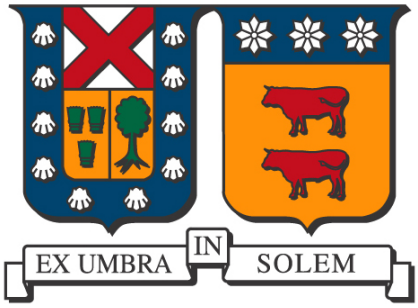
ROBUST SOLUTION SCHEME FOR THE UNIT COMMITMENT: AN ADAPTIVE DATA–DRIVEN LEARNING–BASED APPROACH

JIMÉNEZ BUSTAMANTE, DIEGO NICOLÁS



Universidad Técnica Federico Santa María

Department of Electrical Engineering

Robust solution scheme for the unit commitment: An adaptive data–driven learning–based approach

Author:
Diego Nicolás
Jiménez Bustamante

Thesis Director:
Dr. Alejandro Alberto
Angulo Cárdenas

A thesis submitted in partial fulfillment of the requirements for the degree of

Magíster en Ciencias de la Ingeniería Eléctrica

October 22, 2021

Abstract

Robust optimization models of the unit commitment problem (RUC) have been widely used for the day–ahead calculation of power dispatches and reserves schedules under high penetration of renewable generation. Typically proposed uncertainty sets, as budget–based sets, control the level of robustness of the solutions by the selection of a certain set of parameters. However, the procedures for its calculation are often considered part of a preprocess, ignoring the possible benefits of the dynamic determination of it. In this work, a solution scheme for the RUC problem is proposed, using data–driven–based uncertainty sets, where robustness control parameters are dynamically calculated as a function of previous operation results. The determination of the adaptive robustness level is made using a reinforcement learning approach, resulting in a closed–loop data–driven framework. Besides, an experimental framework that simulates real–time operation is proposed and used to test the proposal. Out–of–sample experiments shown the effectiveness of the proposed scheme against well–known robust formulations with fixed robustness levels, by improving systematic indicators as operational costs, non–served energy, and renewable energy curtailment. Two systems of different scales are analyzed, showing the concept effectiveness and the scalability of the present proposal.

Acknowledgements

A mi familia por su incondicional apoyo y amor, a mis amigos por su cariño y la vida compartida, y al profesor Alejandro Angulo por su no–acotado apoyo, paciencia y dedicación.
A la Agencia Nacional de Investigación y Desarrollo (ANID), que por medio del proyecto basal FB0008 "Advanced Center for Electrical and Electronic Engineering, AC3E" y el proyecto Fondecyt Regular N°1210625, colaboraron con el desarrollo de este trabajo.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The present chapter gives an overall summary of the whole project. Global and national contexts of the operation of power systems under high renewable generation penetration levels are first presented. Then, the state–of–the–art of the disciplines related to the core of the thesis is shortly reviewed. Finally, the central hypothesis, objectives, used methodologies, and contributions of the present project are described.

1.1 Context and motivation

Nowadays, the global concern about environmental safety, besides the constant decrease in costs of technologies used by non-conventional renewable energies (NCRE), has contributed to their progressive inclusion in the electrical power systems (EPS). These new technologies and policies have provided the EPS with new features, making them liable to experiment with new phenomena and, therefore, increasing the complexity of the planning and operation tasks.

In the global context, at the beginning of 2007, the installed capacity of NCRE generation reached up to 182 GW in the world, being equivalent to 4% of the total installed capacity at that moment (4100 GW). For its part, in Chile, the growth and implementation of these technologies arise in 2004, when the government began to motivate the development of NCRE with the creation of the named *Ley Corta I*. Afterward, the promulgation of law N° 20.256 gave place to a series of modifications that strengthened the inclusion of NCRE into the system until its current level and, with the promulgation of law N°20.698, also called *Ley 20/25*, a 20% of NCRE penetration is expected in the year 2025. Fig. 1.1 shows the evolution of the installed power on the Sistema Internectado del Norte Grande (SING) for each generation technology.

The high variability and the intermittent nature of the NCRE are hard to manage when an economic and reliable operation of the system is required at the same time. Due to this difficulty, in October of 2012, the Load Economic Dispatch Center of SING (CDEC-SING) published a *Technical-Economic study about Wind and Solar integration in the SING*, whereby establish that the incorporation of both sources in a moderate way allows the system, in its current state, administrate the NCRE in an efficient and safe way. However, for bigger levels of penetration, modifications will have to be made, both as the development of new software for the operation and the use of new electrical devices, as energy storage ones and active control systems.

According to this, a series of challenges appears looking towards the deep integration of NCRE, especially in the context of short–term planning and real-time operation, both extensively discussed in the literature [4, 5, 6]. In this context, the unit commitment (UC) problem is one of the most impacted. The UC is a mathematical programming problem that defines the state of the generation units (on or off) and their reserve levels within a certain period, which can be a single day until several weeks, in order to achieve an economic operation and meet the operational constraints of the system [7]. Especially in the independent system operator (ISO) managed electricity markets, it becomes essential to have effective methodologies to efficiently account for REG resources in the UC and produce robust decisions that ensure system reliability, specifically on generators setpoints and reserve levels. REG variability, especially from wind units, has very complex dynamics that are hard to predict. Inadequate management of this variability could conduce the system to high–risk level operation, where the computed generators' schedules could be excessively costly or even infeasible [8].

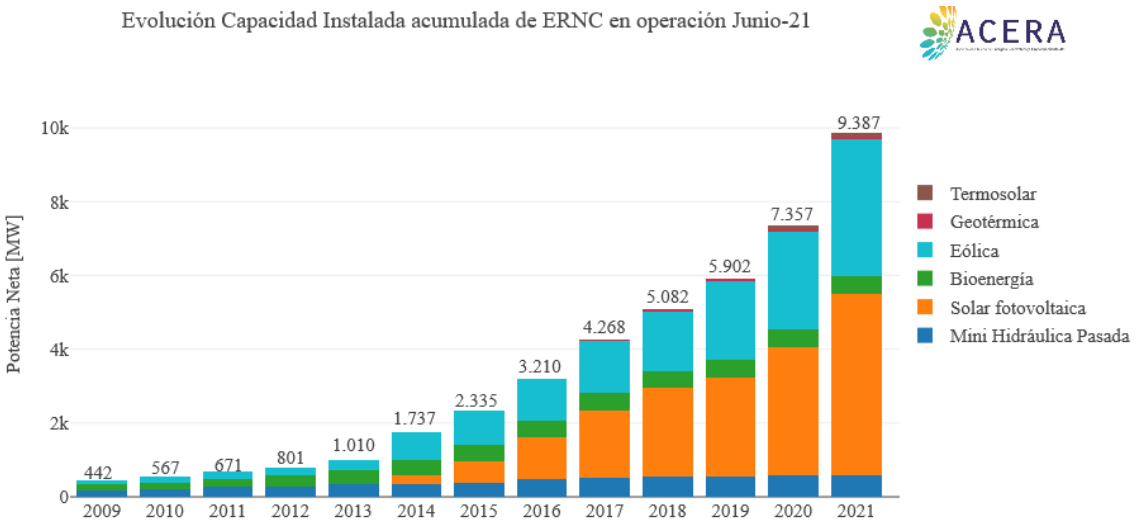


Figure 1.1: Evolution of the installed capacity accumulated of ERNC between 2009 and June 2021. Source: ACERA (`https://acera.cl/estadisticas/`)

The deterministic version of the UC problem, i.e., using fixed values for every uncertainty quantity in the problem, is still an active field of research, and is far to be a well–solved problem. Given the operational requirements and generator features, the UC is a large-scale, non–convex problem, that is required to be solved in a relatively small time, in order to make feasible the implementation of the solution in the real–time operation. This lead to a series of approximations needed to become the problem tractable, raising the well–known trade-off in engineering between the tractability of the problem and an accurate description. Most of the current approaches seek the design of tight and compact mixed-integer linear programming (MILP) formulations, on which remarkable works have been developed [3, 9, 10].

In that regard, considering the difficulties of the solely deterministic version of the UC problem, adding uncertainty to the problem could increase even more its complexity. Most of the current approaches to modeling uncertainty on the UC problem can be categorized into three major categories. The first one is based on stochastic programming (SP), the second on chance-constrained programming (ChCP), and the third on robust optimization (RO). SP approaches principally use scenario trees to capture the underlying distribution of the unknown variables. With this, the uncertainty is discretized and assumed to be known for every scenario, resulting in a large–scale deterministic UC problem[11, 12]. Nevertheless, a typical problem of SP models is the appropriate construction of scenario trees that be able to accurately capture the underlying distributions, which can be very hard, inspiring works that have proposed strategies to tackle it [13, 14]. Concurrently, ChCP models seek to minimize systems costs while ensuring with a certain probability the feasibility of the solution across a set of scenarios. This approach allows for balancing the trade-off between total costs and solution robustness by varying the required probability level of feasibility. However, the major drawback of ChCP models is the non-convexity of probabilistic constraints and its complexity to be evaluated, which has been addressed [15, 16, 17].

On the other way, the RO paradigm applied to the UC, usually called robust unit commitment (RUC) models, appear as an opportunity to handle both tractability and risk management, especially two-stage formulations that have been widely used. Those formulations are characterized for the statement of a three-level optimization problem, where the first level decides the optimal schedule, dispatch and reserves levels for generators, the second level selects the worst-case realization of the uncertainty, contained in the so-called uncertainty set, and third-level decisions define the best re-dispatch of generators. In this way, the construction of the uncertainty set is a key task in the development of RUC models since it defines the complexity of the problem and the robustness of the solution. Due to resolution purposes, most of the proposed uncertainty sets are polyhedral, where the often solution approaches rely on two well–known decomposition algorithms, namely the Benders decomposition and column and constraint generation (C&CG) [18]. However, the design of uncertainty sets with appropriated robustness levels and other desired properties is

still ongoing research.

Concurrently with the development of mathematical programming techniques, a remarkable enhancement of data acquisition and data management has taken place. The increasing amount of available data and the ongoing improvement of hardware technologies have motivated the creation of a new discipline, namely data science, extending its applications over a large spectrum of fields, including mathematical programming. In particular, the *data-driven* paradigm in optimization models with parameters subject to uncertainty has meant the incorporation of data manipulation techniques, leaving aside the need for previous assumptions about the uncertainty involved. For traditional SP and ChCP models, probability distributions are needed, which are commonly derived from the available data and some assumptions of the underlying random process. Nowadays, the amount of data allows researchers to use it directly for the construction of mathematical structures embedded into the optimization models. This results in models that optimize over the data, where statistical requirements on the solution can be directly included into the model as new constraints, changing the typical way of using the SP, ChCP, and RO approach.

Concerning RO models, the data–driven paradigm naturally results in constructing uncertainty sets from the data. One of the most conventional and used uncertainty sets is the budget-constrained set, proposed in [4], where each component is contained in a fixed interval around a nominal point, and the budget constraint controls the deviation from the center by imposing some linear inequalities. Some variations of the conventional budget–set were proposed in [5], in which affine policies were used to model wind variability and the uncertainty set containing the prediction errors. In these formulations, the model robustness is controlled by selecting the budget, eliminating realizations allocated in the set corners, with a very low probability of occurrence. This procedure generates extreme points from the intersection between the box and the linear inequalities, which number grows exponentially with the dimension of the set. Other approaches use scenario-based uncertainty sets, on which the uncertainty set is constructed directly from recent observations. Recent successful implementations of these sets are found in [19, 20], on which the uncertainty sets are constructed as the convex hull of recent observations. An important particularity of the above is that each point of the set represents the observations of a defined number of consecutive hours for each uncertainty parameter. The benefits of this approach are threefold; in the first place allows capturing the intrinsic correlations of wind dynamics on each point, both spatial and temporal. Besides, the convex hull description is well suited for the use of the C&CG algorithm since the number of extreme points can be controlled. Finally, the moving window of recent observations matches with the adaptive construction of the uncertainty sets. The robustness of those sets is controlled by selecting the number of scenarios; if a large number is selected, a more robust solution is expected.

In all of the above-presented methods, the robustness level selection is part of the pre-processing stage and does not change across time. The work [21] shows the differences between different robustness levels for three methods, where an optimal level is found for each method in terms of out–of–sample results. However, these optimal levels are obtained for a particular out–sample dataset and could be inappropriate for others. This fact opens the possibility of dynamically determining the system's appropriate robustness level, given the experience of past operational results.

In this way, this thesis explores the potential benefits of the dynamic determination of the robustness level, constructing a closed–loop framework of data-flow, similar to those widely used in control problems. Besides, the construction of closed-loop frameworks embedding mathematical programming models and machine learning techniques has been already identified in [22] as potentially powerful tools, giving more foundation to the present proposal. In the following section, the principal guidelines of the developed work are presented.

1.2 Summary

This section comprises the principal foundations and methods of the present thesis, from basis hypothesis to main contributions.

1.2.1 Hypothesis of work

Based on the extensive evidence collected from the literature, non–studied topics for dynamical adaptation of uncertainty management policies were identified. Furthermore, existing evidence

suggests possible non–discovered benefits of including real–time data in the definition of uncertainty management policies. In this line, the central hypotheses that support the present work are the following:

1. There exist efficient computational implementations of UC *data–driven* based models, where a desirable risk–averse level in terms of *out–of–sample* results exists, obtained by the dynamical adaptation to the real–time data.

2. An adaptive method for the determination of the risk–averse level of a *data–driven* UC model, based on the historical data and real–time results, could generate planning solutions with better performance in the real–time operation than existing non–adaptive schemes.

1.2.2 Problem statement and objectives

A solution scheme for the RUC problem is proposed in the present work, addressing two primary functions. Firstly, it is used for the dynamic determination of the robustness level of the uncertainty sets. Secondly, it is also used as an experimental framework to simulate real–time operation and evaluate different short–term planning policies. The proposed RUC model uses data–driven–based uncertainty sets with parameters that regulate its sizes. The level of robustness is dynamically obtained as a function of previous real-time results, which constitute a closed-loop data-driven framework, similar to the study in control theory. To this, reinforcement learning techniques are used for the robustness level determination.

In that regard, the main objective of this thesis is to design an adaptive solution scheme for the problem of UC, founded on a two–stage, data–driven based model, where the policies for the determination of the desirable risk–averse level be given by an adaptation to the data and real–time results, in the same philosophy of a closed–loop control architecture. The specific objectives are the following:

1. Design a two-stage, *data–driven* based UC model, which has a set of parameters that regulate the risk–averse level of the decision.

2. Design a methodology for the online determination of the risk–averse level, based on an adaptation of the data and real–time results, constituting a closed–loop control framework.

3. Evaluate the performance of the proposed scheme through extensive out–of–sample computational experiments, made in a simulation platform for real–time operation in EPS, over IEEE instances of different scales.

1.2.3 Methodology

The above presented specific objectives are addressed by applying the following methodology:

1. Construction of a two–stage RUC model, using as a base the deterministic 3–bin model [3], and a modified version of the scenario–based sets presented in [20, 23] for the uncertainty sets, on which a parameter controls the size of the set and, therefore, the robustness of the commitment solution.

2. Construction of an experimental framework to simulate real–time operation of an electric power system. This framework gives place to formulating of a sequential–decision problem for the rolling horizon RUC, using the formalism of dynamic programming. Subsequently, the dynamic programming equations of optimal control are approximately solved using a reinforcement learning agent, resulting in the dynamic adaptation of the robustness level of the uncertainty set used in the RUC model.

3. Conduction of extensive numerical experiments on two different scale power systems, ensuring the better performance of the proposal against current approaches, besides checking scalability. To this end, out-of-sample experiments are performed, comparing operational indexes of the real-time operation simulation model, namely, generation costs, bus imbalances, and violations on bus voltage levels.

1.2.4 Contributions

The contributions of the present thesis are summarized below:

1. This work proposes a closed–loop framework for the resolution of a two–stage RUC problem. To this, a data–driven scenario–based uncertainty set with a variable level of robustness is proposed, which is obtained from the evaluation of real–time operational indexes. The presented scheme is general and can be used in several engineering contexts.

2. A sequential decision–making formulation for the rolling horizon RUC is presented, using the formalism of dynamic programming, allowing the use of several well–studied approximated methods for its resolution. Particularly in this work, an RL agent is used to the dynamic adaptation of the robustness level of the uncertainty set used in the RUC model.

3. This thesis presents extensive numerical experiments on two different scale power systems, both for ensuring the proposal's performance against current approaches and check the scalability of the proposal. Out-of-sample experiments demonstrate the superior performance of the closed–loop scheme over the open–loop version by comparing different operational indexes of the real-time operation simulation model, namely, generation costs and demand–supply violations.

1.2.5 Document structure

In chapter 2, the presentation of the state–of–the–art of UC models with uncertainty management and dynamic adaptivity is made. Consequently, deterministic versions of the UC, optimization paradigms for uncertainty management, and approximations of optimal control methods for the sequential decision–making problems are reviewed. Lately, chapter 3 contains the core of the proposal, beginning with a motivational example that illustrates the scheme's spirit. Then, the proposed closed–loop framework under analysis is decomposed and analyzed in four parts. A two–stage UC model is presented in the first one, besides the dynamic uncertainty sets of the proposed two–stage RUC model. The formulation of the sequential decision–making problem is presented in the second part, adapting the components of the proposed scheme with a dynamic programming description. Lately, the platform for the simulation of real–time operation is presented, describing the models used to test and evaluate the UC methodologies. Lastly, the strategies used to the resolution of the robust UC model are presented, besides a proposal for the construction of the feedback loop based on reinforcement learning techniques. Chapter 4 contains the computational experiments performed. Evaluation methodologies and algorithm settings are firstly described. Then, results of the experiments over an illustrative 4–bus system and the IEEE 118–bus system are exposed. Finally, chapter 5 presents the conclusions of the present thesis, and some discussion about future research directions is developed.

Chapter 2

Background

In the present chapter, the main topics related to the thesis work are reviewed. To better understand the relevance of the unit commitment problem, a brief description of the structure of electricity markets is given. Then, deterministic optimization models of the unit commitment are presented, focusing on mixed–integer linear formulations. Subsequently, optimization under uncertainty approaches are described, analyzing classical and novel paradigms of uncertainty management in optimization, under the variations given by the use of multiple stages, rolling horizon schemes, and data–driven approaches. Finally, short descriptions of dynamic programming formulations and reinforcement learning basics are given, focusing on Q–learning algorithms, and finishing with an overall review of the existing applications of reinforcement algorithms in the context of power management problems.

