can be used. It is one of the most widely used technique due to its implementation simplicity and computational efficiency, moreover the solution provided is a global optimal result.

A practical example on the Linear Programming implementation and solution of an optimal control problem of hydro-electric power stations in cascade is discussed in [26]. The system was considered with a possibility of turbining and pumping in some of the power stations and was translated into a discrete-time optimal control problem that has been solved numerically, adopting a penalty function method. Computation were done with real data and pointed out that the reservoirs only produce electricity when the price is high enough to justify that production, this is index of a good formulation and solution of the problem. The optimization has been computed over a period of 24 hours, no informations about the computational costs were provided but according to [30] the methods used are particularly computational demanding.

A simple model developed by Arthur E. Mc Garity in [3] describes the fundamental behaviours of a single reservoir system and the description of the constraints and variables are similar to the eqs. (1.1) and (1.3) to (1.7) used in this thesis.

Another demonstration of LP application to an optimal reservoir operation problem is presented by Maass et al. in [16], the goal was to maximize the revenues coming from water exploitation while satisfying the constraints of the system. Maass et al. analysed the problem in its dual form: a first implementation used as decision variables the volumes in reservoirs, while in the second implementation the decision variables where the target releases from the basins.

In [34] is described the use of a model called HYDROSIM, implemented for the Tennessee Valley reservoir system, that employ LP to compute the optimal level of basins water level for hydro-power generation.

A technique of separation of long term problems in smaller time domain sub-problems based on sequential LP optimization is cited by Labadie in [15] but no actual implementation examples have been found.

2.2.2 Non-Linear Programming

Non Linear Programming technique is generally used when either the objective function and the constraints are non-linear functions. Non-linear programming techniques refer to the generic problem:

$$\text{Minimize:} \quad L = f(x_1, x_2, ..., x_n) \tag{2.1}$$
$$\text{Subject to:} \quad g_i(x) = 0 \qquad i = 1, ..., m \tag{2.2}$$
$$\text{Subject to:} \quad c_j(x) \leq 0 \qquad j = 1, ..., m \tag{2.3}$$

Yeh in his state of the art review [35], lists the NPL algorithms generally utilized due to their robustness and efficiency:

(i) Sequential Linear Programming (SLP)

(ii) Sequential Quadratic Programming (SQP)

(iii) Augmented Lagrangian Methods, or Method of Multipliers (MOM)

(iv) Generalized Reduced Gradient Method (GRG)

SQP and can be applied in case the objective function is quadratic and the constraints are linear, it is computational efficient and generally, the result is a global optimum. If the problem can be demonstrated as convex (objective function must be concave and constraints set must generate a convex space), efficient methods for convex optimization can be used. In the case study considered in this thesis the constraints aren't convex and so this techniques cannot be applied.

Methods "i" and "ii" require the eqs. (2.1) and (2.3) to be differentiable, moreover sufficient conditions for optimality are difficult to demonstrate and the solution of the necessary Karush–Kuhn–Tucker (KKT) conditions is computationally heavy.

Hiew, in [12], performed a comparative evaluation of SLP and MOM algorithms in a large-scale multi-reservoir hydro power system and concluded that the two methods differ only for 1% in final optimization value but SQL is about two orders of magnitude faster than MOM algorithms. The comparison between these methods has been deepened by Labadie in [15], in his works is reported that the Taylor approximation of non-linear function, typical of Sequential Programming methods, brings to a reduction in process convergence.

NLP is not very popular in the solutions of optimal reservoirs operation due to the computational complexity and the difficult theoretical formulation although some successful cases have been reported by Husain in [13] where a quadratic optimization model for the California Central Valley Project has been compared with a LP model, model was compared with an LP model, and it was found that a significant increase in the total energy production could be obtained using SQP models. Another example of application, based on the Augmented Lagrangian Method is proposed in [4], where a model for solving the short term management of water reservoirs with variable waterfall is proposed. Even with a short time problem is reported that the solution is obtained after a moderate number of iterations with all constraints being satisfied.

2.2.3 Dynamic Programming

Dynamic programming (Principle of optimality by Bellman 1957) for discrete space-time system and the Hamilton-Jacoby-Bellman (HJB) equations that constitutes the continuous form counterpart are well know and largely used sufficient conditions for optimal control problems. In particular the Dynamic Programming (DP) technique, due to it's easiness of implementation and powerfulness in describing complex systems is largely used in multi-reservoir operational optimization. DP can deal with non-convex, non-linear and discontinuous objective and constraint functions, moreover it reaches a global optimum. On the other hand the computational cost scale with a power of two proportion in function of the complexity of the system considered.

Being DP suitable to describe the most complex systems, the technique has been adopted in this thesis but it's usage has been limited to a smaller set of sub problems due to the demanding computational power of the algorithm.

As can be seen in table 2.1, Dynamic Programming is the second used method after Linear Programming. An dissertation is reported by Opricovic S. in [25], where a first rough optimization if followed by the application of Dynamic Programming to solve the problem of optimal control. A similar two stage optimization method for Dynamic Programming has been implemented in this thesis although the optimal reservoir problem and the objective function are completely different. Opricovic S. implemented the method to solve the optimal planning of a single reservoir in which water was needed as primary need to irrigation and power production was considered as secondary benefit.

Although computationally heavy, DP has been compared with Mixed Integer Programming by Labadie in [15], and his works pointed out that the two formulation provided comparable accuracy, but at a fraction of the computer execution time required by the MIP model.

2.2.4 Computational Intelligence

Computational intelligence techniques (fuzzy logics, neural networks, neuro-fuzzy systems and evolutionary algorithms) are successfully applied in various fields of engineering. A lot of these methods have been implemented for solving control problems in packet switching network architectures. Therefore, as stated in [13], a generic technique for the optimisation of complex systems is yet to be identified. Different cases of computational intelligence techniques applied to reservoir optimization can be found in literature although they refer to very specific and non generalized problems.

An example is from [24], in which an extended two-stage stochastic programming with fuzzy variables is developed for water resources management under uncertainty. In the first optimization stage a MIP technique is used, while for the second stage of optimization the problem is implemented by using fuzzy variables and solved using fuzzy chance-constrained programming. The model is suitable to deal with cases under uncertainty, however the main motivation for the use of fuzzy logic is due to the particularity of the object to be optimized: the water was needed to be allocated at different stakeholders, the value function keeps track of the utility to allocate water to one actor to another. This problem significantly differ from the objective of this thesis, where the optimization object is related to hydro power production.

The most significant difference of computational intelligence models with respect to the traditional ones are that they use probabilistic transition rules, not deterministic rules.

As reported by Labadie in [15] and also by Husain in [13] computational intelligence is mainly used when multiple actors enter in the value function and strong non-deterministic components affect not only the inputs of the model but also its structure. Moreover the use of these models often requires a training of the algorithm, and so the need to provide already optimal solutions.

2.2.5 Conclusions

Reservoir system operations has a very rich and extended history for what concerns the application of optimization models. However the use in real-word of these techniques is still limited, as stated by Labadiein [15]: "system operators states that they don't like being told what to do and that they prefer to make decisions in his own way. This is due also to the fact that avoidance of difficulties and perceived system failures are dominant goals with respect to improving efficiency. Labadie also lists 3 key-factors for a successful implementation of reservoir systems optimization models:

1. Improving levels of trust by a more interactive involvement of decision makers in system development

2. A good implementation of these systems in the actual Enterprise Resource Planning

3. Improved linkage with simulation models that are easily interpreted by operators

2.3 Time series analysis

Historical time series about water streamflows in each of the reservoirs are already available for our case study, also the whole time history of electricity price in Italy is available at http://www.mercatoelettrico.

org/it/ [17]. Streamflows time-series are defined over a daily basis while electricity price for the MGP (see 1.1.1) has an hourly resolution. In order to test the behaviour of the model implemented in this thesis, historical data are not enough and the generation of synthetic time series is needed. In particular it is fundamental to generate a realistic time series for the streamflow, since this input directly affect the feasibility of the solution. To simulate electricity price a simple model is adopted, although a rich literature about this topic is available, complex and reliable models require a deep electrical market analysis and must be fed with a big amount of exogenous data. A refined implementation of an electricity price generator is therefore out of the scopes of this thesis, the argument could be certainly among the ones to consider for a future development of this work.

2.4 Streamflow identification and synthetic generation

A good prediction and simulation of daily synthetic streamflow data has a significant importance in water resource management, the simulation of real-world alike streamflows is necessary to analyse the robustness of the optimization models implemented in different scenarios. Moreover, the data generated incorporate a certain degree of uncertainty that is then transferred by the optimization model to the final value of the objective function.

It is important to note that, for the scopes of this Thesis, the focus is on *synthetic generation of streamflow* and *not* in their prediction, therefore in the following state-of-art-review is focused on statistical analysis and fitting methods rather than forecasting techniques. For similar reasons given in 2.3, this consideration is valid also for electricity price and allows to consider more simple and analytical models. The assumption made is mandatory to prevent the work of this thesis from going out of scope, complex models are often multiple input systems that include terrain characterization, atmospheric and oceanic patterns, an example is [7] and can be seen how the work is out of vision for the purposes of this thesis.

In his report [8], D.J. Holtschlag identifies two main branches of methodologies for daily streamflow identification, classified as follow:

1. Statistical models for estimating daily Streamflow
   (a) Trend and seasonal components identification
   (b) Ordinary-least-squares regression

2. Stochastic models
   (a) Autoregressive moving-average model
   (b) Transfer function-noise model
   (c) Composite model

According to [8], this models are parsimonious for a statistical point of view, it means that the number of parameters required for the estimation is relatively small, given the accuracy of the estimates.

Trend and seasonal components identification

Trend and seasonality are the main components of a time-series, as shown by the additive model [5] in section 2.4.

$$y(t) = L(t) + T(t) + S(t) + \varepsilon(t) \tag{2.4}$$

Where $L(t)$ is the level of the time series, $T(t)$ is the deterministic trend component of $y(t)$, $S(t)$ is the seasonal component and $\varepsilon(t)$ is the model-error component. In case of trend and seasonal component decomposition model the erratic component is generally considered to be normally distributed.

In time-series identification, the trend and seasonal components in equation 2.4 are generally approximated by the use of a deterministic function of time and then subtracted from the streamflow data before the analysis of the erratic component. This procedure is described by D.J. Holtschlag in [8] and applied for streamflow analysis, but is also commonly used in general statistics methods, for example in [9]. The technique is also, in part, used for the time history identification in this Thesis.

D.J. Holtschlag in [8], analysed the time series of 20 streamflows related to different reservoirs is U.S.A. and pointed out that, although no evident trend can be often spotted, a strong yearly seasonal component can be always identified. Due to variability in seasonal components along the year, they have been identified by means of a moving average along the time history. The same approach has been applied in this Thesis.

2.4.1 Ordinary-least squares regression

Ordinary-Least-Square Regression Models (OLSR) are extensively used in streamflow identification due to their easiness of application. In the investigation about Cost Effectiveness of the Stream-Gaging Program in

Maine, [28] shown that this method could bring to inaccuracies of the resulting streamflow estimates. As also stated in [8], the full equation for OLSR tends to include explanatory variables that had little computed statistical significance and the to break the principle of parsimony.

2.4.2 Autoregressive moving-average

Autoregressive moving-average (ARMA), autoregressive integrated moving-average (ARIMA) and seasonal autoregressive moving average (SARMA) models are commonly used in streamflow identification thanks to their capability to describe the phenomenon and their relatively widespread diffusion also in other fields.

ARMA models joined with trend and seasonality identification are used in this thesis, the combination of these two techniques is proved to be effective by a widespread literature about streamflow identification and synthetic flow generation.

In the study conduced in [8], about 20% of streamflow analysed presented an Auto Regressive (AR) part only while the remaining 80% were described by autoregressive moving-average equations.

Mondal and Uddin Chowdhury in [21], introduced a time series model to be used in river hydrology for synthetic generation, a model called "deseasonalized Autoregressive Moving Average" (deseasonalized ARMA) is introduced for the generation of decadal (10-day) flows. The model proposed has strongly influenced the method used for the identification and generation of streamflow in this Thesis. Differently from the Thesis work, Mondal and Uddin Chowdhury used a Fourier analysis to remove the seasonal component, then the an ARMA(1,3) model is fitted to match with historical data. The validation synthetic flow generated demonstrated that the deseasonalized ARMA model was able to capture the decadal variability of the Brahmaputra River taken as case study.

A parameter estimation of an ARMA model for river flow forecasting with the objective of managing long-term reservoir operation is proposed by Mohammadi, R. Eslami, and Kahawita in [20]. An important consideration made in the paper is that the characteristics of a time series may vary within a decade, so it’s important to maintain the model updated and historical data should not be taken too far in the past.

A comparison between ARMA and ARIMA model is performed in [14], the statistical index mean absolute percentage error (MAPE), root mean squared error (RMSE) and nash efficiency (NE) were calculated and no substantial difference between the two models emerged.

2.4.3 Transfer function noise model

Transfer function model is a statistical model describing the relationship between an output variable and one or more input variables. Many applications can be found both in hydrology and economics models. Due to their simplicity, discrete-time linear models are the most widespread in literature, an exhaustive dissertation about this family of models is reported in [22] notes.

In [2] the original Box and Jenkins formulation of the noise transfer function has been used to estimate parameters for streamflow time series. The model provided satisfactory results in short term and is defined as particularly interesting for the small number of parameters needed to be estimated and the possibility to implement it even if a short period of records is available.

Single and multiple-input transfer function models and their applications in modelling streamflows systems is reviewed also in [32]. A multiple input model has been introduced, snow-melt and effective rain are considered the two most important input and have been modelled too. A practical application to the Saugeen River system in Canada has been taken as case study, the method implemented showed promising results and a demonstrated to be a quite straightforward approach for multiple-input models.

The transfer function noise model is mainly applied in literature as a simple method for modelling multiple-input systems in a short term forecasting window. Although providing promising results, the model field of application in quite opposite to the interests of the thesis, where a long term synthetic streamflow must be generated starting from a single input (streamflows time history). Moreover the focus of this methodology is about forecasting a time series rather than generating a synthetic one that could represent a realistic real-world behaviour.

2.5 Currently adopted base model

This thesis takes as starting point an already existing model, developed in Politecnico di Milano. The model is based on a matricial representation of the system and uses as optimization technique a linear programming approach. The optimizer structure is intrinsically linear, although different techniques to deal also with non linear constraints and objective function have been introduced. The programming language used is Matlab.

A description of the actual implemented model is necessary to understand the work done in this thesis, mainly for what concerns chapter 4. In the following section an overview of the base model is given, the main concept about the working principle, model limitations and performances.

2.5.1 Working principle

The structure of the model is capable to describe and solve a generic water system by changing its inputs but, for the sake of simplicity, all the practical examples are referred to the case study 1.1.3 taken in consideration.

In order to deal with the multiple non-linear constraints and non-linear objective function (sections 1.1.2 and 1.1.3) a two stage optimization has been introduced. In the first stage only linear constraints are applied and the objective function is linearised around the mean water level in the reservoirs. The second stage uses the results coming from the first optimization as base point to linearise all functions around the first solution, moreover to deal also with strongly non-linear constraints (such as the number of turbine's activations or the minimum turbine power) some control variables are arbitrarily imposed before starting the second optimization. The diagram in figure 2.1 represents the schematized working principle of the algorithm.

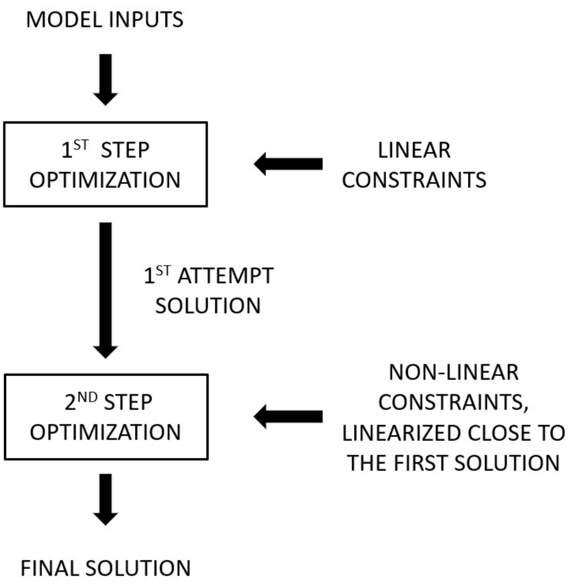


Figure 2.1: Schematized optimizer's working diagram

Model inputs

To describe how the algorithm is structured the main inputs are here introduced, considering the system in the case study with: $I\_reservoirs = 8$ the number of reservoirs, $N\_variables$ the number of variables and T_hours the number of optimization hours (I, N *and* T are used for simplicity).

(i) *price_vect* $[T \times 1]$: is the vector containing all the future hourly electricity price, in €/kWh

(ii) $streamflow\_matrix$ $[T \times I]$: is a matrix containing the predicted streamflows in $\mathrm{m^3\,h^{-1}}$, in columns for each reservoir

(iii) *structure_matrix* $[I \times N]$: is a matrix defined as: $M_{i,x} = -1$ if control variable x takes water from reservoir i, $M_{i,x} = +1$ if control variable x discharge water from reservoir i, zero otherwise. Note that in case a control variable takes water from two or more reservoirs the value -1 is split in fractions, an example in 1.5 is P_4 that takes water from Q_4 and Q_5. To better understand, the matrix for case study is drawn:

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}	X_{19}
Q_1	0	0	0	0	0	-1	1	0	0	0	0	-1	0	1	0	0	0	1	0
Q_2	-1	-1	-1	-1	-1	1	0	0	0	0	1	1	-1	0	0	0	1	0	0
Q_3	0	0	0	0	0	0	-0.3	0	0	0	0	0	0	-1	0	0	0	0	0
Q_4	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	-1	0	0	0	0
Q_5	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	-1	0	0	0
Q_6	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	-1	0	0
Q_7	0	0	0	0	0	0	-0.7	1	0	1	0	0	0	0	1	0	0	-1	1
Q_8	0	0	0	0	0	0	0	0	1	-1	0	0	0	0	0	1	0	0	-1

(iv) *power_vector* $[N \times 1]$: $= +1$ if the variable produces power(is a turbine) or $= 0$ if the variable doesn't produce power(is an emergency bypass). In the case study variables X_{12} to X_{19} are bypass, while the others are turbines.

(v) *concentration_time_vector* $[N \times 1]$: contains for each variable the concentration time in h

(vi) *max/min_flow_matrix* $[T \times N]$: contains the maximum/minimum flows that each variable can have at each hour, in m^3

(vii) *max/min_power_matrix* $[T \times N]$: contains the maximum/minimum power that each variable can have at each hour, in kW

(viii) *max/min_volume_matrix* $[T \times I]$: contains the maximum/minimum volumes in m^3 that each reservoir can have in each hour

(ix) *max_height_rate_matrix* $[\#_{constraints} \times 3]$: contains the maximum height rate of change in m on which a reservoirs can undergoes in a certain period. First column address to the reservoir index, second column the hours in which the constraint is active and the third column the maximum height change during the period

(x) *initial/final_volume_vector* $[I \times 1]$: contains the initial/target volume in m^3 that each reservoir must have

(xi) *volume/height_vector* $[I \times 1]$: contains the description of each reservoir's shape, basically describe the curve $H_i(t) = f_i(Q_i(t))$ (with ref to 1.1.2).

In the list above only the essential and most relevant inputs have been described, but there are also other inputs to take account of: combined min/max flow limits, combined min/max power limits, maximum height rate of change and maximum number of activations plus other service input values.

Model structure

In order to relate the different flows from the control variables X_v with the volume of water in each reservoir Q_i along the time history, a way to represent the water balance equation 1.11 is needed. Moreover, since the model must be optimized through a linear programmer, a matricial representation is necessary. A generic linear programming algorithm finds the minimum of a problem specified by:

$$\min_x f^T \cdot x \text{ such that } \begin{cases} A_{max} \cdot x \leq b_{max} \\ A_{eq} \cdot x = b_{eq} \\ lb \leq x \leq ub \end{cases} \tag{2.5}$$

Thus matrix A, when multiplied by control variables x should provide a vector containing the volumes of all reservoirs at any hour. To do this, matrix A is generated according to the following formulation:

$$A_{m,k} = subdiagonal\_matrix(CT(j)) \cdot structure\_matrix_{i,j}$$

$$\text{with: } (i-1)T \leq m \leq i \cdot T$$

$$\text{with: } (j-1)T \leq k \leq j \cdot T$$

Where the variable $CT(j) = concentration\_time\_vector(j)$ and variable $subdiagonal\_matrix(CT(j))$ is a subdiagonal matrix of ones shifted from the diagonal by a number of cells defined by the concentration time. Thus the final matrix dimensions is $[I_{reservoirs} \cdot T_{hours} \times X_{variables} \cdot T_{hours}]$ A visual example of the matrix composition is shown in figure 2.2.

The state vector x represents instead all the control actions $X_v(t)$ described at any time t, the dimension of the vector is thus $[X_{variables} \cdot T_{hours} \times 1]$.

It is fundamental to focus on the meaning of the multiplication $b = A \cdot x$, the resulting vector b, of dimensions $[I_{reservoirs} \cdot T_{hours} \times 1]$, contains the marginal water volumes increments/decrements for each reservoir for all the hours considered. In practice, the multiplication results is an expression of how the control variables marginally affect the volume of water inside the reservoirs.

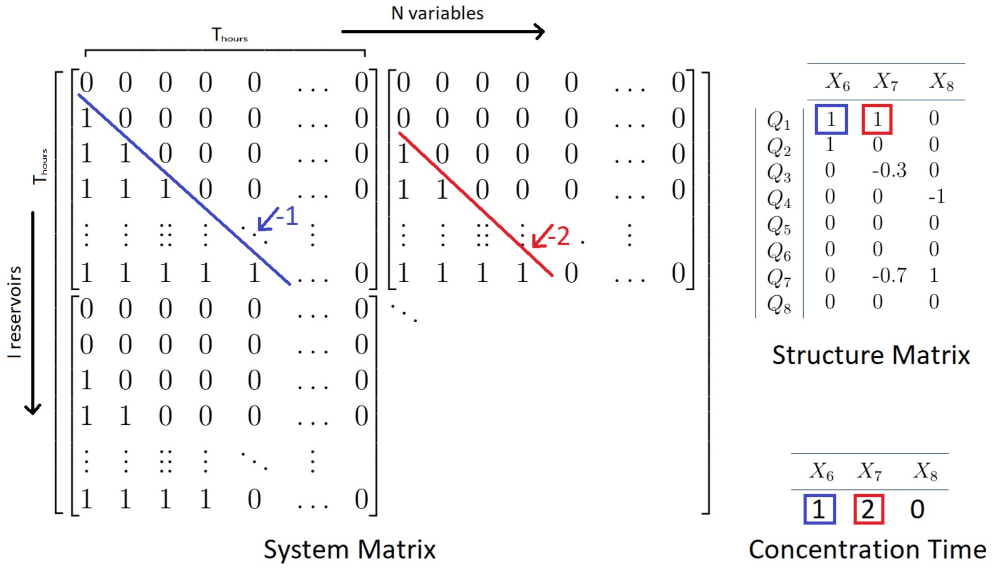


Figure 2.2: Composition of the system matrix

First optimization stage: linear constraints

Once the water increment/decrement in reservoirs is defined as consequence of the decisional variables, its limits must be assessed in order to respect the constraints. In particular the constraints that affects b_{max} are eqs. (1.2) and (1.3), which corresponds to the model inputs items (viii) and (ix).

For what concerns the maximum volume, b_{max} is calculated as:

$$b_{max}(j) = max\_volume\_matrix(:,r) - initial\_volume\_vector(r) - \\ - Cumulatives[streamflow\_matrix(:,r)]$$

$$\text{with: } (T_{hours}(r-1)) \leq j \leq (T_{hours}r) \;\; ;\text{r res: } 1 \leq r \leq I_{res}$$

For the constraint of minimum volume the same procedure is followed but sign of A and b_{min} are inverted.

Other constraints to be introduced are the maximum and minimum flow that each control variable must respect at any hour, in the first step only capping on the flow is considered disregarding the turbine power limits. This is due to the fact being $ken(H_{v,t})$ function of the reservoir height $H_i(t)$, it's implicitly affected by the decision variables $X_v(t)$; even if a linear relation between ken and $X_v(t)$ exists the final power expressed as $P_v(t) = X_v(t) * ken(H_{v,t})$ would be a quadratic function of $X_v(t)$ and no quadratic functions are allowed by the linear programming algorithm.

The flow limits are easily expressed by the direct relations:

$$X_v(t) \leq max\_flow\_matrix(t, v) \qquad \text{maximum flow}$$
$$X_v(t) \geq min\_flow\_matrix(t, v) \qquad \text{minimum flow}$$

Equality constraint is also introduced in the first stage and represents the target volume of each reservoir (reference to constraint 1.9). The problem cannot be open ended: if no final volume constraint are imposed the optimization converge always to a solution in which the final reservoir volume is equal to its minimum since this represents always an optimal choice to maximize profit in the time window considered. In the algorithm this constraint introduced by the relation $A_{eq} \cdot x = b_{eq}$, where b_{eq} is the final volume of the reservoir, while A_{eq} is a matrix containing all the row of A corresponding to T_{end} for each single reservoir. In mathematical formulation:

$$A_{eq}(r, :) = A(r * T_{hours}, :)$$

The result of $A_{eq} \cdot x$ is a vector $[I_{reservoirs} \times 1]$ containing the volume at T_{end} for all the reservoirs.

Maximum height rate of change in a reservoir can be introduced according to the following expression:

$$[A(T, :) - A(T_s, :)] * x \leq \sum_{t=T_{start}}^{T_{end}} streamflows(t, r) + max\_rate(r, 3)$$

Where T_{start} and T_{end} are the initial and final time of the constraint, to add this limitation to the linear programmer is enough adjoining to A_{max} and b_{max} the elements on the left and on the right of the equation, respectively. As can be noticed the constraint is expressed in reservoir height, to express it in volume an interpolation of the *volume/height_vector* is done to extract the correspondent volume of water.

In some cases the maximum rate of change is function of reservoir height,often, due to the conical shape of the reservoir this equation is non-linear and creates the necessity of an approximation; moreover the constraint is expressed through a step function. The first approximation could be treated in the second stage of the optimization, while, in the case of a step function the strategy adopted is to limit the maximum rate of change according to its lowest value, in a prudential way.

The objective function 1.10 is rewritten considering energetic coefficient constant and neglecting the effects of power load on efficiency, this simplification is valid since the variation in height of the free water level is negligible with respect to the total water head.

$$L(X(t), Q(t), P(t)) = \sum_{t=1}^{T} Price(t) \cdot \sum_{v=1}^{N_v} [X_v(t) * ken(H_{v,mean})]$$

Where $ken(H_{v,mean})$ is the energetic coefficient, taken as a constant calculated over the mean height of the reservoirs. This formulation allows to write the function as required from the linear programming algorithm: $x \cdot f^T$, where $f^T = Price(t) \cdot ken(H_{v,mean})$.

Second optimization stage: introducing non-linear constraints

In the second optimization step the matrices A, A_{eq} and the vectors $b_{max}, b_{eq}, lb \ and \ ub$ are adjoined with secondary constraints that, due to their non-linear dependency from x couldn't be considered. A distinction between two main typologies of non-linearity can be made:

(i) Soft non-linear: are constraints that can be linearised after considering an approximation in the dynamic of the system

(ii) Hard non-linear: are constraints that include boolean or step functions that cannot be linearised

A soft non-linear constraint, neglected in the first step , is the limit on the minimum and maximum power that a turbine can produce. Since both flow and power limit the same variable, only the most restrictive constraints is taken into consideration. In the case of maximum capping on a generic variable $X_v(t)$ the two limits are expressed as follows:

$$X_v(t) \leq max\_flow\_matrix(t, v) \qquad \text{max flow}$$
$$X_v(t) * ken(H_v(t)) * \eta_v(X_v(t)) \leq max\_pw\_matrix(t, v) \qquad \text{max power}$$

The effects of efficiency are neglected, $\eta_v(X_v(t)) = 1$ and the energetic coefficient is approximated to a constant value calculated over the reservoir height coming from the previous optimization, $ken(H_{1st\,step}(t))$. Thanks to this simplification the maximum power constraint is translated into a maximum flow constraint:

$$X_{v,max,power}(t) = max\_power\_matrix(t, v)/ken(H_{1st\,step}(t))$$

Finally, to set the upper bound the smaller values is taken:

$$ub(v,t) = \min(X_{v,max,flow}(t), X_{v,max,power}(t))$$

The simplification applied hold if the height of reservoirs undergoes little variations from the first step to the second. Similar procedure is applied in case of a minimum constraint. In the case of bypass control variables, $X_{12,13,...,19}$ only the maximum flow represents a limit since the energetic coefficient associated is null. It's important to notice that at this level no minimum idle power has been introduced yet, the turbine power can varies continuously from a minimum value(set equal 0) to its maximum rather than assume the power interval: $[0; P_{min} \div P_{max}]$.

In order to include this non-linearisable behaviour all the variables that present a power value between 0 and P_{min} after the first optimization step are arbitrarily constrained to at 0 or P_{min} according to the following rule:

$$X_v(\bar{t}) = 0 \qquad \text{if } P_{v,1st\,step} \leq 0.5 * P_{v,min}(\bar{t}) \tag{2.6}$$

$$X_v(\bar{t}) = P_{v,min} \qquad \text{if } P_{v,1st\,step} \geq 0.5 * P_{v,min}(\bar{t}) \tag{2.7}$$

Where $P_{v,1st\,step}$ is the power generated by the flow $X_v(\bar{t})$ at step 1.