2.1 Electricity markets fundamentals

In few words, an electricity market is an arrangement constructed to transfer electric energy from producers to consumers. This transmission is made via a power system, which is a specialized infrastructure on which several market agents are connected. Even considering that most of the electricity consumption has to be made at low voltage levels, varying from some hundreds of volts to some tens of kilovolts, generation stations and electricity customers are usually placed at very long distances. This fact makes necessary a transmission system operating at high voltage levels, generally of the order of hundreds of kilovolts, keeping the power losses due to Joule heating at minimum values. This spatial location generates a primary natural subdivision of power systems into the named distribution and transmission levels, respectively, depicted in Fig. 2.1.

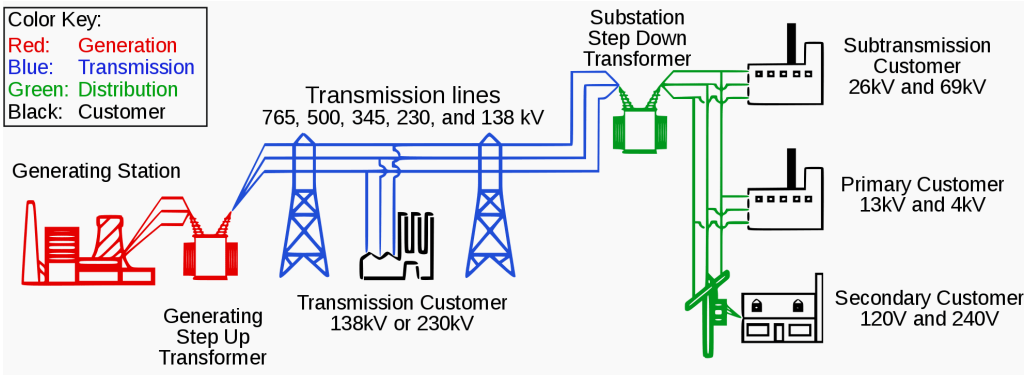


Figure 2.1: Power system basic schema. Different market agents are connected at different levels of the power system. Source: `https://electrical-engineering-portal.com/electric-power-systems`

In addition to generators and customers, others agents also participate in the electricity market, accomplishing different functions that vary from system to system. However, some of the

functions are widely shared between different power systems, particularly energy production, energy consumption, system owning, and system operation, where an agent can fulfill more than one function.

Concerning the system operator, this agent is responsible for maintaining a safe and economical operation of the power system, by assuming the technical task of coordinating the balance between the generation and consumption of energy. Since the economical facet of the task, system operators are independent of the rest of the agents, and are often referred to as independent system operators (ISO).

In this sense, among other obligations as keeping the real–time balance of consumption and generation, ISOs have to manage a trading system that guarantees that producers are paid for the energy they deliver into the system, and consumers pay for the energy they extract from it. This procedure has to be formal, known by every agent, where many possibilities exist in large–scale systems. As an example, Fig. 2.2 shows different interactions between different agents of the wholesale market of energy of Chile, where G1 to G3 refer to generators. In the diagram, three kinds of interaction are observed: regulated contracts, contracts under direct negotiation, and buy–sell trading into the spot market, which the ISO manages.

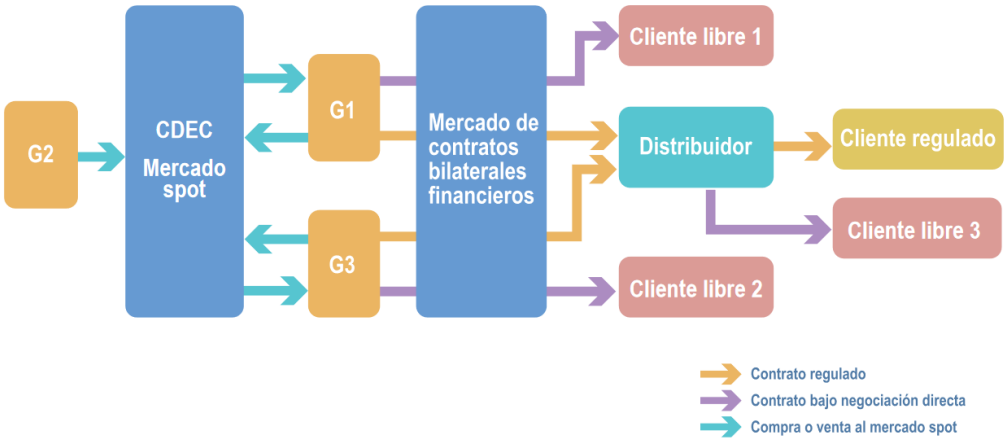


Figure 2.2: Diagram of Chile's market structure. Source: `https://mercadoernc.minenergia.cl/`

Generally, long–term contracts between generators and distributors are regulated for controlling and stabilizing the price paid for the consumed energy by regulated clients, e.g., residential consumers. In many systems, as in Chilean, the adjudication of the contracts is made via public auctions. The entity in charge looks for obtaining economic deals while satisfying supply objectives. As the prices agreed in the contracts are regulated for a considerable amount of years, they help reduce the volatility and uncertainty of the energy price on the spot market.

The spot market of energy is one of many systems used for managing the real–time trading of energy. Every buyer and seller of energy interact with a pool of power on which every bid is presented. This market solves the imbalances not covered by contracts. The resolution of which offers are selected is generally based on the marginal utility theory. In the particular case of Chile, all generators are forced to participate by publishing their generation costs, able to be audited by the ISO. Then, based on forecasts of future power demanded by system loads and forecasts of renewable–based generation, the ISO decides the generation schedule of generators for different periods, pursuing the most economical possible operation restricted by the physical limitations of the system.

In general, two types of sub–markets are present in the operation of power systems, namely the day–ahead (DA) and the real–time (RT) market, both characterized by the different periodicity of resolution. In the DA market, offers are made daily. In contrast, in the RT market, offers are presented in minutes or hours, resulting in different levels of uncertainty and then different levels of risk. Fig. 2.3 shows the timeline of the above–mentioned markets.

On this basis, the ISO solves short-term planning problems based on bids submitted in both trading stages, and the power system is dispatched according to the solution of those problems. In particular, the UC problem decides the ON–OFF schedule of generators and the reserve levels using the information delivered in the ahead trading. The economic dispatch (ED) problem decides

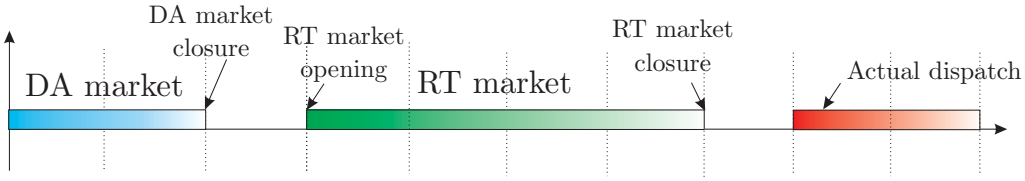


Figure 2.3: Timeline of trading periods for DA and RT markets.

real–time generators setpoints using the information proportioned in the RT market. It is essential to highlight that all imbalances between what was planned before and what happened during the trading period must be cleared when a dispatch is applied. Each player in the electricity market must either be balance responsible or agree with somebody else to be balance responsible for him. The ISO calculates this balance every time t where the operation is determined and for every market participant:

$$balance_t = total\_generation_t + energy\_purchase_t - total\_consumption_t - energy\_sales_t$$

In this context, mathematical programming problems for the short–term planning of the system are crucial for its operation and the resolution of the markets. Both the UC and ED are far away to be well–solved problems and are still a matter of interest for private and public institutions. This work focuses on the UC problem, where a detailed description is given in the section below.

2.2 Deterministic models of the UC

2.2.1 Brief review of typical methods

The UC is a mathematical programming problem that seeks the least cost generators schedule and its dispatch levels to meet the system demand during a specific period. UC problems involve different types of generators connected to the network, being the most usual hydro and thermal generators. Features of thermal generators are typically the hardest to deal with since they include time coupling between variables and non–linear functions. Some of the features commonly considered in UC problems are minimum–maximum power limits, minimum on/off periods, power ramps, non-linear generation costs, start-up costs as a function of the previous operation of the generators, among others.

Four significant categories group most of the existing methods for the deterministic unit commitment, namely dynamic programming, decomposition methods, meta-heuristics approaches, and MILP approaches [24, 25, 13]. Before introducing specific MILP formulations, a review of the rest of the approaches is presented for completeness.

Dynamic programming (DP) represents one of the classical approaches to deal with the UC problem. Early uses of DP had to combine it with heuristics and operator's criteria to overcome possible drawbacks of its use, as the known *curse of dimensionality* [26, 27]. However, most recent contributions that include DP techniques use it to solve subproblems of the UC, usually derived from decomposition techniques as Lagrangian-decomposition ones. Typically on those approaches, side constraints that link the different generator operations schedules, as demand and reserves requirements, are relaxed, resulting in subproblems of single–UC (1UC), which are scheduling problems where the unit behavior responds to price signals [28, 29, 30, 31].

In the same line, decomposition approaches take advantage of the structure of the problem, which can be exploited via suitable resolution algorithms. As said in the paragraph above, one of the methods is the Lagrangian–decomposition (LR), via the relaxation of demand and reserves requirements. In this way, the crucial points in LR methods are resolving the maximization problem involved (dual problem) and recovering feasible solutions [30, 32, 33]. Another decomposition technique is Benders' decomposition, which calculates *complicated variables* [13] and generates structures that allow different decompositions Problems often suitable for this decomposition are security-constrained UC (SCUC) and UC problems with network constraints. [34, 35, 36].

Concurrently, meta–heuristic-based methods can be described as attempting to solve the UC without giving optimality guarantees on the solution [37, 38]. Priority lists–based methods are popular heuristics methods that include operator's criteria in searching the solution [39, 40]. Other

approaches use typical algorithms of guided random exploration such as taboo search, simulated annealing, and genetic algorithms, which have been shown to provide feasible solutions [41, 42, 43].

With the development of new computational and mathematical advances, resulting in the improvement of off–the–shelf solvers, these kinds of formulations were deprecated, giving new methods based on mathematical programming.

2.2.2 MILP formulations

MILP formulations for the UC took the researcher's attention due to the continuous improvement of the area's theoretical results and computational tools. Early applications of these models can be found in the '70s decade by Garver [44] and Muckstadt J. and Wilson R. [45], and other posterior proposals based principally on exhaustive searching and complete enumeration [46, 47, 24], which was able to solve only small instances. However, the most significant progress has been reached in the last 20 years [13], allowing them to overcome these initial difficulties.

The usual approach to model the UC through a MILP formulation is considering binary variables for the generators' states and continuous variables for the quantities related to the network's power flow, namely, dispatch levels, bus voltages, and bus angles. A general structure for a MILP-UC model is the following:

$$\min_{\boldsymbol{x},\boldsymbol{y}} \ \boldsymbol{c}^T\boldsymbol{x} + \boldsymbol{d}^T\boldsymbol{y} \tag{2.1}$$

$$\text{s.t. } A\boldsymbol{x} \leq \boldsymbol{b} \tag{2.2}$$

$$H\boldsymbol{y} \leq \boldsymbol{h} \tag{2.3}$$

$$G\boldsymbol{x} + E\boldsymbol{y} \leq \boldsymbol{g} \tag{2.4}$$

$$\boldsymbol{x} \in \{0,1\}^{|\mathcal{G}|\times|\mathcal{T}|} \tag{2.5}$$

The binary variable vector $\boldsymbol{x}$ corresponds to the commitment decisions of units, containing the on/off states of conventional generators and other related variables. The vector of continuous variables $\boldsymbol{y}$ includes the dispatches and reserves decisions, besides bus voltage variables. $\mathcal{T}$ is the set containing the indexes of each period considered in the planning horizon, while $\mathcal{G}$ is the set of indexes for conventional generators.

The objective function (2.1) seeks to minimize the total operation cost for the whole planning horizon, composed of the start–up/shutdown costs and the dispatch costs. Equation (2.2) groups the minimum up/down times constraints for generators and the logical relations between binary variables. Equation (2.3) includes power ramps constraints, and equation (2.4) contains load system balance requirements, reserves requirements, and generation limits.

Recent advances in MILP formulations have been showing somewhat two trends, apparently opposites. One of them focuses on developing *accurate* models, trying to incorporate the real–world behavior and characteristics of the units, and, therefore, looking for the obtaining of better operational decisions able to be applied in practice. For example, technical constraints of hydro units and the modeling of the water–to–produced–energy is addressed in [32, 48], and for thermal units, detailed descriptions of start–up and shut–down trajectories have been made in [49, 50]

On the other side, *tight* formulations are pursued, improving the efficiency of the models and allowing them to solve big instances. As the UC models used in this proposal are part of this stream, tight models are detailed in the paragraphs below.

Tight formulations and polyhedral study of the UC

The importance of obtaining good polyhedral descriptions in big-scale problems, as the UC, resides principally in reducing the computation times. Besides, since the UC is an operational problem, it has to be solved continuously. Its solutions have to be implemented according to the schedule of the system operator, setting a limit to the maximum computation time available.

In this line, tight and compact formulations seek the reduction of computation time by developing efficient models. Tightness is related to descriptions of feasible regions closest to the convex hull of feasible integer points, whereas compactness does with the number of variables used in the description. Since most of the solution approaches used for MIP/MILP problems are based on branch and cut algorithms, tight formulations are relevant to obtain stronger lower bounds and accelerate the algorithm's convergence [51].

The description of feasible schedules on thermal generators is one of the most challenging tasks due to the intertemporal relationships between variables. In the first place, the formulation of minimum up/down constraints was studied firstly by Lee et al. [52] and lately by Takriti et al. [53]. The main result of this series of contributions is the convex hull description for min up/down time inequalities, given by:

$$u_{gt} - u_{gt-1} = v_{gt} - w_{gt} \quad \forall t \in \mathcal{T} \tag{2.6}$$

$$\sum_{i=t-TU_g+1}^{t} v_{gi} \leq u_{gt} \qquad \forall g \in \mathcal{G}, \forall t \in [TU_g, T] \tag{2.7}$$

$$\sum_{i=t-TD_g+1}^{t} w_{gi} \leq 1 - u_{gt} \quad \forall g \in \mathcal{G}, \forall t \in [TD_g, T] \tag{2.8}$$

where u_{gt} represents the state of the generator g at time t, v_{gt} takes the value of 1 if generator g was turned on at time t and 0 if not, and w_{gt} has the same definition as v_{gt} but considering shutdowns. TU_g and TD_g are the minimum up and down times of generator g, respectively.

A few years later, other works took these contributions to build specific UC models. For example, in [54] the 1–bin model was developed for the thermal generators, including a piecewise linear description for startup costs. Lately, based on the results obtained by Takriti et al. [53], the works [3, 55] developed more efficient descriptions, namely 3–bin models, including new formulations for ramp constraints and novel approximations for startup costs and non-linear production costs. In the same line, the work developed by Bendotti et. al. [56] provides an extension of [53] by constructing a new set of valid inequalities. In [57] the computational complexity of the UC is analyzed, founding that UC is strongly NP–hard and NP–hard even for instances with a time horizon $T = 1$.

On the other hand, network–flow-based formulations use a slightly different approach. In [28], Guan et al. described the generation schedule of each unit is as an acyclic network. Fig. (2.4) shows a graph representation of a generator schedule. The number of on/off states is defined by the minimum up/down times, respectively, and a feasible path represents a feasible operation. The method's key idea is to use a Lagrangian Decomposition (LR) scheme for solving the multi-generator problem with relaxed side constraints. In contrast, the individual schedule for each generator is solved independently by using Dynamic Programming (DP). This approach did not include ramping constraints because of the time–coupling that those insert in the single generator problems, making the DP algorithm no longer helpful.

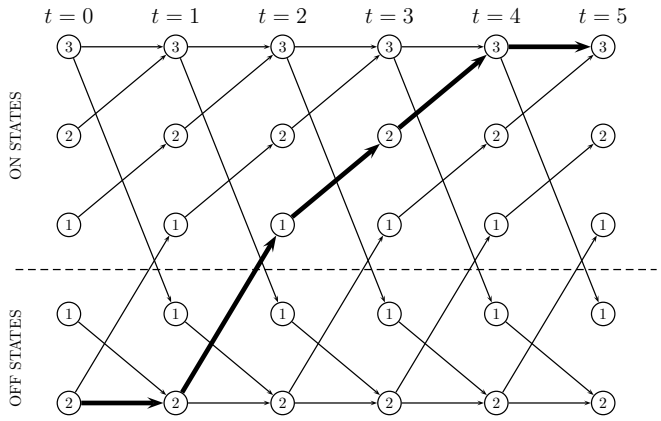


Figure 2.4: Graph representation of a generator schedule

The incorporation of ramping constraints was addressed lately by the same authors in [29], but only considering piecewise linear cost functions for the generators and showing expensive computation cost when dealing with general convex functions. Frangioni and Gentile solve this issue[30], where a method to solve the single UC (considering a single unit) with ramps constraints and general convex costs functions is presented. The main idea of this proposal is constructing *interval graphs*, where each node represents a feasible on/off cycle for each unit in the sense of

meeting minimum up/down times, and lately solving the economic dispatch (ED) for every node. This approach allows introducing ramps constraints in the problem without affecting the typical structure of a DP problem since the costs obtained by solving the individual ED's for every feasible operation are introduced in the graph as fixed costs. The resulting problem has the structure of the shortest path problem, which is solvable in polynomial time for acyclic graphs [51].

Although this approach showed being helpless when dealing with multi-generator UC compared to other tights and compact formulations (as 3-bin and related models), the single generator network–flow description has inspired the development of new tools in the representation of the polyhedral structure of the UC. In this line, Ostrowski and Knueven [58] constructed a compact extended formulation for a single generator. Then, a cut-generating linear program uses this extended formulation that can be used as a callback in a UC mixed-integer programming model.

Other recent works also have developed a polyhedral study of the UC. In [59, 60], a convex hull description of the minimum up/down polytope with ramping constraints is formulated for exceptional ramp rates values and different planning horizon lengths, showing good theoretical results but still ineffective for general instances.