Another non-linearisable constraint is the limit over the turbine number of activations in a given period. No variable in the model controls the effective state switch (powering on/off) of the turbine but it can be only deduced from the control vector x once optimization has been completed. Since it's not possible with a linear programmer to deal with this constraint, between the first optimization step and the second, arbitrary decisions are taken on the control variables ($X_v(\bar{t})$) that don't respect the limit. The logic with these decisions are takes is the following:

(i) The turbine state change is identified over the control variables obtained from the first step $X_{v,1st\,step}$.

(ii) In case of a "short shut down", defined as a turbine shut down which is shorter or equal than one hour, the turbine is forced to produce at least the minimum power by imposing $P_{v,2nd\,step}(\bar{t}) \geq Pmin_v(\bar{t})$. This strategy reduces the number of activations, but if the constraint is still not respected the further steps are necessary.

(iii) In case the power plant in which the variable $X_v(overlinet)$ exceeds the number of power on contains more than one turbine, part of the daily power production is transferred to the other variables in order to respect the constraints. This allows the respect the activations limits without changing the optimal solution.

(iv) If no turbines are available to share the power load, the variable $X_v(\bar{t})$ is forced to keep at least the minimum power at time $\bar{t}$ in order to reduce the number of activations. The constraint of minimum power is introduced with the criterion of shortest down-time first in order to reduce at maximum the perturbation around the optimal solution.

For what concerns the value function, in the second optimization step is expressed as follows:

$$L(X(t), Q(t), P(t)) = \sum_{t=1}^{T} Price(t) \cdot \sum_{v=1}^{N_v} [X_v(t) * ken(H_{1st\,step}(t))]$$

Where $ken(H_{1st\,step}(t))$ has been introduced before and the turbine efficiency, function of $X_v(t)$ is still neglected and fixed constant to 1. Once the second step of optimization is concluded, the optimal solution x is used to calculate all the actual values of ken and η and the different non-linear variables are updated with their correct value.

2.5.2 Performances

The results obtained with the current model reflect a behaviour which is similar to the actual reservoir operation strategies applied in the real case. Moreover, from a mathematical point of view, linear programming is a tool that allows to find the global optimum of a problem, if it exist, without the requirement of an initial guess. Even though the introduction of a two step approach and the approximation of non-linear function and constraints doesn't allow the system to be optimized in its overall completeness, the sub-optimal solution is near to the global optimum and can be considered a satisfactory trade-off between model complexity and optimal result.

Better results could be obtained with a time discretization refinement but, due to the fact that reservoir operations are managed hourly, no practical advantages are achieved.

2.5.3 Main limitations

The limitations of this algorithm are related mainly to two points of view: necessity to adopt non-linear functions in the model and computational costs.

Non-linear dynamics

The approximations related to *ken* coefficient and turbine efficiency mainly act as a bias on the estimation of the optimal parameters, this issue could be easily solved adding more optimization stages that iteratively refine the optimal solution. But due to the intrinsic uncertainty of the input models, the computational costs required and the small marginal gain obtained the optimization steps are limited to the necessary number of two.

Moreover some step constraints, such as the maximum height change rate, approximated to the minimum step value, could result in a too prudential limitation that biases in an excessive way the operation of the system.

The hard non-linear constraints are the ones that generate major issues, they have been introduced adding arbitrary constraints between the two steps; if the problem is ill-defined and near an unfeasibility region, the addition of these constraints could result in a non feasible solution during the second optimization round. This issue often arises during the initial hours of the optimization, when the system is near the initial boundary condition and cannot easily satisfies the constraints since in these hours the already sold power production is imposed.

Another limitation of this approach is that when the constraints are imposed among the two steps, all the variables in the neighbourhood of the constraint are free to change and no control over new potential un-respected non-linear constraint is applied.

For example, due to the limit over activations a variable is set from off-state to P_{min} at time $\bar{t}$, the optimization process in the second step could shift the deactivation of the turbine at the instant $\bar{t}+1$.

As already stated, despite the model limitation, a satisfactory solution is found for the case study taken in consideration. But for particular cases, where non linearities play a more fundamental role in the dynamic of the system, the approximation made could not hold and a different strategy should be applied. A possible approach to these kind of problem could be the use of the Dynamic Programming implemented in chapter 6.

Computational costs

The main motivation that brought this thesis work to find a modified approach to the problem, with respect to the tool already developed, is the huge amount of computational resources needed to solve the optimal problem. Even if the Linear Programming is known to solve a linear problem with algorithms that are among the most efficient, the problem dimensions are huge. The model requires to solve a number of optimal control variables which is $N_{variables} \cdot T_{hours}$. A typical long term optimization is run over minimum 1 year, to respect the annual cyclicity, for a total hours amount of $T_{hours} = 365 * 24 = 8760$. Thus in the case study considered, where 19 variables are used, the overall number of control variables is $\#CV = 19 \cdot 8760 = 166440$, which is a considerable amount.

Moreover, the matrix needed to characterize the model would easily saturate RAM memory since problem size expands quadratically in with time. This is due to the dimensions of matrix A necessary to describe the system, as seen in 2.5.1, it has a size of $[I_{reservoirs} \cdot T_{hours} \times X_{variables} \cdot T_{hours}]$; considering a long term optimization on the case study problem($T_{hours} = 8760$, $I_{reservoirs} = 8$, $X_{variables} = 19$) the memory needed to store the matrix in RAM is:

$$RAM_{usage} = I_{reservoirs} * X_{variables} * T_{hours} * \frac{8byte}{1024^3} * 2 \text{ Matrixes} = 174Gb$$

Due to the large amount of time and memory requirements that makes the optimization almost impossible for a generic user, an alternative approach to the problem has been developed and successfully applied in chapter 4.

Chapter 3

Generation of synthetic input data

In order to test the performances of the models implemented in this thesis, historical data are not enough and the generation of synthetic time series is needed. A high number of long term simulated inputs is used in chapter 4 to conduce a Montecarlo analysis and to asses the behaviour of the system. Moreover synthetic data are also used in the short term optimization (chapter 5) to identify the optimal operating conditions.

The two data-set taken in input by the model, and that are needed to be simulated, are the streamflows into each reservoir and the electricity price. Realistic time series are needed for a correct evaluation of results, in particular it is important to generate a process capable to reproduce both the variance of historical data and, for what concerns the streamflows, a realistic pattern.

3.1 Streamflow time series identification and generation

As introduced in section 2.3, it is fundamental to generate streamflow time-series capable to characterize the stochastic process since it's necessary to use realistic streamflows in order to correctly evaluate the robustness of the algorithm. As emerged from preliminary studies, the streamflow data strongly affect the results of the solution and, for problems near to the unfeasibility region there is a strong bias on the estimation of the optimal solution. The only identification of variance and the subsequential generation of noisy data is thus not enough, even if it could be an acceptable method for a Montecarlo approach, when applied to the

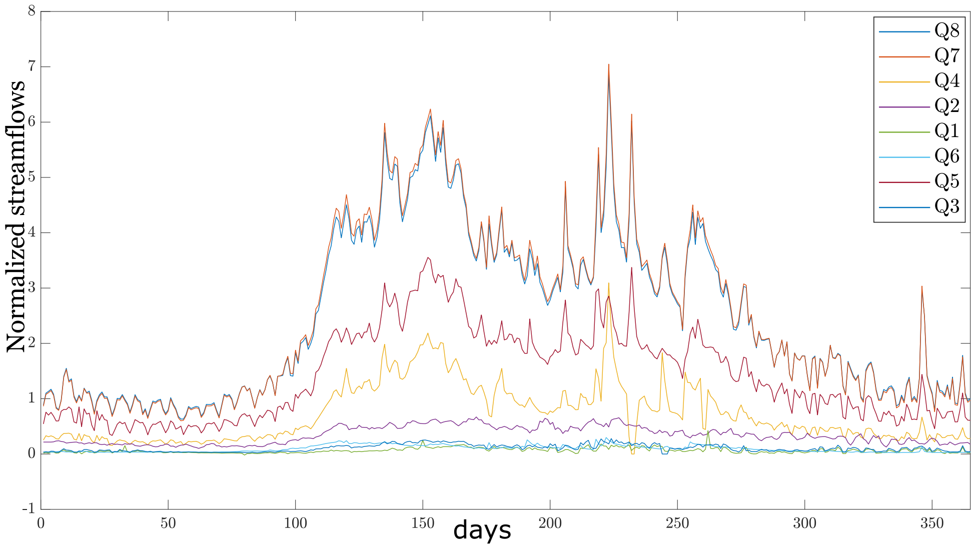


Figure 3.1: Historical streamflows for reservoirs in the case study

model, the dynamic of the system could bring to erroneous results.

All the data used come from real flow measurement and are specifically referred to the case study taken in consideration. The values, in m^3/h, are provided daily and represent the *net streamflow* for each reservoir, for net streamflow is intended the water flow that enter in a reservoir minus the spillover that naturally occurs and minus the already considered streamflows of the upstream reservoirs.

In figure 3.1 time series referred to year 2018 are exhibited, the streamflows refer to all the 8 reservoirs related to the case study taken in consideration, figure 1.5. For disclosure reasons the data have been normalized, anyway this doesn't affects the considerations made.

In the following sections the identification methodology is proposed, for the sake of simplicity the applied procedure is described taking into consideration only a single basin (Q1, in figure 3.2), then the results for all the reservoirs analysed are reported. The methodology adopted is similar to the one proposed in [21, postnote], the theory on which the procedure relies on is the same but some modifications have been adopted in order to better capture the noise modulation along the year.

The main steps are: 1. Trend and seasonality removal 2. Fitting ARMA model 3. Noise envelope and modulation 4. Synthetic flow generation According to the analysis made in 2.3, ARMA models joined with trend and seasonality identification is used in the thesis since this technique is proved to be effective for streamflow identification and synthetic flow generation; with respect to SARMA modelling is capable to better identify noise modulation during the year.

3.1.1 Trend and seasonality removal

Before fitting an ARMA time series model it is needed to clean the historical data from the trend components and from its periodicity (adopting a more specific terminology, periodic component of streamflows will be called also seasonal component).

To remove the trend component, a moving average centred on a window of time length n is performed on the time series:

$$\begin{aligned}\overline{p}_{\text{SM}} &= \frac{p_M + p_{M-1} + \cdots + p_{M-(n-1)}}{n} \\ &= \frac{1}{n}\sum_{i=0}^{n-1} p_{M-i}\end{aligned}$$

The trend component then is subtracted from the data:

$$S_{dt}(t) = S_h(t) - T(t)$$

Where $T(t) = \overline{p}_{\text{SM}}$ is the trend component, $S_{dt}(t)$ is the detrended time series and $S_h(t)$ is the original streamflow data.

In order to remove the seasonal component the period of seasonality is identified by applying the autocorrelation function 3.1 to $S_dt(t)$. Autocorrelation is the correlation of a signal with a delayed copy of itself, it is used as a tool for finding repeating patterns, such as the periodicity in a signal when it's obscured by noise. If a signal contains a periodicity, its autocorrelation is periodic with a time-period equal to the original signal.

$$r_k = \frac{\sum_{i=1}^{N-k}(p_i - \bar{p})(p_{i+k} - \bar{p})}{\sum_{i=1}^{N}(p_i - \bar{p})^2} \tag{3.1}$$

Once the seasonal time-span is evaluated, seasonality coefficients are identified by applying a stable seasonal filter to the detrended series. First the seasonality period is selected (365 days), then all the indices corresponding to each period are stored and finally the average of data corresponding the these indices is computed. The same process is repeated to all the 365 days.

With the same procedure used for trend component, seasonality is removed from $S_{dt}(t)$:

$$n(t) = S_{dt}(t) - S(t)$$

Where $n(t)$ is the detrended and deseasonalized time series and $S(t)$ the seasonal component. From now on, we will refer to the term $n(t)$ as "noise" or "erratic component".

3.1.2 Fitting ARMA model

The erratic part is fitted with an autoregressive moving average model. ARMA(p,q) is one of the most common techniques for time series analysis and also among the most used for streamflow identification. The considered model is in the form:

$$X_t = \varphi_0 + \varphi_1 X_{t-1} + \cdots + \varphi_p X_{t-p} + Z_t - \vartheta_1 Z_{t-1} - \cdots - \vartheta_q Z_{t-q} \quad (3.2)$$

With p is the order of the autoregressive component, q is the order of the moving average component, $\varphi_1, \ldots, \varphi_p$ are the estimated autoregressive parameters, φ_0 is the constant offset, $\vartheta_1, \ldots, \vartheta_q$ are the moving average parameters and Z_t is the actual erratic term.

The model selection process consists two stages: i Model identification (orders p and q are determined) ii Model parameters estimation

Model identification

To determine the moving average order q and the autoregressive order p is necessary to calculate the autocorrelation function and the partial autocorrelation function for the noise, the plot of these two functions provides information about the possible model orders. An exhaustive dissertation about ARMA model identification can be found at `https://people.duke.edu/~rnau/411arim3.htm`,[23].

Model parameters estimation

The bias term φ_0 is known and considered null since the noise has null average(due to the fact that trend has been removed).

For the estimation of the other parameters the maximum likelihood method has been used due to its best performances in comparison to other methods (see [29]). The parameters that maximize the likelihood value are computed in MATLAB environment using "arima" function.

3.1.3 Noise envelope and modulation

The erratic component in streamflow usually exhibits a periodical behaviour in the variance. Autoregressive moving average model is not capable to describe periodicity in the variance. The envelope of noise has been calculated and then the synthetic signal is modulated by the envelope, in this way original noise periodicity is restored.

The procedure applied is more advanced, with respect to the one proposed by Mondal and Uddin Chowdhury in [21], where the noise signal is simply divided by the respective seasonal standard deviation.

The envelope of the noise is obtained by computing its analytic signal with the Hilbert transform:

$$\begin{aligned} &x \Rightarrow \hat{x} \\ &\hat{z} = \begin{cases} \hat{x} \times 2 & \text{if } x > 0 \\ 0 & \text{if } x < 0 \end{cases} \end{aligned} \tag{3.3}$$

Where $\hat{x}$ represents the Fourier transform of x.

Hilbert transform allows the description of noise through its analytic signal, which is basically the representation of the original signal in polar coordinates:

$$S_a(t) = S_m(t)e^{j\varphi(t)}$$

Where $S_m(t)$ is the instantaneous amplitude or envelope of the signal, while $\varphi(t)$ is the instantaneous phase, therefore by taking the module of the analytic signal its envelope can be calculated.

3.1.4 Synthetic streamflow generation

For the final generation of artificial streamflow the opposite path is taken, first of all a synthetic noise is generated thanks to the ARMA model previously fitted, then the noise is modulated over the envelope, finally seasonality and trend components are added. A normalization over the maximum value of the envelope was necessary to modulate noise. The described procedure can be expressed by the following equations:

$$\begin{aligned} &N_s(t) \Leftarrow ARMA(\hat{p}, \hat{q}) \\ &N_m(t) = N_s(t) * Env(t)/max(Env(t)) \\ &S_s(t) = T(t) + S(t) + N_m(t) \end{aligned}$$

Where $N_s(t)$ is the noise generated from the ARMA model, $N_m(t)$ is the modulated noise, $T(t)$ and $S(t)$ are trend and noise component and $S_s(t)$ is the comprehensive synthetic streamflow generated.

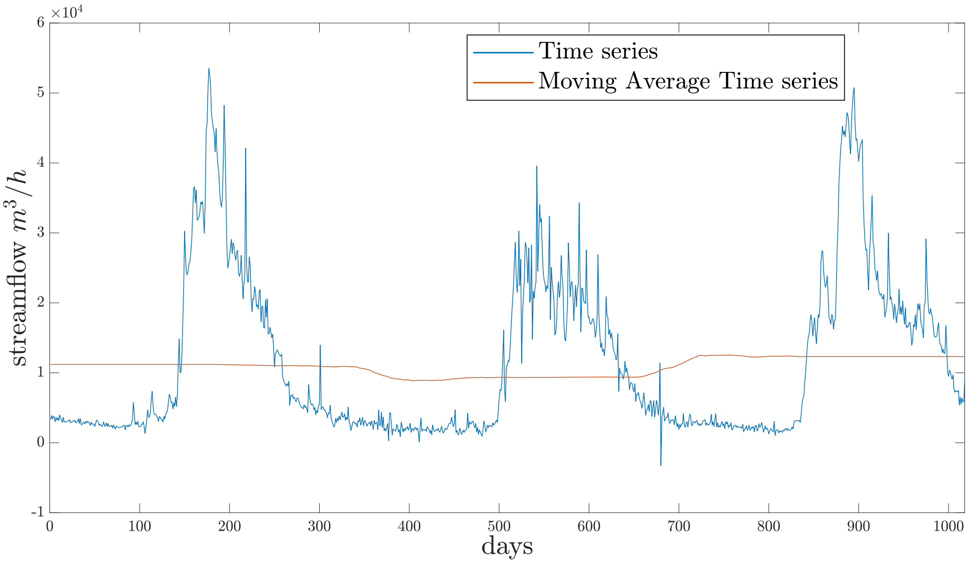


Figure 3.2: Historical streamflow for Q1 reservoir

3.1.5 Application of the model

As can be seen from figure 3.2, a trend component and a seasonal component is contained in the time series analysed. In the data are present some negative spikes, clearly due to errors in measurement, where a negative value is found in historical data is replaced by the mean value of the streamflows in its neighbourhood. By performing a moving average on the time series provided, the mean component of the flow is extracted (the red line in figure 3.2). Even though it shows no particular trend (on three years time-span there is no significant increment or decrement in the water flows) the mean component must be removed to have an unbiased estimation of seasonal component. It is important to notice that the window time-length chosen to perform the moving average lasts 365 days, to avoid removing variance intra-day components that must be modelled later in the ARMA series.

The trend component is removed and the autocorrelation function has been applied as shown if figure 3.3.

Looking at 3.2, it's evident that a one-year seasonality is contained in the time series, a mathematical evidence is provided by the autocorrelation. From figure 3.3, can be clearly seen that the peaks are at 0, 359 and 710 days of lag, this confirms that a seasonality of about one year is present.

Seasonality is removed, figure 3.4 shows the whole process of detrending

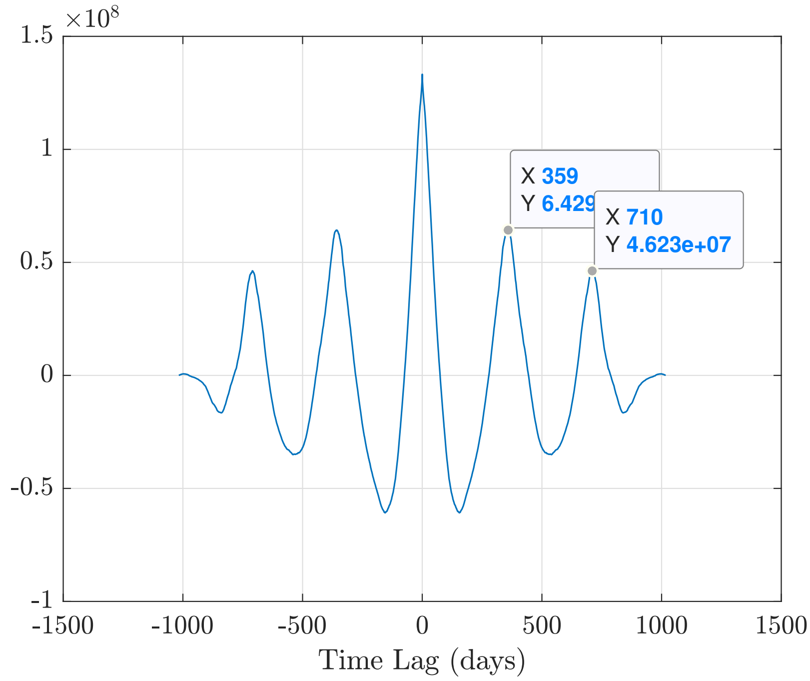


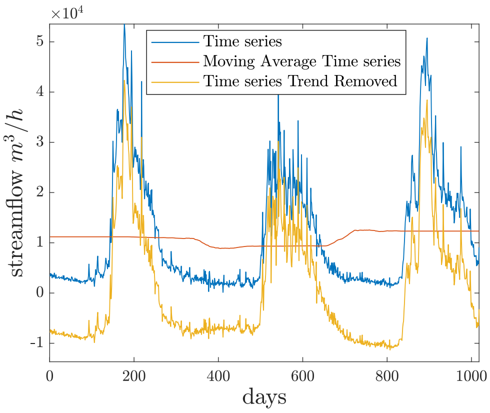
Figure 3.3: Autocorrelation function on Q1 reservoir streamflow

and deseasonalization for the considered the time series.

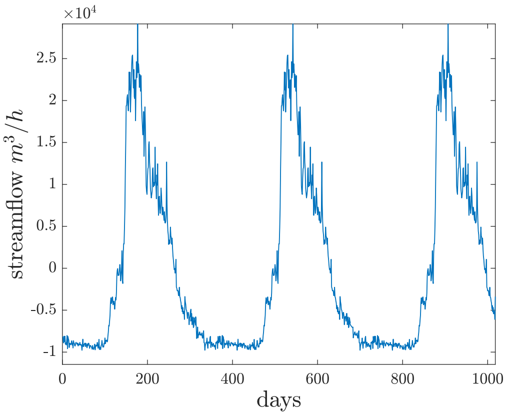
As can be seen the remaining noise 3.4c has a zero mean but the erratic component exhibits a periodical behaviour in the variance. Seasonality 3.4c instead, it's exactly the same over the three years since the periodic components estimated have no variance.

Looking at the autocorrelation function (ACF), 3.5a and at the partial-autocorrelation function (PACF), 3.5b it's possible to see how autocorrelations are significant for a large number of lags, but the autocorrelation at lag 2 and above are mainly due to the propagation of the autocorrelation at lag 1. This can be seen from the PACF plot where PA at lag 1 is significantly higher than the others, meaning that all the higher-order autocorrelations are effectively explained by the lag-1 autocorrelation, [23].

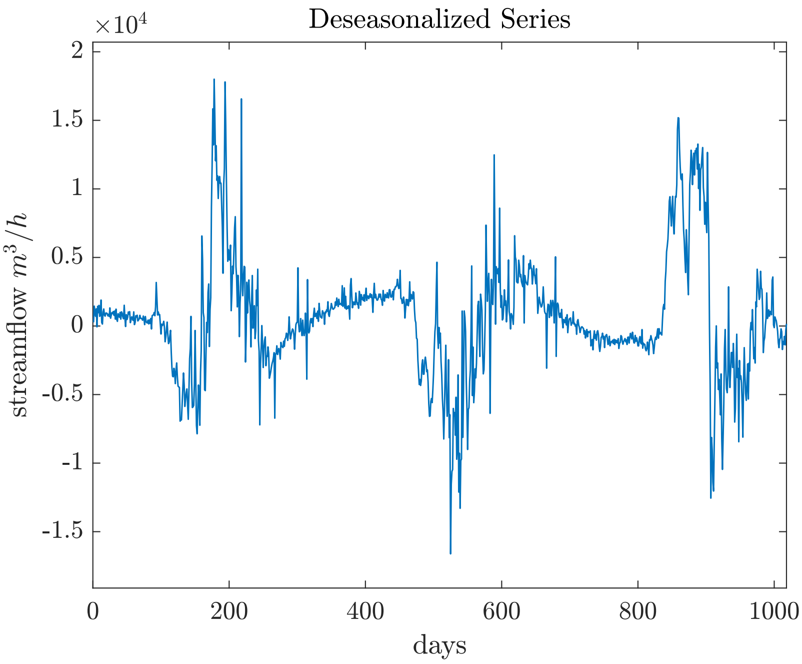
According to the theory, partial autocorrelation at lag k is an estimation of the autoregressive coefficient φ_p in equation 3.2, thus by inspecting the PACF it is possible to determine which and how many autoregressive terms are needed. In the case taken in consideration the PACF clearly cuts-off at lag $k = 2$, so the order of the AR part is 2. Looking at the ACF, due to slow decay of autocorrelations value, [23] suggests the fitting with a model containing a MA order of 1.



(a) time series, trend removed

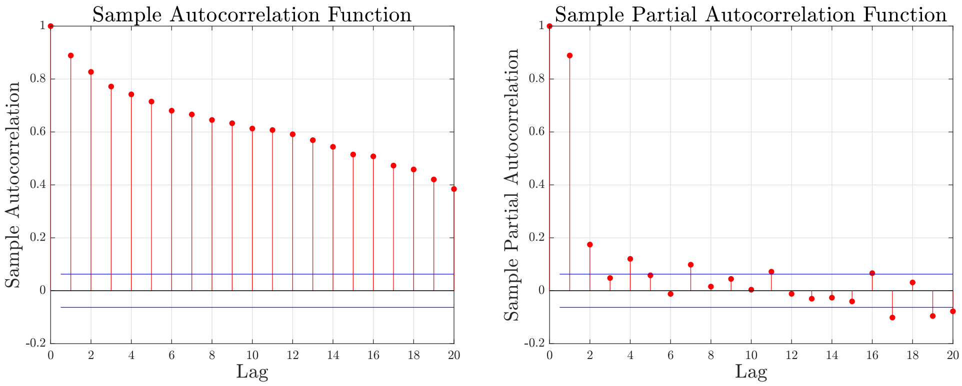


(b) time series, seasonal component



(c) time series, seasonality and trend removed

Figure 3.4: Detrending and deseasonalization process



(a) Autocorrelation function **(b)** Partial-autocorrelation function

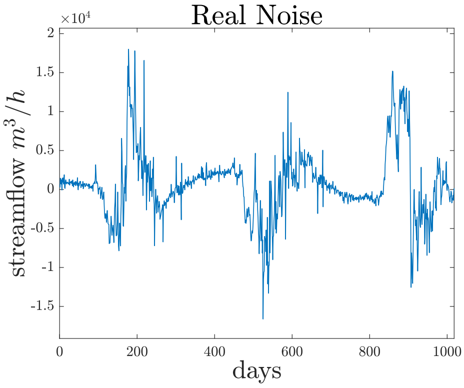
Figure 3.5: Autocorrelation and Partial-Autocorrelation Functions

In respect of the considerations made, the model to be fitted is and ARMA(2,1), a comparison between real noise and synthetic noise is reported in figure 3.6. As can be seen, although following a similar process, the generated noise is quite different in term of amplitude, this is due to the fact that ARMA cannot simulate a periodic behaviour.

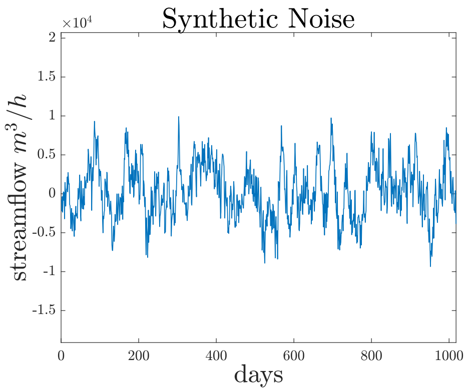
Thus, to adjust the generated noise according to the original variance, the modulation signal is identified. Original noise and the envelope are shown in figure 3.7. In order to smooth the envelope curve a moving root mean square window over 50 samples have been applied *before* calculating the analytical signal.

The final synthetic noise generated results in a correct representation of the measured erratic component, as seen in figure 3.6c.

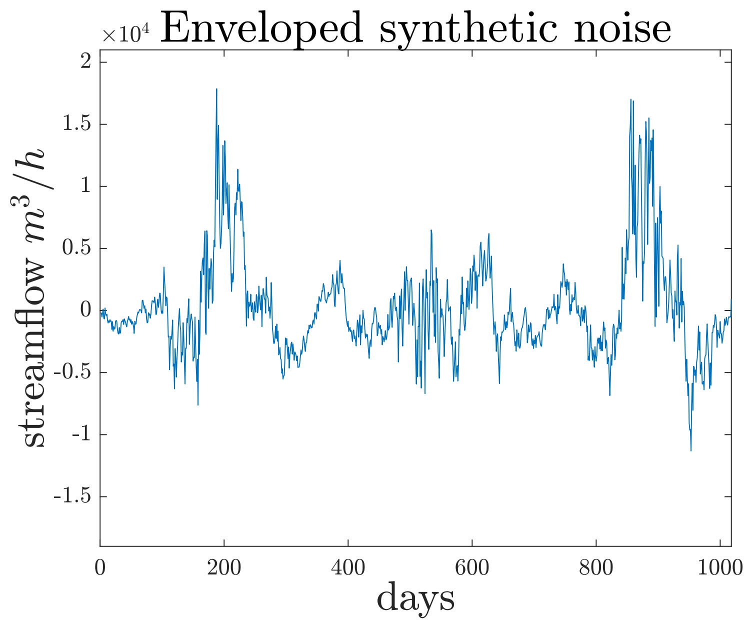
The final synthetic streamflow is presented in figure 3.8, compared to 3.2 can be asserted that the streamflow generated synthetically with the adopted method is a good representation of the real phenomenon. The same conclusions can be drawn for all the others synthetic streamflows generated for the case study and reported in 3.9. Negative values could be present since the data are generated synthetically, in these cases the negative streamflow is set to zero.



(a) Real noise



(b) Synthetic noise



(c) Enveloped synthetic noise

Figure 3.6: Real noise vs Synthetic noise

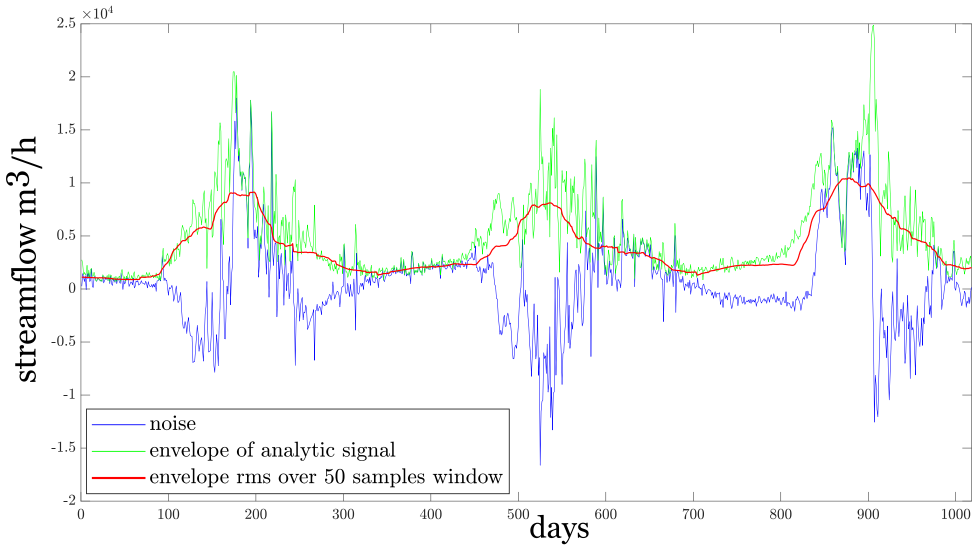


Figure 3.7: Noise envelope for Q1 streamflow

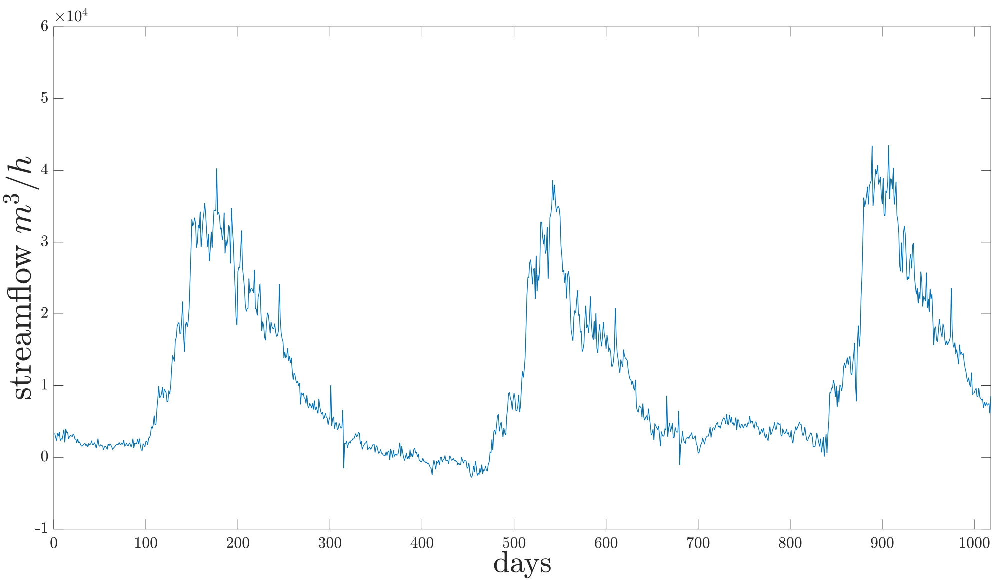


Figure 3.8: Synthetic streamflow Q1 reservoir

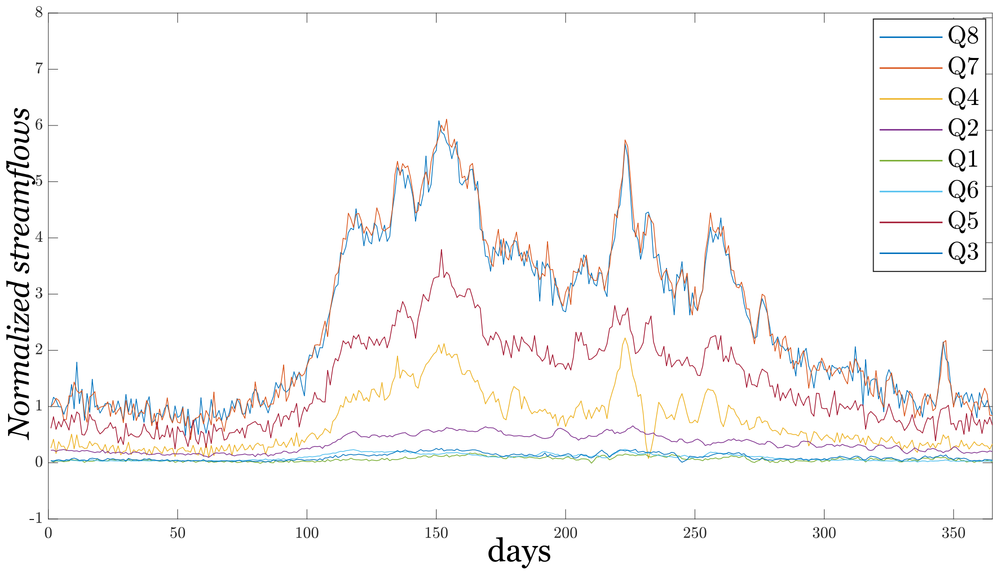


Figure 3.9: Normalized synthetic streamflows of case study reservoirs

3.2 Electricity price perturbation

The second input to be modelled is the price of electricity, the whole time history of electricity price in Italy is available at [17]. For the time series analysis has been taken into consideration a period of data that ranges from 2013/01/01 to 2018/12/31, unlikely streamflow data that are given daily, price time series resolution is of one hour.

To simulate electricity price a statistical model is adopted, although a rich literature about this topic is available, complex and reliable models require a deep electrical market analysis and must be fed with a big amount of exogenous data such as production and consumption forecasts. A refined implementation of an electricity price generator is therefore out of the scopes of this thesis, the argument could be certainly among the ones to consider for a future development of this work.

The procedure is described in the following steps.

3.2.1 Identification of the main periodic components

Through a Fourier analysis on the whole set of data, periodic components are identified in figure 3.10. As can be seen two main periodic components are present and correspond to 1 cycle and 2 cycles per day. It's important to notice that Fourier analysis is performed on the whole set of data of 6 years length since longer time series allows for a better frequency resolution.

Once certified that exists a daily component, a Normal distribution is fitted for each of the 24 hours in a day. The number of samples per each our corresponds to the number of days considered.

Two cases has been inspected: Normal distribution computed on the whole time history, considering a data set of $6\,yrs \times 365\,days$, and Normal distribution computed over the last 200 days. The comparison between the two cases is fundamental to understand which data use to generate a synthetic electricity price, plot 3.11 clearly shows how energy price has risen in the last six years. This suggests to use only recent data for price generation in order to avoid having a biased synthetic time history, moreover in case only the last 200 days are taken into account normal distributions are fitted with much less dispersion.

3.2.2 Synthetic energy price generation

Box plot in figure 3.12 represents the identified daily statistics that are then used to generate energy prices. It's important to remark that

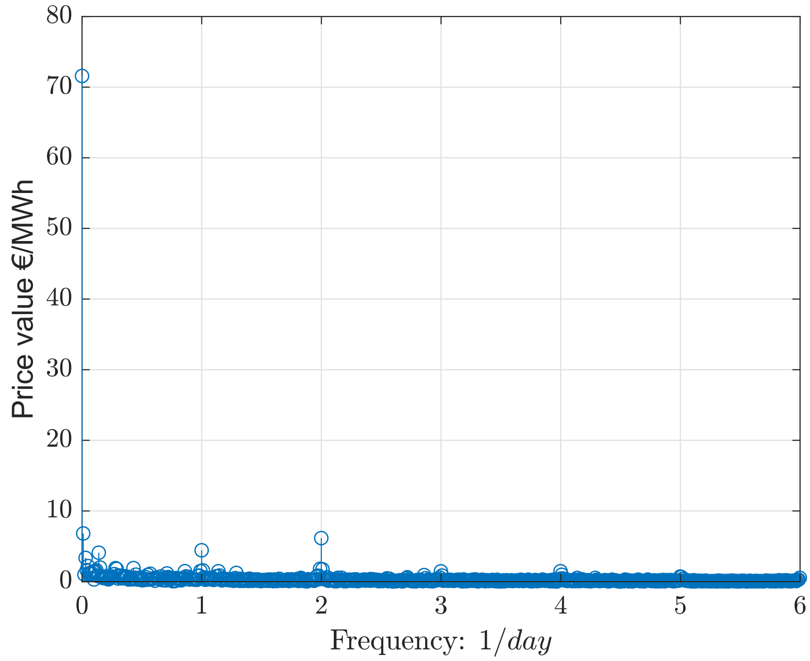


Figure 3.10: Fourier analysis on the whole time history

this simple approach doesn't allow to reconstruct the real price behaviour, it can only generate a time series that has a similar periodicity and that correctly represent hourly mean value and variance. Marked dynamic is not considered, a limitation example could be that usually prices are higher/lower than the mean for a certain group of hours (if a price is high at time t it's likely to be high also at time $t + 1$), this fact is not described by the model.

Figure 3.13a refers to the mean price calculated considering the 200 days prior to the date considered, the two cycles/day identified in Fourier analysis are here shown with two peaks at about 9:00 and 20:00. Figure 3.13b presents 3 synthetic time series generated for the next 24 hours, as can be seen, the mean value on average is respected but the process is completely random since is governed by random Gaussian variables.

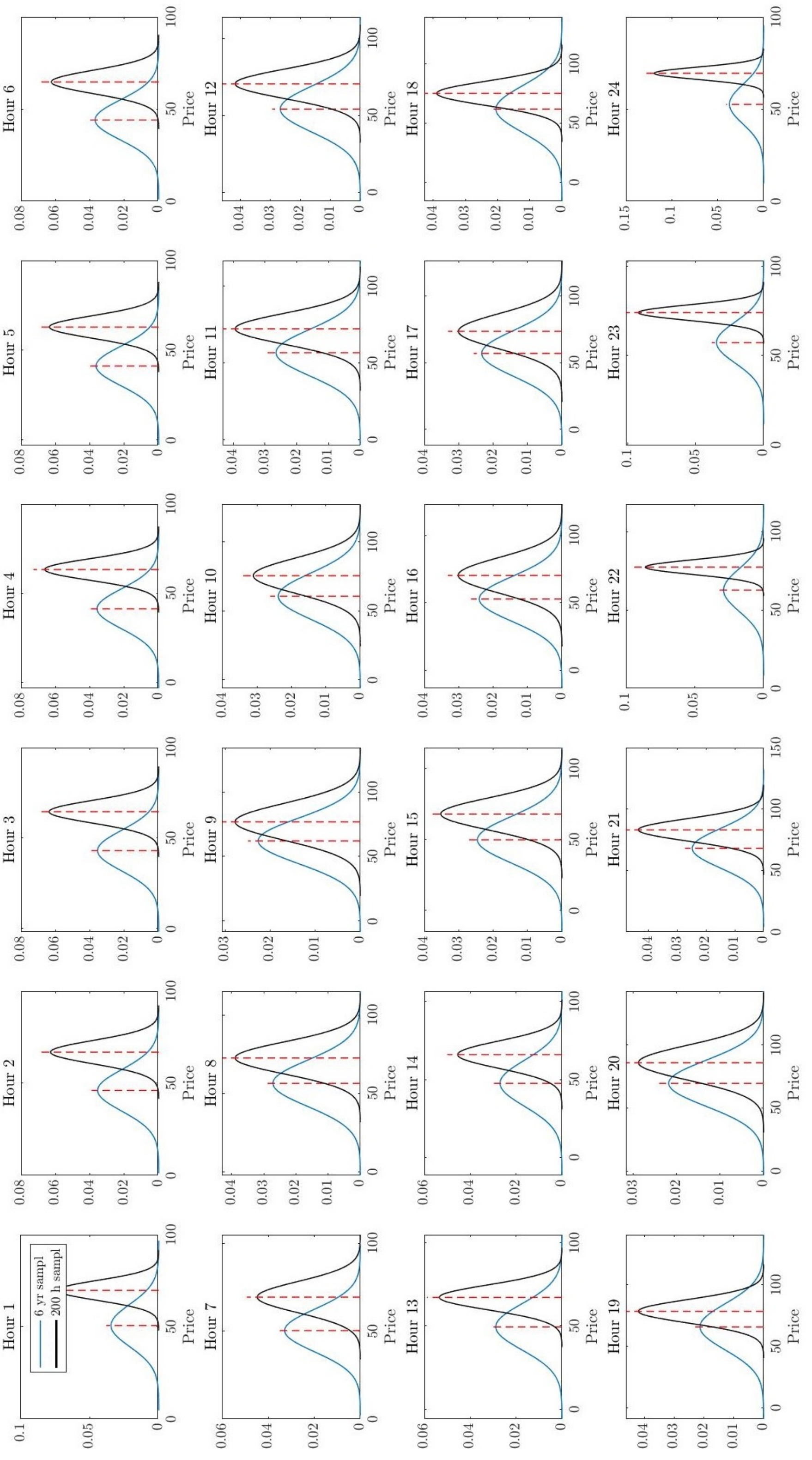


Figure 3.11: Normal price distribution hourly comparison

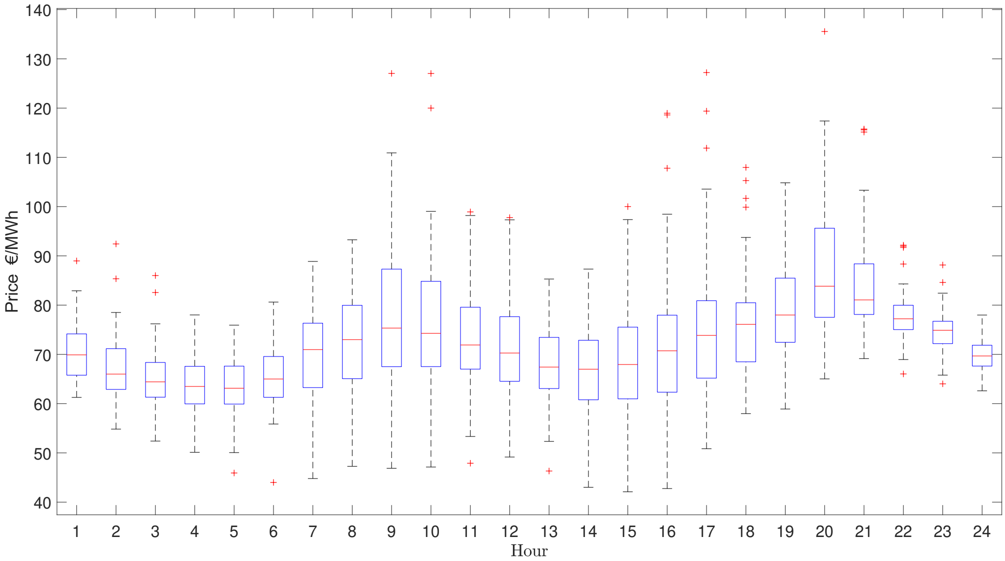
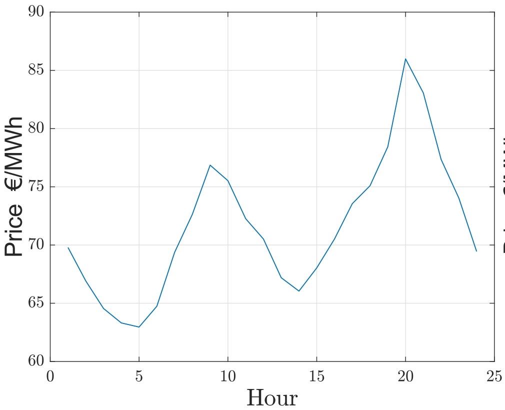
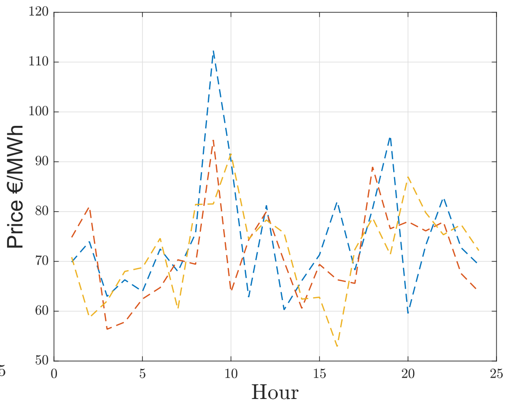


Figure 3.12: Energy price distribution during the day



(a) Mean hourly price



(b) Set of 3 syntetic prices over 24 hours

Chapter 4

Long term optimization

In literature, referring to chapter 2, most of the applied optimization methods regard the optimal management of reservoirs in the short term periods (weeks at most), however from a practical perspective it is necessary to run the simulation over period of at leas one year. Being the terminal boundary conditions of the system (the reservoirs water levels) unknown, they must be arbitrarily fixed and if a short optimization period is chosen they affect considerably the optimal solution outcome. The period of at leas one year has been selected the streamflows, and therefore the system operation, presents an annual cyclicity as demonstrated in 3.1.

Due to the long term optimization and the annual cyclicity of the problem, final reservoirs water level are set to values similar to the initial ones without greatly affecting the optimal result.

It's important to notice that the problem cannot be open-ended since the optimal solution would be always to use all the water contained in the reservoirs and at T_{end} all the basins will result in reaching their minimal value.

4.1 Objective

Long term optimization has a dual purpose: first of all it can be used for budgeting reasons since can be considered as a predictor of a good reservoir operation management; then, it is also used to draw the nominal height trajectory that reservoirs should follows along the year, this is necessary to furnish an indication for ending reservoirs volumes in the short therm optimization problem, as seen in chapter 5.

Due to the computational limits described in 2.5.3, it's necessary to find an alternative method to solve the problem over the long term. The objectives of the work described in this chapter are thus to implement a new resolution method based on the algorithm already developed, capable to solve the optimal problem and to test its performances compared to the actual approach. The focus is mainly on the study of the trade-off between computational costs and final value of the optimal solution, taking into account not only the robustness of the new approach but also its behaviour with respect to the input uncertainties modelled in chapter 3. It's important to understand to which extend uncertainties affect the optimal solution and, as consequence, how much computational costs can be reduced at the expenses of the solution optimality.

4.2 Adopted methodology

The new optimization strategy uses the same model structure described in section 2.5 but, in order to reduce computational costs, the problem is segmented into smaller sub-problems that are easier to solve; the adopted technique is known to be called "divide and conquer", since thanks to the problem decomposition its complexity is strongly reduced. The two dimensions in which the problem is broken down are space and time, for space segmentation is intended the problem simplification in its structure: from the optimization of a complex network of plants and reservoirs to the optimization of a single reservoir isolated from the others. Time simplification instead refers to the problem description in its overall structural complexity but solved on smaller time windows sequentially concatenated.

The segmentation strategy adopted is a two step process, first the problem is solved in his space discretized form, then the solution of the first step is used to solve the problem segmented in time. The process can be described according to the following steps, it is represented also schematically in figure 4.1, where the working principle's logics is diagrammed.

(i) Identification of seasonal reservoirs

(ii) Spatial segmentation of the problem

(iii) Optimization of individual seasonal reservoirs sub-problems on the whole time span

(iv) Time segmentation of the system in its completeness

(v) Optimization of the time segments

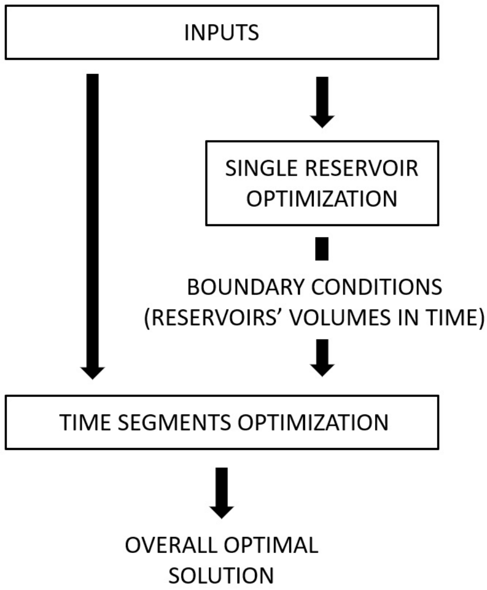


Figure 4.1: Schematized segmented optimizer's working diagram

Even if the overall number of optimizations needed is higher, being the complexity of the sub-problems much smaller than the original one, the total computational effort is strongly reduced. The optimization algorithm used is the one described in 2.5; what changes is the framework in which has been used: instead of the overall problem, the single sub-problems are fed into the base algorithm and the results assembled in a secondary moment.

4.2.1 Identification of seasonal reservoirs

Are denominated as "seasonal" all the reservoirs that follow a particular water level path along the year, typically these reservoirs have a very large capacity with respect to the maximum flow allowable to the turbines in the related power plant. The large capacity smooth out all the daily and weekly variations of the free surface of the basin, this means that once the optimal solution has been computed, variations in the control variables around the optimal solution only marginally affect the reservoir free surface height. The lecturer may notice that no considerations have been made about the positioning of the basins inside the system, a seasonal reservoir has not to be necessarily upstream with respect to the others, it is only required a sufficient water capacity.

In the case study taken into consideration, represented in fig. 1.5 on page 9 , the reservoirs considered to be seasonal are Q_1 and Q_2 due to their big dimensions. Also Q_4 and Q_5 could be included among seasonal reservoirs but due to their particular common relation with P_4 (described in section 1.1.3 on page 8) cannot be optimized independently.

4.2.2 Spatial segmentation of the problem

In order to be optimized independently, the seasonal reservoirs must be isolated from the rest of the model, all the inputs needed for optimization, described in chapter 2.5.1 must generated by taking only the parts related to the reservoir considered. For what concerns the time dimension, the single basin is optimized over the overall time span, so no cuts to input matrix in time direction are performed. The main changes in the inputs are listed below, considering a seasonal reservoir of index $\overline{r}$

(i) $S\_streamflow\_matrix = streamflow\_matrix\,(:, \overline{r})$ $[T \times 1]$: only the streamflow considering reservoir $\overline{r}$ are taken into consideration

(ii) $S\_structure\_matrix$ $[1 \times N_r]$: the structure matrix collapses in a vector of ones, since only one reservoir and the related turbines are taken into account

(iii) S_power_vector $[N_{\overline{r}} \times 1]$: only the variables related to reservoir $\overline{r}$ are taken into consideration

(iv) $S\_concentration\_time\_vector = null$: with a single reservoir case the concentration time loses its meaning

(v) S_max/min_flow_matrix $[T \times N_{\overline{r}}]$: only the variables related to reservoir $\overline{r}$ are taken into consideration

(vi) S_max/min_power_matrix $[T \times N_{\overline{r}}]$: only the variables related to reservoir $\overline{r}$ are taken into consideration

(vii) $S\_max/min\_volume\_matrix = max/min\_volume\_matrix(:, \overline{r})$ $[T \times 1]$: only limits on reservoir $\overline{r}$ are considered

(viii) $S\_max\_height\_rate\_matrix$ $[\#_{constraints} \times 3]$: only the constraints referred to reservoir $\overline{r}$ are copied in the matrix

(ix) $initial/final\_volume\_value = initial/final\_volume\_vector(\overline{r})$ $[1 \times 1]$

In case the seasonal reservoir is not upstream but downstream with respect another control variable to the $S\_streamflow\_matrix$ must be summed up the flows coming from that variables. Two cases have to be considered, if the upstream variable is related to a non-seasonal reservoir all the input streamflow of that reservoir are directly summed to the streamflow $S\_streamflow\_matrix$ matrix with a time lag corresponding to the concentration time. In case the upstream control variable is related to a seasonal reservoir, optimized flows coming from the sub-problem of the upstream reservoir are added to the $S\_streamflow\_matrix$ of the basin considered.

4.2.3 Optimization of single reservoirs

It can be noticed that, according to the adjoining procedure of streamflows described, the single sub-problems must be necessarily optimized in series starting from the most upstream reservoir, thus no parallel computing is possible in the spatial segmented optimization.

The sub-problems to be optimized have a much smaller number of variables, from $\#CV = N_{tot} \cdot 8760$ to $\#CV = N_{reservoir,\overline{r}} \cdot 8760$ where $N_{reservoir,\overline{r}}$ is the number of control variables related only to reservoir $\overline{r}$.

In the case study considered, for reservoir Q_2 the variables number pass from $\#CV = 19 \cdot 8760 = 166440$ to $\#CV = 2 \cdot 8760 = 17520$, it's evident how the problem complexity is reduced. Moreover the dimensions of the system matrix A passes from $[I_{reservoirs} \cdot T_{hours} \times X_{variables} \cdot T_{hours}]$ to $[1 \cdot T_{hours} \times X_{var,\overline{r}} \cdot T_{hours}]$, for the case study studied this means a matrix 76 times smaller which results in a much smaller ram consumption.

4.2.4 Time segmentation of the problem

Once the single seasonal reservoirs has been optimized the problem discretization is shifted to the optimization of sub-problems that contain the overall structure of the problem but over just a portion of the total optimization time. For these sub-problems the boundary conditions are provided from the solution of the single reservoirs optimization, in fact, the inputs of the original problem only specify initial and final set-points, no information about intermediate reservoir levels are provided. From the previous optimization the optimal profiles for the singles seasonal reservoirs are known, these profiles are considered as a base curve over which extrapolate initial and final water reservoirs volumes for the different sub-problems in time. Please note that only seasonal reservoirs are optimized in the previous step, smaller basins are considered to be cyclical within a day or more, thus for these reservoirs initial and final state is always set to be equal to the initial boundary conditions.

The input of the sub-problem are rewritten according to the time segment considered, starting at time T_{start} and finishing at T_{end}. Note that $\Delta T = T_{end} - T_{start}$ is a constant value and represents the length of the time step in which the overall optimization window is divided: $\Delta T = T_{total}/N_s$, where N_s is the total number of segments. An exception is made the last segment duration, as will be explained later. Inputs are written as follows:

(i) $S\_price\_vect = price\_vect(T_{start} : T_{end})$ $[\Delta T \times 1]$: only the prices corresponding to the actual segment hours are selected

(ii) $S\_streamflow\_matrix = streamflow\_matrix\,(T_{start} : T_{end}, :)$ $[\Delta T \times I]$: the whole reservoirs data are used but only for the corresponding segment time

(iii) $S\_power\_vector = power\_vector$ $[N \times 1]$: since is not time dependent remains unvaried

(iv) $S\_concentration\_time\_vector = concentration\_time\_vector$: in opposition to the space segmentation, now the whole concentration time vector is used

(v) $S\_max/min\_flow\_matrix = max/min\_flow\_matrix\,(T_{start}, : T_{end}, :)$ $[\Delta T \times N]$: all the variables but only constraints related to the time segment

(vi) $S\_max/min\_power\_matrix = max/min\_power\_matrix(T_{start} : T_{end}, :)$ $[\Delta T \times N]$: all the variables but only constraints related to the time segment

(vii) $S\_max/min\_volume\_matrix = max/min\_volume\_matrix(T_{start} : T_{end}, :)$ $[\Delta T \times I]$: all the reservoirs but only constraints related to the time segment

(viii) $S\_max\_height\_rate\_matrix$ $[\#_{constraints} \times 3]$: only the constraints referred to the related time segment are copied in the matrix

(ix) $initial/final\_volume\_value$:

$$init\_volume\_value(r) = \begin{cases} opt\_volume(T_{start}, r) \text{ if r seasonal} \\ init\_volume\_value(r) \text{ if r is cyclic} \end{cases}$$

$$fin\_volume\_value(r) = \begin{cases} opt\_volume(T_{end}, r) \text{ if r is seasonal} \\ fin\_volume\_value(r) \text{ if r is cyclic} \end{cases}$$

Where *optimal_volume* is a matrix containing, for each column, the optimal volumes of water for the seasonal reservoirs along the whole optimization period. This matrix is generated thanks to the first optimization stage over the single basins.

4.2.5 Optimization of time segments

It is of fundamental importance to evaluate the problem complexity and understand if it's necessary to adopt the described divide and conquer technique in place of the global optimization. An algorithm has been introduced to establish if a problem is too complex and needs to be decomposed.

The described algorithm effectively reduce the computational costs only if the problem structure presents at least two reservoirs, of which one of the two must be seasonal. This because in case of only one basin the problem is already in it's minimum spatial subdivision and if non-seasonal reservoirs only are present no optimization is run over the whole period and thus no set points are available for the time segmentation.

If the problem respects the previous necessary conditions to be decomposed, its complexity is evaluated according to the problem dimensions: $P\_dim = I_{reservoirs} * N_{variables} * T_{hours}^2$. From the problem dimensions the cost in term of RAM memory usage is computed, if the maximum memory installed is exceeded the problem is solved in its decomposed form. Another condition that triggers the divide and conquer approach is that the optimization time cannot be longer than the maximum time segment length ΔT_{max}. For further informations about the evaluation algorithm, a part of it is reported in listing A.1 on page 127.