This series of contributions, both in the polyhedral study and linear approximations, were collected and gave birth to the 3-bin model presented in 2013 by Morales-España et al. This is considered one of the best MILP formulations in the literature at the moment, showing remarkable performance against previous formulations [3]. A specific description of the *3–bin* model is made in the below subsection.

The *3–bin* model

In this subsection, a detailed presentation of the *3-bin* model is made to use it in the next chapter. These equations describe the feasible operation of thermal generators and a piece–wise approximation of the startup costs. Variables u_{gt}, v_{gt}, and w_{gt} have the same meaning that in equations (2.6) - (2.8).

1. Generation limits:

$$p_{gt} + r_{gt} \leq (\overline{P}_g - \underline{P}_g)u_{gt} - (\overline{P}_g - SU_g)v_{gt}, \qquad \forall g \in \mathcal{G}^1, t \tag{2.9}$$
$$p_{gt} + r_{gt} \leq (\overline{P}_g - \underline{P}_g)u_{gt} - (\overline{P}_g - SD_g)w_{g,t+1}, \qquad \forall g \in \mathcal{G}^1, t \tag{2.10}$$
$$p_{gt} + r_{gt} \leq (\overline{P}_g - \underline{P}_g)u_{gt} - (\overline{P}_g - SU_g)v_{gt} - (\overline{P}_g - SD_g)w_{g,t+1} \quad \forall g \notin \mathcal{G}^1, t \tag{2.11}$$

2. Ramps constraints:

$$(p_{gt} + r_{gt}) - p_{g,t-1} \leq RU_g \quad \forall g, t \tag{2.12}$$
$$-p_{gt} + p_{g,t-1} \leq RD_g, \qquad \forall g, t \tag{2.13}$$

3. Minimum up/down times:

$$(2.6), (2.7) \text{ and } (2.8).$$

4. Start–up costs:

$$\delta_{gst} \leq \sum_{i=T_{gs}^{SU}}^{T_{g,s+1}^{SU}-1} w_{g,t-i} \quad \forall g, t \in [T_{g,s+1}^{SU}, T],\ s \in [1, \mathcal{S}_g) \tag{2.14}$$
$$\sum_{s \in \mathcal{S}_g} \delta_{gst} = v_{gt} \qquad \forall g, t \tag{2.15}$$

The above equations model the startup costs of thermal generators. These costs often have an exponential dependence on the offline time of the unit, usually modeled via a piecewise linear approximation. Variable δ_{gst} represents the startup type s of the generator g in the time t, $T_{g,s+1}^{SU}$. Equation (2.14) determines the time since the last shutdown, and (2.15) activates at most one startup cost C_{sg}^{SU}.

5. Total costs:

$$\sum_{g\in\mathcal{G}}\sum_{t\in\mathcal{T}} \quad [C_g^{NL}u_{gt} + C_g^{LV}(\underline{P}_g u_{gt} + p_{gt}) + \sum_{s\in\mathcal{S}_g} C_{sg}^{SU}\delta_{gst} + C_g^{SD}w_{gt} + C^{NSE}nse_t] \tag{2.16}$$

The production costs (2.16) are formulated as linear functions of the dispatched power, however, there exists tighter approximations [61]. $C_g^{NL}u_{gt}$ and C_g^{LV} represent the *no–load costs* and the linear production cost, respectively. The second part of the costs in equation (2.16) is composed of 3 parts: $C_{sg}^{SU}\delta_{gst}$ is the startup cost, $C_g^{SD}w_{gt}$ the shutdown costs and $C^{NSE}nse_t$ the cost for non–served demand.

The performance of the *3–bin* model was tested for different variations of equations (2.9)-(2.16) against the well–known *1–bin* model [54]. In particular, in [3], extensive computational experiments were developed, showing the better performance of the *3bin* model against the *1bin* one: tables 2.1 and 2.2 show tables extracted in their entirety from [3]. Experiments were developed for four different formulations, namely *1bin*, *3bin*, *P1*, and *P2*. *1bin* formulation is the same as the presented in [54], and *3bin* uses the equations of *1bin* but considering minimum up/down times formulated by equations (2.6)-(2.8). *P1* is the same as *3bin*, but using start–up costs definition as in (2.14)-(2.15), and *P2* is the complete formulation (2.6)-(2.16).

On the other hand, test cases were separated into two groups of different scales. Cases 01-10 posses several total generators varying from 28 to 54, while the number of generators in cases 11-20 vary from 132 to 187. Finally, *x7–day* and *x10–gen* refer to different methodologies to evaluate the cases. The first run a UC problem for 7 days with variable demand values. In contrast, the second methodology replicates 10 times the existing generators, allocating the copies at the same buses as the original instances and solving for a single day.

As can be seen, both tables show that all models involving the generator's operation description using three binary variables over–perform the *1bin* model. It is interesting to appreciate that only by describing minimum up/down constraints with three binary variables, computational times improve by 40% on average. Then, working on this base, *P1* and *P2* models add more reformulations, improving, even more, the presented indexes, with *P2* resulting the better formulation concerning computational times.

		CPU Time			Integrality Gap			Opt. Tolerance			Nodes			Iterations		
		3bin	*P1*	*P2*	*3bin*	*P1*	*P2*	*3bin*	*P1*	*P2*	*3bin*	*P1*	*P2*	*3bin*	*P1*	*P2*
x7-day	Cases 01-10	120.9	33.7	9.7	73.3	59.8	45.8	97.3	59.9	47.0	88.9	105.6	57.7	42.9	33.0	14.5
	Cases 11-20	140.6	15.4	4.9	75.3	58.1	38.5	130.1	5.1	5.4	169.8	69.1	96.2	72.4	21.4	11.7
	Cases 01-20	130.4	22.8	6.9	74.3	59.0	42.0	112.5	17.4	16.0	122.8	85.4	74.5	55.7	26.6	13.0
x10-gen	Cases 01-10	36.3	22.0	12.2	64.0	47.6	34.7	71.2	8.0	11.1	136.5	122.6	121.7	56.2	45.8	35.7
	Cases 11-20	45.4	8.4	4.2	74.5	55.2	42.1	57.0	0.6	0.7	189.7	99.1	160.4	56.9	19.5	15.6
	Cases 01-20	40.6	13.6	7.1	69.1	51.3	38.3	63.7	2.3	2.9	160.9	110.2	139.7	56.6	29.9	23.6

Table 2.1: Table V of [3]. Computational performance compared with *1bin* (%).

	3bin over 1bin	*P1 over 1bin*	*P2 over 1bin*	*P1 over 3bin*	*P2 over 3bin*	*P2 over P1*
Cases 01-10	1.5	3.7	9.2	2.4	6.1	2.5
Cases 11-20	1.3	8.8	22.1	7.0	17.7	2.5
Cases 01-20	1.4	5.7	14.3	4.1	10.4	2.5

Table 2.2: Table VI of [3]. Overall speedups.

Given these results, the present study on the UC problems uses the *P2* formulation as a base for the analysis considering parameters subject to uncertainty. As is shown in the exposition made above, the study of better every time formulations of the UC is an active field of research. The development of tighter polyhedral descriptions and cutting planes generator algorithms are still of great interest since UC is the core of even more complex problems used in industry. For example, problems considering possible system contingencies or with the presence of parameters with unknown values could need the resolution of several instances of the deterministic UC, making the chosen base formulation more dramatic.

2.2.3 The rolling horizon solution scheme

As was seen, the UC problem gives the solution of units schedules and reserves levels for a period that can go from few hours to several days. This calculation needs the generation of values for the uncertain parameters at every planning hour, usually obtained by forecasting models, which use observed stored values.

When dealing with systems with massive penetration of non–conventional REG sources, i.e., with a considerable level of uncertainty, two main problems take place when calculating and implementing schedule solutions for a significant number of hours. [62]. Firstly, the current solution could not capture the dynamics of the uncertainty values during the implementation period. Then, it is mandatory to wait until the next calculation stage to incorporate the new information. In the second place, the forecasting errors grow with the number of prediction steps, even with perfect knowledge of the random process [63], resulting in high prediction errors for the values of the most distant hours.

Therefore, if the forecast errors are significant, the probability of having big differences between planned and actual generators dispatches is considerable, increasing the use of system reserves, producing high costs of operation, and even compromising its security [62].

In such wise, the rolling horizon scheme takes over these drawbacks by increasing the frequency of the calculation process and by diminishing the number of hours on which solutions will be implemented.

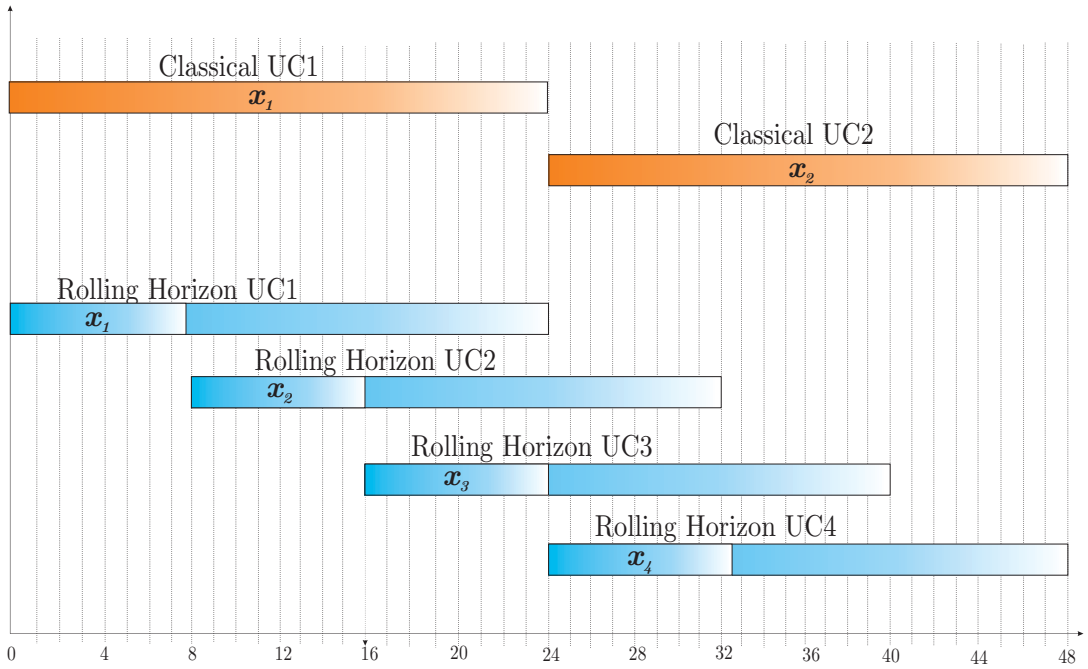


Figure 2.5: Rolling horizon UC schedules vs classical UC schedules. In this case, rolling horizon UC models are solved each 8 hours, obtaining solutions for 24 hours ahead, where only the first eight are implemented. Implementation delays are also show, being lower for the rolling horizon scheme since its higher solving frequency.

In Fig. 2.5, solutions calculation and implementation are illustrated for both classical and rolling horizon solution schemes. As can be seen, the rolling horizon scheme keeps the original horizon of planning for the UC, in this case, equal to 24 hours, but only the solutions of the first eight hours are implemented. In this sense, 3 UC instances have to be carried out to obtain the daily schedule, representing a more intensive use of computational resources. Notwithstanding, the benefits are three-fold: updating data is more often, errors on values forecast of distant hours have less impact on the solutions, and the delay between calculation and implementation stages is reduced.

Rolling horizon's schemes have been studied in control problems on dynamic systems [64], the same as in the context of the UC problem [62, 65, 66].

2.3 Optimization under uncertainty

Reviewed the importance of choosing a suitable formulation for the UC problem. In general, for every optimization problem, let us analyze what happens when some problem parameters have unknown values. Optimization under uncertainty, classically referred to as *stochastic optimization* (SO), is a branch of mathematics that studies optimization problems involving parameters described as random variables. As reviewed in [67], a general enough framework to include all the existing explored branches of study on optimization under uncertainty does not exist. Just for the mention, topics such as *optimal control*, *stochastic dynamic programming* and *multistage optimization* are only three of many approaches that can be taken to model a system subject to uncertain events. Generally speaking, each branch uses different modeling approaches, resulting in different solution techniques that often diverge from one to another. Therefore, the engineer or mathematician who models a particular system chooses one or combines various existing approaches to capture the system's behavior and the decisions that have to be made.

In power systems planning and operation, uncertainty can come from different sources, namely fuel prices, components failure rates, renewable sources, demand levels, and others. In particular, the present thesis is focused on the analysis of the last two elements in the context of the UC problem, where the typical approach to include the uncertainty is through stochastic programming (SP), which study the case when random variables describe the parameters of a standard optimization problem. Therefore, objective functions and left/right–hand side of constraints also becomes random variables, making necessary the incorporation of some criteria for defining appropriately the new objective and new constraints that will give helpful solutions. For a better understanding, consider a general linear problem:

$$\min \quad \boldsymbol{c}^T \boldsymbol{x} \tag{2.17}$$

$$\text{s.t.} \quad A\boldsymbol{x} \leq \boldsymbol{b} \tag{2.18}$$

$$x_i^{min} \leq x_i \leq x_i^{max}, \quad \forall i \tag{2.19}$$

If $\boldsymbol{c}$, A and $\boldsymbol{b}$ are random variables, the problem (2.17)-(2.19) turns ambiguous since the interpretation of the objective and constraints is unclear. This ambiguity occurs because probabilistic distributions now describe $\boldsymbol{c}^T\boldsymbol{x}$, $A\boldsymbol{x}$ and $\boldsymbol{b}$. Then, even fixing $\boldsymbol{x}$ to some value $\boldsymbol{x}^*$, the values of $\boldsymbol{c}^T\boldsymbol{x}^*$, $A\boldsymbol{x}^*$ and $\boldsymbol{b}$ could differ with different probabilities. At this point, the mathematician or engineer solving the problem has to decide, with some criteria, how to interpret the new problem and give it some sense to make it clear and well defined. Then, concerning the objective function, it is often decided to focus on some probabilistic cost indicator, e.g., the expected cost. Concerning the constraints, they are usually forced to be met with a certain level of probability $1-\varepsilon$, with $\varepsilon \in [0,1]$ often small. With this considerations, the problem (2.17)-(2.19) becomes:

$$\min \quad \mathbb{E}[\boldsymbol{c}^T \boldsymbol{x}] \tag{2.20}$$

$$\text{s.t.} \quad \mathbb{P}[A\boldsymbol{x} \leq \boldsymbol{b}] \geq 1-\varepsilon \tag{2.21}$$

$$x_i^{min} \leq x_i \leq x_i^{max}, \quad \forall i \tag{2.22}$$

$$(\boldsymbol{c}, A, \boldsymbol{b}) \sim \mathcal{D} \tag{2.23}$$

This problem has a precise meaning, with both expressions. $\mathbb{E}[\boldsymbol{c}^T\boldsymbol{x}]$ and $\mathbb{P}[A\boldsymbol{x} \leq \boldsymbol{b}]$, being deterministic functions of $\boldsymbol{x}$, and depending on the density probability function of $\mathcal{D}$. The formulation above represents a general case in which every parameter is a random variable. In general, problem (2.20)-(2.23) is tough to solve, motivating the allocation of the parameters subject to uncertainty either in the objective function or on some constraints. Different mathematical programming structures are generated depending on the selected criteria, needing special care and attention for each separated case. These structures have been widely studied, giving place to the classical paradigms in optimization under uncertainty, namely SP and chance-constrained programming (ChCP). On the other hand, more novel approaches with other interpretations and mathematical structures have been proposed, specifically robust optimization (RO) and distributionally robust optimization (DRO), which will be addressed correspondingly in subsection 2.3.2.

Most of its applications in engineering systems are used on the framework of sequential–decision optimization problems. There, the main task is sequentially deciding how to act in light of the sequential realization of unknown parameters. One of the branches of mathematical optimization that study this problem is called multistage optimization, widely used in power systems planning and operation.

2.3.1 Multistage optimization

In planning problems, when decisions have to be made sequentially, multistage optimization raises as a powerful tool for decision-makers, especially when data uncertainty is involved. In the UC problem, generator statuses and dispatches levels are determined without knowing the future values of demand and renewable generation levels. Then, after observing these values, the system has to be re-dispatched due to the discrepancies between predicted and actual values, keeping a safety balance between energy generation and consumption.

The most typical framework used is the two-stage formulation, composed of three principal elements: the first stage variables $\boldsymbol{x}$, also called "here and now" decisions, the parameters subject to uncertainty $\boldsymbol{\xi}$, and the second stage variables $\boldsymbol{y}$, informally known as "wait and see" decisions. The philosophy behind this structure tries to represent a dynamic decision-making process as a function of the available information. "Here and now" decisions have to be made before the realization of the uncertainty. Lately, when complete information is available, "wait and see" decisions fulfill the role of corrective actions.

$$\text{here and now} \rightarrow \text{observation of data} \rightarrow \text{corrective actions}$$
$$\text{first stage } (\boldsymbol{x}) \longrightarrow \text{data realization } (\boldsymbol{\xi}) \longrightarrow \text{second stage } (\boldsymbol{y})$$

The critical point of the formulation is how to determine a good value for the first stage variables only with previous data or limited information of the uncertainty parameters and anticipating possible future decisions. Many criteria can be chosen to measure the importance of future decisions. Usually, these criteria have a relationship with the objectives pursued and the risk level accepted by the decision-maker. However, a typical mathematical structure is shared, and a general formulation can be stated, which is following presented for a linear problem considering two stages:

$$\min_{\substack{\boldsymbol{x}\in X \\ \boldsymbol{\xi}\in\Xi}} \; \boldsymbol{c}^T\boldsymbol{x} + \rho\left(Q(\boldsymbol{x},\boldsymbol{\xi})\right) \tag{2.24}$$

where $Q(\boldsymbol{x},\boldsymbol{\xi})$ is the optimum value of the second stage problem:

$$\min_{\boldsymbol{y}\in Y(\boldsymbol{x},\boldsymbol{\xi})} \; \boldsymbol{d}^T\boldsymbol{y} \tag{2.25}$$

In this setting, the set Ξ contains all the possible realizations of the uncertainty vector parameter $\boldsymbol{\xi}$. X is a polyhedron containing the feasible values of $\boldsymbol{x}$ and $Y(\boldsymbol{x},\boldsymbol{\xi})$ is the polyhedron of the feasible second–stage values, which depends on $\boldsymbol{x}$ and $\boldsymbol{\xi}$. The function ρ represents the criteria taken by the decision-maker to measure the impact of second stage costs, accordingly with the possible realizations of the uncertainty and the selected first–stage variables.