In the case divide and conquer approach is chosen it is necessary to set rules for the division in time of the optimization problem. The segment length ΔT must respect the following constraints:

1. $\Delta T \leq \Delta T_{max}$
2. ΔT must be multiple of 24 hours
3. Remainder of $T_{hours}/\Delta T$ at least $0.4 * \Delta T_{max}$
4. If the remainder is smaller than $0.4 * \Delta T_{max}$ but $remainder + \Delta T \leq \Delta T_{max}$ the selected ΔT is a valid value but the last segment length is $remainder + \Delta T$

5. Among the whole set of ΔT that respect the constraints, $\Delta \overline{T} = \max \Delta T$ is chosen

Condition 2 is necessary to respect the cyclicity of non-seasonal reservoirs that usually have one day frequency variation (filling up overnight and discharge during peak price hours). Conditions 3 and 4 are necessary to ensure a sufficient length of the last time time segment, if the resulting step is too short boundary conditions strongly affect the optimal solution and there is the risk that no solutions to the problem can be found due to lack of flexibility. Last condition is to select the maximum time window size possible in order to remain in the neighbourhood of the optimal solution that would be obtained from a global optimization. The algorithm is reported in listing A.2 on page 128.

The last parameter that remains to be identified is the maximum period length ΔT_{max}, the choice of this variable is of fundamental importance since it directly affects the trade-off between computational performances and optimal solution obtained. For bigger ΔT_{max} the problem is divided into a smaller number of more complex sub-problems, this is beneficial from an optimality point of view but detrimental in terms of computational costs since the solution of a bigger number of smaller sub-problems is much faster. The analysis conduced in this chapter are not only useful to validate the divide and conquer strategy but also to determine a ΔT_{max} value that guarantee a good trade-off between costs and results.

4.3 Simplifying assumptions and considerations

Since the final optimization of the sub-problems is based on the approach described in 2.5, this methodology shares also all its the limitations, except for the ones related to the computational costs.

Considering that the model input are affected by uncertainties; in reality, over the long term, the exact global optimum solution is not needed but it's satisfactory to be in its neighbourhood. This consideration allows to apply a divide and conquer strategy knowing that the solution obtained with the implemented model will be certainly sub-optimal to the global optimization, performed with the base algorithm.

For what concerns more operative assumptions, the seasonal reservoirs are considered not only to have small daily/weekly variations but also to have a considerable weight in the optimization of the system. The importance of a reservoir is not only defined by the nominal power of the

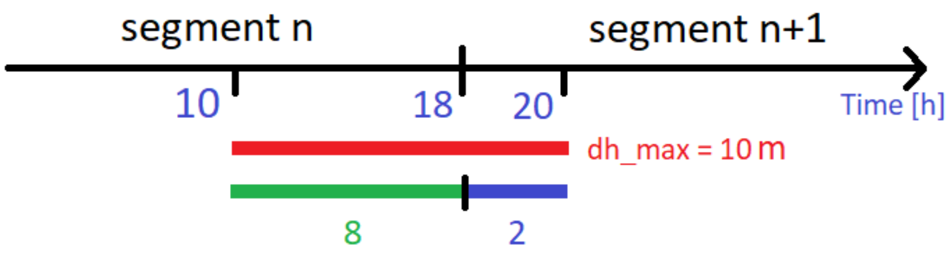


Figure 4.2: Example of split constraint

turbines related but also from their positioning into the system, an upstream reservoir for example has much more importance than a downstream reservoirs, since its water has an higher intrinsic value and due to the fact that thank to its dominant position it directly affect all the water flows downstream.

Smaller reservoir are instead considered to play a secondary role into the system, their main purpose is to decouple the daily demand of energy during the day from the water availability thanks to accumulation and discharge during a daily/weekly cycle. No long term effects of these reservoirs are considered, since their boundary conditions are set equal for each time segmented sub-problem.

An added approximation with respect to the base model is that if a constraint that last for more than an hour is defined in coincidence with the separation of one time segment to another, also the constraint is split into two parts. A graphical example is provided in figure 4.2, it shows how a constraint on the maximum change in height of 10m over a period that goes from hour 10 to hour 20 is proportionally divided into the two time segment. Due to the subdivision of constraints along the separation of two segments and to the reduced time-span between two boundary condition the model has loss of flexibility in the possible solution field.

4.4 Performances comparison with original approach

In this section is evaluated the behaviour of the divide and conquer approach with respect to the original approach, before analysing the effects of input uncertainties, results consistency and robustness must be assessed. Four different deterministic cases have been investigated, each case is solved according to: a complete optimization, a 3 segmented optimization and a 12 segments optimization. The case study used as a base is 1.1.3, the constraints introduced and the physics of the system is set as a constant, what changes from one case to another are the initial and final boundary conditions and the streamflows in input for each reservoir. The total optimization time length is 2170 h, which is considerably smaller than the number of hours in a year, this choice is necessary in order to compute the optimal solution also in case of the global optimization, 2170 h is in fact the maximum allowable optimization length with the original approach solution given the computational hardware available.

Table 4.1 describes the characteristics of each case:

Table 4.1: Deterministic cases description table

case	1	2	3	4
initial volume	half res. capacity	half res. capacity	half res. capacity	half res. capacity
final volume	minimum res. capacity	half res. capacity	half res.r capacity	half res. capacity
streamflows	nominal	nominal	x2	x4

In case 1, final volume equal minimum reservoir capacity only for basins $Q_{1,2,3,4,5}$, reservoirs Q_6 and Q_7 are set to half of reservoir capacity to not compromise the feasibility of the solution.

As can be seen, with the only exception of case 2, case 1,3 and 4 represents particular conditions that are very unlikely to happen in reality. These stress-tests of the systems are used to highlight the differences in the global and segmented optimizers behaviours. An important key performance indicator that can be measured for the different cases is the quantity of wasted water value, for wasted water is intended the difference between the amount of water which is discharged from the bypass variables in the segmented optimization and the water bypassed in the global optimization. Mathematically it's represented as:

$$wasted\_watervalue = \sum_{t=1}^{T_{end}} P_{rice}(t) * \sum_{v=12}^{19} [flow\_matrix\_global(t,v) - flow\_matrix\_segmented(t,v)] * KEN\_matrix(t,v)$$

Losses due to bad allocation can be obtained subtracting wasted water value from the total revenue losses, bad allocation losses represents the diminished value of the revenue function due to "bad choices" in allocating water along the time profile. Thus total revenue losses can be decomposed according to its two main components, wasted water and bad allocation.

Nominal streamflows are intended to be similar with respect to the average streamflow values encountered in reality for the case study, as can

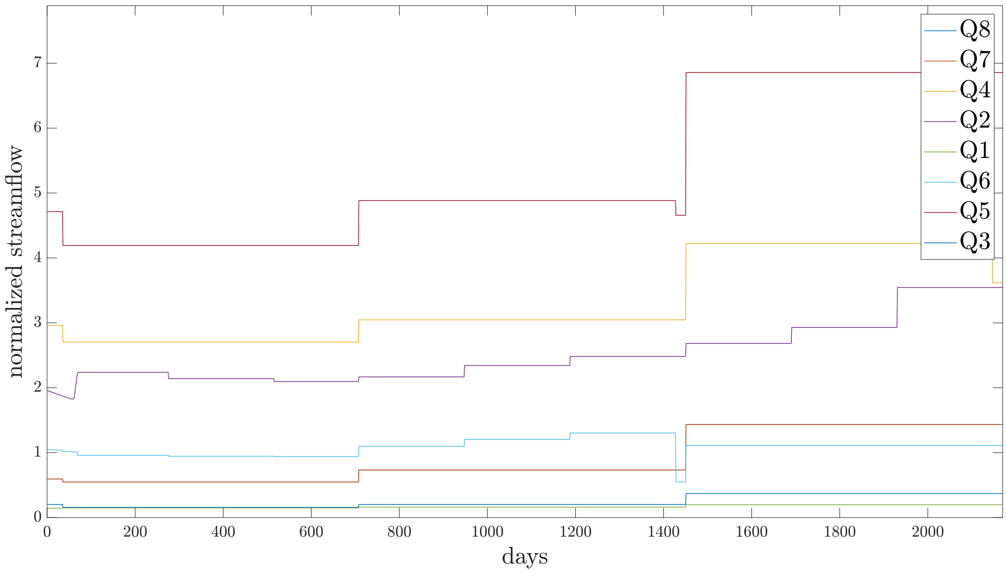


Figure 4.3: Deterministic nominal streamflow, normalized

be seen from figure 4.3, in deterministic case the streamflow adopted are composed only by long steps functions since in the deterministic study only the comparison among the different cases and approaches is evaluated. Please note that, for disclosure reasons, the order of magnitude along y axis is omitted.

4.4.1 Optimization results

A typical optimization outcome produces different outputs, results are provided in a matricial form similar to the inputs described in 2.5.1. The most important ones are described below:

(i) *flow_matrix* $[T \times N]$: it contains the flow rate of the different control variables in m^3/h defined at each time step

(ii) *power_matrix* $[T \times N]$: the energy produced by each control variable in the given hour expressed in MW h

(iii) *volume_matrix* $[T \times I]$: the volume of water contained in each reservoir at each hour expressed in m^3

(iv) *height_matrix* $[T \times I]$: the height of the free surface water level contained in each reservoir at each hour expressed in m, it's the dual of *volume_matrix*

(v) KEN_matrix $[T \times N]$: it contains the values of the energetic coefficient for a given control variable at a given hour, the coefficient accounts for the final reservoir height contained in *height_matrix* and associated to the given control variable

(vi) *optimal_value* $[1 \times 1]$: it is the final optimization value coming from the linear programmer, expressed in €

It is important to notice that for both approaches, overall optimization or divide and conquer, the output structure is exactly the same. For the sake of simplicity the results about reservoir levels will be always expressed in reservoir water volume rather than reservoir water height.

As can be noticed the optimal value is a redundant information since it can be calculated also according to the following equation:

$$optimal\_val = \sum_{t=1}^{T} Price(t) * \sum_{v=1}^{N_v} [flow\_matrix(t, v) * KEN\_matrix(t, v)]$$

This equation is used to calculate the optimal values since it is more representative of the real process.

Case 2 - nominal case

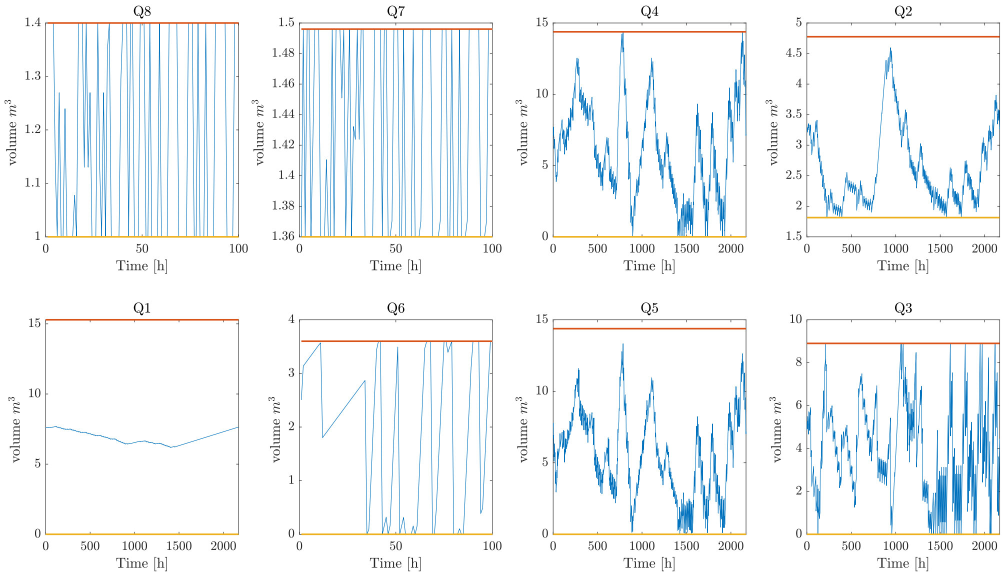
Figure group 4.3 shows an example of outputs, the solution for *case 2* which is considered the nominal case. Results presented are computed optimizing on the overall time length.

Plot 4.4a: represents the optimal flow profile in time for control variable. Red and yellow lines represent the maximum and minimum capacity values allowed and,as represented, are fully respected by the optimization. As can be seen the boundary conditions for all the reservoirs are consistent with the constraints imposed, the start at half total volume capacity and end at the same value. It's important to notice that the reservoirs have a periodicity which is proportional with their capacity, the bigger ones $(Q_1, Q_2, Q_3, Q_4, Q_5)$ show slow volume variation along time, while the smaller ones are characterized by a very high cyclicity (Q_6, Q_7, Q_8). An important observation useful to validate the hypothesis of annual cyclicity in seasonal reservoirs is that the two basins Q_1 and Q_2, assumed to be seasonal in the case study considered, present no repetition pattern in the optimization period (around 3 months). Since the figures refer to the global optimization, can be said with certainty that the reservoirs considered have a seasonal cycle. Better investigations could have been done considering a longer optimization period but, as already stated for the global optimization, computational costs would have been too high.

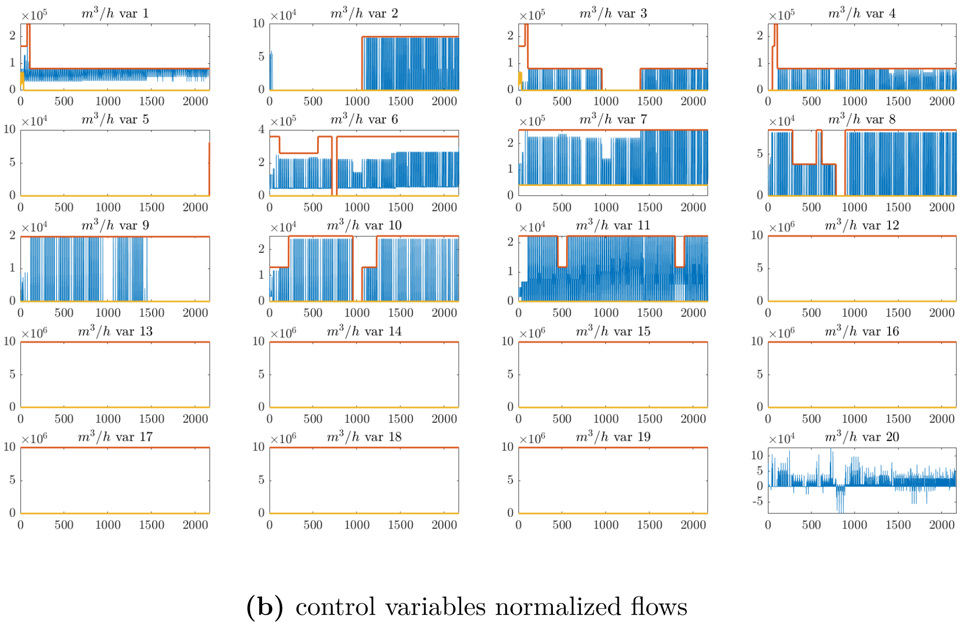
Plot 4.3b: represents the optimal water flow rate in time for each control variable. Red and yellow lines represent the maximum and minimum flow rate allowed for a certain control variable, as shown, they can vary in time, for example a maintenance stop is simulated for variable 3 between hours 1000 and 1500. According to table 1.2 on page 10 variables from 1 to 11 are turbines meanwhile variables from 12 to 19 are the emergency bypass of the different reservoirs, variable 20 is just a service bypass that has no physical meaning and can be neglected. As can be seen the bypass variables have always zero flow rate since, from an optimization point of view they don't produce revenues.

Plot 4.3c: represents the optimal energy to be produced for each control variable. As can be seen, for variable 9 and 11 there seems to be a capping over the maximum power value that a turbine can generate which is smaller that the actual constraint, this could be confused with an optimization error, but looking at 4.3b can be seen that that what limits the maximum productivity of the turbine is its maximum flow rate. What determines which of the two constraints is active is the value of the energetic coefficient (ken) in the corresponding hours, for high ken more power is produced with a lower flow rate, so in that case the power produced will be the effective limitation for the variable considered. Variables from 12 to 19 instead produce always zero power, since they are emergency bypass.

Plot 4.3d: represents a detail in the flow rate for variable 10, can be clearly seen that during the day a typical optimal power production profile is an on/off scale function where production peaks load are in coincidence with the two daily peaks of energy price.

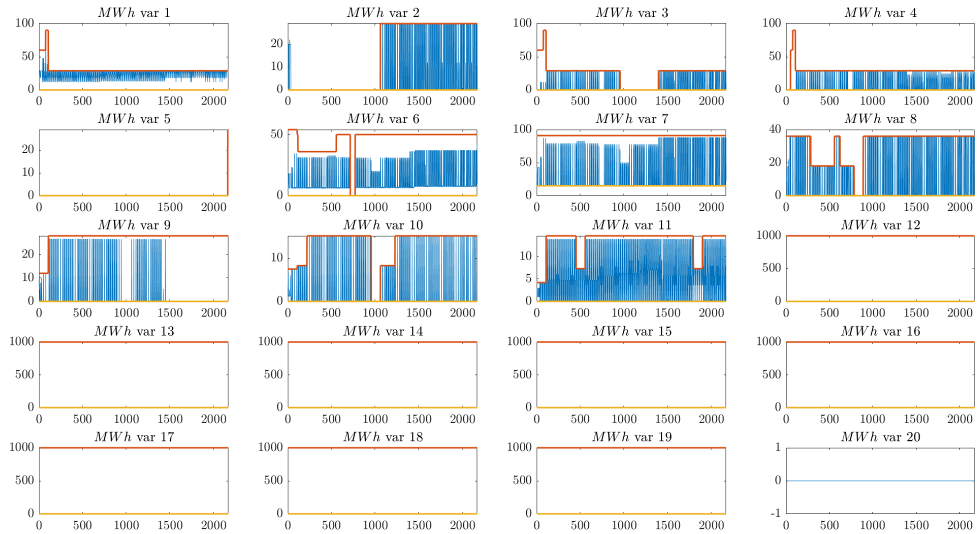


(a) reservoirs normalized volume in time



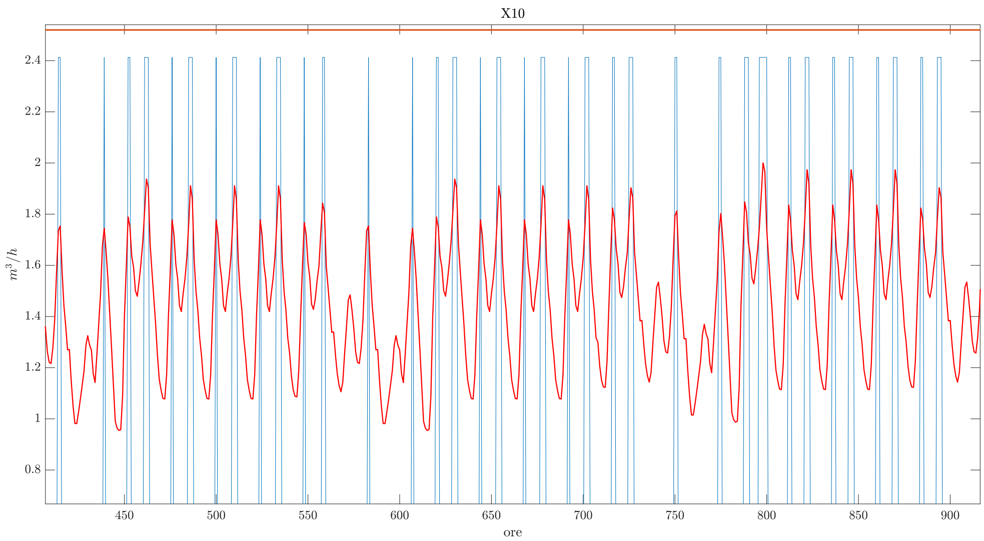
(b) control variables normalized flows

Figure 4.3: Case 2 optimization outcome, on overall time length



(c) energy produced

Figure 4.3: Case 2 optimization outcome, on 12 segments optimization

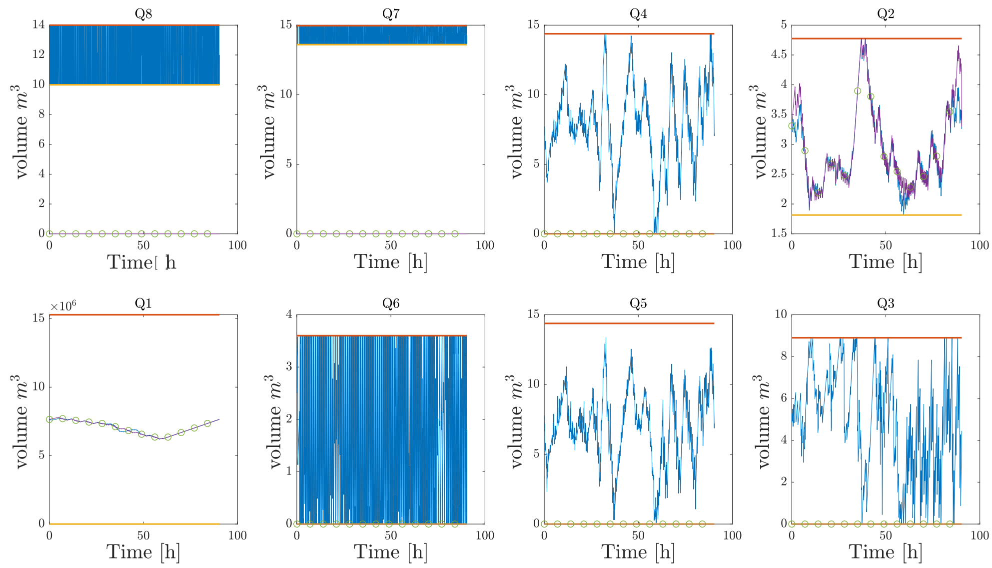


(d) Variable $X10$, turbine flows (blue) vs price (red)

Figure 4.3: Case 2 optimization outcome, on overall time length

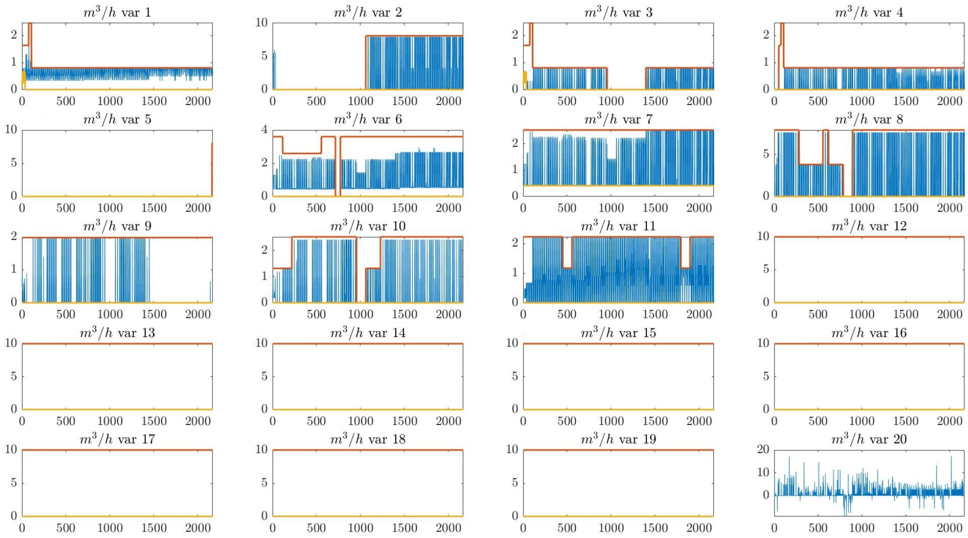
Figure group 4.4 shows the solution for *case 2* but this time, instead of a global optimization, the applied method is the divide and conquer approach. As can be better seen from 4.5, reservoirs Q_1 and Q_2 have been at first optimized singularly (violet line) and in the second optimization round optimized on the overall system but on 12 independent segments (blue line). Green circles represent the separation point of the different segments, as shown, optimal solution for single seasonal reservoirs and for seasonal reservoirs in the overall system coincide. Green circles represents the set point over which the segmented optimization in time is based.

A comparison between 4.3 and 4.4 shows how the two solution, although different, have a very similar behaviour. A more detailed comparison between the global approach and the segmented approach is given in the next section.



(e) reservoirs normalized volume in time

Figure 4.4: Case 2 optimization outcome, on 12 segments optimization



(f) control variables flows

Figure 4.4: Case 2 optimization outcome, on 12 segments optimization

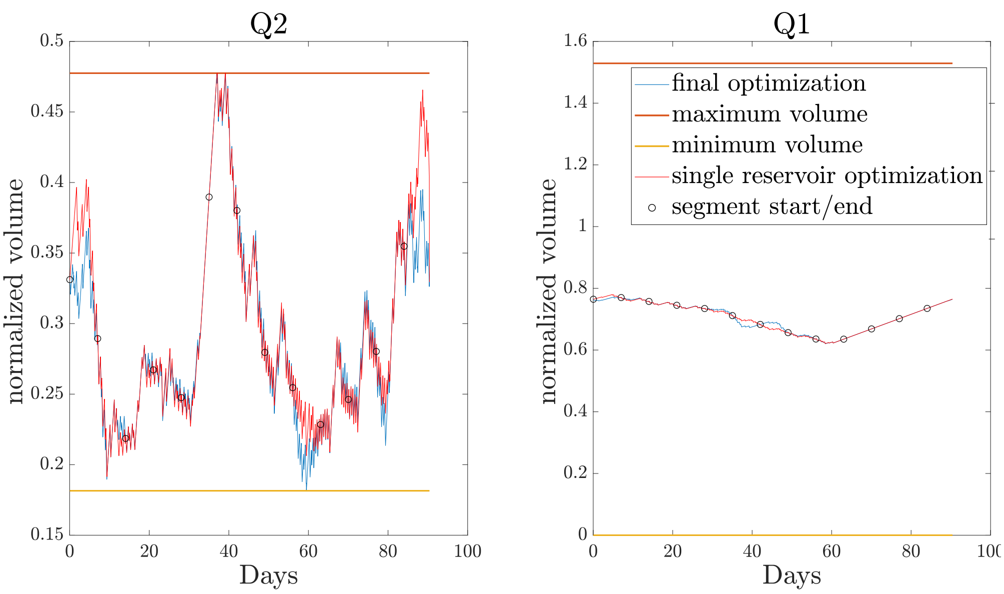
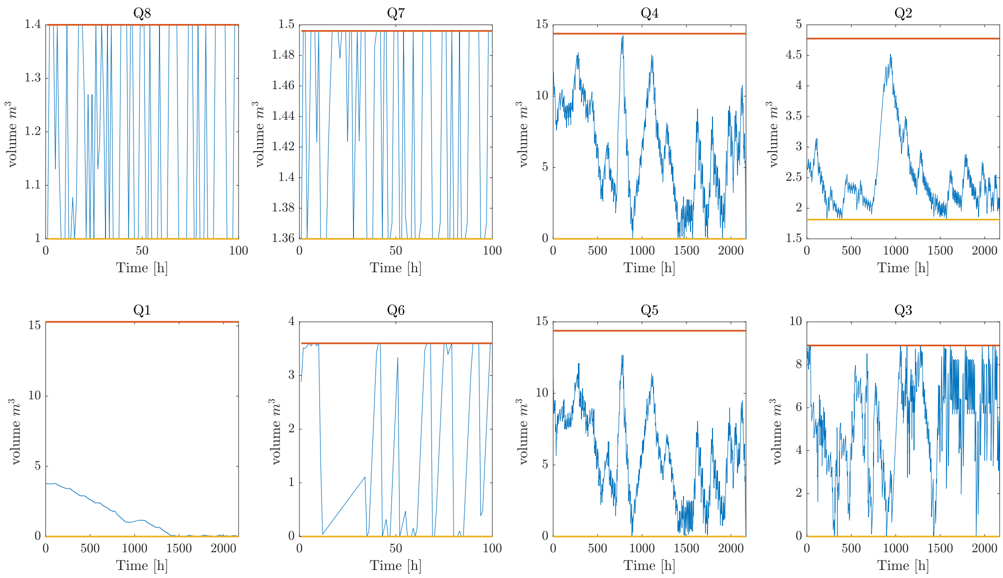


Figure 4.5: Case study principle scheme

Case 1 - reservoirs emptying

This case is not representative of reality since in normal operating condition no final volume target is set to minimum on all the reservoirs but it's useful to understand the behaviour of the optimizer, moreover this case allows to better highlight the differences in the final optimality value since it's require to exploit all the water resources available in the basins.

In all the three optimization typologies (global, 3 segments and 12 segments) similar results are obtained, for the sake of simplicity only the results from the global optimization are shown, a comparison among these methods is the performed in the section below. Figure 4.6 shows how the reservoirs $Q_{1,2,3,4,5}$ are set to assume their minimum value at t_{end}, with respect to *case 2* flow rates in the different are generally higher, this is due to the fact that a greater water amount can be used to produce electric energy.