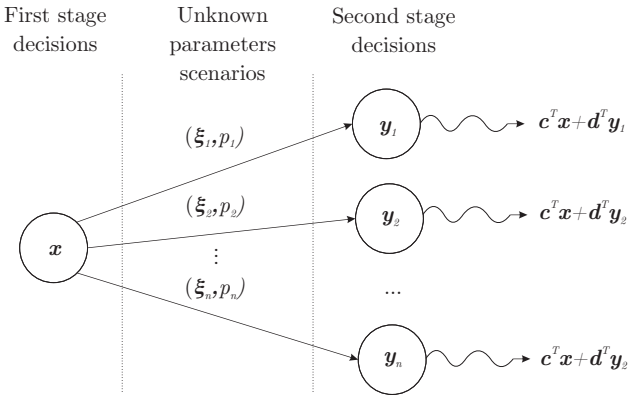


Figure 2.6: Diagram of a two–stage decision sequence for a problem with discrete number of scenarios.

In order to see the structure of a two-stage problem graphically, Fig. 2.6 shows a diagram of decision three for a two–stage problem with a discrete number of scenarios. As can be seen, the first stage decision $\boldsymbol{x}$ holds for every scenario k, in which everyone has a second stage variable $\boldsymbol{y}_k$ associated. In this case, every scenario is defined by a parameter value $\boldsymbol{\xi}_k$ and a probability

value p_k. Every second–stage decision $\boldsymbol{y}_k$ is optimal, given the values $\boldsymbol{x}$ and $\boldsymbol{\xi}_k$, resulting in a second–stage cost $\boldsymbol{d}^T\boldsymbol{y}_k$.

This formulation can be extended to more stages when the uncertainty information is revealed sequentially, and more sequential decisions must be made. The resulting model consists of a series of nested problems with the following structure:

$$\min_{\boldsymbol{x}_0 \in X_0} \; \boldsymbol{c}_0^T \boldsymbol{x}_0 + \rho \left(\min_{\boldsymbol{x}_1 \in X_1(\boldsymbol{x}_0, \boldsymbol{\xi}_1)} \; \boldsymbol{c}_1^T \boldsymbol{x}_1 + \rho \left(.... + \rho \left(Q(\boldsymbol{x}_{T-1}, \boldsymbol{\xi}_T) \right) \right) \right) \tag{2.26}$$

Considering the structure of the problem (2.26), detailed formulations and specific solution methods are strongly dependent on the function ρ. Besides, uncertainty modeling is crucial since the computational complexity. Considering continuous probability distributions is generally intractable, being necessary discretizations of the underlying random process. To this end, scenario trees are usually constructed, depicting all the possible sequences of the random process and formally formulating the multistage problem. Fig. 2.7 shows an extension of Fig. 2.6 for a multi–stage problem. As can be seen, the number of variables grows exponentially with the number of stages. A deep analysis of these problems is made in [68].

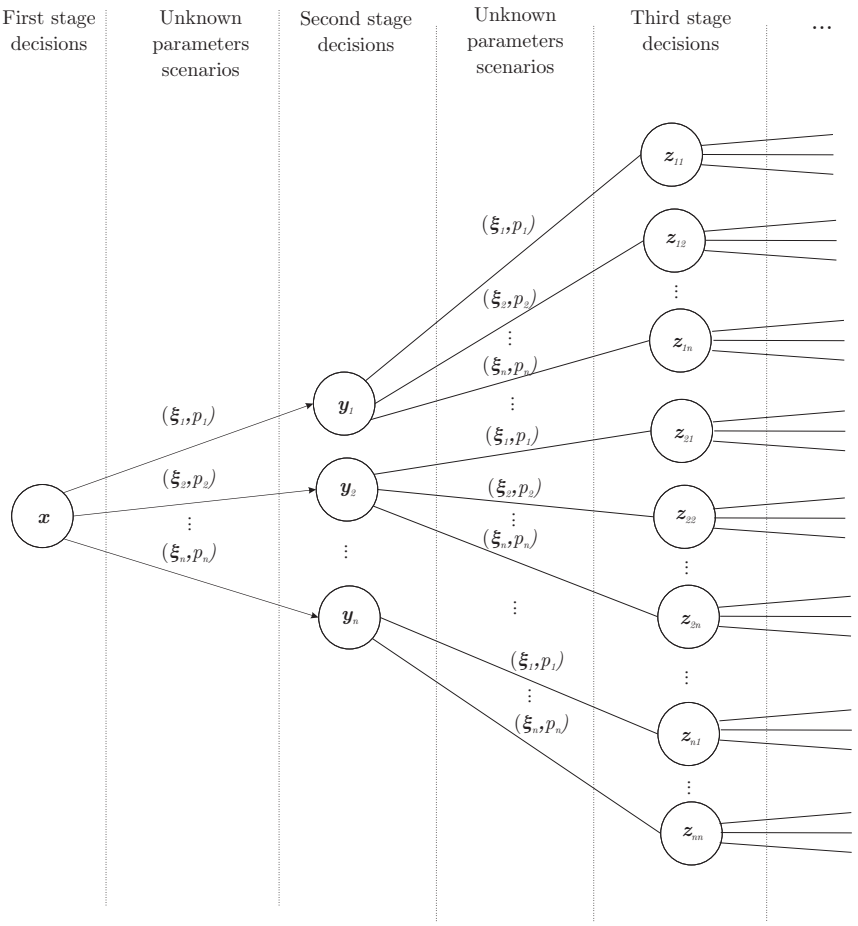


Figure 2.7: Diagram of a two–stage decision sequence for a problem with discrete number of scenarios.

Given the interpretation of the above formulations, applications of multistage formulations are extensive in sequential decision problems. A well–known approach is the Stochastic Dual Dynamic Programming (SDDP), widely used in the context of hydrothermal coordination [69, 70, 71, 72]. On the other hand, a two-stage formulation often addresses the UC problem considering uncertain demand and renewable generation values. As was reviewed in section 2.1, in the day ahead calculations, i.e., the UC resolution, generators schedules, and reserves levels are calculated based on the available information. However, in the RT market, new offers are made to supply the imbalances given by prediction errors. In this way, two-stage UC selects schedules and reserves

levels based on different possible values of future renewable generation. To this end, different modeling approaches are reviewed in the context of two-stage formulations.

2.3.2 Traditional paradigms in optimization under uncertainty

As described in the formulation of problem (2.20)-(2.23), is necessary to include some criteria when working uncertainty data. Each criterion will produce different problem interpretations and formulations, which require different resolution methods, producing different solutions. In this section, traditional methods for handling the uncertainty into short–term planning problems are presented. This presentation will be made using two-stage formulations.

Stochastic programming

aims to optimize the expected value of the cost function across all the possible realizations of the unknown parameter. The principal feature of SP is using probability distributions to include uncertainty into the problem, usually discretized in scenario trees. The structure of a two–stage stochastic problem is the following:

$$\min_{\boldsymbol{x}\in X} \ \boldsymbol{c}^T\boldsymbol{x} + \mathbb{E}[Q(\boldsymbol{x},\boldsymbol{\xi})] \tag{2.27}$$

Accordingly, the first-stage variable is calculated to minimize the costs of the first stage plus the expected costs of the second stage. In terms of solvability, the problem is convex if the recourse variable is linear. If the probability distribution is discrete, the general structure for a two-stage linear stochastic problem is the following:

$$\min \ \boldsymbol{c}^T\boldsymbol{x} + \sum_{i\in\mathcal{S}} p_i \boldsymbol{d}^T\boldsymbol{y}_i \tag{2.28}$$

$$\text{s.t.} \quad A\boldsymbol{x} \leq \boldsymbol{b} \tag{2.29}$$

$$H_i\boldsymbol{y}_i \leq \boldsymbol{h}_i \qquad \forall i \in \mathcal{S} \tag{2.30}$$

$$G_i\boldsymbol{x} + E_i\boldsymbol{y}_i \leq \boldsymbol{g}_i \quad \forall i \in \mathcal{S} \tag{2.31}$$

with $\mathcal{S}$ the scenario set and p_i the associated scenario probabilities. As can be seen, the objective costs involve the calculation of the second stage cost for every scenario, which could be problematic when the number of them grows. This issue has inspired the design of decomposition algorithms [73] or scenario reduction algorithms [74], which are often needed.

Chance–constrained programming

ChCP models were proposed for the first time by Charnes and Cooper in [75], and as the SO approach, the randomness of uncertainty parameters are managed with probability distributions. The main difference is that ChCP models seek to minimize the objective function while meeting a certain number of constraints with a given degree of probability. The structure of a single-stage linear ChCP model is the following:

$$\min_{\boldsymbol{x}\in \mathrm{X}} \ \boldsymbol{c}^T\boldsymbol{x} + \mathbb{E}[\boldsymbol{d}^T\boldsymbol{y}] \tag{2.32}$$

$$\text{s.t.} \quad \mathbb{P}[G(\boldsymbol{x},\boldsymbol{y}(\boldsymbol{\xi})) \leq 0] \geq 1-\varepsilon \tag{2.33}$$

$$\boldsymbol{\xi} \sim \mathcal{P} \tag{2.34}$$

In the model above, equation (2.33) ensures that the solution satisfies at least, with probability $1-\varepsilon$, under probability function $\mathbb{P}$, that the set of constraints $G(\boldsymbol{x},\boldsymbol{y}(\boldsymbol{\xi})) \leq 0$ are jointly meet. ChCP is often used since it gives flexibility to the decision-maker to control the balance between the cost and the reliability through ε.

Despite the flexibility and advantages of modeling, the framework possesses serious drawbacks in computational implementation and resolution. Firstly, the probability calculation in equation (2.33) involves the computation of a multivariate integral, which is generally computationally intractable. In the second place, (2.33) in non-convex, even for discrete probability distributions, X convex and the polyhedron defined by $G(x,\boldsymbol{y}(\boldsymbol{\xi})) \leq 0$ being convex in $\boldsymbol{x}$ [76].

In light of the facts, a large amount of the associated literature has focused on reformulations, approximations, and solution algorithms for chance-constrained problems. In [77, 78], a

ChCP formulation is solved by using the *sample average approximation* (SAA) method. In [79], a conservative reformulation of the chance constraint (2.33) is made for the single constraint case. The conditional value–at–risk (CVaR) measure is used to achieve a convex relaxation. However this convexification has been proved to single chance–constraints, where the application to joint constraints is an active field of research [19].

Robust optimization

One of the most novel and promising approaches to deal with uncertainty in optimization problems is RO. The first big difference with previously described paradigms is that RO avoids using probability distributions. Instead, RO focuses on designing uncertainty sets containing all the possible realizations of the unknown parameters. Depending on the context and the decision-maker's desire, the robustness sense can vary. It can be the realization that provokes the largest constraint violation, the one producing the biggest cost function value, or the realization that produces the lowest asset return [80]. A two-stage robust formulation can be stated as follows:

$$\min_{\boldsymbol{x}\in \mathrm{X}} \left\{ \boldsymbol{c}^T \boldsymbol{x} + \max_{\boldsymbol{\xi}\in\Xi} Q(\boldsymbol{x}, \boldsymbol{\xi}) \right\} \tag{2.35}$$

A remarkable feature of RO is the high dependence of the solution's quality on selecting the uncertainty set, turning its design into a critical point. A set containing unlike-to-happen realizations could produce over-conservative solutions, i.e., with high reliability and high cost [80, 4]. A typical example of an over–conservative choice of uncertainty set is the conventional box uncertainty set, which is defined as follows [81]:

$$\Xi_{\text{box}} = \left\{ \boldsymbol{\xi} : \xi_i \in [\xi_i^L, \xi_i^U],\ \forall i \in I \right\} \tag{2.36}$$

The box uncertainty set Ξ_{box} contains all the values of $\boldsymbol{\xi}$ such that every component ξ_i, with $i \in I$ the set of indexes of the parameters subject to uncertainty, is bounded between a minimum and maximum value, ξ_i^L and ξ_i^U respectively. In this case, the extreme points of Ξ_{box} are unlike–to–happen scenarios that give rise to costly solutions. Moreover, with this kind of set, the conservativeness of the solution can barely be controlled. Motivated by this issue, Bertsimas and Sim [82] proposed the budget uncertainty set:

$$\Xi_{\text{budget}}(\gamma) = \left\{ \xi_i : \sum_{i\in I} \frac{|\xi_i - \bar{\xi}_i|}{\hat{\xi}_i} \leq |I|\gamma,\ \xi_i \in [\bar{\xi}_i - \hat{\xi}_i, \bar{\xi}_i + \hat{\xi}_i],\ \forall i \in I \right\} \tag{2.37}$$

In this case, $\bar{\xi}_i$ is the nominal or expected value for the *ith* component, while $\hat{\xi}_i$ is the maximum deviation of the measures from $\bar{\xi}_i$. The parameter γ, also called uncertainty budget, controls the size of $\Xi_{\text{budget}}(\gamma)$ and concedes to the decision–maker to move from a full robust box set ($\gamma = 1$) to the deterministic case ($\gamma = 0$).

Distributionally robust optimization

The distributionally robust optimization (DRO) approach uses similar criteria to robust optimization, considering that the uncertainty is present in the probability distribution. In this way, DRO assumes that the real distribution belongs to a set called ambiguity set, defined by specific probabilistic requisites.

For example, consider that the only available information about a multidimensional random variable $\boldsymbol{\xi}$ is its expected value $\bar{\boldsymbol{\xi}}$ and its covariance matrix K. In this case, the unknown distribution $\mathbb{P}$ can be modeled as belonging to an ambiguity set $\mathcal{P}$ defined by:

$$\mathcal{P} := \left\{ \mathbb{P} \in \mathcal{M} \mid \mathbb{E}_{\mathbb{P}}[\boldsymbol{\xi}] = \bar{\boldsymbol{\xi}},\ \mathbb{E}_{\mathbb{P}}\Big[(\xi_i - \bar{\xi}_i)(\xi_j - \bar{\xi}_j)\Big] = K_{ij} \right\} \tag{2.38}$$

Then, similar to RO, DRO seeks the works case distribution in $\mathcal{P}$, optimizing a specific cost function. In particular, a two–stage DRO model is usually formulated as:

$$\min_{\boldsymbol{x}\in \mathrm{X}} \left\{ \boldsymbol{c}^T \boldsymbol{x} + \max_{\mathbb{P}\in\mathcal{P}} \mathbb{E}_{\mathbb{P}}[Q(\boldsymbol{x}, \boldsymbol{\xi})] \right\} \tag{2.39}$$

At first sight, the problem (2.39) is very similar to (2.35) since the max–min relationship between the unknown parameters and decision variables. Indeed, the same as RO, the complexity of solving (2.39) heavily depends on the definition of $\mathcal{P}$. However, reformulations and solution approaches could be very different due to the different nature of uncertainty and ambiguity sets.

The four above–mentioned paradigms are fundamental, and each one has its limitations. However, these paradigms serve as a base for constructing more complex approaches, involving combinations of them or implementing other techniques. In the next section, more complex development of the presented methods is described in the context of the vast amount of data availability.

2.4 *Data–driven* optimization models

The explosion of the available data witnessed in the last decade has motivated the development of new procedures in operation research. In the particular field of optimization under uncertainty, *data–driven* optimization raises as a powerful paradigm for the modeling of uncertainty on optimization problems. Unlike the conventional mathematical programming approaches, data-driven techniques do not presume the existence of an exact pre–given model for the unknown parameters. Instead, they all focus on the practical setting using only uncertainty available data.

Fig. 2.8 shows how the available data is used in classical optimization models versus the use in data-driven models. The first use the data to construct probability distributions, needing assumptions that could not relate to with the underlying random process. Conversely, data-driven models directly use the data to construct mathematical structures embedded in the optimization model. This methodology leads to solutions heavily dependent on the in–sample data and the intern criteria selected to evaluate it.

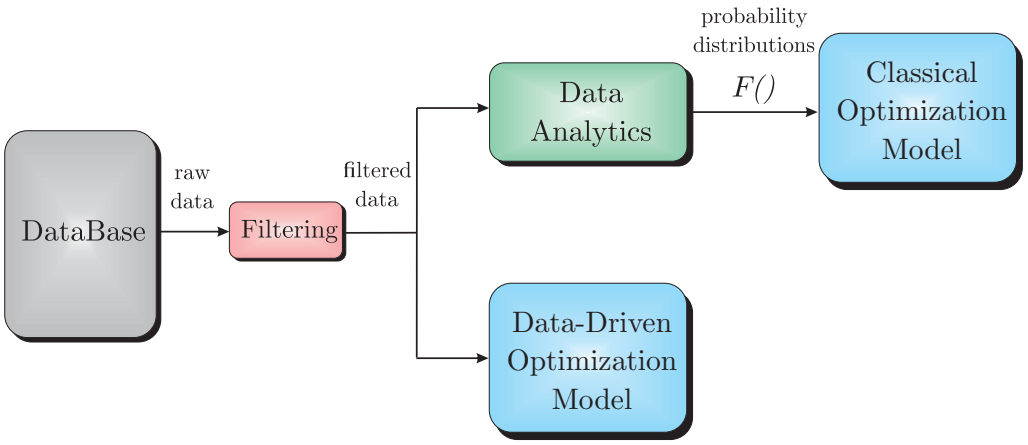


Figure 2.8: Two paradigms in optimization models with uncertainty management.

Consequently, the data-driven paradigm could be used along with each of the three previously described methods of uncertainty management. Therefore, a review of the data-driven versions of SO, ChCO, and RO models is made in the following lines.