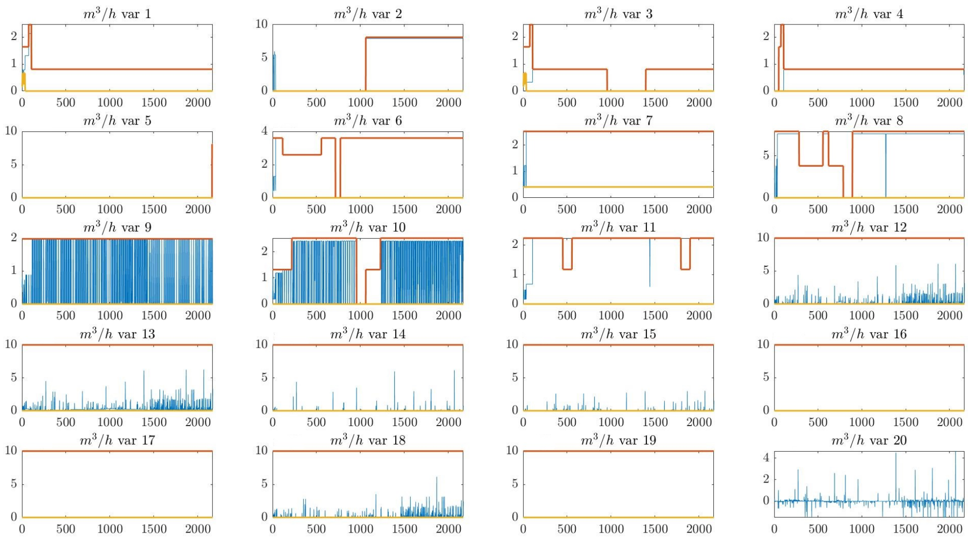
(a) reservoirs normalized volume in time

Figure 4.6: Case 1 optimization outcome, global optimization

Cases 3 and 4 - high streamflows

In cases 3 and 4 nominal streamflows are multiplied by x4 and x6 respectively, these cases are unlikely to happen in a real system but they are useful to characterize the behaviour of the different optimizers, if case 1 is useful to understand how the global optimization exploit resources better than the segmented one, case 3 and 4 are used to understand how much water is wasted in critical conditions and how the flexibility of the system is affected by the segmentation.

In section 4.4.1 can be seen how the variables $X_{12,13,14,15,16,17,18,19}$, related to bypass, are active and a noticeable amount of water is allowed to flow through them. The amount of wasted water is shown to be higher in the 12 segments optimization due to the reduced flexibility of the optimization. The two figures demonstrate also how the water flows for the different turbines is always at its maximum value for both the optimizations, this is due to the large amount of water available in the system.



Case 4: control variables flows, global optimization

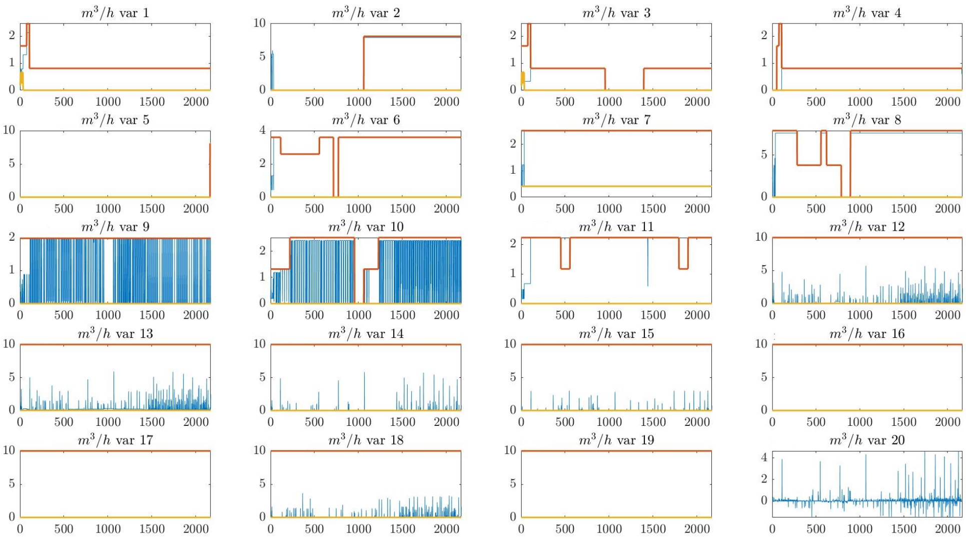


Figure 4.7: Case 4: control variables flows, 12 segment optimization

4.4.2 Cases comparison

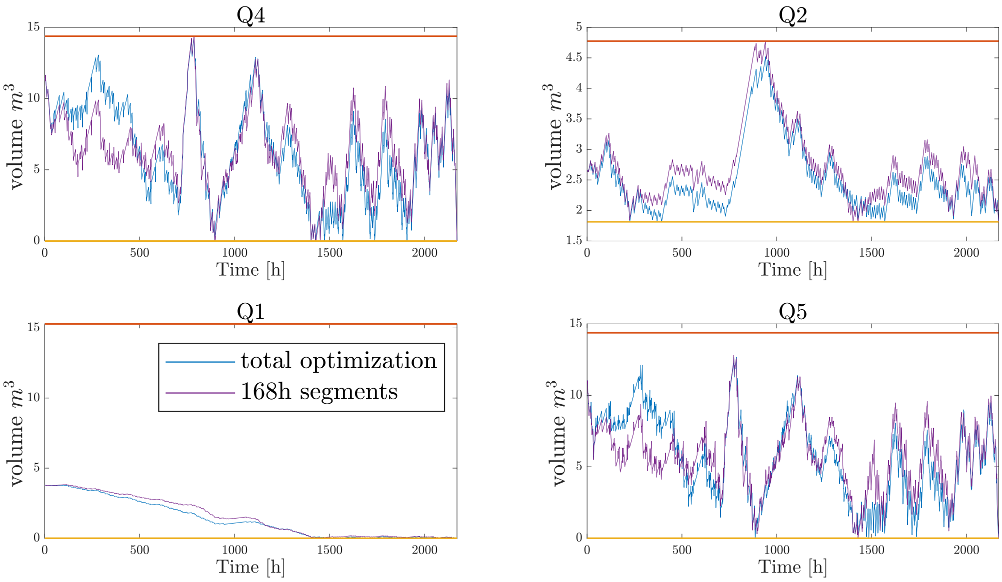
Commented optimization results are provided in the previous section, to effectively validate and compare the different optimization approaches a direct comparison between the results is needed. Since global optimization is considered to provide the best knowledge of optimum for the problem and, given the fact that the divide and conquer approach is a sub-optimum solution, all the results on which a differential analysis is needed are based on the solution provided by the global approach.

In figure 4.8, by directly superimposing solutions of the global optimization with the 12 segments optimization (168h time segments) it's shown how the two optimization results are similar but not exactly the same. Global optimization is compared with the 12 segments one neglecting the 3 segments optimization since, as expected, it shows a behaviour which is intermediate between the two. Reservoirs $Q_{1,2,4,5}$ are reported in figure, being among the bigger, they have a slower variation cycle and differences are better appreciated.

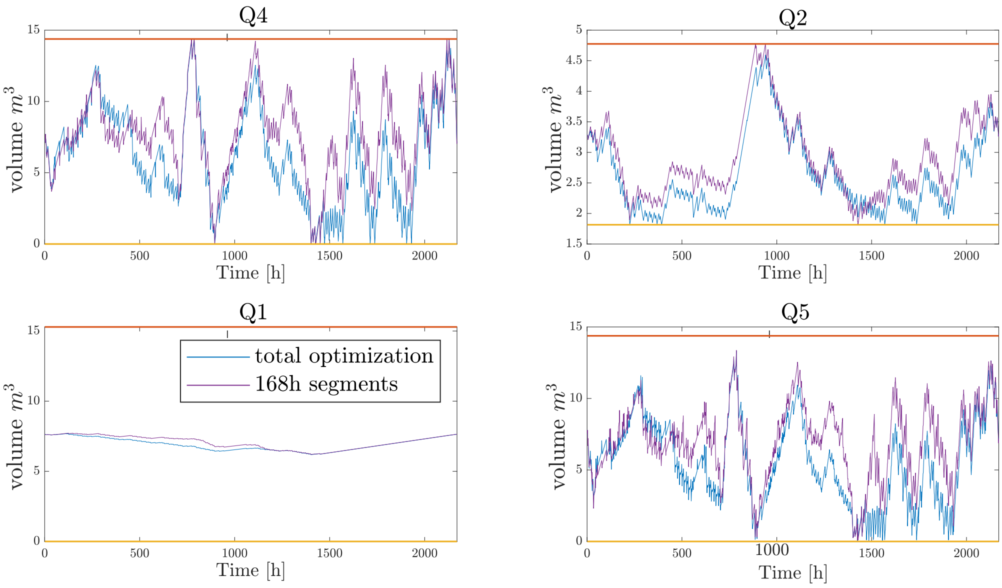
In both cases the constraints and target values are respected by the two optimization typologies. It's important to notice that in all the cases presented the reservoirs water levels are higher for the case of a 12 segments optimization, this is consistent with the fact that a global optimization considers the total system from the first instance. Upstream reservoirs are affected the downstream portion of the system that can use the same water to produce additional electric energy, thus water in the upstream portion of the system has greater economical value and tends to be used earlier with respect to the global optimization.

It is also of fundamental importance to compare the losses in the maximization of value function, figure 4.9 shows the different 4 cases and the associated revenue loss in percentage points (losses considered differential with respect to the global optimization). As expected the revenue losses are higher in the 12 segments optimization; this is mainly to the fact that, although the different segmented optimizations share the same optimal single seasonal reservoir policy, if the overall optimization time is divided in only three segments the solution space is bigger due to the higher flexibility of the system, moreover the effects of boundary conditions on the segments only partially influence the optimum due to the big length of time segments.

A can be seen, revenue losses in cases 1 and 2 are way bigger than case 3 and 4, this is due to the fact that the augmented streamflows introduce a big quantity of water into the system that partially hides the sub-optimal water management. Overall performances are shown to be still very good,



(a) case 1 - reservoirs normalized water level comparison



(b) case 2 - reservoirs normalized water level comparison

Figure 4.8: Comparison between optimal and segmented optimization

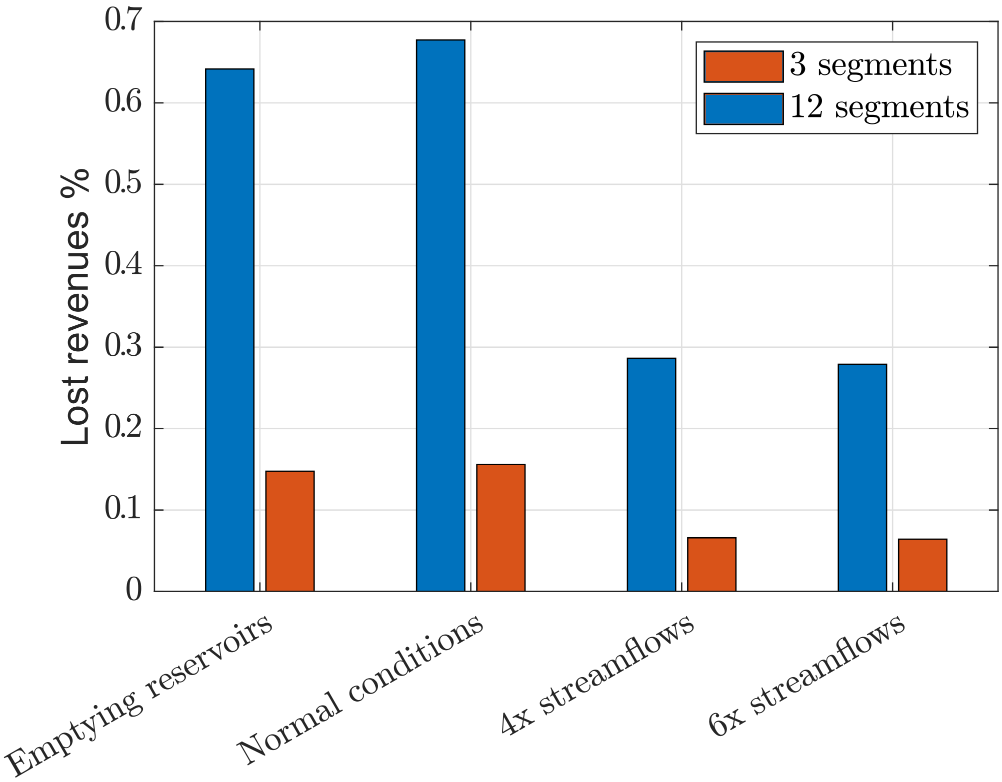


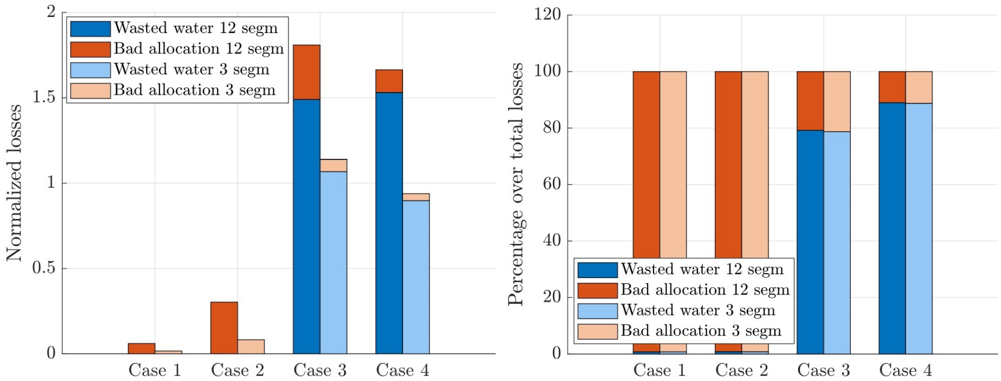
Figure 4.9: Percentage of revenue losses with respect to the global optimization

even in the worst case, only a 0.68 % reduction of the final value function is observed.

It is interesting to analyse how the total revenue loss is divided into its components (wasted water and bad allocation) for the different cases taken into consideration and for the two different optimizations divided respectively in 3 and 12 segments. Total revenue is thus decomposed according to its two main components, wasted water and bad allocation, as shown in figure 4.10.

From 4.10a be noticed that case 1 and 2 presents a very small absolute loss and it is only due to the bad allocation component, this demonstrates that in case of no augmented streamflows the segmented optimization is sufficiently flexible to avoid wasting water. Bad allocation is anyway presents and, looking at figure 4.9, can be concluded that in nominal conditions it is the main loss component in both absolute and percentage terms.

Cases 3 and 4, with augmented streamflows, have big absolute losses (see 4.10a); in particular case 3, with intermediate augmented streamflows



(a) Absolute normalized values **(b)** Percentage values over total revenue losses

Figure 4.10: Total revenues decomposition in wasted water and bad allocation values

seems to suffer the major losses, this is due to the fact that in case 4 streamflows are set so high that also the global optimization has to waste big amounts of water and consequently the differential wasted water value decreases. Figure 4.10b shows the increasing weight of water losses due to the reduced flexibility of the segmented optimization.

4.5 Effects of uncertainty on optimal results

In the previous section, where deterministic results are compared, the consistency of the new methodology has been assessed. It is now of fundamental importance to understand how uncertainty in the input models affect the optimal results, if the uncertainty of the modelled synthetic data in chapter 3 is strongly reflected on the final results, the meaning of getting a global optimal solution is reduced. Objective of this section is to set a correct value for the maximum segment length δT_{max} that allows to reduce computational costs without affecting the final outcome, considering that the problem is subject to uncertainty.

To perform this analysis Monte Carlo method is used, a single base case has been considered and different sets of synthetically generated data are used as input for the optimizations, each optimization (adopting the divide and conquer approach) is solved according to a variable segment length δT in order to analyse the differences in optimal results and computational costs. All results obtained are normalized from 0 to 1 in case of computational cost or optimal value and from 0 to 100 in case of segment length.

The base case considered is a long term optimization of the case study 1.1.3, the total length of the optimization is 7840 hours, considerably higher than in the deterministic case since with the segmented optimization computational costs are lower. The optimizations are run for 12 different lengths of time segments, respectively of: 96, 168, 216, 264, 336, 432, 528, 576, 672, 792, 888, 984, 1152, 1344, 1584, 1680 hours; segment length of 1680 hour corresponds to 100% length when normalization is applied. The 12 optimizations are run on the same set of data, for 100 different set of data in order to have sufficient values to fit a distribution.

The approach just introduced is performed in 3 main different cases: streamflows uncertainty, price uncertainty and both streamflows price uncertainty; for a total number of optimizations equal to $12 * 100 * 3 = 3600$.

In the first case only streamflow data are perturbed, according to the procedure described in chapter 3, a set of 100 different streamflows similar to the ones in figure fig. 3.9 on page 52 is generated.

In figure 4.11 the result of optimal value and computational costs, evaluated as optimization time, is reported for all the 100 optimizations and for the different time segments length. As can be clearly seen the optimization time increases with an exponential trend while the optimal value has a strong increment at the beginning and then tends to an asymptote with increasing time segment length. A peak in simulation time around 50% of maximum segment length is found, this is mainly due to the last segment, that as described in 4.2.4, can varies and if it is longer than the previous time segments causes an optimization time increase.

Box-plot in figure 4.12 represents the variance of the optimal solution final value and of computational time. As expected, computational time variance is very small since all tests are performed on the same hardware, while variance in the optimum is much larger due to uncertainties in the streamflows. Can be also noticed how, for small time segments, the variance is higher due to the lower system flexibility. From the box-plot is clear how for segment length longer than the 53% of the maximum length no effective benefit is obtained, the optimal value curve presents an asymptote which is largely inside the variance interval of the optimum solution. Execution time instead increases exponentially, thus the limitation of the segment length to the 53% of the maximum strongly reduces the computational effort, with a reduction of 60% on the execution time.

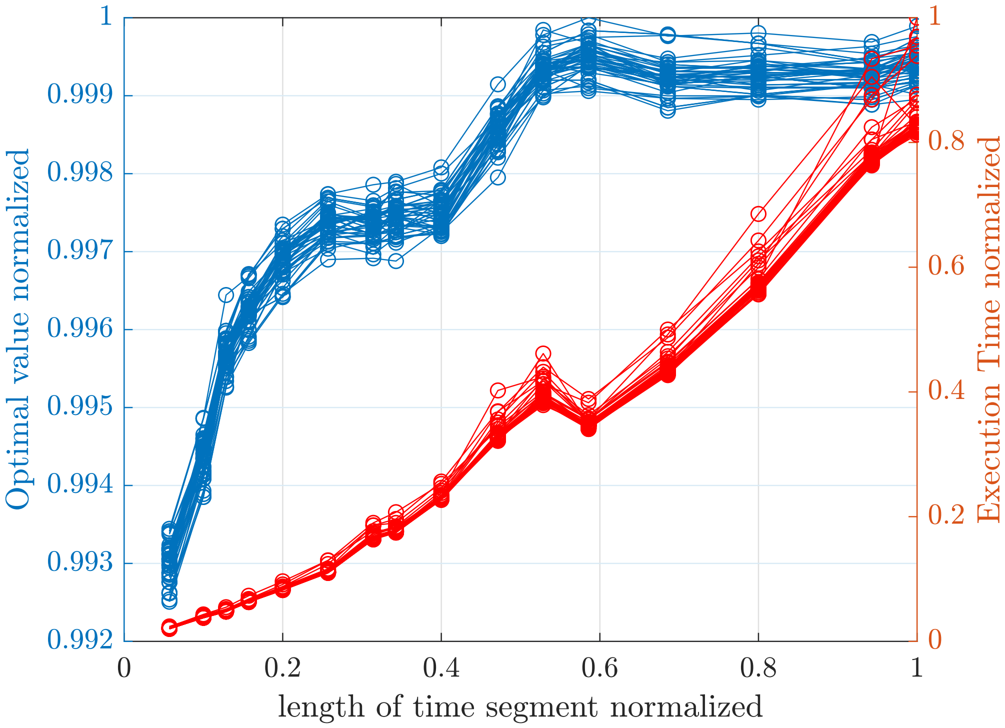


Figure 4.11: Normalized trade-off between computational costs and optimum, streamflow uncertainty

The presence of some outliers for higher segment length are mainly due to the shrink in variance of the optimal value which no more hides some simulation results that, due to the synthetic data generations, are distant from the mean value.

The lower flexibility of the optimization for smaller time segments is evidenced also in the variation of reservoirs volumes, figure 4.13 compare the probability density function of the reservoir volumes along the optimization time for a short segments optimization and a long segments one. X axis represent the time, Y axis the volume of the reservoir and the colour intensity the probability density function associated. Can be clearly seen how the volume is dispersed over a wider area for longer time segments, this is due to the fact that the optimizer has a larger space of action, the final value function is in fact characterized by a higher value and lower variance.

Introducing also the price uncertainty in the model, the final variance of the optimal solution increases as demonstrated in figure 4.14. As can be seen the incrementing trend of the optimal value is no more evident as before, when also the uncertainty in prices is introduced the optimal

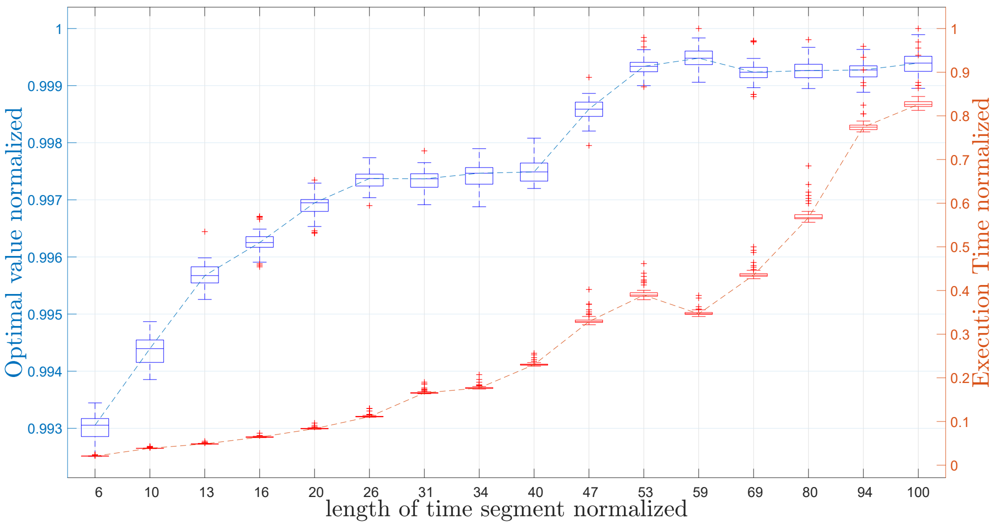
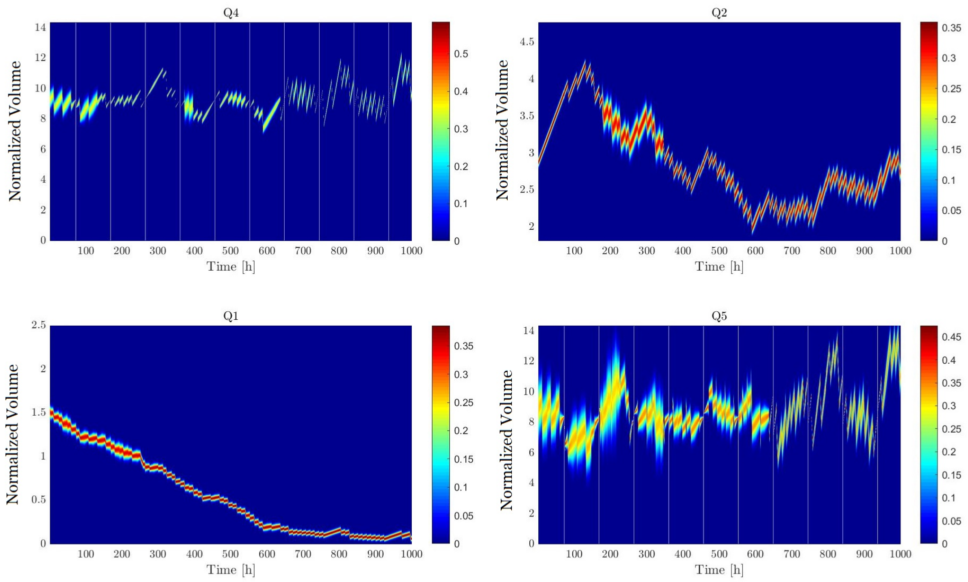


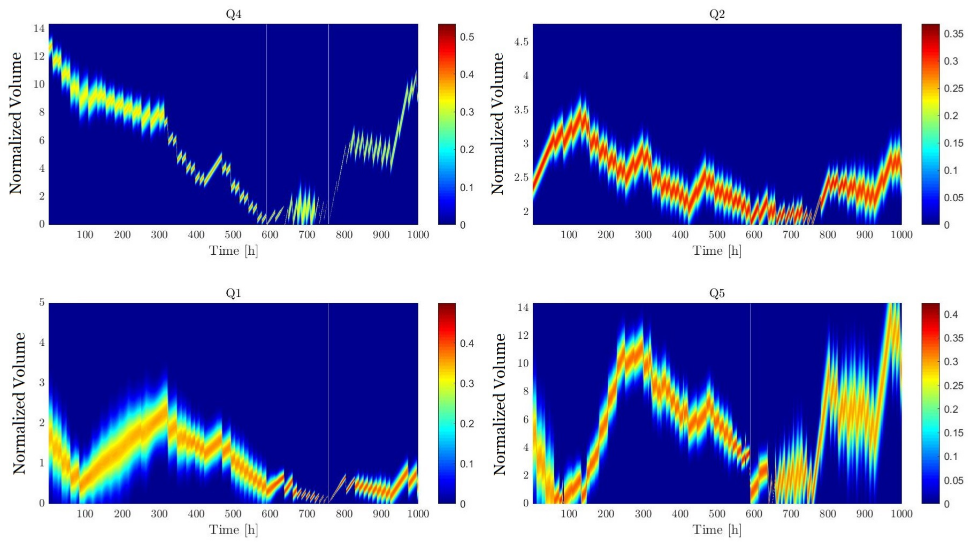
Figure 4.12: Box-plot trade-off between computational costs and optimum, streamflow uncertainty

value behaves more like a step function, with great uncertainty for short time segments optimization. Is thus fundamental to select a segment length above the 53% of the maximum length, where value function is characterized by an asymptotic behaviour and low variance, moreover computational costs are 60% less than the ones related to the maximum segment length. Selecting a length much higher than the 53% is thus not beneficial from an optimal solution point of view but it is only detrimental for what concerns computational costs.

The variance of reservoirs operation is shown in figure 4.15 for the case of minimum time segment (96h) and maximum time segment length (1680h). As shown before, the variance of reservoirs with longer time segment is higher due to increased flexibility, moreover variance in case of uncertainty on both streamflows and prices is much higher due to their combined effect. In some cases can be seen how the variance collapses to a single value, it happens in correspondence of a set-point where a seasonal reservoir is forced to have a volume equal to the one obtained from the single reservoirs optimization.