2.4.1 Data–driven stochastic optimization and distributionally robust optimization

The core of SO is the use of probability distributions to model the behavior of uncertain parameters. As studied in [83], estimating the true probability distribution from the data is challenging, and considering a single distribution could drive poor out-of-sample performance. In this line, SAA methods are suitable solution approaches that can help to overcome these drawbacks. SAA produces a series of problems and solutions, on which a convergence criterion defines when a solution is optimal [84, 85, 86]. Another famous approach is distributionally robust optimization (DRO), which assumes a particular grade of uncertainty on the empirical distribution obtained from the data. To this end, consider a set of probabilities distributions $\mathcal{P}$ that share some properties. Then,

the DRO approach, applied to a two-stage model, can be stated in the following form:

$$\min_{\boldsymbol{x}\in \mathrm{X}} \boldsymbol{c}^T\boldsymbol{x} + \max_{\mathbb{P}\in\mathcal{P}} \mathbb{E}_{\mathbb{P}}[Q(\boldsymbol{x},\boldsymbol{\xi})] \tag{2.40}$$

As can be seen, DRO combines RO and SO paradigms by considering the work–case of the probability distribution on the set $\mathcal{P}$, also called the ambiguity set. Then, the decision-maker can include partial information of distribution from the data while maintaining the tractability of the formulation. As in RO, the selection of the ambiguity set is a critical point in the performance of DRO. According to [87], the decision-maker needs to consider tractability, statistical meaning, and performance. Tractability accounts for a solvable structure of the problem. On the other hand, the statistical meaning is accounted for by various proposals of ambiguity sets. For example, DRO approaches using mean and second–moment matrix information is solvable in polynomial time [88], where $\mathcal{P}$ is defined as:

$$\mathcal{P} = \left\{ \mathbb{P}\in\mathcal{M} \;\middle|\; \begin{array}{l} \mathbb{P}[\boldsymbol{\xi}\in\Xi] = 1 \\ (\mathbb{E}_{\mathbb{P}}[\boldsymbol{\xi}] - \boldsymbol{\mu})^T\Sigma^{-1}(\mathbb{E}_{\mathbb{P}}[\boldsymbol{\xi}] - \boldsymbol{\mu}) \leq \psi_1 \\ \mathbb{E}_{\mathbb{P}}\left[(\boldsymbol{\xi}-\boldsymbol{\mu})(\boldsymbol{\xi}-\boldsymbol{\mu})^T\right] \leq \psi_2\Sigma \end{array} \right\} \tag{2.41}$$

In the above definition, $\mathcal{M}$ is the set of all probability distributions, $\boldsymbol{\xi}$ is the uncertainty vector, Ξ is the support set, i.e., the set containing the historical observations, and $\boldsymbol{\mu}$ and Σ are the empirical mean vector and covariance matrix, respectively. The first constraint ensures that the probability distribution is supported by Ξ. The inequality constraints define the sizes of the first and second moment confidence regions, respectively, controlled by the parameters ψ_1 and ψ_2. As can be seen, the moment–based ambiguity set achieve tractability and statistical meaning. However, it has been reported that a considerable amount of data is necessary to ensure convergence to the real probability distribution, resulting in conservative solutions for a moderate amount of data. Conversely, other kinds of sets, based on metric between probability distributions, could be defined as is showed below:

$$\mathcal{P} := \{\mathbb{P}\in\mathcal{M} \mid d(\mathbb{P},\mathbb{P}_0) \leq \delta\} \tag{2.42}$$

In (2.42), $\mathbb{P}_0$ is a referential distribution, which usually is selected as the empirical, $d(.,.)$ is some metric between probability distributions, and δ is a constant controlling the distance from the referential distribution. A novel metric in the context of power systems is the Wasserstein metric $d_W : \mathcal{M}(\Xi)\times\mathcal{M}(\Xi)\to\mathbb{R}_+$, defined as follows:

$$d_W(\mathbb{P}_1,\mathbb{P}_2) = \inf_{\boldsymbol{\pi}} \left\{ \mathbb{E}_{\boldsymbol{\pi}}[d(\boldsymbol{\xi}_1,\boldsymbol{\xi}_2)] : \boldsymbol{\xi}_1\sim\mathbb{P}_1\,,\ \boldsymbol{\xi}_2\sim\mathbb{P}_2 \right\} \tag{2.43}$$

Here, $\mathcal{M}(\Xi)$ represents the space of probability distributions supported by the dataset Ξ, $d(\boldsymbol{\xi}_1,\boldsymbol{\xi}_2) = ||\boldsymbol{\xi}_1 - \boldsymbol{\xi}_2||$ is the 2–norm of the difference of the random vectors $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$, and $\boldsymbol{\pi}$ the joint distribution of $\boldsymbol{\xi}_1$, $\boldsymbol{\xi}_2$, and $\boldsymbol{\pi}$, with marginals $\mathbb{P}_1$ and $\mathbb{P}_2$. Tractable reformulations and performance guarantees have been studied in [89]. In this way, based on a historical dataset $\Xi = \{\hat{\boldsymbol{\xi}}^1 ..., \hat{\boldsymbol{\xi}}^N\}$, the decision–maker can derive an empirical distribution by associating $\mathbb{P}_0 = \sum_{i=1}^N \frac{1}{N}\delta_{\hat{\boldsymbol{\xi}}^i}$, where $\delta_{\hat{\boldsymbol{\xi}}}$ is the Dirac distribution concentrating unit mass at $\hat{\boldsymbol{\xi}}$.

Ambiguity sets based on the Wasserstein metric have been used in the context of the UC in [90, 91], using two–stage formulations (2.40).

2.4.2 Data–driven chance–constrained optimization

As was reviewed in past sections, SO approaches are similar to ChCO since the use of probability distributions. In this way, SAA methods for ChCO has also been applied [77, 92]. In the case of DRO, the application to chance–constrained models results straightforward by introducing the concept of worst–case probability distribution appropriately, as in the following two–stage formulation:

$$\min_{\boldsymbol{x}\in\mathrm{X}} \quad \boldsymbol{c}^T\boldsymbol{x} + \mathbb{E}_{\mathbb{P}_0}[Q(\boldsymbol{x},\boldsymbol{\xi})] \tag{2.44}$$

$$\text{s.t.} \quad \inf_{\mathbb{P}\in\mathcal{P}} \mathbb{P}[G(\boldsymbol{x},\boldsymbol{y}(\boldsymbol{\xi}))\leq 0] \geq 1-\varepsilon \tag{2.45}$$

Is needed to say that the formulation above is not unique for data–driven ChCO models, e.g., the definition of the second stage costs may vary from one approach to another. The core of the formulation is the probabilistic constraint (2.45), where the robustness principle applies in finding the worst probability distribution $\mathbb{P}$ that achieves the minimum probability of satisfying $G(\boldsymbol{x}, \boldsymbol{y}(\boldsymbol{\xi})) \leq 0$. Reformulations and approximations for ambiguity sets using the Wasserstein metric are addressed in [93, 94]. Other proposals are present in [95, 96].

2.4.3 Data–driven robust optimization

As a fundamental element of robust optimization models, uncertainty sets implicitly determine the optimal solutions for robust optimization problems. Therefore, a correct design is a critical task. The earliest proposals of uncertainty sets that use fixed shapes, as ellipsoidal or box sets, result in shallow capabilities of capturing the structure and behavior of the historical data since strong assumptions are needed [22]. As an example, consider the sets shown in (2.36) and (2.37). In this case, any modification on the data that satisfies $\xi_i \in (\xi_i^L, \xi_i^U), \forall i \in I$, does not change the shape of the set.

This lack of flexibility prompts the design of more sophisticated sets capable of capturing the complexity of the data. In this way, in [97], sets based on the $\text{CVaR}_\gamma(\cdot)$ function are devised, and lately applied in [98]. Concurrently and in the same line, in [20] an uncertainty set constructed as the convex hull of the data points is proposed, showing better performance than budget–based sets (2.37).

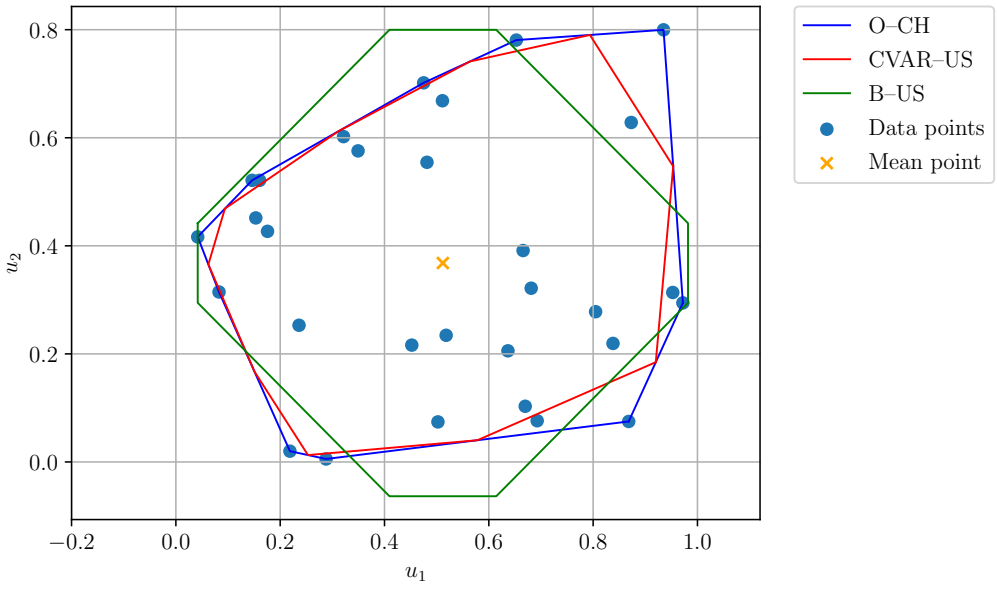


Figure 2.9: Examples of uncertainty sets. O–CH is the convex hull of original data points, CVAR–US is the CVaR–based uncertainty set, and B–US is the uncertainty set defined in (2.37).

Fig. 2.9 contains plots of three different uncertainty sets constructed from the same data, obtained from the random generation of 30 two-dimensional points uniformly distributed between 0 and 1. The green line represents the budget-based set with $\gamma = 0.7$, the red line the CVaR-based set with $\gamma = 0.97$, and the blue line represents the convex hull set. The blue dots are representing the original data, and the orange x is the average of it. It is clear from the Fig., that both CVaR-based and convex hull sets visually follow the shape of the data better than the budget-based set, where out-of-sample experiments made in [98, 20] show its superior performance for different values of γ.

Also, uncertainty sets based on Dirichlet processes and kernel density functions were studied in [99, 100, 101], where principal analysis methods were applied to decompose the correlations involved and then construct the sets based on confidence intervals. Concurrently, in [19], a deep study on uncertainty sets based on hypothesis testing is made, besides the relationship between probabilistic constraints and uncertainty sets.

The conclusions extracted from Fig. (2.9) and the diversity of proposed data–driven uncertainty sets are twofold. In the first place, many types of sets can be constructed from the same dataset, and the performance, statistical inference, and tractability could differ drastically. For example, budget–based sets are constructed from linear inequalities, constituting easy–to–implement formulations, resulting in a model suitable to be solved with Benders decomposition methods [4]. On the other hand, CVaR–based uncertainty sets are constructed from the definition made in [98], which results in costly computational calculations when the number of samples grows. For scenario–based uncertainty sets, implementations are straightforward, and solutions algorithms rely on the column–and–constraint generation technique [102, 20]. Secondly, the variations of the parameters associated with each set definition could significantly change the model's performance without adding mathematical or computational complexity and with similar statistical meaning.

In conclusion, methods for uncertainty management in sequential–decision problems have been extensively studied, and efficient reformulations and solutions algorithms are available for its implementation. However, design parameters that control the solution's robustness are usually determined in the pre–processing stage, ignoring the possible benefits of its dynamical determination.

2.4.4 Illustrative example of optimization under uncertainty applied to power systems operation

A simple and practical example is analyzed to appreciate how the above-described methods to model the uncertainty into optimization problems work. To this, consider the problem of deciding the optimal dispatch of a conventional thermal generator in a single bus system, depicted in Fig. 2.10. A variable load and a wind generator are also connected to the bus.

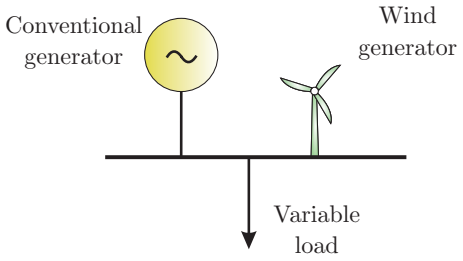


Figure 2.10: Illustrative single bus system, composed by a conventional generator, a wind generator and a variable load.

The problem here is that the decision has to be made without knowing the exact values of demanded power and the renewable generation that will take place when the dispatch be applied. Consider that only there is information of the historical values both of demanded power d_i^p and renewable wind generation r_i^w, with $i \in \mathcal{S}$ the set of observed scenarios. To simplify the analysis, consider the equivalent demanded power $d_i = d_i^p - r_i^w$ that has to be supplied entirely by the conventional generator. The variable cost of the conventional generator is c_p, and the cost associated to non–served demand and power spillage are c_+ and c_-, respectively. With this, the cost function is given by:

$$c(p, d) = c_p p + c_+[d - p]_+ + c_-[p - d]_+$$

where $[z]_+ = \max(0, z)$. Then, given the set of observed data $\mathcal{S} = \{d_i\}_{i=1}^N$, and the maximum dispatch level of the conventional generator p_{max}, a two–stage problem modeling the optimal dispatch is:

$$\min_{0 \le p \le p_{max}} \; c_p p + \rho_{\mathcal{S}}\big(Q(p, d)\big) \tag{2.46}$$

with $Q(p, d)$ the second–stage problem:

$$\min_{s^+, s^- \ge 0} \; c_+ s^+ + c_- s^- \tag{2.47}$$

$$\text{s.t.} \quad p + s^+ + s^- = d \tag{2.48}$$

The problem above seeks for the dispatch p that minimizes the dispatch costs, plus some measure $\rho_{\mathcal{S}}(\cdot)$ of the imbalances costs, measured over the set of observed equivalent demand $\mathcal{S}$. To

observe how the reviewed paradigms of optimization under uncertainty work and the data-driven paradigm, the problem (2.46)-(2.48) will be addressed using each one.

Firstly, consider the SP approach. To model the probability distribution of d, the sample set $\mathcal{S}$ will be considered as its approximation, i.e., all samples are modeled to be equiprobable. Then, the same as (2.28)-(2.31), the resulting formulation is the following:

$$\min \quad c_p p + \frac{1}{N}\sum_{i=0}^{N}(c_+ s_i^+ + c_- s_i^-) \tag{2.49}$$

$$\text{s.t.} \quad p + s_i^+ + s_i^- = d_i \qquad i = 1...N \tag{2.50}$$

$$s_i^+, s_i^- \geq 0 \qquad i = 1...N \tag{2.51}$$

$$0 \leq p \leq p_{max} \tag{2.52}$$

For the statement of an ChCP model, an additional constraint will be considered. In particular, it will be required that $\mathbb{P}_{\mathcal{S}}[s^+ + s^- \leq s^{max}] \geq 1-\varepsilon$, i.e., the sum of non–served demand and power spillage has to be less than s^{max}, with at least probability $1-\varepsilon$. Then, considering the empirical distribution as uniform distributed, the ChCO model can be stated as follows:

$$\min \quad c_p p + \frac{1}{N}\sum_{i=0}^{N}(c_+ s_i^+ + c_- s_i^-) \tag{2.53}$$

$$\text{s.t.} \quad p + s_i^+ + s_i^- = d_i \qquad i = 1...N \tag{2.54}$$

$$s_i^+ + s_i^- \leq s^{max} z_i + M(1 - z_i) \qquad i = 1...N \tag{2.55}$$

$$\sum_{i=1}^{N} \frac{z_i}{N} \geq 1 - \varepsilon \tag{2.56}$$

$$s_i^+, s_i^- \geq 0 \qquad i = 1...N \tag{2.57}$$

$$z_i \in \{0, 1\} \qquad i = 1...N \tag{2.58}$$

$$0 \leq p \leq p_{max} \tag{2.59}$$

where z_i is a binary variable representing the scenarios on which constraint $s_i^+ + s_i^- \leq s^{max}$ is meet, and M a constant with a high value, used for the description of a disjunctive inequality that model the chance constraint (2.55)-(2.56). Note that if $z_i = 1$, then (2.55) is equivalent to $s_i^+ + s_i^- \leq s^{max}$, whereas for $z_i = 0$, the right–hand side is M, which is equivalent to relax the condition. Concurrently, (2.56) accounts for the number of scenarios where the condition $s_i^+ + s_i^- \leq s^{max}$ is meet, and forces the empirical probability be at least $1-\varepsilon$.