(a) 6% of the maximum segment length



(b) 100% of the maximum segment length

Figure 4.13: Volume probability density function in time

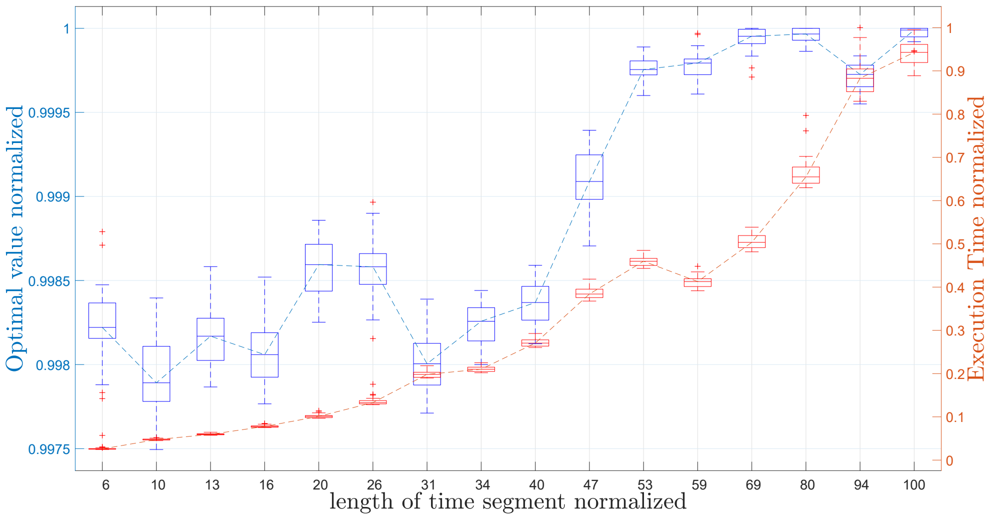
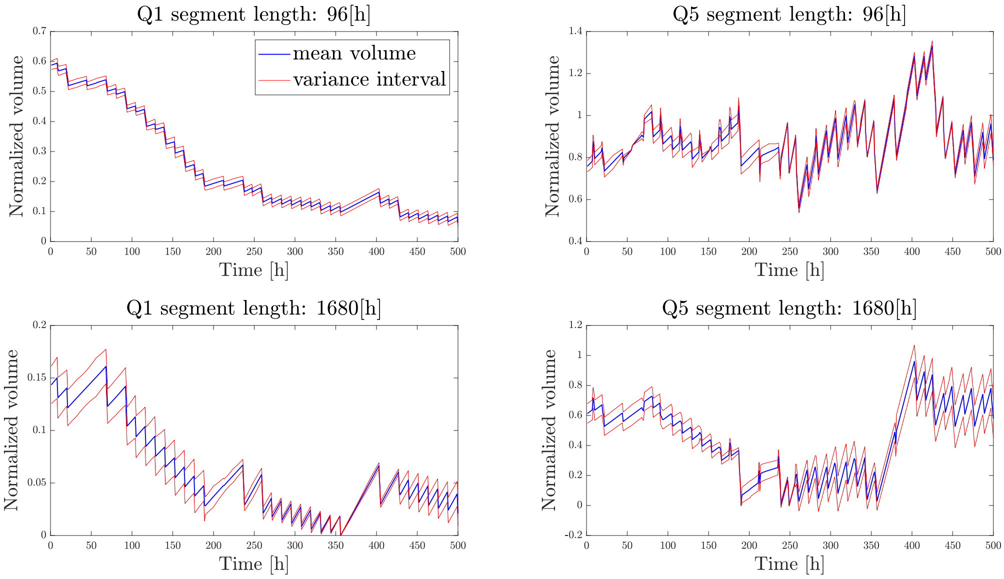
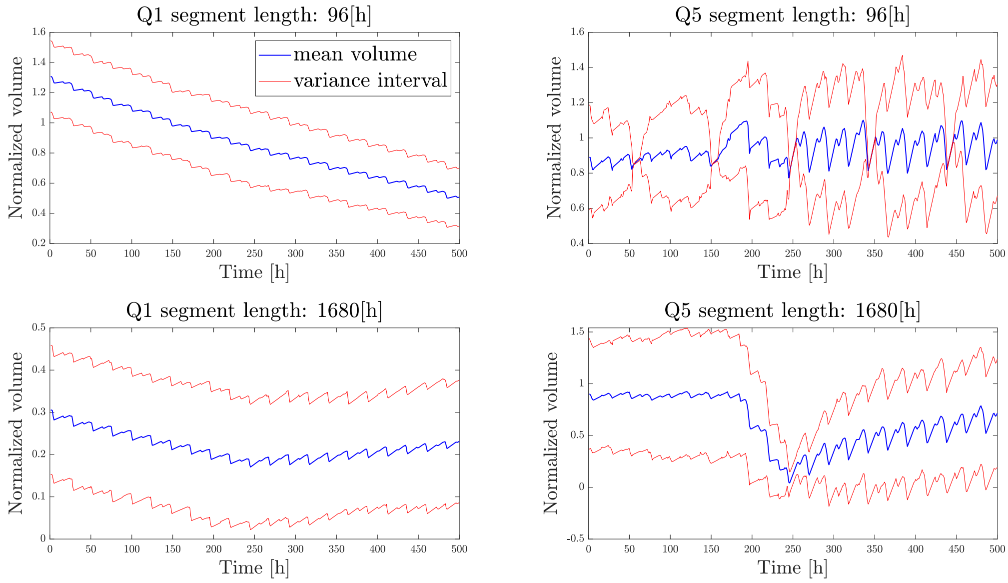


Figure 4.14: Box-plot trade-off between computational costs and optimum, streamflow and price uncertainty



(a) Uncertainty on streamflows



(b) Uncertainty on streamflows and price

Figure 4.15: Variance of reservoir operations

4.6 Discussion on final results

The approach to the problem with a divide and conquer strategy is shown to be sub-optimal with respect to the global optimization. However the results are fully consistent with the system physics and similar to the ones provided from the global optimization. Performances in all the deterministic cases considered are fully satisfactory, the optimality losses are in the order of 5% for worst case scenario and no problems related to shrinking of the possible solution space have been encountered. The computational time required for the solution of a problem with a length of 2170 hours is about 15 times lower with respect to the global optimization. For problem size orders of 1 year (8760 hours) only the segmented optimization is shown to solve the problem since global optimization results in being too demanding even for advanced hardware. Revenue losses, as expected, are higher considering a smaller optimization segment; this is mainly due to the shrinking of the possible solution space that has a negative effect on both the optimal operation of the turbines and on the system flexibility. The reduction in flexibility is clearly visible in the higher wasted water value of the 12 segments optimization with respect to the 3 segment, moreover the effects of boundary conditions strongly influence the optimum solution in the case of shorter segments

The stress tests 3 and 4 where the streamflows have been considerably increased demonstrated that, in all the optimization methods, a solution can be found thanks to the activation of emergency bypass. The revenues losses, with respect to the global optimization, are reduced from case 3 to case 4 (streamflows increased of x4 and x6 times respectively), this is due to the fact that over a certain limit the water system tends to saturate and all the results converge to the common solution of maximum power load in all the hours.

Montecarlo approach suggested that, for the case study taken into consideration, the optimal choice for the time length segments results to be the 53% of the maximum segment length, in absolute value: 793 hours, corresponding to around the 10% of the global optimization time window. Variance analysis demonstrated that no effective benefits are obtained in further increasing the segment length, however particular attention must be made in selecting values below the optimal one (53% of the maximum segment length), since the optimal value shows a step behaviour in case both price and streamflow uncertainties are introduced in the model. For longer time segments is also observed a reduction in the variance of the optimal solution due to the increased system flexibility, for the same motivation, the variance in operational range of reservoirs is instead increased. Thus for a smaller number of time steps not only the optimal value is higher but presents also a smaller uncertainty which is useful to better asses operational strategies on the long term.

The choice of dividing the problem into an higher number of smaller sub-problems is demonstrated efficient from a computational cost point of view. Global execution time cannot be measured due to the impossibility to solve the global problem, however the choice of selecting a 10% segment length of the total optimization window has brought to a 60% computational time reduction with respect to the solution of a problem with a 20% segment length of the total optimization window.

Chapter 5

Short term optimization

In the previous chapter the importance to run a long term optimization and the related strategies are described, in reality the effective operational strategy is based on a short-term vision of the system. For short term optimization is intended a time horizon of about 96 hours (4 days), electrical markets are open for a maximum of 34 hours ahead of the actual time but extra optimization time is added to reduce boundary condition effects.

Brokers are required to make multiple offers on the different electric markets more time per day, as described in sections sections 1.1.1 and 1.1.4. The auctions can have different outcomes according to the offer profile proposed and to the market conditions itself. Thus if in the long term is useful to understand, on average, which is the optimal set of strategies to arrive in a certain point, in the short term the optimization outcomes cannot be directly applied since market dynamics strongly affect the desired optimal production strategy calculated.

The desired optimization result can be completely different from an auction outcome, a showcase is reported in table 5.1. According to the market equilibrium price the total energy sold (and so the energy that must be produced) differs significantly from the optimal strategy computed.

Table 5.1: Example of electricity market auction outcome compared to optimal strategy

Hour		1	2	3
Optimal strategy				
	Optimal production MW h	**60**	**80**	**20**
	at price €/MW h	65	70	50
Offer profile				
Offer 1	energy offered MW h	10	20	5
	at price €/MW h	55	60	45
Offer 2	energy offered MW h	20	30	15
	at price €/MW h	60	65	50
Offer 3	energy offered MW h	30	40	20
	at price €/MW h	65	70	55
Offer 4	energy offered MW h	60	50	25
	at price €/MW h	70	75	65
Outcome				
	Market equilibrium price	63	75	40
	Total energy sold MW h	**30**	**140**	**0**
	at price €/MW h	63	75	-

Cell shading in the energy offered rows — accepted: Offer 1 hours 1, 2; Offer 2 hours 1, 2; Offer 3 hour 2; Offer 4 hour 2. Rejected: Offer 1 hour 3; Offer 2 hour 3; Offer 3 hours 1, 3; Offer 4 hours 1, 3.

(pink)	rejected
(green)	accepted

5.1 Objective

According to the considerations made, the short term optimization must not only be able to furnish an optimal production profile for a given deterministic case but, more importantly, should provide informations about a possible bid profile to be offered on the market. The main objective is thus to provide a tool that can help the market brokers to integrate the knowledge they have of the market with the optimal operating conditions and actual state of the system.

5.2 Assumptions

Given the intrinsic complexity of the problem and its aleatory nature, for the short term analysis different simplification have to be made. Looking at the auctions structure in figure fig. 1.3 on page 3 it can be clearly seen how the different auctions time slots are superimposed, this adds further complexity to the system since the optimal offer profile is not related only to the 4 offers bundle in a given auction but is a combination of all the offers coming from the intersecting auctions. To simplify the problem only MGP (Mercato del Giorno Prima) is take into account, since it's the biggest market in volume and the less influenced from the other markets.

An important assumption made for the analysis is that in the short time period the only uncertainty that affects the system is on the equilibrium price. Streamflow are considered to be known since a good knowledge of meteorological and soil conditions is available in the short term.

Moreover, an actor can both offer or buy energy from the electrical market, this allows to apply different trading strategies on different markets, however this kind of operations are not in the interest of this thesis, therefore the buy option will be neglected and an energy producer is assumed to be only able to sell energy over the market.

Another assumption made is that, often, each power plant is accounted as a single producer on the market, all the considerations made are about the total power production however, they could be easily extended to take into account the power plants singularly.

5.3 Identification strategy for the bidding profile

As asserted in the objectives description, the focus is to provide useful informations about the offers to be proposed in the market. Due to the intrinsic uncertainty in the results a Montecarlo method is used to evaluate the system behaviour in the short term. It's of particular interest spotting repetitions in operating patterns for different input conditions, from an operating point of view these patterns suggest that a good solution to the problem could be in its neighbourhood.

The adopted strategy can be summarized as:

(i) Synthetic electricity price time series generation

(ii) Computation of the optimal strategy for each input data

(iii) Results aggregation and analysis

Contrarily to the uncertainty analysis performed for the long-term now Montecarlo method is used to compute possible solution for the short-term problem, is thus necessary to generate a much larger set of perturbed optimizations, moreover this kind of analysis is performed several times a day by te brokers on different markets. Due to the computational effort required and the short time available to obtain the results the optimizations are run over a very short period of 96 hours, in contrast with the long-term optimization considerations. However results coming from the long term optimization are available and are used, in a similar way to what has been done for the seasonal reservoirs, to set the 96 hours ahead target volume values for the reservoirs. This is possible considering the fact that short term optimization doesn't affect the long term optimal results which can be updated on a weekly basis.

For the Montecarlo analysis the global optimization technique adopted since, due to the short time length of the problem, computational costs are low and in the short term operative phase it's of interest to reach the maximum optimum possible. Small problem means also small memory usage so parallel computing is adopted to cut down optimization times.

For the short term analysis 1000 different optimizations are carried out, the nominal operating case (*case 2*, in section 4.4.1 on page 67) of the long term optimization has been adopted as base case. Streamflows are kept fixed for all the optimizations while electricity prices are generated according to 3.2. Boundary conditions for the reservoirs volume are set as follows:

(i) $initial\_volume\_vector(r) = [max\_volume\_matrix(1, r) - min\_volume\_matrix(1, r)]/2$:

initial volume is half of maximum capacity of the reservoirs

(ii) $final\_volume\_vector(r) = volume\_matrix(96, r)$: final volume is equal to the optimal reservoirs' volume obtained from the long term optimization in the hour 96

Results of all the simulations are then aggregated, the distribution in time, price and power produced is analysed. As will be seen, the distribution of power/price in a determined hour in indicative of the offer profile that should be proposed in the market.

In all the analysis, as shown in figure 5.1, where each circle represents the optimal power production in a determined hour for the first 35 hours, circles marked in blue seems to overlap on a single value, this behaviour reflects the fact that power production for these hours is decided by a previous auction which is already closed. The blue sector thus represents the amount of energy already sold in the past closed auctions which has yet to be dispatched. Therefore in the first 35 hours the output of the optimizer is closer to a simulation rather than an optimization since constraints of minimum and maximum power production are set equal to produce the exact amount of energy sold. The simulation is necessary to adapt the first deterministic part of initial boundary conditions and energy sold with the effective optimization (in green colour).

In the first hours of a ST optimization is thus not necessary to generate an offer profile and the analysis is performed over the subsequent hours.

5.4 Interpretation of results

Contrarily to the long term optimization the single optimum result now loses its meaning, due to the high uncertainty in the auctions' outcome it is of major interest the study of aggregated results. For this reason all the results presented are in aggregated form and no particular attention was given to the single optimization.

Figure 5.1 reports all the 1000 optimization outcomes on a single plot, each circle represents the optimal power production in a determined hour, so for each hour there are 1000 circles that can overlap to each other. Red line represents the average electricity price calculated over the 1000 synthetic time series; as can be clearly seen, even with some differences,

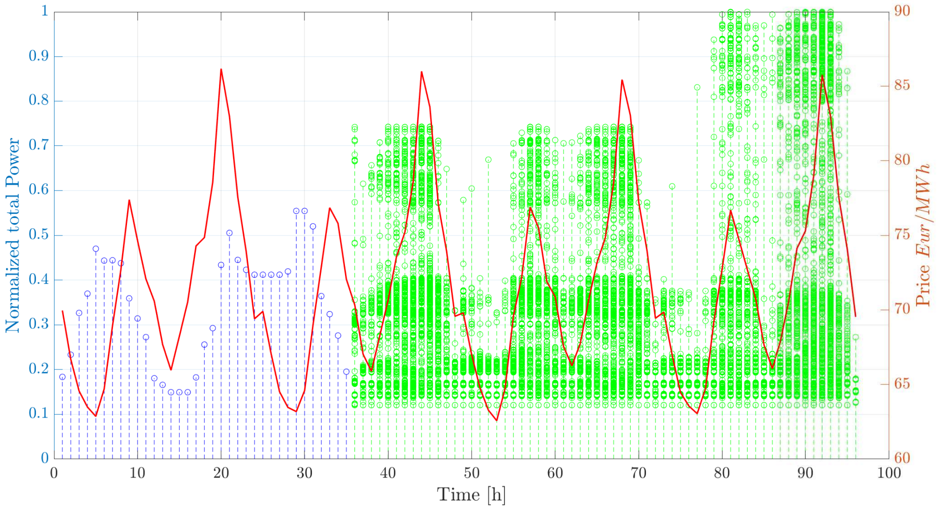


Figure 5.1: Aggregated optimization results for power production

power production in the optimized part tends to be concentrated in the period where the electricity price is higher.

Figure's group 5.2 represents how the different optimization results are distributed if a single hour is taken into consideration. Looking at figure 5.2a for example, in the x axis is reported the realized market equilibrium while on the y axis is reported the total energy produced from the overall system, every circle represents the single result of one optimization that at a given price indicates which is the optimal energy amount to produce.

The patterns and the accumulation areas that the circles generate provide useful suggestions on how to create the bidding offers profile for the hour taken into consideration. To better identify how the circles are distributed an algorithm that generate a centroid line has been developed. In figure 5.2b the same data of 5.2a are reported but with the addition of the centroid line, in red. Centroid line, as the name suggests, is representative of the optimizations' outcome's density and is calculated according to the following procedure:

(i) Electricity price distribution is divided into 20 equals segments of range:

$P_range = [\max_{pp} Price(\overline{h}, pp) - \min_{pp} Price(\overline{h}, pp)]/20$

Where $pp = 1, 2, \cdots, 1000$ is the index of the optimization while $\overline{h}$ is the the hour considered.

(ii) For each optimization *pp* the segment in which the correspondent electricity price belongs is identified

(iii) The mean of the optimal energy productions related to the optimizations that belong to the same price segment is calculated and corresponds to the Y value of the centroid for that segment

(iv) X value of the centroid is calculated,for a certain segment, as the mean value of all the electricity prices that belong to that same segment

It's important to notice that the choice to divide the x axis into 20 segments is arbitrary and has been choose considering the best compromise between centroid curve points resolution and averaging effects.

Centroid line is a useful tool to evaluate the bidding offers characteristics, as said, the offer are identified by the amount of energy to be sold and the related price; thus an offer can be represented by a point in the figure's group 5.2.

All the combinations that are below the centroid line (violet line) are considered as "conservative offers" since the same amount of energy is offered to an higher price with respect to the majority of possible outcomes and therefore the tendency is to sell energy only if a price higher that the mean of the market. On the opposite side, all offers identified by a point above the centroid line (green line) are considered as aggressive offers, since they have an higher propensity to sell energy on the market due to the fact that the offered price is below the mean electricity price for that hour.

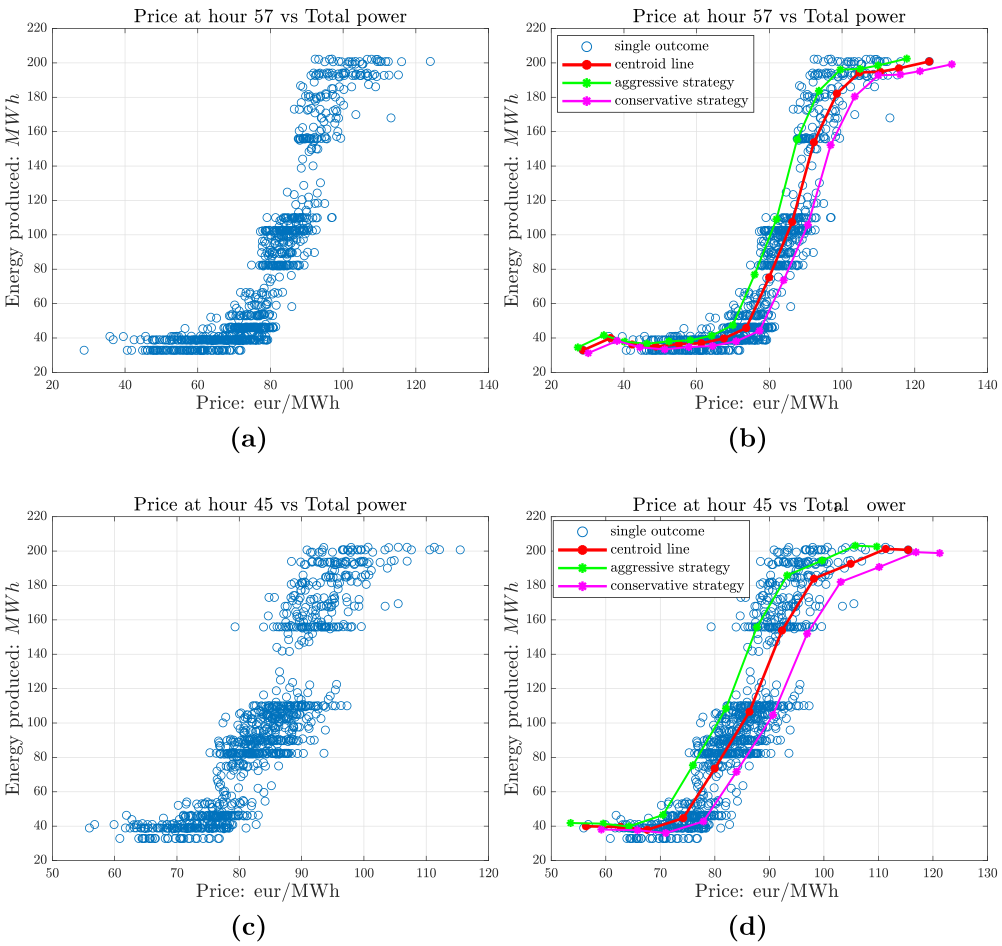


Figure 5.2: Optimal result dispersion in power produced vs price for a determined hour

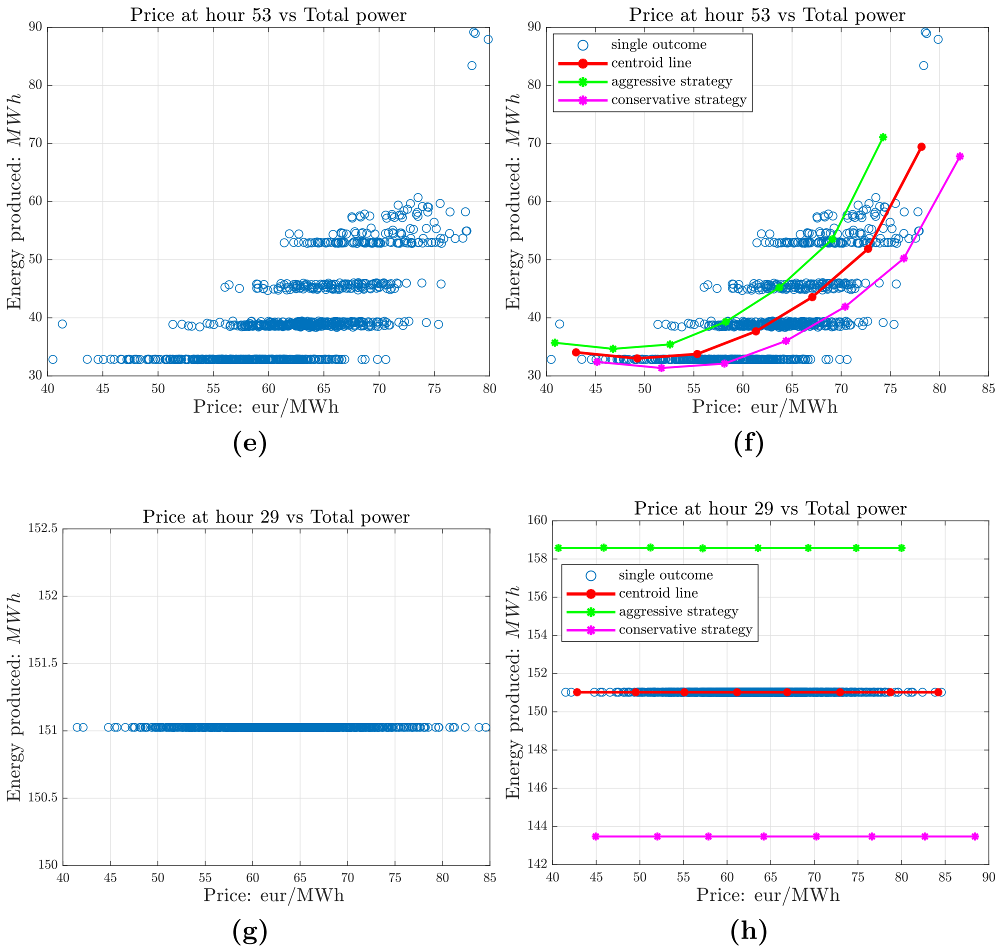


Figure 5.2: Optimal result dispersion for total power produced vs price in a determined hour

5.4.1 Examples of strategy evaluation

Figures 5.2b and 5.2d show a ”S” shape curve with an homogeneous power production distribution along increasing market price. In this case a possible offer profile could be identified from a low value of energy production, corresponding to the initial part of the ”S” curve, an intermediate value of energy offered and an high value of energy offered corresponding to the final part of the curve.

Figure 5.2e instead shows a completely different situations, a stratification of electricity prices over precise values of energy production. Due to the fact that the different aggregated ”lines” of energy production are superimposed from a price point of view there are different prices combinations that can be made keeping the energy offered fixed to those values. The offers could be for example identified by intersecting the centroid curve with the different levels of optimal energy to be produced. Since each offer correspond to a specific amount of energy, reducing the price related to that amount of energy (moving left in the graph) increases the probability to win that offer.

From the last case observed, in figure 5.2h, it is clear how, for every situation considered, the optimal solution is to produce only the determined amount of energy, disregarding the price. This happens during cases in which the price is particularly low and doesn’t justify the production of an extra amount of energy, but the minimum flow constraints (for example the minimum vital outflow of a river) forces the system to use at least a minimum amount of water that, instead of being wasted, is used to produce electricity even at very low prices.

It is of fundamental importance to notice how, in al the cases there is a minimum energy production, even for very low prices. This is due to the fact that the constraints of the system (as happens in 5.2h) forces the production of a minimum energy quantity, thus one bid in the offer profile can be composed by the identified minimum energy to be produced in any case, at a zero price. In this way there is the certainty to sell that energy; please note that offering electricity at no price doesn’t mean to giveaway free energy (see: 1.1.1), the final electricity price is anyway the one established form the market equilibrium.

An example of applied strategy can be found in table 5.2 , where all the considerations made are applied to the situations in figure 5.2

Table 5.2: Examples of applied offer profiles, with respect to figure 5.2

Hour		57	45	53	29
Offer profile					
Offer 1	energy MW h	40	40	32	151
	at price €/MW h	40	60	45	0
Offer 2	energy MW h	80	100	40	-
	at price €/MW h	80	90	63	-
Offer 3	energy MW h	160	160	45	-
	at price €/MW h	100	85	65	-
Offer 4	energy MW h	200	200	55	-
	at price €/MW h	120	105	70	-

5.4.2 Analysis of single power plants

As anticipated in the assumptions; often different power plants, even if referred to the same reservoirs system, are operated independently from a market bidding point of view. This means that for each plant is necessary to identify the optimal offer profile for the different hours, the considerations made up to now are still valid, what changes is the data aggregation level: now the different variables (the turbines) contribute to the total energy production only within the same power plant, and no more from a global system point of view.

An important consideration to be made is that the physical characteristics of the power plant, such as the installed power or the related reservoir capacity, strongly affects its operational behaviour and so the different offer profiles that are suitable for a certain plant.

From figure 5.3, where all the optimization results are grouped for each plant and displayed, it is clear how for a determined hour (hour 54) different plants behave in different ways. Power plants $P_{1,2,3,5}$ present an on/off behaviour where the optimal solution is to not produce energy or to produce at maximum load. Looking at figure fig. 1.5 on page 9, can be clearly seen how plants 1 and 5 takes water from relatively big reservoirs, thus they can modulate the output between maximum and minimum according to the electricity price. Plant 3 shows the same behaviour even though the associated reservoir Q_3 is relatively small, this is due to the fact that at the upstream of Q_3 there is plant P_1 which works with an on/off logic and forces the downstream plant to work accordingly.

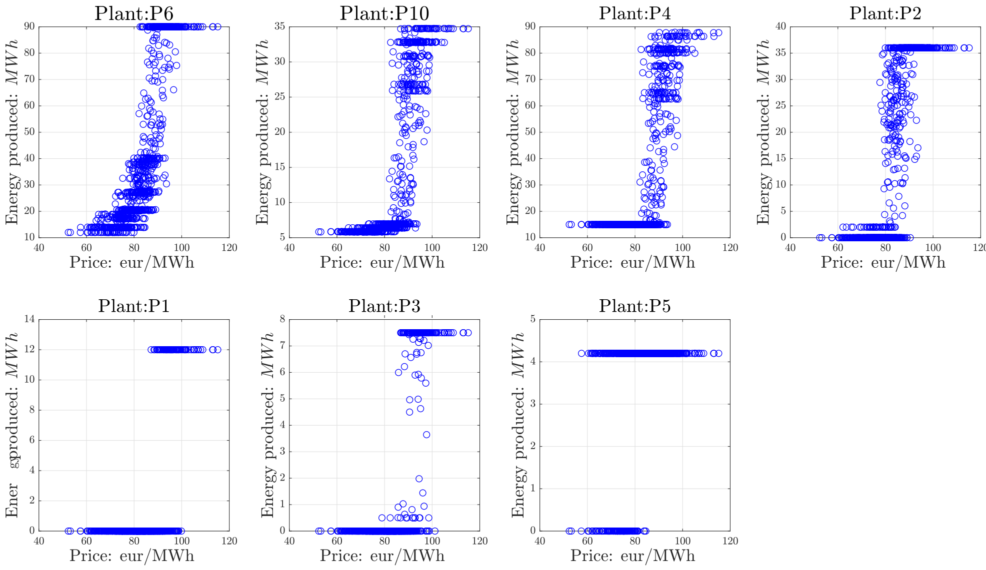
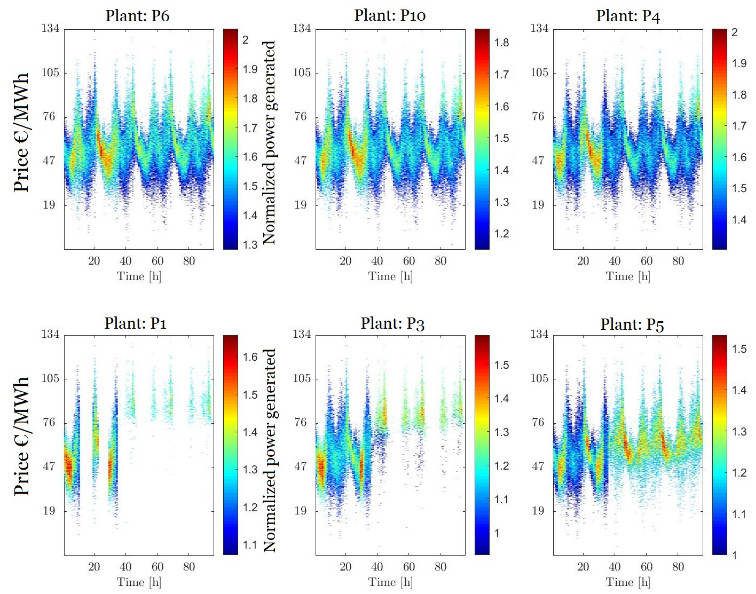


Figure 5.3: Optimal result dispersion for single plant power produced vs price for hour 54

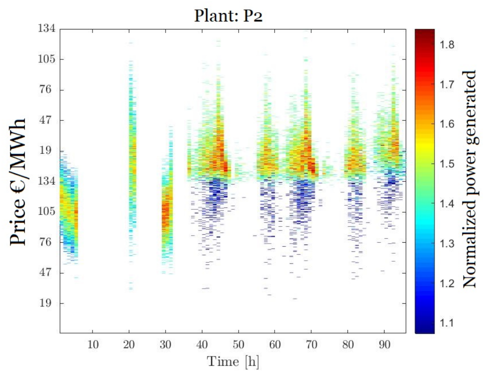
Plants $P_{4,6,10}$ instead, due to the smaller reservoir associated, have a much more varied results distribution in power, moreover these plants are downstream in the system, thus they are also subjected to the variability of the upstream power plant. Another thing to notice is that their power production never goes to zero, this is due to the fact that, discharging the water into a river, they are subjected to environmental limitations.

Figures' group 5.4 demonstrate that the typical behaviour of the different power plants persists for the overall optimization time. On the x axis is reported the timeline, on the y axis the electric energy price, while the colour-bar represents the density of energy generated for a given price at a given hour. Red colour indicates high energy density while white is associated to zero energy production.