On the other hand, concerning a RO model, the second stage cost will be equal to the worst–case scenario. In this particular case, the uncertainty set is considered exactly the same as the scenario set $\mathcal{S}$:

$$\min \quad c_p p + \eta \tag{2.60}$$

$$\text{s.t.} \quad \eta \geq c_+ s_i^+ + c_- s_i^- \qquad i = 1...N \tag{2.61}$$

$$p + s_i^+ + s_i^- = d_i \qquad i = 1...N \tag{2.62}$$

$$s_i^+, s_i^- \geq 0 \qquad i = 1...N \tag{2.63}$$

$$0 \leq p \leq p_{max} \tag{2.64}$$

where the variable η captures the worst case cost of $c^+ s_i^+ + c^- s_i^-$, imposed by the inequality (2.61). Finally, for the statement of the DRO model, is necessary describe the ambiguity set at first. It will consider every discrete distribution of N values that has a mean equals to $\bar{d}$ and variance σ^2, just as the same of the one–dimensional case of (2.38). This can be write explicitly imposing the conditions over the probability values of the distribution, by:

$$\mathcal{P} := \left\{ \{k_i\}_{i=1}^N \;\middle|\; \sum_{i=1}^{N} k_i d_i = \bar{d};\ \sum_{i=1}^{N} k_i (d_i - \bar{d})^2 = \sigma^2;\ \sum_{i=1}^{N} k_i = 1;\ k_i \in [0,1], \forall i \right\} \tag{2.65}$$

Thus, the DRO model can be explicitly formulated as:

$$\min \quad c_p p + \max_{k_i} \left\{ \sum_{i=0}^{N} k_i (c_+ s_i^+ + c_- s_i^-) \right\} \tag{2.66}$$

$$\text{s.t.} \quad p + s_i^+ + s_i^- = d_i \qquad i = 1...N \tag{2.67}$$

$$(\alpha) \qquad \sum_{i=1}^{N} k_i = 1 \tag{2.68}$$

$$(\beta) \qquad \sum_{i=1}^{N} k_i d_i = \bar{d} \tag{2.69}$$

$$(\gamma) \qquad \sum_{i=1}^{N} k_i (d_i - \bar{d})^2 = \sigma^2 \tag{2.70}$$

$$(\delta_i) \qquad 0 \leq k_i \leq 1 \qquad i = 1...N \tag{2.71}$$

$$s_i^+, s_i^- \geq 0 \qquad i = 1...N \tag{2.72}$$

$$0 \leq p \leq p_{max} \tag{2.73}$$

In this formulation, the second stage problem seeks maximizing the expected value of the failure cost, by founding the worst–case values of k_i, with $i \in \{1, 2, ..., N\}$, where k_i are the probabilities associated of the sample i. In this sense, optimizing over the weights k_i is equivalent to optimize over probability distributions. Besides, constraints (2.68)-(2.72) define explicitly the ambiguity set (2.65). Then, to recast the model (2.66)-(2.73) to a standard optimization problem, is useful obtaining the dual of the second stage problem, and write the reformulation on p, s_i^+, s_i^-, and dual variables α, β, γ and δ_i, for $i = 1...N$. The equivalent DRO model results:

$$\min \quad c_p p + \alpha + \beta \bar{d} + \gamma \sigma^2 + \sum_{i=1}^{N} \delta_i \tag{2.74}$$

$$\text{s.t.} \quad p + s_i^+ + s_i^- = d_i \qquad i = 1...N \tag{2.75}$$

$$\alpha + \beta d_i + \gamma (d_i - \bar{d})^2 + \delta_i \geq c_+ s_i^+ + c_- s_i^- \qquad i = 1...N \tag{2.76}$$

$$s_i^+, s_i^-, \delta_i \geq 0 \qquad i = 1...N \tag{2.77}$$

$$0 \leq p \leq p_{max} \tag{2.78}$$

$$\alpha, \beta, \gamma \in \mathbb{R} \tag{2.79}$$

which is now a linear problem. Even looking for the same objective, these four models employ different policies to deal with the uncertainty present in d. Moreover, using the same set of scenarios $\mathcal{S}$, the way of how measure the importance of the future operation conditions, i.e., the values of non–served demand and power spillage, and the criteria for the consideration of the uncertainty vary on each one. In particular, typical arguments used to choose one approach or another are computational tractability, statistical meaning, and out–of–sample performance [22]. However, there is no single correct answer.

To empirically tests the models, a set of observed samples S is defined:

i	1	2	3	4	5	6	7	8	9	10
d_i	11	12	9	10	13	8	7	13	7	10

with defined values of d_i, now is possible to solve each model, however, is still necessary define the values of some parameters in ChCO and DRO models. In the ChCO model, s^{max} and ε will be defined as 3 and 0.1, respectively, imposing the chance constraint that the sum of non–served demand s_i^+ and power spillage s_i^- should be less than 3 in the 90% of the cases. On the other hand, the values $\bar{d}$ and σ^2 of the DRO problem will be fixed as 10 and 5.11, respectively, being exactly the sample mean and variance of $\mathcal{S}$. Finally, problem parameters were fixed as $p_{max} = 20$, $c_p = 1$, $c_+ = 1.5$, $c_- = 1$. Then, solving each model, optimal results of p, the second–stage cost, and the total cost are presented below:

Model	p	$\rho(Q(p,d))$	**Obj. Cost**
SO	8	3.5	11.5
ChCO	10	2.25	12.25
RO	10.6	3.6	14.2
DRO	7	4.5	11.5

Table 2.3: In–sample results using the set $\mathcal{S}$ under different policies of uncertainty management, with problem parameters $p_{max} = 20$, $c_p = 1$, $c_+ = 1.5$, $c_- = 1$.

From the previous analysis, it is clear that the ChCO model would give a higher optimal cost than SO since it is more constrained. It can be seen that the RO model produces both the higher optimal value p and the higher optimal cost, which can be associated with a higher risk–averse level.

It is important to highlight that the order relationship between the different objectives costs could be somehow deduced before solving the models since each is a deterministic function of p_{max}, c_p, c^+, c^-, $\mathcal{S}$, and specific parameters. However, the results shown in table 2.3 have no practical relevance since they were obtained using past observations. An effective evaluation of the methods has to be made via out-of-sample experiments, i.e., evaluating the operation cost on a set of instances of demand not contained in the original scenario set. Therefore, consider three test sets, namely $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$, showed below:

i	1	2	3	4	5	6	7	8	9	10	Mean	Variance
$\mathcal{S}_1$	11	13	11.5	9	9.5	11.5	13	7.8	7.6	12	10.59	3.99
$\mathcal{S}_2$	9	12	13	11.5	9.4	11.2	8.7	14.5	12.6	8.3	11.02	4.36
$\mathcal{S}_3$	12	8	12.3	9.4	8.1	9.6	10	11.6	7.6	12	10.06	3.30

Applying the solutions of table 2.3 on the three evaluation sets, the obtained out–of–sample values of $C_k = \frac{1}{N}\sum_{i \in S_k}(c^+ s_i^+ + c^- s_i^-)$ are showed below:

Model	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$
SO	12.04	12.53	11.19
ChCO	12.41	12.68	11.92
RO	12.71	12.98	12.52
DRO	12.39	13.03	11.59

Table 2.4: Out–of–sample average costs of applying solutions of table 2.3 under realization of sets $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$.

Table 2.4 show the average out–of–sample costs of applying each p value to each test set $\mathcal{S}_i$. As can be checked, SO formulation achieves the lowest values of C_k for the three tests sets. In contrast, the RO solution results being the most expensive. However, this evidence is not enough to make any conclusion about what method is the best. In this regard, it is interesting to make some sensitivity analyses on some of the parameters. In particular, keeping the in-sample set S and the values of every parameter, but increasing the value of c_+ from 1.5 to 3, the new results are:

Model	p	$\rho(Q(p,d))$	**Obj. Cost**
SO	10	3.6	13.6
ChCO	10	3.6	13.6
RO	11.5	4.5	16
DRO	8	8	16

Table 2.5: Results using the set $\mathcal{S}$ under different policies of uncertainty management, with problem parameters $p_{max} = 20$, $c_p = 1$, $c_+ = 3$, $c_- = 1$.

With this modification, SP and ChCP methods give the same solutions and constitute the cheapest ones for the in–sample set $\mathcal{S}$. On the other hand, RO and DRO approaches result in the same value on the objective function, but with different values of the dispatch solution p, with the

RO case being the most conservative. Hence, applying these new solutions on the test sets, the new obtained out–of–sample average costs are:

Model	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$
SO	14.21	14.9	13.1
ChCO	14.21	14.9	13.1
RO	13.81	14.42	13.7
DRO	16.01	17.06	14.34

Table 2.6: Out–of–sample average costs of applying solutions of table 2.5 under realization of sets $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$.

With these new configuration, the RO approach results in the best out–of–sample performance on sets $\mathcal{S}_1$ and $\mathcal{S}_2$, whereas DRO achieves the overall worst. These examples demonstrate three relevant aspects that have to be considered when treating optimization problems subject to uncertainty.

1. It is not correct to account for the performance of a method through in–sample results. In this sense, out–of–sample evaluation is mandatory when dealing with problems subject to uncertainty.

2. The performance of each method is strongly dependent on the problem parameters and evaluation sets.

3. Using policies of this kind in real-time operation could result in optimal policies dynamically changing.

To appreciate the meaning of the third point, consider that sets $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ constitute sequential realizations of real values. Suppose the operator fixes the operation under the RO model's solutions and keeps the in-sample set s. In that case, it fails to operate with the lower cost policy when the realization of $\mathcal{S}_3$ would occur. In this sense, the concept of dynamic adaptation of uncertainty management policies could be profitable.

On the other hand, a feature that is not appreciated in the above examples is the computational complexity of each method, since the simplicity of the tested instance. By contrast, computational times could be higher enough to become a method inapplicable in big instances, making it necessary to either develop dedicated solution algorithms or change the method.

As is reviewed in this section, optimization under uncertainty is a wide branch of mathematics that is still in continuous development. The construction of unified frameworks for studying data-driven optimization problems has been explored [67, 19, 103], but is far to be a well–established theory. The present section has given an overall overview of one approach for modeling systems subject to uncertainty, and the approaches described are usual in the context of power systems operation and planning.

An interesting fact evidenced in the developed example is the possible benefits of changing the uncertainty management policy dynamically, as new data and real operational results are measured. Then, formulations able to capture the dynamic evolution of the systems could be suitable to adapt the uncertainty management policies. The dynamic programming approach is reviewed in the next section, which constitutes another strategy to model systems and deals with uncertainty. In conjunction with the above-presented theory, the special features of dynamic programming will be used to construct the central methodology of the present thesis.

2.5 Dynamic programming and reinforcement learning

2.5.1 Basic concepts

Dynamic programming (DP) can be described as a mathematical framework that models sequential decision–making problems involving discrete–time dynamic systems. This structure generates a sequence of states on which its values are influenced by control. At every stage, the decision-maker takes a control decision, balancing the immediate and future expected costs. In the below presentation, the stochastic DP approach is addressed, where a random process is present in the

system dynamic, and the horizon of examination is considered infinite. In this way, the system evolution is described through the transition function:

$$\boldsymbol{s}_{t+1} = \boldsymbol{f}_t(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{w}_t), \quad t = 0, 1, 2, ... \tag{2.80}$$

where t is the time index, $\boldsymbol{s}_t \in \mathcal{S}$ is the state of the system, $\boldsymbol{u}_t \in \mathcal{U}(\boldsymbol{s}_t)$ is the control vector to be selected at time t, $\boldsymbol{w}_t$ is a random process, and $\boldsymbol{f}_t$ is a function that describes the evolution of the system from time t to $t+1$. As is depicted in Fig. 2.11, equation (2.80) defines the resultant state that the system reaches $\boldsymbol{s}_{t+1}$ if a control action $\boldsymbol{u}_t$ is taken at the state $\boldsymbol{s}_t$, and the random variable takes the value $\boldsymbol{w}_t$.

Control actions $\boldsymbol{u}_t$ are assumed to belong to some set $\mathcal{U}(\boldsymbol{s}_t)$, also called action space, and are obtained from a policy function of the current state $\boldsymbol{u}_t = \boldsymbol{\mu}_t(\boldsymbol{s}_t)$, widely known as control or action policy. This function represents some knowledge about the system, which indicates the optimal control action to be taken if the state $\boldsymbol{s}_t$ is reached. The sense of optimality is originated from the definition of transition costs $\boldsymbol{g}_t(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{w}_t)$ associated with costs incurred due to the system's evolution from $\boldsymbol{s}_t$ to $\boldsymbol{s}_{t+1}$.

Then, suppose the problem has a time horizon associated. In that case, the total cost incurred is defined as the weighted sum of the costs of the individual transition. Therefore, considering a pre–defined set of policies $\pi = \{\boldsymbol{\mu}_t\}_{t \in \mathcal{T}}$ that dictate how to act when certain states are reached, the expected total cost of the process starting at the initial state $\boldsymbol{s}_0$ is given by

$$J_\pi(\boldsymbol{s}_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t g_t(\boldsymbol{s}_t, \boldsymbol{\mu}_t(\boldsymbol{s}_t), \boldsymbol{w}_t)\right] \tag{2.81}$$

In equation (2.81), the parameter $\gamma^t \in [0, 1]$ is the so–called discount factor, which accounts for the importance of the future costs. With $\gamma^t = 0$, a greedy policy is selected, i.e., the decision–maker is focused on minimizing the immediate transition cost, without caring about the future implication of the current decision. The quantity $J_\pi(\boldsymbol{s})$ is commonly called as the cost–to–go function, which is an estimation of the future expected total cost of starting from the state $\boldsymbol{s}$, and acting according to the policies set π.

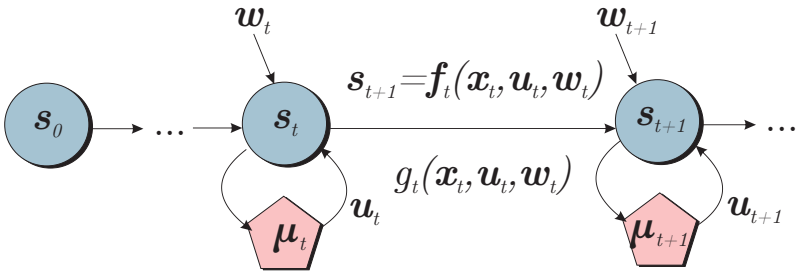


Figure 2.11: System dynamics from the DP approach.

Thus, the objective of the DP approach is finding the optimal set of policies that minimizes the total expected cost, i.e.

$$J_{\pi^*}(\boldsymbol{s}_0) = \min_\pi J_\pi(\boldsymbol{s}_0) \tag{2.82}$$

One of the key aspects of this formulation is the capacity to obtain decompositions of problem (2.82), based on the principle of optimality of the tail subproblems. The principle states that, given a starting state $\boldsymbol{s}_0$, and full knowledge of the series of the uncertainty realization $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3, ...\}$, an optimal control sequence $\{\boldsymbol{u}_0^*, \boldsymbol{u}_1^*, \boldsymbol{u}_2^*, ...\}$ can be deduced, which determines the future state sequence $\{\boldsymbol{s}_1^*, \boldsymbol{s}_2^*, \boldsymbol{s}_3^*, ...\}$ via equation (2.80). Then, consider the subproblem of minimizing the expected future cost (2.81) from the state $\boldsymbol{s}_k^*$, over the set of policies $\{\boldsymbol{u}_k, \boldsymbol{u}_{k+1}, \boldsymbol{u}_{k+2}, ...\}$. The truncated optimal control sequence $\{\boldsymbol{u}_k^*, \boldsymbol{u}_{k+1}^*, \boldsymbol{u}_{k+2}^*, ...\}$ is optimal for this subproblem, i.e., we can keep with only the final portion of the optimal sequence. Briefly, the principle says that the tail of an optimal sequence is optimal for the tail subproblem [104].

This result gives rise to the Bellman equation, where its derivation can be found in [104]. Then, considering the sequence of optimal total expected costs $J^* = (J^*(\boldsymbol{s}_0), J^*(\boldsymbol{s}_1), ..., J^*(\boldsymbol{s}_t), ...)$, the following recursion is achieved

$$J^*(\boldsymbol{s}_t) = \min_{\boldsymbol{u}_t \in \mathcal{U}(\boldsymbol{s}_t)} \mathbb{E}\left[g_t(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{w}_t) + \gamma J^*(\boldsymbol{s}_{t+1})\right] \tag{2.83}$$

The above result states that the optimal control $\boldsymbol{u}_t^*$ of the equation (2.83) belongs to the optimal sequence of the tail subproblem, allowing the individual calculation of it rather than the calculation of the whole sequence.

Alternative formulations of the Bellman equation are also stated in terms of the Q–function, which definition is given by:

$$Q(\boldsymbol{s}_t, \boldsymbol{u}_t) = \mathbb{E}\left[g_t(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{w}_t) + \gamma J^*(\boldsymbol{s}_{t+1})\right] \tag{2.84}$$

where $Q(\boldsymbol{s}_t, \boldsymbol{u}_t)$ can be interpreted as the cost of starting at the state $\boldsymbol{s}_t$, applying the control $\boldsymbol{u}_t$, and then acting optimally. Therefore, the same as equation (2.83), the above definition permits isolating the calculation of single control action instead of calculating the whole sequence. In this way, the optimal control action at the state $\boldsymbol{s}_t$ is given can be obtained from:

$$\mu^*(\boldsymbol{s}_t) \in \underset{\boldsymbol{u}_t \in \mathcal{U}(\boldsymbol{s}_t)}{\arg\min}\, Q(\boldsymbol{s}_t, \boldsymbol{u}_t) \tag{2.85}$$

And then, the recursion of the Bellman equation can be recovered by stating

$$Q^*(\boldsymbol{s}_t, \boldsymbol{u}_t) = \mathbb{E}\left[g_t(\boldsymbol{s}_t, \boldsymbol{u}_t, \boldsymbol{w}_t) + \gamma \min_{\boldsymbol{v} \in \mathcal{U}(\boldsymbol{s}_t)} Q^*(\boldsymbol{s}_{t+1}, \boldsymbol{v})\right] \tag{2.86}$$

According to the information available of the system under study, these equations constitute the basics of DP formulations, where several solution algorithms and approximations have been proposed to solve them. It is well known that solving DP equations to optimality could be very difficult for many systems, as some instances of the shortest path problem (SPP) with resource constraints [105].

Consequently, many approximation techniques and algorithms have been developed to approximately solve the problem of optimal control stated by DP (reference). In particular, reinforcement learning (RL) algorithms have reached significant popularity given the constant growth of the computation power, and the performance showed in some systems, as the AlphaGo Zero agent [106]. In the next section, the basics of RL and its principal algorithms are described.

2.5.2 Reinforcement learning basics and Q–learning algorithms

RL is a machine learning (ML) method which, according to [107], consists of giving an agent the capacity to learn what to do, how to map states to actions, to maximize a numerical reward signal. Every RL framework has two principal elements: *(i)* the *agent* and *(ii)* the *environment*, corresponding to the object the agent interacts with [107]. The dynamic between them can be summarized as follows: given a particular state of the environment, the agent takes actions and the environment responds by presenting a new state and giving rise to rewards, which are numerical values that the agent aims to maximize over time through its choice of actions [107]. In the optimal control context, the agent is the controller, and the environment is given by the system under control. The controller is given by a parametric set of policies $\boldsymbol{\mu}_t(\boldsymbol{s}_t, \boldsymbol{r}_t)$, on which $\boldsymbol{r}_t$ is a vector parameter that is somehow updated (depending on the specific RL method) on each stage, accordingly with some principle of optimality. This is illustrated in Fig. (2.12).

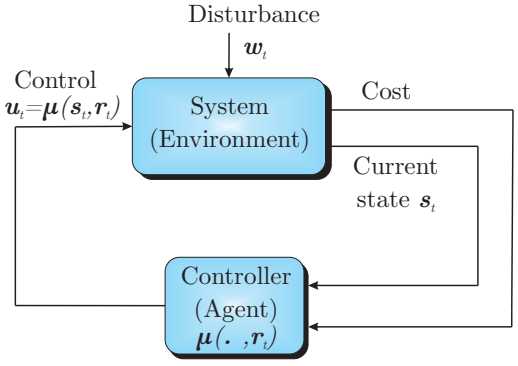


Figure 2.12: Interaction scheme between the agent and the environment

Accordingly, RL methods seek an approximation of the optimal control sequence of problem (2.83). Before introducing specific RL methods, it is needed to say that most of the algorithms rely on the accomplishment of the Markov property on the system definition. This property states that all the information of past control actions and past states is contained in the current state $\boldsymbol{s}_t$, which means that past information is irrelevant to the choices of future controls as long we know $\boldsymbol{s}_t$. The name of Markov property is related to the similarity with the definition of states of Markov chains, where by definition, the conditional probability distribution of the future states depends on the history of the chain only through the present state, i.e. :

$$P(\boldsymbol{s}_{t+1}|\boldsymbol{s}_0, \boldsymbol{u}_0, .., \boldsymbol{s}_t, \boldsymbol{u}_t) = P(\boldsymbol{s}_{t+1}|\boldsymbol{s}_t, \boldsymbol{u}_t), \quad \forall t = 1, 2, ... \tag{2.87}$$

This property is not strictly applied in RL algorithms, but the concept is always desired in the states' definitions. On the other hand, the state domain S and the control actions domain $\mathcal{U}(\boldsymbol{s})$ are considered to be discrete in the RL context. In that sense, both sets can be defined as $\mathcal{S} = \{\boldsymbol{s}^i\}_{i\in\mathcal{I}}$, and $\mathcal{U}(\boldsymbol{s}) = \{\boldsymbol{u}^j\}_{j\in\mathcal{J}}$, $\forall \boldsymbol{s} \in \mathcal{S}$, with $\mathcal{S} = \{1, 2, ..., N\}$, and $\mathcal{U}(\boldsymbol{s}) = \{1, 2, ..., N\}$. Then, without loss of generally, DP equations can be expressed in terms of the states and control actions indices, i.e., $\boldsymbol{s}_t^i \to i_t$ and $\boldsymbol{u}_t^j \to j_t$, giving place to cost–to–go functions and Q–values definitions $J(\boldsymbol{s}_t^i) \to J(i_t)$, and $Q(\boldsymbol{s}_t^i, \boldsymbol{u}_t^j) \to Q(i_t, j_t)$, respectively.

Roughly speaking, RL methods are divided accordingly to the knowledge available of the system. In this way, methods are classified as model–based or model–free methods. The last ones do not need an explicit description of the system dynamics and are often used when dealing with complex dynamic equations.

Besides, the approximate resolution of the Bellman equation can also be classified into two main types for both model–based and model–free methods. In the first place, policy iteration algorithms are designed to construct a sequence of policies, each one better than its predecessor. The algorithm starts with an initial policy $\boldsymbol{\mu}_0$, and in the first stage iteration, the cost–to–go function acting under that policy is evaluated. This process is also called policy evaluation, and consist of the solution of the following system of Bellman equations:

$$J_{\boldsymbol{\mu}_k}(i_t) = \mathbb{E}\left[g_t(i_t, \boldsymbol{\mu}_k(i_t), \boldsymbol{w}_t) + \gamma J_{\boldsymbol{\mu}_k}(i_{t+1})\right], \quad \forall i \in \mathcal{I} \tag{2.88}$$

Lately, the new policy $\boldsymbol{\mu}_{k+1}$ is obtained as

$$\boldsymbol{\mu}_{k+1}(i_t) \in \underset{j_t\in\mathcal{U}(i_t)}{\arg\min}\, \mathbb{E}\left[g_t(i_t, j_t, \boldsymbol{w}_t) + \gamma J_{\boldsymbol{\mu}_k}(i_{t+1})\right], \quad \forall i \in \mathcal{I} \tag{2.89}$$

and the process is repeated until $J_{\boldsymbol{\mu}_{k+1}(i)} - J_{\boldsymbol{\mu}_k(i)} \leq \epsilon$, $\forall i \in \mathcal{I}$, with ϵ a fixed small tolerance. Additionally, under some conditions convergence is proved [104].

Secondly, value iteration algorithms iterate, getting better approximations of cost–to–go functions, based on the Bellman equation formulated as a recursive definition:

$$J_{k+1}(i_t) = \min_{j_t\in\mathcal{U}(i_t)} \mathbb{E}\left[g_t(i_t, j_t, \boldsymbol{w}_t) + \gamma J_k(i_{t+1})\right] \tag{2.90}$$

where $J_k(i_t)$ is the approximation of the optimal cost–to–go function of the state i_t, at the iteration k. In this way, given any initial conditions $\{J_0(i)\}_{i\in\mathcal{I}}$, the convergence of algorithm (2.90) to the optimal cost–to–go function is guaranteed [104].

As a type of value iteration algorithm, Q–learning methods directly approximate the Q–values of the optimal policy and then avoid the multiple policy evaluation steps of the policy iteration methods. One great feature of them is that can be implemented in model–free fashion and its implementation is quite straightforward. The original Q–learning algorithm was proposed by Watkins [108], and is based on equation (2.86). Accordingly, a sequence of state–control pairs (i, j) are generated, and using a step–size $\alpha_t \in (0, 1]$, the Q–values are updated according to:

$$\tilde{Q}_{t+1}(i, j) = (1 - \alpha_t)\tilde{Q}_t(i, j) + \alpha_t(F_t(\tilde{Q}_t, v))(i, j) \tag{2.91}$$

where $\tilde{Q}_{t+1}(i, j)$ is the optimal Q–value approximation at iteration t, and $F_t(\tilde{Q}_t, v)$ is the updating operator, defined as:

$$(F_t(\tilde{Q}_t, v))(i, j) = \begin{cases} g_t(i_t, j_t, \boldsymbol{\xi}_t) + \gamma\tilde{Q}_t(i_{t+1}, v) & (i, j) = (i_t, j_t) \\ \tilde{Q}_t(i, j) & (i, j) \neq (i_t, j_t) \end{cases} \tag{2.92}$$

where v is obtained from $v \in \arg\min_{j \in \mathcal{U}(i_{t+1})} \tilde{Q}_t(i_{t+1}, j)$. Concerning to the convergence of the Q–learning algorithm, it has been proved that exist necessary conditions on the stepsize [109]. Specifically, the sequence α_t should accomplish $\sum_{t=0}^{\infty} \alpha_t = \infty$, and $\sum_{t=0}^{\infty} (\alpha_t)^2 \leq \infty$. For example, a stepsize of the form $\alpha_t = \frac{c_1}{t+c_2}$ addresses both conditions, for c_1 and c_2 some positive constants.

One drawback of the above–mentioned Q–learning algorithm is that the approximations are made over biased estimators of the cost-to-go function, leading to slow convergence to the optimal policy and poor performance in some stochastic systems. Van Hasselt [110] solved this by using two different estimators for the Q–values, namely $\tilde{Q}_t^1$ and $\tilde{Q}_t^2$, constructed from different sample sets. Then, to update the value of $\tilde{Q}_t^1$ in the equation (2.91), the value of v used in the updating operator $F_t(\tilde{Q}_t^1, v)$ is obtained from the second estimator as $v \in \arg\min_{j \in \mathcal{U}(i_{t+1})} \tilde{Q}_t^2(i_{t+1}, j)$. In that work, it is proved that this method generates unbiased estimators of the cost–to–go functions. Besides, as the samples on each estimator have to be independent, only one estimator can be updated at each iteration of the algorithm, which can be done by using some random policy selection. However, both estimators can be used to determine the optimal control value that will be applied in the system. These modifications give place to the Double Q–learning algorithm, which can be roughly described in the pseudo–code below:

Algorithm 1: Double Q–Learning

Initialization: $\tilde{Q}_0^1$, $\tilde{Q}_0^2$, $\boldsymbol{s}_0$;
while $t \leq |\mathcal{T}|$ **do**
  Obtain j_t based on $\tilde{Q}_t^1$ and $\tilde{Q}_t^2$;
  Observe $g_t(i_t, j_t, \boldsymbol{\xi}_t)$ and i_{t+1} ;
  Select either update $\tilde{Q}_t^1$ or $\tilde{Q}_t^2$;
  if *update* $\tilde{Q}_t^1$ **then**
    Select $v \in \arg\min_{j \in \mathcal{J}} \tilde{Q}_t^1(i_t, j)$;
    Update $\tilde{Q}_{t+1}^1(i,j) = (1-\alpha_t)\tilde{Q}_t^1(i,j) + \alpha_t(F_t(\tilde{Q}_t^2, v))(i,j)$;
  else
    Select $v \in \arg\min_{j \in \mathcal{J}} \tilde{Q}_t^2(i_t, j)$;
    Update $\tilde{Q}_{t+1}^2(i,j) = (1-\alpha_t)\tilde{Q}_t^2(i,j) + \alpha_t(F_t(\tilde{Q}_t^1, v))(i,j)$;
  end
  $t \leftarrow t+1$;
end

Despite the effectiveness of the Q–learning and the double Q–learning algorithms in some contexts, their applicability is limited to domains in which useful features can be extracted or with too low dimensional state spaces. In 2015, the recent breakthroughs on the development of deep neural networks were combined with reinforcement learning to overcome these limitations. This conjunction originated agents capable of capturing features from high dimensional spaces and mapping them to concrete actions [111]. The resultant framework was called Deep Q–Network (DQN), which is illustrated in Fig. (2.13). Results showed that the agent reached professional human performance across 49 games, using the same algorithm, network architecture, and hyperparameters.

The key idea of DQN algorithms is to approximate the Q–function (2.86) using a deep convolutional neural network, namely, $Q(\boldsymbol{s}_t, \boldsymbol{u}_t; \boldsymbol{\theta}_i)$, in which $\boldsymbol{\theta}_i$ are the parameters or weights of the Q–network at the iteration i. As was reported in [112], Q–learning algorithms suffer from instability or even diverge when non–linear functions are used to approximate the Q–function, principally due to several classes of correlations between observations and estimators. To this end, the DQN algorithm relies on two protocols, named experience replay and periodic update. Experience replay is based on storing agent's experiences $\boldsymbol{e}_t = (\boldsymbol{s}_t, \boldsymbol{u}_t, g_t, \boldsymbol{s}_{t+1})$ at each time–step, constructing a set of experiences $\mathcal{D}_t = \{\boldsymbol{e}_k\}_{k=1}^t$. Then, the Q–table update is made over samples of experiences, obtained from a uniform distribution over the set of stored transitions $(\boldsymbol{s}_k, \boldsymbol{u}_k, g_k, \boldsymbol{s}_{k+1}) \sim U(\mathcal{D}_t)$, eliminating the correlations between consecutive observations. This results in an update iteration that minimize the following loss function:

$$L_i(\boldsymbol{\theta}_i) = \mathbb{E}_{(\boldsymbol{s},\boldsymbol{u},g,\boldsymbol{s}') \sim U(\mathcal{D}_t)} \left[\left(g + \gamma \min_{\boldsymbol{v} \in \mathcal{U}(\boldsymbol{s}')} Q(\boldsymbol{s}', \boldsymbol{v}; \boldsymbol{\theta}_i^-) - Q(\boldsymbol{s}, \boldsymbol{u}; \boldsymbol{\theta}_i) \right)^2 \right] \tag{2.93}$$

where $\boldsymbol{\theta}_i^-$ is are the network weights used to compute the target. Finally, the periodic update

ensures that the update of the parameters from $\boldsymbol{\theta}_i^-$ to $\boldsymbol{\theta}_i$ is made only every a specific amount of steps, and kept fixed between them. Remarkably, the design of deep network architecture could be adapted to address different tasks in different contexts. However, finding an adequate structure is often a complex procedure.

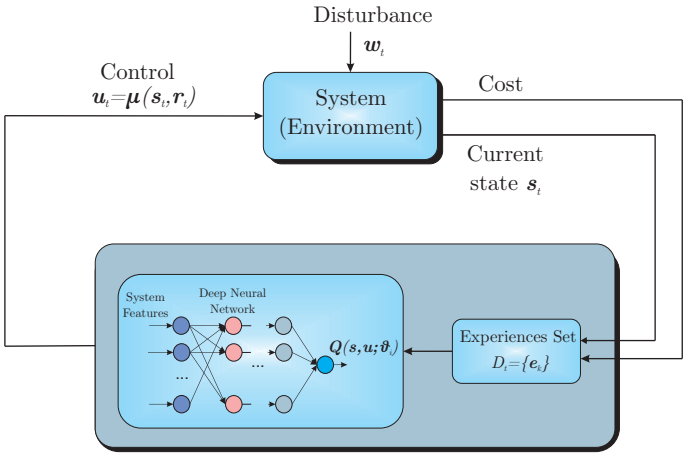


Figure 2.13: Deep Q–Network scheme.

It is important to highlight that Q–learning algorithms represent only a single branch of RL methods. The published applications of RL on power systems planning, operation and control are numerous, using different algorithms, and solution schemes. In the following subsection, a brief review of RL applications on power systems is made.

2.5.3 Reinforcement learning on energy management applications

RL applications in power systems cover a wide range of decision, control, and optimization problems. Besides, the number of RL methods applied to solve these problems is also huge. The diagram of Fig. 2.14, extracted in its entirety from [1] shows a classification of the algorithms usually used in power systems problems. In the following lines, some applications on energy management, demand response, and operational control are presented, to establish the state of the art of the inclusion of RL in power systems operation and control, based on a recent review [113]. Finally, proposals pointing in the same direction of the present thesis are listed, fulfilling the base on power management models involving closed–loop frameworks.

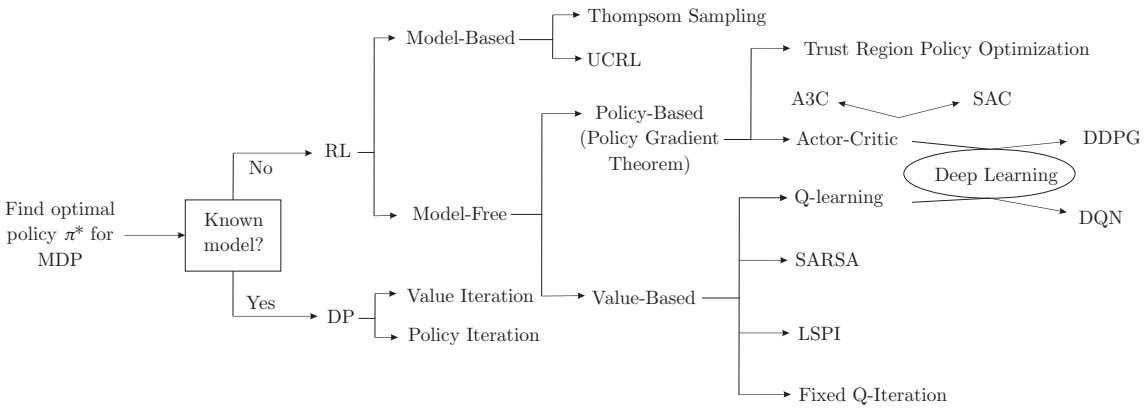


Figure 2.14: Diagram of most known RL methods [1]

Energy Management

The growing inclusion of distributed energy resources (DERs) and storage devices across microgrids has settled the problem of the optimal exploitation of renewable resources. In this sense, the

generation levels of DERs, battery storage levels, and demand consumption are now degrees of freedom on the system, which can be determined following some sense of optimality. Accordingly, these energy management problems can be formulated as sequential decision-making problems and, therefore, can be solved using RL methods. In the literature can be founded applications on residential appliances [114, 115], optimal energy management of electric vehicles [116, 117], and optimal operation of microgrids [118, 119, 120].

Demand Response

Demand response is a problem in smart grids, which keep the balance of the consumed power of customers and supply of utility companies, using the energy price as an incentive. In this sense, the consumption of energy is motivated by its price, being necessary to incorporate the consumers' behavior into the control loop to achieve grid stability. Here, deep RL techniques have been used to create data–driven models to solve these kinds of problems [121, 122].

Operational Control

Given the natural relationship between RL and optimal control, RL methods have been considered in different electric power system control–decision problems: Transient angle stability [123, 124], oscillatory angle instability [125, 126], voltage control [127], automatic generation control (AGC) [128, 129], economic dispatch [130, 131], wide-area control, households control [132, 133], and wind generation control [134, 135]. Concurrently, the relation between model predictive control (MPC) and RL has been investigated [136], and its relationships have been explored formally in [137] [124]. Some ideas that have combined the RL and MPC fields are: Addressing infinite-horizon optimization in MPC via Q-learning [138] and employment of the value function (stage cost) in economic MPC as a parametrization approximator in Q-learning [139].

Cloosed–Loop Frameworks on Power Management

The construction of closed-loop frameworks embedding mathematical programming models and machine learning is a novel approach, as has been described in [22], identifying potentially powerful tools. A framework of this kind has been implemented in the context of refinery procurement and production planning [140]. In the context of energy management, in [141], recurrent neural networks are used for the construction of uncertainty sets, in [142] a closed-loop scheme was proposed for the UC problem on which neural networks are used to approximate cost–to–go functions. Finally, establishing a closed-loop architecture, without using RL, in [143], the AC optimal power flow (OPF) problem is solved using a feedback scheme, keeping optimal operation into the power flow manifold. In the same direction, the works [144, 145] follow the construction of closed–loop controllers for the optimal dispatch problem.

Chapter 3

Mathematical formulations

The present chapter minutely describes the core of the thesis work. The developed framework is firstly presented, introducing its principal features and components, along with a motivational example that helps foresee the framework's potential benefits by analyzing a simple case. Then, the different components of the framework are introduced, stating the assumptions taken and deriving the corresponding mathematical expressions. Optimization models for the unit commitment subject to uncertainty are formulated in the first place. Later, the generation of scenarios for the probabilistic distributions approximations, besides the construction of uncertainty sets, are presented. Subsequently, the dynamic programming formulation that gives place to the adaptation of the risk–aversion level is stated, followed by the presentation of the framework used for the simulation of the real–time operation of the power system. Ultimately, the solution methodologies for the formulated problems and the corresponding solution algorithms are described.

3.1 Problem statement

The objective of the present proposal is to generate a solution framework for the UC problem subject to uncertainty, able to achieve better operational results than existing methods while keeping scalability to be applicable in big systems and generate practical, real–world solutions.

As reviewed in the previous chapter, the most recent approaches to managing the UC problem's uncertainty rely on classical optimization under uncertainty techniques, namely SP, ChCP, RO, or DRO approaches, and using a data-driven methodology [4, 20, 6, 12, 146, 14, 17, 98, 91].