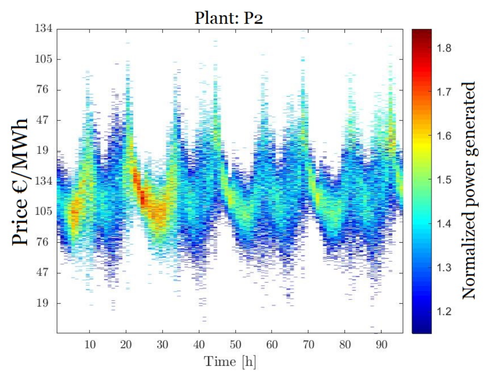
Figure 5.4b clearly shows how power plant P_2 operates according to an on/off logic since high power density is associated only at high electricity price, moreover for the hours in which the mean price is very low (between 45÷55 and 70÷80) there are significantly large white bands, meaning that no power is produced at all. Power plant $P10$ in figure 5.4c, instead, shows an opposite behaviour, with no white bands and a power production much more distributed along the price y axis. For the first 35 hours a major energy density is visible but this is due to the effect of the pre-allocated energy to be sold in the simulation period.



(a) Aggregated power plants



(b)



(c)

Figure 5.4: Power production density according to electric energy price in time

Chapter 6

Single reservoir optimization - Dynamic Programming

The optimization models introduced up to now are proven to be consistent with the physics of the model, however different simplifications have been made in order to deal with non-linear constraints and functions (see 2.5.3). In particular the linear optimization introduced up to now neglects that the energy coefficient variations is function of the reservoir height and that the efficiency is not constant but function of the turbine's power load. These considerations directly affect the value of the optimal function that has to be maximised and consequently the behaviour of the overall system. Moreover some non-linear constraints are taken into account performing arbitrarily decisions that, even if reasonable, are not directly included in the optimization process.

The optimization technique of dynamic programming (DP) is here introduced, dynamic programming allows to overcome most of the previous model's limitations. Energetic coefficient, efficiency, activation cost of the turbines, constraints that shows step functions and boolean constraints can be introduced.

Due to the powerfulness of this model and the relatively implementation easiness, combined to the fact that a global optimum is reached, in literature (see 2.2.3) different cases are demonstrated to be successfully solved with this approach. DP is, on the other side, quite computationally expensive but, as will be shown, the dimension of the state space variables can be reduced with a first attempt optimization thus reducing computational costs.

6.1 Objective

In accordance with the observations done in chapters 4 and 5 the importance of reaching the perfect optimum to a certain extend loses of meaning. However a different approach to the problem is investigated in order to understand how the simplifications affect the solution and if the Dynamic Programming could be a viable solution to solve optimal reservoirs management problems.

Since DP optimization requires a completely different approach to the problem, the objective of this chapter is to develop and apply a dynamic programming optimizer on a specific case, as proof of concept of its performances, and to evaluate the possible introduction of this new approach inside a generalized framework for the solution of a generic reservoir system, as done for the global and segmented optimizers.

6.2 Assumptions

The case study 1.1.3 considered up to now for the analysis of the model is too complex for the purposes of this chapter. A simple water system based on a single reservoir and related power plant is instead adopted, for the sake of simplicity, the power plant is considered to have only one turbine and one emergency bypass.

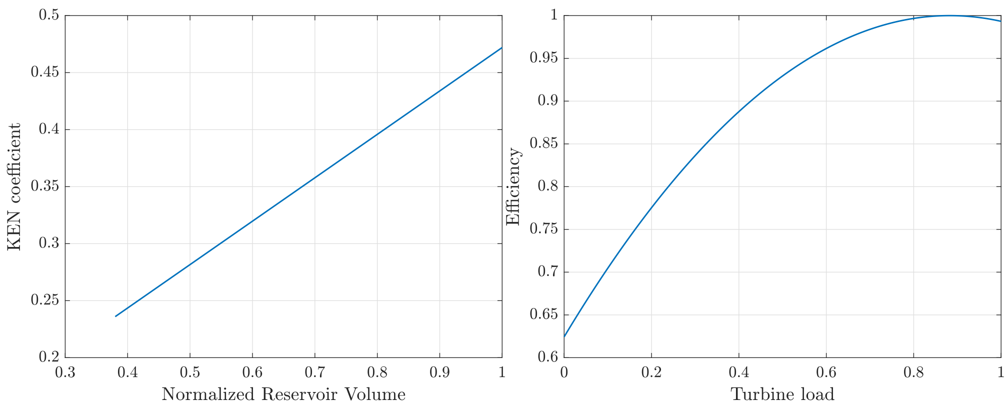
All the analysis are carried out on a short term period in order to contain computational costs, a short term optimization is anyway proved to be sufficient for the analysis needed in this chapter.

Activation cost of the turbine is introduced into the model but, for a direct results comparison with the linear optimizer, it has been neglected; also the implementation of other non-linear constraints is demonstrated but, for the same reasons, is neglected during the comparison of the two optimizations models. Thus, only the non-linearities that act directly of the final value function are considered for the analysis (the energetic coefficient and the efficiency values), since they represents the main limitations of the linear model.

6.3 Model of the problem

The case study taken into consideration is a single reservoir single plant water system, it can be considered as a portion of the main case study (1.1.3) since reservoir's and power plant's data used are the one related to Q_2 and P_2 respectively, main data are reported in table 1.2 on page 10.

In addition to the already existing data that characterize the system, now energetic coefficient and efficiency of the turbine are introduced. For the sake of simplicity the energetic coefficient is expressed as a linear function of the reservoir volume but any other different function could have been implemented. For what concerns the turbine efficiency instead, it is expressed by a polynomial curve which is function of the power load. No real data about turbine's efficiency are available, so a typical efficiency curve for a Francis turbine is taken from Sangal, Garg, and Kumar in [27] and normalized to fit maximum and minimum power load of the considered turbine. Ken and efficiency curves normalized are shown in figure 6.1. Boundary conditions for the reservoir's volumes are set to half of the maximum capacity for bot the initial and final target volumes. Streamflow introduced in the reservoir is equal to Q_2 streamflow in the nominal case of figure fig. 4.3 on page 67, also electricity prices are taken from the same nominal case.



(a) *ken* curve **(b)** Efficiency curve

Figure 6.1: Energetic coefficient and efficiency normalized curves

6.4 Adopted methodology

Dynamic optimization is implemented following a completely different approach with respect to the linear optimization developed in the previous chapters. The first big difference is that the model is now seen as a discrete dynamic systems that evolves in time, the state of the system is represented by the water volume contained in the reservoir that evolves from one time step to another according to the control variable which is the combined outflow of turbine and bypass. For each time interval the cost(gain) of going from one state to another is evaluated by calculating the necessary control action and evaluating the relative cost.

It's important to notice that the problem formulation adopted is the dual of the classic linear formulation used up to now. In the previous case the optimization algorithm seeks for the optimal combination of control variables to maximize revenues and defines as consequence the state of the system. With DP the optimal combination of states of the system is assessed and, as consequence, the control variables are defined.

6.4.1 Dynamic programming generalities

Dynamic programming technique is based on the Principle of Optimality developed by Richard Bellman (1949), which states that "If the optimal solution for a problem passes through an intermediate point X_1, T_1 , then the optimal solution to the same problem starting at X_1, T_1 must be the continuation of the same path". This principle allows to solve a discrete dynamic optimization problem (with known final conditions) starting from the end and, thanks to backward calculations, finding the optimal states path. The method, synthetically described, allows for an efficient solution of dynamic problems, the most known example in literature is the "travelling salesman problem", (see [33]).

In order to apply the optimality principle it is necessary to discretize both time and space, this is an important consideration since it means that state variable (i.e. the water level in the reservoir) must be also discretized; in the linear optimization this is not necessary, only time discretization was needed.

Once the system is dicretized in time and space its possible evolutions can be represented by the grid in figure 6.2, representing all possible states in the different time instants. Evaluating all options with forward calculation would require the computations of all the possible combinations, the principle of optimality is thus of fundamental importance to reduce the computational cost. When the cost function is evaluated backwards, the computation of all the possible combination is not required but the final optimum is exactly the same to the one calculated with an Exhaustive research. If the problem is thus dicretized on the overall state space the principle of optimality represents a global optimization technique.

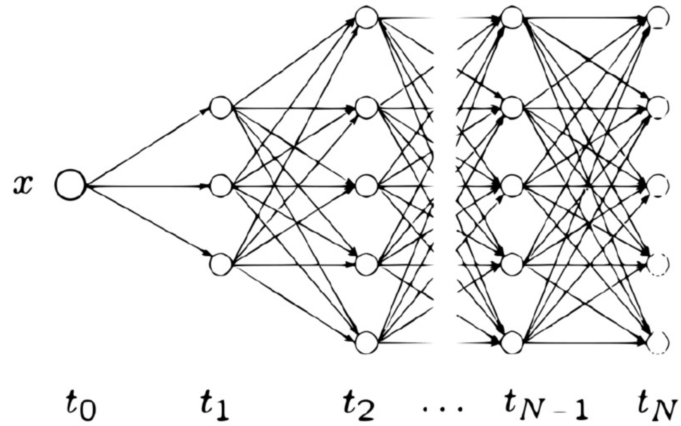


Figure 6.2: Example of system evolution grid

6.4.2 Model of the problem

A generic dynamic programming problem can be formulated as follows:

Minimize:

$$J = \int_{t_0}^{t_1} f\left(\mathbf{x}(t), \mathbf{u}(t), t\right) \mathrm{d}t \tag{6.1}$$

Subject to:

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}\left(\mathbf{x}(t), \mathbf{u}(t), t\right) \\ X(t_0) = X_0 \\ X(t_{end}) = X_{end} \end{cases} \tag{6.2}$$

It is important to notice that, differently from before, X is a vector that contains the states of the system for a determined time step. J is the cost(revenue) function that is minimized(maximized), it is determined by the function $f(\mathbf{x}(t), \mathbf{u}(t), t)$ which describes its marginal increment in time. Function g instead describes the evolution dynamic of the system, while $u(t)$ is the control action.

In order to apply the optimality principle it is necessary to discretize both time and space, time discretization follows the hourly segmentation already described in 2.5. For the space discretization (*state* X), instead, different techniques have been inspected and evaluated in the following sections.

The discretized problem, with direct reference to the case study considered, is described by the minimisation of:

$$J(X(t), u(t), u) = \sum_{t=1}^{T} Price(t) \cdot u_1(t) \cdot ken(X) \cdot \eta(u_1(t)) \tag{6.3}$$

Subject to:

$$\begin{cases} X(t+1) = X(t) + A(t) - u_1(t) - u_2(t) \\ X(t_0) = X_0 \\ X(t_{end}) = X_{end} \end{cases} \tag{6.4}$$

Considering the case study taken into account X is the state variable that describes volume of the reservoir at time t (previously has been defined as $Q_i(t)$), $u_1(t)$ is the control variable related to the turbine's flow, while $u_2(t)$ refers to the outflow due to the bypass control variable. $A(t)$ is the streamflow of water coming into the reservoir, while X_0 and X_f are the final target volumes.

As can be noticed, the discretization of 6.1 into 6.3 results in an equation very similar to the functional eq. (1.10) on page 8, with the only difference that now a single turbine is considered. If the cost function J is calculated in backward direction (the sum index i goes from T_{end} to 1) takes the name of *residual cost function* J^*. The system dynamics in 6.4 is the dual of the water volumes balance equation in eq. (1.1) on page 6, with the difference that no upstream flows are considered to discharge water into the reservoir.

Constraints eqs. (1.2) to (1.7) can be easily implemented into the model since due to the nature of DP both the state $X(t)$ and the control variables $\overline{u}(t)$ are directly accessible during the optimization process. In particular, direct limitations on the state space and on the control variable space can be imposed:

$$X_{min}(t) \leq X(t) \leq X_{max}(t) \qquad \text{max and min volumes} \tag{6.5}$$

$$u_{1,min}(t) \leq u_1(t) \leq u_{1,max}(t) \qquad \text{max and min turbine flow} \tag{6.6}$$

$$u_{2,min}(t) \leq u_2(t) \leq u_{2,max}(t) \qquad \text{max and min bypass flow} \tag{6.7}$$

$$|X(t+1) - X(t)| \leq \delta X_{max}(X(t), t) \qquad \text{max volume change rate} \tag{6.8}$$

Inequality 6.8 can be now easily implemented even for step functions of $\delta X_{max}(X(t), t)$, since the state $X(t)$ is an explicit value in the DP optimization. As proof of concept, also strongly non-linear constraints such as turbine's activation cost or minimum idle power are implemented into the model and described in the further sections.

6.4.3 Dynamic programming algorithm implementation

A Matlab algorithm has been developed to solve the DP problem, which is built according to the following steps:

(i) Calculation of the cost increment at each step for each possible state transition

(ii) Backward evaluation of the optimal residual cost function J^*

(iii) Forward evaluation of the state's path correspondent to the optimal residual cost function

cost increment calculation

In a determined time step, accordingly to the state transition considered the cost function undergoes to an increment/decrement, the increment mus be calculated for each state combination at each time step. With reference to the figure 6.3, where each circle represents a possible value of the state X of the system, the cost of going from state $x^a(t_k)$ to state $x^b(t_{k+1})$ is defined by the following equation.

$$cost\_matrix(k, a, k+1, b) = opt\_alloc\_func(x^a(t_k), x^b(t_{k+1}), t) \quad (6.9)$$

Where $cost\_matrix$ is a four dimensional matrix containing all the costs of switching from one state to another with a size of $[T_{hours} N_{states} T_{hours} N_{states}]$, while the optimal allocation function defines the effective cost value and the correspondent control action to be applied for the state switch.

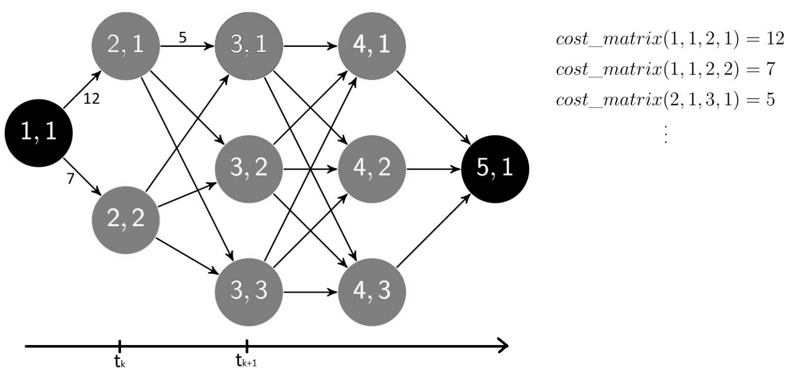


Figure 6.3: Example of cost function composition matrix

Evaluation of the cost matrix is performed by the algorithm in listing A.3 on page 128, only the core of the matrix composition algorithm is shown.

residual cost function evaluation

Once that all the cost increments have been evaluated the optimal residual cost function is calculated backward for each time step, according to the formula:

$$J^*(x_k^a, t_k) = \min_{x_{k+1}^b}[cost\_matrix(k, a, k+1, b) + J^*(x_{k+1}^b)]$$

The portion of algorithm that perform the backward computation can be found in listing A.4 on page 129. It is important to notice that computing the residual cost function in backward direction, thanks to the optimality principle, a lot of computational cost is saved. The overall states combinations that needs to be evaluated are $T_{hours} * N_{states}^2$ which is only a small number with respect to a comprehensive calculations of all the possible states combination in time, that would have required $N_{states}^{T_{hours}}$ computations.

optimal state's path calculation

Although the optimal residual cost function has been calculated, the algorithm is still incomplete since the minimum cost is known but the states through which the optimal solution passes through are yet to be identified. Starting now from the beginning the cost of going from time t_k to each of the states at t_{k+1} is added to the cost function (which is set to have a starting value equal to zero at t_{start}). The state corresponding to the cost increment that minimize the function $J(t_{k+1})$ is chosen as optimal state for t_{k+1}. The procedure is expressed by the equation:

$$\text{find:} \quad x_{k+1}^i \quad \text{that:} \quad \min_{x_{k+1}^i}[J(tk) + cost\_matrix(k, a, k+1, i)]$$

and reported, more in detail, in listing A.5 on page 129. Please note that the objective is the minimization of the function since, for resolution methods, the revenues are defined to have a negative value.

6.4.4 Discretization of the state variables space

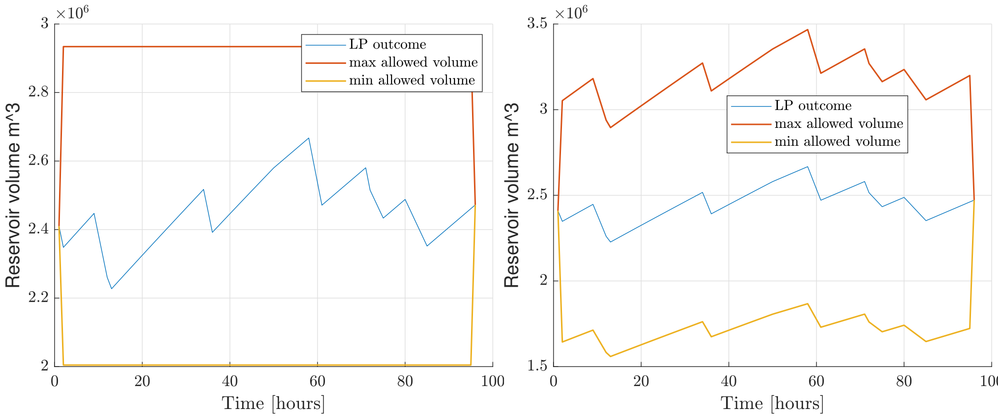
In order to apply DP, the problem has to be discretized both in space and time, in the case of time discretization, as done up to now, hourly segments are used. Discretization in space is much more critical for two main motivations: it directly affects the convergence of the solution to an optimal results and, according to the calculations required by DP ($T_{hours} * N_{states}^2$), quadratically affects the computational costs.

Discretization level is strongly determined by the system taken into consideration, in particular the ratio between reservoir capacity and maximum turbine flow. In order for the DP to work properly the maximum discretization interval of the reservoir should be at least one order of magnitude smaller than the maximum turbine flow. In this way, from one time step to another, the state space is sufficiently dense to allow a fine regulation of the turbine outflow. Thus the constraint on the minimum discretization level can be written as:

$$\overline{N}_{min} \geq N_{states} \text{ so that } \frac{X_{max} - X_{min}}{N_{states}} \leq \frac{u_{1,max}}{10}$$

In the case study considered the reservoir capacity is 63 times the maximum turbine flow. Thus the discretization level should at least be $\overline{N}_{min} \geq 630$, which is demanding from a computational point of view.

In order to save computational cost the discretization on the whole state space is not performed but two different strategies are introduced. Since the linear model, even with all the limitations introduced, returns a solution which is in the neighbourhood of the optimal one; it is used to perform a first trial solution in order to determine a base water level profile, on this profile are then constructed two tolerance regions in which the water level can varies in the subsequent DP optimization. With this exploit the result optimality has not been compromised, but the computational power required is greatly diminished since now the interval of variation $X_{max} - X_{min}$ of the water volume is reduced. In particular two different strategies have been tested, the first one, in figure 6.4a , takes the minimum and maximum water level measured during the linear optimization and creates a constant minimum/maximum limit. The second strategy, in figure 6.4b , is more advanced since maximum and minimum allowable volume curves are made to follow the linear optimization result.



(a) Fixed min/max variation **(b)** Dynamic min/max variation

Figure 6.4: Volume discretization strategies for DP

6.4.5 Optimal allocation function logic

The passage from one state to another is not straightforward since the control action u is composed by two variables, the turbine flow u_1 and the bypass flow u_2. Being the control action of dimension 2 and the dynamic equation 6.2 of dimensionality 1, there could be infinite possible solutions to the problem. Moreover, the optimal allocation logic must take into account also for the constraints and for the unfeasible state transitions.

basic logic introduction

Due to the fact that water could be allocated both to the turbine or to the bypass flow, a base logic to maximize the profits is needed. In the transition from state x_k^a to state x_{k+1}^b , the amount of water to be discharged from the reservoir is calculated as:

$$\Delta X(t_k) = x_k^a + A(t) - x_{k+1}^b \qquad \text{(volume reduction if positive)}$$

According to the different values that ΔX can assume the strategy is modified as consequence:

(i) $\Delta X < 0$: this means that the target volume to be reached in the state transition is too high with respect to the initial state volume and input streamflows. No control action is set and the revenues coming from transition are set to -10^9.

(ii) $0 \leq \Delta X < u_{1,min}$: the water to be discharged from the reservoir is positive, but not enough to reach the minimum idle load of the turbine, so water is all routed to the bypass.

$$\begin{cases} u_1 = 0 \\ u_2 = \Delta X \\ revenues = 0 \end{cases}$$

(iii) $u_{1,min} \leq \Delta X \leq u_{1,max}$: the water to be discharged is in the operational range of the turbine, thus the flow is entirely allocated to it.

$$\begin{cases} u_1 = \Delta X \\ u_2 = 0 \\ revenues = u_1 \cdot ken((x^b_{k+1} + x^a_k)/2) \cdot \eta(\Delta X) \end{cases}$$

(iv) $u_{1,max} < \Delta X \leq (u_{1,max} + u_{2,max})$: the water to be discharged is above the operational range of the turbine, turbine load is saturated and the remaining water discharged from the bypass.

$$\begin{cases} u_1 = u_{1,max} \\ u_2 = \Delta X - u_{1,max} \\ revenues = u_1 \cdot ken((x^b_{k+1} + x^a_k)/2) \cdot \eta(\Delta X) \end{cases}$$

(v) $\Delta X > (u_{1,max} + u_{2,max})$: the water to be discharged is above the maximum combined outflow of both bypass and turbine, the transition is thus unfeasible.

$$\begin{cases} u_1 = 0 \\ u_2 = 0 \\ revenues = -10^9 \end{cases}$$

As can be noticed the evaluation of energy coefficient and of turbine efficiency now correctly affect the value function to be optimized. In the case of an unfeasible transition instead, a high and negative value of revenues is set to avoid considering the transition as optimal during the residual cost function evaluation. The algorithm used to evaluate the optimal allocation is reported in listing A.6 on page 130.

non-linear constraint introduction

The optimal policy to pass from one state to another can be coded according to the requirements of the system, it is possible to implement non-linear functions and boolean constraints in an easy way since, for each transition the initial and the target state of the system is known.

Maximum change rate in reservoir height or volume according to a step function related to reservoir capacity is introduced as:

$$\Delta X \leq \Delta X_{max}(t_k)$$

If the constraint is not respected, as in the other cases, revenues value is set to -10^9. The constraint, which was set to the most restrictive value in the linear optimizer, is no more approximated but follows the exactly imposed step function.

During the basic logic introduction is already kept into account the constraint of minimum turbine load, which was among the most problematic constraints in the linear optimization.

Maximum number of turbine activations is substituted by the turbine's cost of activation since, as described in 1.1.3, it is not a hard constraint but just a preference and the cost of activation introduction is better suited for the real system operation. Cost of activation is added during the backward evaluation of the residual cost function, starting from the end the count number of activations has been substituted with the number of deactivations, which is the same.

A deactivation is defined as:

$$\text{deactivation count } +1 \text{ if:} \quad u_1(t_k) > 0 \wedge u_1(t_{k+1}) = 0$$

The deactivation cost is immediately added to the residual cost function in the transition from one state to the other:

$$J^*(x_k^a, t_k) = J^*(x_{k+1}^b) + cost\_matrix(k, a, k+1, b) + act\_cost$$

Please note that for the comparison with the linear optimizer, the activation cost is set to zero in order to not introduce a bias in the estimation of final revenues.

6.5 Limitations of the model

Although DP optimization method seems to perform better for what concerns the real system modelling, it is not free from limitations.

The main concern is about the computational effort required. The model scales linearly in time ($T_{hours} * N_{states}^2$) and, for long term optimizations, has a relative advantage over the linear optimization method that scales quadratically. However this advantage is maintained only in the case of a single reservoir systems, for multiple-reservoirs system the number of states combinations increases exponentially according to:

$$(N_{states,single\,reservoir})^{2\cdot I_{reservoirs}}$$

thus the problem scales exponentially with the number of reservoirs. This makes computationally heavy to optimize a big water system such as the case study in section 1.1.3 on page 8. Another factor that strongly affects the computational costs is the presence of concentration times, to optimize systems in which the water takes more than one hour to flow from upstream to downstream reservoir is necessary to include inside all the possible state transitions not only the ones related to states X_k and X_{k+1} but also the effect of states X_{k-c}, where c indicates the concentration time value. The introduction of states related to earlier time lags exponentially increases the possible combinations. The final number of incremental costs to be calculated for each time step in case of multi reservoir system with concentration time is thus:

$$(N_{states,single\,reservoir})^{2\cdot(I_{reservoirs}+\sum_r c(r))} \tag{6.10}$$

Where $\sum_r c(r))$ is the sum of the concentration time related to the different reservoirs.

As can be noticed, shifting to the complementary problem (from the optimization on the control variables to the optimization on the states path) the problem computational cost has been shifted from the time dimension to the space dimension. With the new approach the problem has less limitations in maximum optimization time but it grows exponentially in function of the system complexity.

6.6 Analysis of the performances

All the analysis are performed on the single reservoir - single power plant taken into consideration, optimizations are run over a time period of 100 hours. Comparison of results with the linear global optimization model is performed considering a nominal streamflow and boundary conditions are set for initial and target volumes as half of the maximum reservoir capacity.

Before analysing the performance of the DP optimization an assessment on the optimal choice of the number of discretized states is needed. Figure 6.5 shows the trade-off between optimal value and computational costs in function of the discretization density. According to the observations made

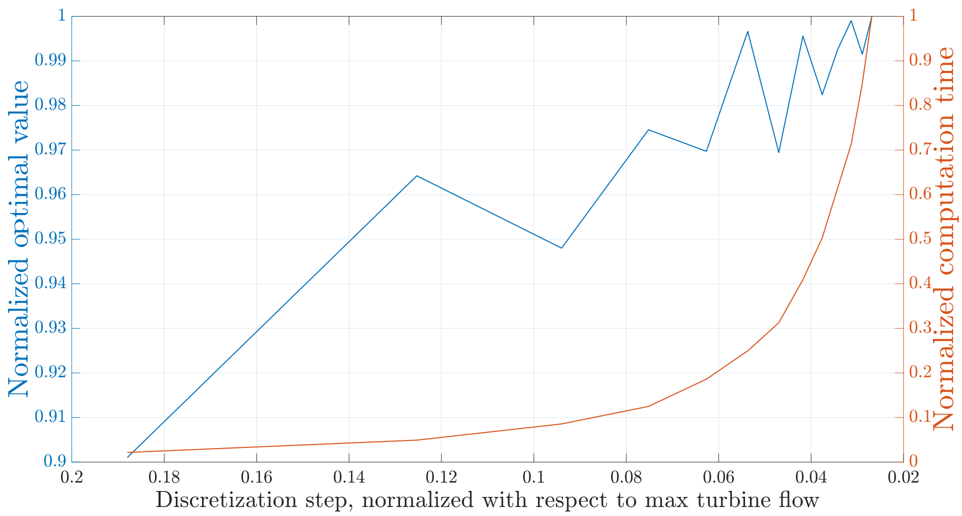


Figure 6.5: Normalized trade-off between computational costs and optimal value

in 6.4.4, the dimension of the segmented reservoir space is normalized by dividing the segmented capacity for the maximum flow allowed by the turbine. As can be seen, the optimal value has not reached a stable asymptotic behaviour for the segmentation density considered, but considering the very high exponential rate of the computational time needed a discretization level of 6% of the maximum turbine load is selected, this choice corresponds to a discretization of the state space into 300 segments. The state space variation method used is the one related to figure 6.4a since implies a constant discretization range.

It's important to notice how the computational cost has a similar behaviour with respect to fig. 4.11 on page 82, both the curves increase with a quadratic trend but, in this case, to obtain a satisfactory result an higher increment in the discretization level is needed compared to the increment in time length segment of the previous case.

The performances of the two discretization methods in figure 6.4 are compared considering variations in the number of state space levels and in maximum range of variation. Optimization outcomes are summarized in table 6.1, from the results can be seen how the dynamic variation around the first attempt optimization computed with the linear optimizer performs generally better that the fixed variation range. In line with this consideration, further optimizations are run with a 300 values state discretization and adopting a dynamic reservoir volume variation range of 30%.

Table 6.1: Segmentation method performance comparison

number of states	200			300		
variation range +/- %	10	30	100	10	30	100
normalized value - static variation	0.94	0.96	0.93	0.97	0.99	0.96
normalized value - dynamic variation	0.95	0.98	0.95	0.96	**1**	0.99

Once determined the discretization levels and strategy for the case study taken in consideration, it is of interest to determine to what extend the introduction of non-linear functions in the optimization process makes the results differs from the approximated linear optimization model. In particular the results presented below are affected by a correct estimation of energy coefficient and efficiency, moreover the allocation of power to respect the minimal turbine flow is varied in the optimization model instead of being decided arbitrarily.

In figure 6.6 are reported the optimization results for both the linear optimizer and the dynamic programmer. As can be immediately noticed from 6.6b the DP optimization results are about 20% higher than the LP approach. Better performances are mainly due to the fact that dynamic programming considers the energetic coefficient *ken* as variable with respect to the reservoir's water level, this can be clearly seen in figure 6.6a, where the optimal level of the reservoir in DP is made to increase as soon as possible in the first part of the optimization and then decreased to respect final target volume only in the final optimization period. The same behaviour can be observed also in 6.6c where, contrarily to the LP turbine flow, DP optimization keeps the turbine deactivated for the initial period to make the reservoir increasing in level, even in time period in which the electricity price is high. Water is then discharged in the latter period with an higher intensity even if prices are not particularly competitive since, due to the higher reservoir level, the energetic value of the water is higher.

Correct selection of the solution space is demonstrated for this problem by looking at 6.6a, the reservoir volume for the DP optimization is well defined within the boundaries, so no need of an increment in the solution space is needed.

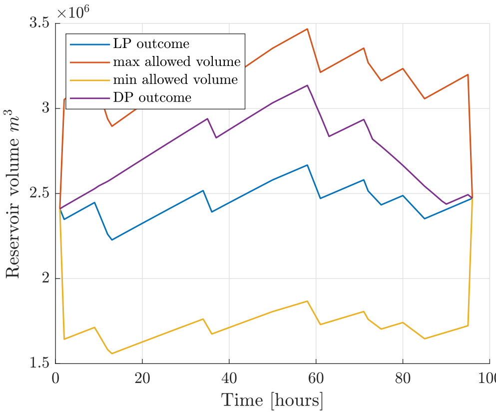
Consistency of the model is demonstrated by figure 6.6c, all the constraints of minimum and maximum turbine flow are respected, moreover the optimal solution tends to modulate the power production according to the on/off behaviour in proximity of electricity price peaks, as already seen from linear optimization. Small residual values of turbine and bypass flows are present on the whole optimization time, even if the minimum turbine flow is respected they are to be considered as numerical errors introduced by the state discretization. An more dense discretization would reallocate the small residual production near the already existing flow peaks, this is the main explanation of the optimal solution increasing trend seen in 6.5.

Effects of efficiency value implementation on the final optimum are considered to be negligible since LP optimization mainly works with an on/off logic which tends to mitigate the efficiency losses caused by working conditions of the turbine far from the nominal ones.

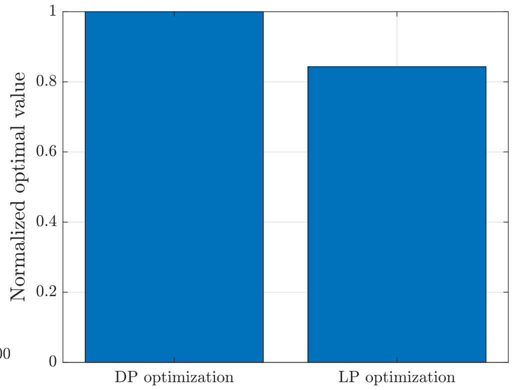
With reference to figure 6.7, the effect of energy coefficient function on the reservoir level has been investigated, optimization results are plotted on the same graph for different *ken* curves. It can be clearly notice how, for an increasing slope of the curve in figure 6.1a, the reservoir optimal water level tends to increase in the initial part due to a postponement of the production in the final optimization part, this is due to the higher weight that *ken* value assumes with a steeper curve.

Another advantage of DP is that a grid, containing the optimal solution for all the possible states starting from a time t_k, shown in figure 6.8, is implicitly calculated by the algorithm. Each node represents a state discretization value, its value is proportional to the y height of the point, state values are organized in columns according to the different time steps on x axis. The arrow that connects each state value to the next value in time represents the optimal path, if no arrows are present the solution is unfeasible starting from the correspondent state.

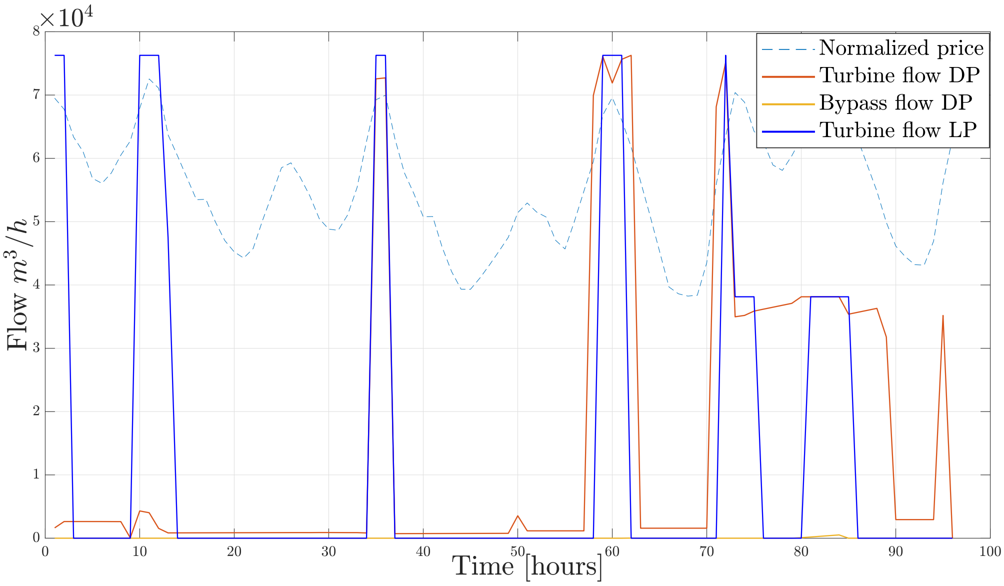
This graph is very useful since it shows how certain states, marked in red, represent to some extend a waypoint in which most of the optimal paths tend to pass trough. Moreover, in case of system uncertainty, if the real state in a future time step won't coincide with the expected one the optimal solution with respect to the new real state is already available.



(a) Reservoir level comparison



(b) Normalized optimal value



(c) Decisional variables flow

Figure 6.6: Comparison between LP and DP optimization

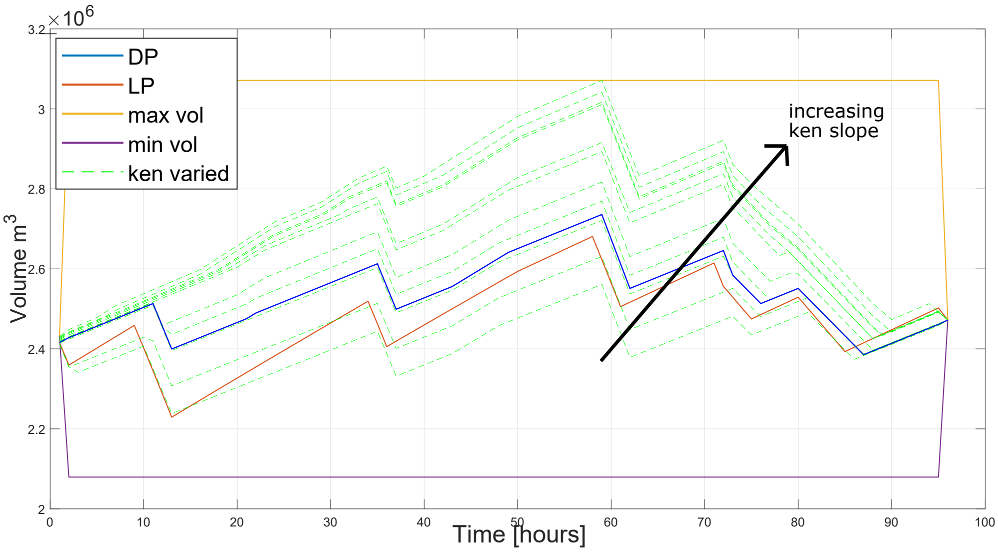


Figure 6.7: Optimal reservoirs level in function of increasing *ken* slope

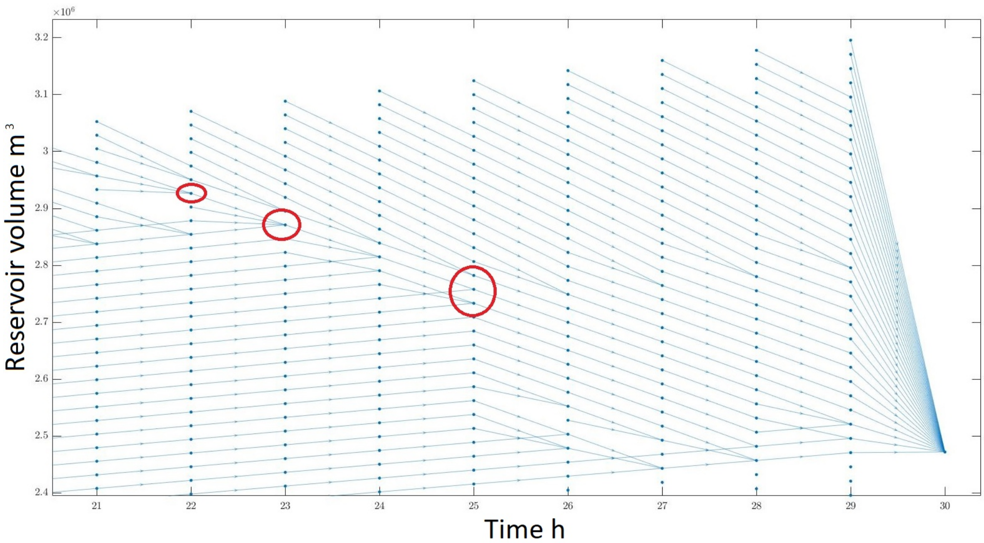


Figure 6.8: State space - Time optimal path grid

6.7 Discussion on final results

Dynamic programming optimization is demonstrated to perform considerably better with respect to the previous linear model in cases in which the problem presents a small (or a singular) number of reservoirs. The possibility to introduce non linear functions and constraints improves the optimal result of about 20%, moreover for long term optimizations the problem complexity scales linearly and not quadratically as in the first optimization approach, the possibility to parallelize the computation of different states further increases the computational advantage on long term problems.

The discretization of the state space has been turned out to be a fundamental choice for the effective functionality of the DP model. A too wide discretization space would require an excessive number of states resulting in a computationally demanding optimization. If a narrow state space is instead selected, the solution could converge to an optimum which is not the global one; if the space instead is not sufficiently discretized the solution does not converges at all. The segment dimensions have been expressed in function of the nominal turbine flow, which has been demonstrated to be an important index for the selection of the discretization level. To solve the discretization problem an efficient two steps method, based on an initial linear optimization, is used effectively. The optimal volume range, selected for the case study considered, is a dynamic variation of +/- 30 % on the linear optimization value. The trade-off curve between optimal value and computational costs showed that a discretization level of at least the 6% of the turbine nominal flow is needed to achieve satisfactory results. Further increments in the discretization levels are demonstrated to only marginally improve the revenue function value and to strongly increase computational costs

Although DP method outperforms LP optimization in simple long term problems, in case of complex water systems, with multiple reservoirs and concentration times, the number of states increases exponentially according to 6.10, this makes almost impossible to solve the optimization for problems similar to the case study considered in section 1.1.3 on page 8.

Conclusions

This work of Thesis has led to the implementation of a comprehensive computational framework which is capable to optimize a multi-reservoirs hydroelectric system. In order to test the performances and validate the model developed, identification of input historical data and generation of synthetic time series has been performed. For what concerns the streamflows identification and generation, a de-seasonalized ARMA model with residual noise envelope has been applied (3.1). The daily streamflows generated with this method are shown to be a good representation of the real phenomenon, the assumption of non-correlation among different streamflows is made since reservoirs are geographically separated. For electricity price perturbation the data are generated with an hourly frequency, a simple statistical model has been adopted due to the fact that more complex and reliable models would have required a deep electrical market analysis, which out of the scopes of this Thesis. Through a Fourier analysis two main periodic components has been identified, corresponding to 1 and 2 cycles per day. The adopted method of normal distribution identification for the same hour on different days demonstrated to be effective in reproducing price statistics. The method pointed out the necessity to use only recent data (maximum 200 days in the past) in order to avoid a biased estimation of the phenomenon.

The approach to the problem with a ”divide and conquer strategy” is shown to be sub-optimal with respect to the global optimization, however the results are fully consistent with the physics of the system. During the analysis of the deterministic cases, performances are shown to be fully satisfactory with an optimality loss in the worst case scenario of only 0.5%. The computational time required for the solution of the same problem is about 15 times lower with respect to the global linear optimization, moreover, the divide and conquer approach is the only one of the techniques considered capable to solve problems with a length order of 1 year. Montecarlo analysis suggested that, for the case study 1.1.3 taken into consideration, an optimal choice for the length of the time segment is at least 793 hours, roughly corresponding to the 10% of the global optimization time window. The computational time saved, with respect to the longest segment that could have been optimized (due to hardware limitations), is 60%, with an optimal value loss of less than 0.5%. For longer time segments is also observed a reduction in the variance of the optimal solution, due to the increased system flexibility; for the same motivation, the variance in operational range of reservoirs is instead increased. Thus, for longer steps not only the optimal value is higher but presents also a smaller uncertainty.

Variance analysis demonstrated that no effective benefits can be obtained in practice if the time segment length is further increased, due to uncertainty in input data. Even if for longer segments no benefits are obtained, results have shown that particular attention must be taken in not underestimating the minimum time length since optimal value shows a step behaviour.

For the short term optimization instead, the focus is shifted on the generation of a possible bid profile to be offered on the market. The analysis, based on a Montecarlo method, allows the identification of recurrent solutions with a similar price/power generation profile; the extrapolation of this results allows the evaluation of an aggressive or conservative strategy. Of particular importance is the possibility to identify the minimum basal power production that a certain water system must produced due to the imposed constraints. Short term analysis allows also to describe the physic of the different power plant from a price/production curve point of view.

A different model and optimization approach based on dynamic programming has also been investigated. DP is demonstrated to perform considerably better with respect to the previous linear model in cases in which the problem presents a singular reservoir. The discretization of the state space has been turned out to be a fundamental choice for the effective functionality of the DP model. A too wide discretization space would require an excessive number of states resulting in a computationally demanding optimization. If a narrow state space is instead selected, the solution could converge to an optimum which is not the global one The segment dimensions have been expressed in function of the nominal turbine flow, which has been demonstrated to be an important index for the selection of the discretization level.

The optimal result is improved of about 20% with respect to the linear model; moreover, for long term optimizations, the problem complexity scales linearly and not quadratically, resulting in a computational advantage on long term problems. Although DP method outperforms LP optimization in simple long term optimizations, in case of complex water systems the

DP model grows exponentially making impossible to find a solution even for small time windows. This result would suggest that a "divide and conquer" approach better suits complex and long term problems while DP performs way better in the solution of single reservoir problems with long optimization windows. In case of simple system and short term optimization both approaches are valid, taking into account that linear programming in this case is demonstrated to be more efficient but less effective.

6.8 Future developments

On the basis of the analysis performed and of the considerations made, future activities have been identified

Electricity price advanced modelling. To simulate the electricity price a simple statistical model is currently adopted. The model is suitable for the Montecarlo analysis since it is capable to correctly represent mean and variance of the historical time series. Although, given the huge amount of public data available on the electricity market, more complex models could be implemented in order to improve the uncertainty analysis and, more importantly, the operational strategies in the short term. A more advanced modelling means the introduction of new inputs to the system, in literature numerous studies can be found about the prediction of electricity price in correlation with other factors, such as power consumption, country's principal energy source, etc...

Analysis of the brokers actual strategy. Bidding strategy evaluation is based on a statistical and quantitative approach, which is often different from the real strategy adopted by the brokers, mainly based on expertise and knowledge of the system. Possible improvements in the short term problem modelling could be done analysing real broker's choice in different scenarios, trying to implement the considerations and strategies that they adopt in an algorithm.

Bidding strategy on the overall electricity markets. In 5.2, the assumption made is that only MGP is considered for the generation of a bidding profile. Although approximative, the assumption is valid since MGP is the main reference market and the less perturbed by the others. Thus, for a comprehensive offer profile the whole set of auctions should be taken into account in order to provide better results and a wider vision of the system. This possible refinement of the model should be coupled with the advancements in electricity price modelling, since a global bidding profile estimation makes sense only if the electricity price is correctly predicted.

Integration of the DP approach within the existing framework. Dynamic programming has been shown to perform better than LP on single-reservoirs problem, however at the actual development stage, the model has not yet been generalized. Automatic selection of optimal state space dimension and discretization should be developed and the integration of DP with the exiting framework, created for the linear optimizer, should be performed.

Appendix A

Codes and algorithms

In the following pages are reported fragments of the code developed and used in this thesis, all the programs are written in MatLab language. This is not a full description of the scripts but just the most relevant portion of code are included.

A.1 Divide and Conquer evaluation algorithm

Listing A.1: Divide and Conquer evaluation algorithm

```
ott_spezzata = 0;
dim_prob = m_bacini*n_variabili*n_ore^2;  %system complexity
consumo_ram_stimato = dim_prob * 8 * 7.5 / 1024 / 1024 / 1024 ; %[Gb]
m_bacini_stagionali = sum(BACINI_STAGIONALI);

if consumo_ram_stimato>RAM_max && m_bacini_stagionali>0 && m_bacini>1   %limit on system complexity
    ott_spezzata = 1;
end

if n_ore>periodo_max && m_bacini_stagionali>0 && m_bacini>1   %limit on maximum allowable segment length
    ott_spezzata = 1;
end
```

A.2 Period length calculation

Listing A.2: Period length calculation algorithm

```
resti = [];
unisco_ultimo_periodo = 0;
divisori = 24:24:floor(periodo_max/24)*24;
if ott_spezzata == 1
    for kk = divisori
        resti = [resti, rem(n_ore,kk)];  % must be
            multiple of 24 hours
    end
    n_ore_periodo = max(divisori(((divisori+resti)
       <=periodo_max) | (resti>periodo_max*0.6)));
    resto = resti(n_ore_periodo == divisori);
    if resto + n_ore_periodo <= periodo_max
        unisco_ultimo_periodo = 1;  % add remain
           hours to the last segment
    end
end
```

A.3 Cost matrix computation

Listing A.3: Cost matrix computation

```
for tt = 1:n_ore-1
    states_in = linspace(Q_low(tt),Q_up(tt),
       n_stati_V(tt));
    states_out = linspace(Q_low(tt+1),Q_up(tt+1),
       n_stati_V(tt+1));

    for s_in = 1:n_stati_V(tt)
        for s_out = 1:n_stati_V(tt+1)
            [f_val,~] = rev_eval(states_in(s_in),
               states_out(s_out),X_max(tt,:),X_min
               (tt,:),dQ_max(tt),afflusso(tt),ken(
               states_in(s_in)),VETTORE_PREZZI(tt)
               );
            cost_matrix(tt,s_in,tt+1,s_out) =
               f_val;
```

```
        end
    end
end
```

A.4 Residual cost function calculation

Listing A.4: Residual cost function calculation

```
%% best path calculation (backward calculation)
V(n_ore,n_stati_V(end)) = 0;

for tt = n_ore-1:-1:1
    for s_in = 1:n_stati_V(tt)
        rev_earned = zeros(n_stati_V(tt+1),1);
        for s_out = 1:n_stati_V(tt+1)
            rev_earned(s_out) = gain_M(tt,s_in,tt
               +1,s_out) + V(tt+1,s_out);
           end
        end
        V(tt,s_in) = min(rev_earned);
    end
end
```

A.5 Optimal state path calculation

Listing A.5: Optimal state path calculation

```
%% find states of minimal path (forward
   calculation)
percorso_ottimo = zeros(n_ore-1,1);
pos_attuale = 1;
for tt = 1:n_ore-1
    rev_earned = zeros(n_stati_V(tt+1),1);
    for s_out = 1:n_stati_V(tt+1)
        rev_earned(s_out) = gain_M(tt,pos_attuale,
           tt+1,s_out) + V(tt+1,s_out);
    end
    [maxv, pos_attuale] = min(rev_earned);
    percorso_ottimo(tt) = pos_attuale;
```

```
    %clear rev_earned
end
percorso_ottimo = [pos_attuale;percorso_ottimo];
```

A.6 Optimal allocation function

Listing A.6: Optimal allocation function

```
%% constraints
if dQ < 0  % negative reservoir variation
    f_val = -1e20;
end
if dQ > dQ_max % maximum change rate
    f_val = -1e20;
end
if dQ > X_max(1) + X_max(2) % non riesco a
   svuotare il bacino in tempo
    f_val = -1e20;
end
%% optimal choice
if f_val==0
    if X_max(1) >= dQ
        X(1) = dQ;
    else
        X(1) = X_max(1);
        X(2) = dQ - X_max(1);
    end
    if dQ < X_min(1)
        X(1) = 0;
        X(2) = dQ;
    end
    eta = funzione_rendimento(X(1),X_min(1),X_max
       (1));
    f_val = ken(Q)*eta*X(1)*prezzo;
end
```

Acronyms

ACF	Auto-Correlation Function
ARIMA	Auto Regressive Integrated Moving Average
ARMA	Auto Regressive Moving Average
DP	Dynamic Programming
GRG	Generalized Reduced Gradient Method
HJB	Hamilton-Jacoby-Bellman
KKT	Karush–Kuhn–Tucker conditions
LP	Linear Programming
LT	Long Term optimization
MGP	Mercato del Giorno Prima
MI	Mercato Infragiornaliero
MILP	Mixed Integer Linear Programming
MOM	Method of Multipliers
NLP	Non Linear Programming
OLSR	Ordinary-Least-Square Regression Models
PACF	Partial Auto-Correlation Function
PO	Principle of Optimality
RAM	Random Access Memory
SARMA	Seasonal Auto Regressive Moving Average

SLP Sequential Linear Programming

SQP Sequential Quadratic Programming

ST Short Term optimization

Bibliography

[1] Asmadi Ahmad et al. "Reservoir Optimization in Water Resources: a Review". In: *Water Resources Management* 28 (Sept. 2014). DOI: 10.1007/s11269-014-0700-5.

[2] Virgilio Anselmo and Lucio Ubertini. "Transfer function-noise model applied to flow forecasting / Modèle Box-Jenkins pour la prévision des débits des cours d'eau". In: *Hydrological Sciences Journal* 24 (Sept. 1979), pp. 353–359. DOI: 10.1080/02626667909491874 (cit. on p. 29).

[3] Charles ReVelle Arthur E. Mc Garity. *Design and Operation of Civil and Environmental Engineering Systems*. Jan. 1997 (cit. on p. 22).

[4] A. Bensalem et al. "Optimal Management of Hydropower Systems". In: *Physics Procedia* 55 (2014). 8th International Conference on Material Sciences, CSM8-ISM5, pp. 138–143. ISSN: 1875-3892. DOI: https://doi.org/10.1016/j.phpro.2014.07.020. URL: http://www.sciencedirect.com/science/article/pii/S1875389214000820 (cit. on p. 23).

[5] Jason Brownlee. *How to Decompose Time Series Data into Trend and Seasonality*. 2017. URL: https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/ (cit. on p. 27).

[6] MAS Consulting. *MAS Consulting*. URL: https://www.mas-co.it/.

[7] Samir Córdoba-Machado et al. "Seasonal streamflow prediction in Colombia using atmospheric and oceanic patterns". In: *Journal of Hydrology* 538 (2016), pp. 1–12. ISSN: 0022-1694. DOI: https://doi.org/10.1016/j.jhydrol.2016.04.003. URL: http://www.sciencedirect.com/science/article/pii/S0022169416301846 (cit. on p. 26).

[8] Habib Salehi D.J. Holtschlag. *Statistical models for estimating daily streamflow in Michigan*. Jan. 1992. DOI: 10.3133/wri914194 (cit. on pp. 26–28).

[9] A.M. Davey and B.E. Flores. "Identification of seasonality in time series: A note". In: *Mathematical and Computer Modelling* 18.6 (1993), pp. 73–81. ISSN: 0895-7177. DOI: https://doi.org/10.1016/0895-7177(93)90126-J. URL: http://www.sciencedirect.com/science/article/pii/089571779390126J (cit. on p. 27).

[10] Clayton Forknall. "Optimal management strategies for a cascade reservoir system". Sept. 2014, p. 204 (cit. on p. 20).

[11] Aris Georgakakos, Huaming Yao, and Yongqing Yu. "Control Models for Hydroelectric Energy Optimization". In: *Water Resources Research - WATER RESOUR RES* 33 (Oct. 1997), pp. 2367–2379. DOI: 10.1029/97WR01714.

[12] K.L. Hiew. *Optimization algorithms for large-scale multireservoir hydropower systems*. Jan. 1987 (cit. on p. 23).

[13] Azhar Husain. "An Overview of Reservoir Systems Operation Techniques". In: (2012) (cit. on pp. 17, 20, 21, 23–25).

[14] Rana Ikram et al. "Application of Time Series Models for Streamflow Forecasting". In: *Civil and Environmental Research* 9 (Mar. 2017), pp. 2224–5790 (cit. on p. 28).

[15] John Labadie. "Optimal Operation of Multireservoir Systems: State-of-the-Art Review". In: *Journal of Water Resources Planning and Management* 130 (Mar. 2004), pp. 93–111. DOI: 10.1061/(ASCE)0733-9496(2004)130:2(93) (cit. on pp. 18, 20–25).

[16] Arthur Maass et al. *Design of Water Resource Systems*. Vol. 45. Jan. 1962. DOI: 10.2307/1236007 (cit. on p. 22).

[17] Gestore dei Mercati Energetici S.p.A. *GME - Gestore Mercati Energetici*. 2019. URL: http://www.mercatoelettrico.org/it/ (cit. on pp. 25, 26, 53).

[18] Gestore dei Mercati Energetici S.p.A. *Guida al Mercato Elettrico*. 2003 (cit. on p. 2).

[19] Gestore dei Mercati Energetici S.p.A. *Manuale utente per operatori di Mercato*. 2003 (cit. on p. 2).

[20] Kourosh Mohammadi, H R. Eslami, and Rene Kahawita. "Parameter Estimation of an ARMA Model for River Flow Forecasting Using Goal Programming". In: *Journal of Hydrology* 331 (Nov. 2006), pp. 293–299. DOI: 10.1016/j.jhydrol.2006.05.017 (cit. on p. 28).

[21] M Mondal and Jahir Uddin Chowdhury. "Synthetic Stream-Flow Generation with Deseasonalized ARMA Model". In: *Journal of Hydrology and Meteorology* 8 (Aug. 2016), p. 32. DOI: 10.3126/jhm.v8i1.15570 (cit. on pp. 28, 42, 45).

[22] S. Tsay Mr. Ruey. *Transfer Function Models*. Jan. 2009 (cit. on p. 29).

[23] Robert Nau. *Identifying the numbers of AR or MA terms in an ARIMA model*. 2018. URL: https://people.duke.edu/~rnau/411arim3.htm (cit. on pp. 44, 47).

[24] Javad Nematian. "An Extended Two-stage Stochastic Programming Approach for Water Resources Management under Uncertainty". In: *Journal of Environmental Informatics* 27 (Jan. 2015). DOI: 10.3808/jei.201600334 (cit. on p. 24).

[25] Isailovski M. Opricovic S. Eccher L. "An optimization procedure for water reservoir planning". In: *Yugoslav Journal of Operations Research* 225 (Jan. 1991) (cit. on p. 24).

[26] A.F. Ribeiro et al. "On the optimal control of a cascade of hydroelectric power stations". In: *Electric Power Systems Research* 88 (July 2012), pp. 121–129. DOI: 10.1016/j.epsr.2012.02.010 (cit. on p. 21).

[27] Saurabh Sangal, Ashish Garg, and Dinesh Kumar. "Review of Optimal Selection of Turbines for Hydroelectric Projects". In: 3 (Apr. 2012), pp. 424–430 (cit. on p. 105).

[28] W. O. Thomas. "Cost Effectiveness of the Stream-Gaging Program in Maine". In: *United States Geological SurveyWater-Supply Paper 2244* 2244 (Mar. 1984) (cit. on p. 28).

[29] Umberto Triacca. *Estimation of the parameters of an ARMA model*. Jan. 2009 (cit. on p. 44).

[30] Marnix van der Vat. "Optimizing reservoir operation for flood storage, hydropower and irrigation using a hydro-economic model for the Citarum River, West-Java, Indonesia". Aug. 2016. DOI: 10.13140/RG.2.2.22550.55363 (cit. on pp. 17, 18, 22).

[31] Jacques W. Delleur and M Kavvas. "Stochastic Models for Monthly Rainfall Forecasting and Synthetic Generation". In: *Journal of Applied Meteorology* 17 (Sept. 1978), pp. 1528–1536. DOI: 10.1175/1520-0450(1978)017<1528:SMFMRF>2.0.CO;2.

[32] T. Astatkie W. E. Watt. "Multiple-input transfer function modeling of daily streamflow series using nonlinear inputs". In: *Water resources research* 34 (Oct. 1998), pp. 2717–2725 (cit. on p. 29).

[33] Wikipedia. *Travelling salesman problem*. 2018. URL: https://en.wikipedia.org/wiki/Travelling_salesman_problem (cit. on p. 106).

[34] Ralph Wurbs. *Computer Models for Water Resources Planning and Management*. Jan. 1994 (cit. on p. 22).

[35] William W.-G. Yeh. "Reservoir Management and Operations Models: A State-of-the-Art Review". In: *Water Resources Research* 21.12 (1985), pp. 1797–1818 (cit. on pp. 20–22).

[36] U Zessler and Uri Shamir. "Optimal Operation of Water Distribution Systems". In: *Journal of Water Resources Planning and Management-asce - J WATER RESOUR PLAN MAN-ASCE* 115 (Nov. 1989). DOI: 10.1061/(ASCE)0733-9496(1989)115:6(735).