The present proposal is constructed over this approaches, adding and adaptation layer based on the feedback of operational results. In [98], uncertainty management policies with different levels of risk-aversion were tested, identifying that different risk-averse levels reach minimum out-of-sample costs for each method and each test data set. This evidence suggests that operating in real-time with this kind of policy, its optimal risk–aversion level is dynamically changing. Concurrently, in [22], the idea of constructing closed–loop data–driven frameworks, machine learning algorithms, to feedback mathematical programming models is foreseen as a promising proposal.

In this line, the proposed adaptation layer is designed for the dynamical calculation of the optimal risk–averse level through an RL agent, which is fed with real–time operational results. To this end, a two–stage RUC model is used, where the uncertainty set is defined by the existing data and the calculated risk–averse level. Then, the operational decisions calculated are applied to the power system, giving place to new real–time results, defining a new risk-aversion level for the next period. The proposed adaptation schema is depicted in Fig. 3.1.

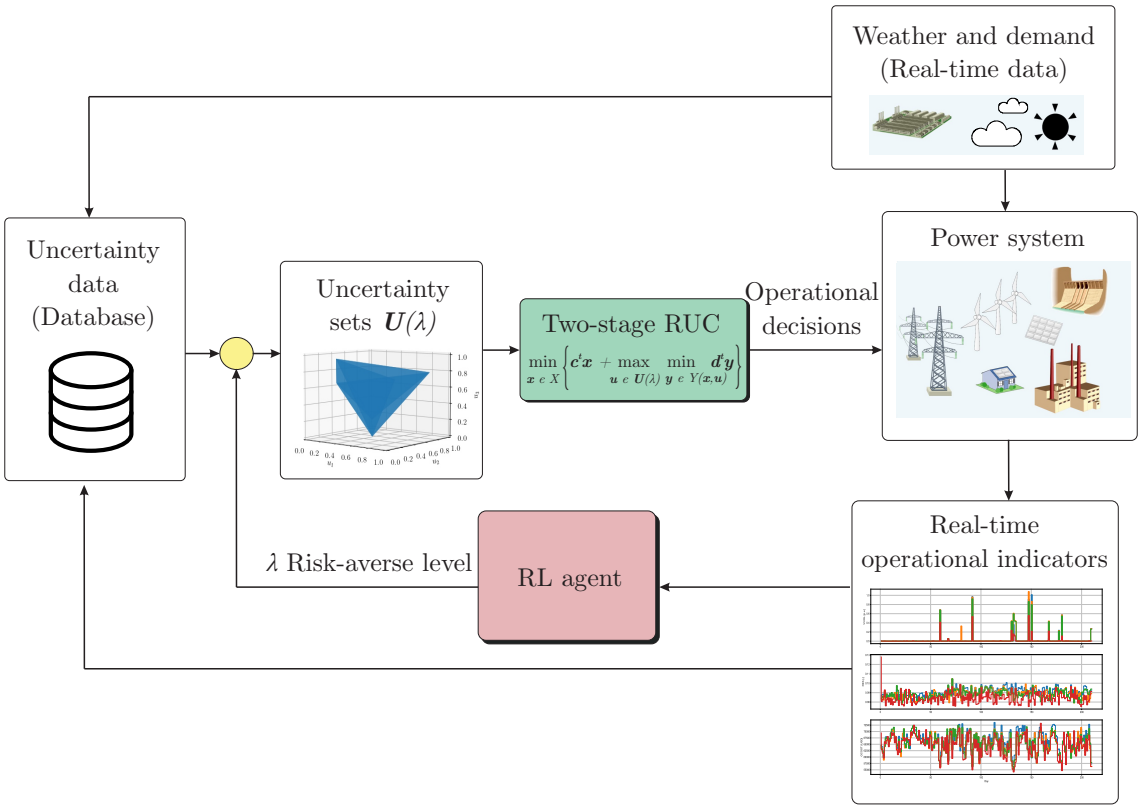


Figure 3.1: Schematic diagram of the proposed framework. Operational results feedback the construction of robust uncertainty sets through a RL agent, achieving a closed–loop data–driven framework.

3.2 Motivational example

The example of section 2.4.4 is analyzed using the proposed adaptive approach to illustrate the spirit of the present proposal.

Firstly, for the sake of comparison, let us review an analytical result. If d has a normal distribution $\mathcal{N}(\mu, \sigma)$, then the solution for the stochastic problem, i.e., minimizing over x the expected value $\mathbb{E}[c(x, d)]$, is equal to $x^* = F^{-1}(\frac{c_+ - c_x}{c_+ + c_-})$, with F the cumulative distribution function of the above–mentioned normal distribution.

Consider that the operation problem has to be solved sequentially, i.e., each day k it has to be decided the amount of power to be dispatched by the conventional generator, knowing the information of past days. In this way, the demand of the day k is represented now by a stochastic process d_t. Therefore, the daily cost now is indexed by k, $c_k(x_k, d_k) = c_x x_k + c_+[d_k - x_k]_+ + c_-[x_k - d_k]_+$, where x_k is the amount of power dispatched by the conventional generator on the day k. Then, for a horizon of $|\mathcal{K}|$ days, using the results of DP formulations (see section 2.5), the overall cost is given by $J^*(x_0) = \min_{x_t} \mathbb{E}[\sum_{k=k_0}^{|\mathcal{K}|} c_k(x_k, d_k)]$

As was seen, this problem accomplishes the tail–optimality principle, expressed by Bellman's equation (2.83). To simplify the analysis, consider that there are no dynamics involved, or equivalently, there are no constraints between x_t and x_{t+1}. Hence, the total solution can be obtained simply by solving the stage–wise problem, i.e.,

$$J^*(x_0) = \sum_{t=t_0}^{\mathcal{T}} \min_{x_t} \mathbb{E}\Big[c_t(x_t, d_t)\Big] \tag{3.1}$$

Then, two different methods for the resolution of the sequential optimal dispatch (3.1) will be compared, considering the data set of the historical observed demand at day k as $\mathcal{O}_k = \{d_k, ..., d_{k-1}\}$. The first method obtains the x_k solutions by solving a two–stage robust model,

which uses box uncertainty sets, centered in the observed mean $\bar{d}_k$, $\mathcal{U}_k = [\bar{d}_k - \hat{d}_k, \bar{d}_k + \hat{d}_k]$, where $\hat{d}_k$ is the maximum observed deviation, the same as in (2.36). The model is described below:

$$\min_{x_k} \; c_x x_k + \max_{d \in \mathcal{U}_k} \; \min_{s_d^+, s_d^+ \geq 0} \; c_+ s_d^+ + c_- s_d^- \tag{3.2}$$

$$\text{s.t. } x_k + s_d^+ + s_d^- = d \qquad \forall d \in \mathcal{U}_k \tag{3.3}$$

The second method uses the same model, but with a modified uncertainty set $\mathcal{U}_k(\lambda_k) = [\bar{d}_k - \lambda_k \hat{d}_k, \bar{d}_k + \lambda_k \hat{d}_k]$, where $\lambda_k \in [0, 1.5]$ is a parameter controlling the robustness of the set. The value of λ is determined at every day k as an output of a proportional–integral (PI) controller $\lambda_{k+1} = \lambda_k + \beta_p e_k + \beta_i \sum_{j=k_0}^{k} e_j$, with $e_k = x_k^* - d_k$ the difference between the dispatched and the demanded power at day k, and β_p, β_i, constants. This second method adds some flexibility in the definition of the uncertainty set, allowing more robust solutions than the obtained from the first method for $\lambda_k > 1$, and less robust for $\lambda_k < 1$.

To compare both methods, the dynamics of d_k will be described by an autoregressive–moving average process of order (p, q), ARMA(p, q) for short, defined as:

$$d_k = \alpha + \varepsilon_k + \sum_{i=1}^{p} \phi_i d_{k-i} - \sum_{i=1}^{q} \theta_i \varepsilon_{k-i} \tag{3.4}$$

where ε_k is the error at time k, which is assumed to be distributed as white noise with variance σ_ε^2. Coefficients ϕ_i and θ_i are the coefficients of the autoregressive and moving–average polynomials, respectively, and α is a constant.

ARMA models are one of the most simple stochastic linear processes, which combines the dependence on past observations with the influence of some perturbation, described by the white noise. The objective is to compare both above–presented methods, in the light of the performance of an agent with complete information of the process. As is developed in [63], for an ARMA process, the h–step–ahead forecast density of d_{k+h}, for $h = 1, 2, ...$, is given by $d_{k+h}|d_1, ..., d_k \sim \mathcal{N}(\hat{d}_{k+h|k}, \text{MSFE}(e_{k+h|T}))$, with $\hat{d}_{k+h}$ the minimum mean square error forecast of d_{k+h}, obtained recursively by:

$$\left(1 - \sum_{i=1}^{p} \phi_i L^i\right) d_{k+h|k} = \bar{d} + \left(1 - \sum_{i=1}^{q} \theta_i L^i\right) \varepsilon_{k+h|k} \tag{3.5}$$

where L^i is the lag operator of order i, $L^i d_k = d_{k-i}$, $\varepsilon_{k+j|k} = 0$ for $j > 0$, $\varepsilon_{k+j|k} = \varepsilon_{k+j}$, and $d_{k+j|k} = d_{k+j}$ for $j \leq 0$. In the other hand, MSFE$(e_{k+h|k})$ is the mean square forecast error for the h–step–ahead forecast error, $e_{k+h|k} = d_{k+h} - \hat{d}_{k+h|k}$, given by $\text{MSFE}(e_{k+h|k}) = \sigma_\varepsilon^2 \sum_{i=0}^{h-1} \psi_i^2$, where ψ_i are the coefficients of the polynomial $\Psi(L) = \frac{1 - \sum_{i=1}^{q} \phi_i L^i}{1 - \sum_{i=1}^{p} \theta_i L^i}$.

For the case of the optimal power dispatch, the forecast is needed only for 1 step ahead, resulting in a normal distribution with mean $\hat{d}_{k+1|k}$ and variance σ_ε^2 for the demand of the day $k + 1$. Hence, having complete information of the ARMA process, the optimal solution of the stage–wise stochastic problem $\min_{x_k} \mathbb{E}[c_t(x_k, d_k)]$ is $x_k^* = F_k^{-1}\left(\frac{c_+ - c_x}{c_+ + c_-}\right)$, with F_k the cumulative distribution function of the normal distribution $\mathcal{N}(\hat{d}_{k|k-1}, \sigma_\varepsilon)$.

Hereafter, the two robust methods, with limited information contained in $\mathcal{O}_k$, are compared in terms of a backtest examination of the daily costs. The method using the robust set $\mathcal{U}_k = [\bar{d}_k - \hat{d}_k, \bar{d}_k + \hat{d}_k]$ is hereinafter referred as *Fixed*, and the method considering the parameter λ_k will be referred as *Adaptive*. In Fig. 3.2, the costs of the two methods are shown for 300 days of backtest simulation. Besides, the solution obtained from the 1–step forecast distribution with full knowledge of the process is also depicted, which is referred as *Analytical*. In this example, an ARMA$(2, 2)$ process with mean equal to 10 was selected to describe d_k, the number of scenarios considered for the construction of both $\mathcal{U}_k$ and $\mathcal{U}_k(\lambda_k)$ was 5, and different initial conditions where tested for the second method, namely $\lambda_0 = 1.5$ and $\lambda_0 = 0$. Concurrently, the costs parameters involved are defined as $c_x = 1$, $c_+ = 1.5$ and $c_- = 1$.

The summary of the results is shown in table (3.1). As can be seen, the Adaptive method over–perform the Fixed method for both initial conditions, in average cost and variance. No method was used for the determination of the controller parameters, only by examination. The Adaptive method even reaches a lower variance than the Analytical method when using an initial condition $\lambda_0 = 0$. In this way, this example reveals instances where the adaptation on uncertainty sets improves the performance over fixed approaches.

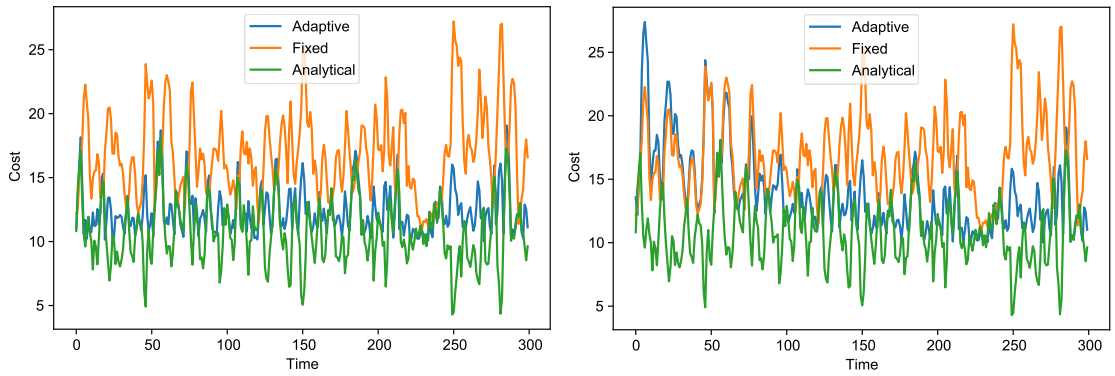


Figure 3.2: Daily costs for the three methods. In right and left plots, the value of λ_0 was set for the *Adaptive* method as 0 and 1.5, respectively.

Method	**Fixed**	**Adaptive**($\lambda_0 = 0$)	**Adaptive**($\lambda_0 = 1.5$)	**Analytical**
Average Cost [-]	16.831	12.420	13.915	10.717
Std. Deviation [-]	3.290	1.825	3.243	2.470

Table 3.1: Average and standard deviation of the three methods across the 300 days of backtest simulation.

In the context of control problems, the superior performance of closed-loop architectures is well-known over the open-loop ones. However, in energy management problems, as the UC, this relationship has been barely explored. This example serves as a motivation for developing a closed–loop scheme for the resolution of the UC. Before any claim, it is necessary to note that the differences between the newsvendor and the UC problems are huge, and these results may not be extendable. However, as the spirit of both problems is the same, there is doubt about the effectiveness of this concept in more complex systems. Nevertheless, this example states a precedent.

3.3 Proposed framework of study

The present study is made in the context of the real-time operation of electric power systems, considering the penetration of renewable generation. To that end, an analysis framework is proposed, which has a threefold purpose: (i) modeling the sequential decision–making process from which the operation is determined, (ii) evaluating the quality of the operation based on system indicators, and (iii) use the evaluation to apply corrective actions in the next period. As the focus is the design of efficient policies of uncertainty management on the UC problem, the framework is divided into three major components, depicted in Fig. (3.3).

The first is the short-term planning model, namely the UC model with uncertainty management policies. The second component is the power system environment, involving the dynamics for a fixed status of generators. On that basis, this component considers the calculation of generators dispatch levels, besides the corrective actions given by discrepancies derived from the realization of the uncertainty. Those deviations and the system responses give place to an evaluation of the operation based on operational indicators. Finally, the third component represents a module designed to modify the uncertainty management policies of the planning model, following the results obtained from the evaluation.

Mainly, we assume that the short-term planning model possesses a set of parameters representing the risk–averse level taken by the system operator, defining the robustness of the solution, similar to the second method presented in section (3.2). To summarize the order of the sequential actions, the decision-maker takes a value for the risk–averse level at each period, inducing a short-term planning decision. Lately, this decision, along with the realization of the uncertainty, defines the operation and, consequently, a set of system indicators. At last, the system indexes'

evaluation is used to determine the risk-averse level at the next period. This scheme is very similar to a closed-loop architecture, commonly studied in control problems.

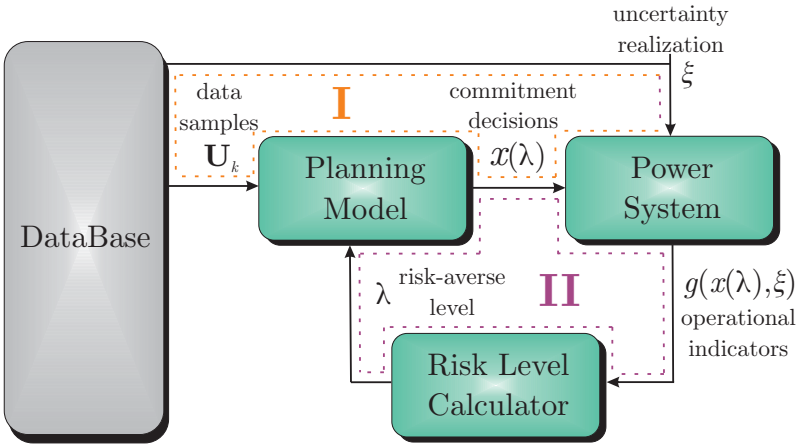


Figure 3.3: General proposed adaptive scheme

This framework is quite general and can be used for many problems with parameters subject to uncertainty. In this sense, the module *Database* represents the stored historical information, mainly used to extract samples for the planning problem and store the performance of the past operation. Many works have proposed adaptive schemes making use of the closed–loop I [5, 20, 21], depicted in Fig. (3.3). This closed–loop has the property of adapting the samples used in the planning stage by updating them according to the last observed realizations. In contrast, the closed–loop II considers also the behavior of the real–time operation, i.e., capturing information of the system and the performance of past control actions. In this sense, the risk-averse level x is expected to add more information to the planning problem, improving the indices derived from it. This risk-averse level is assumed to be part of a specific risk measure, which accounts for the impact of the selected samples. The exact influence of λ on the commitment decisions is addressed in the next section.

3.3.1 Two–stage UC model subject to uncertainty

For the UC problem subject to uncertainty, we use a standard two–stage formulation [147, 148], using the deterministic *3–bin* model as the base [3] for the integer region description, which is presented in the section (2.2.2). Before introducing a specific formulation, a more general approach is presented, introducing the risk measure $\rho_\lambda(.)$ for the evaluation of the second–stage costs, where λ is a parameter representing the risk–averse level of the decision–maker.

As was reviewed in section 2.3.1, the risk measure $\rho_\lambda(\cdot)$ is applied over the second stage costs distribution, which is function of first stage variables values and the distribution of the parameters subject to uncertainty. In the present work, these parameters are the values of REG availability and demand levels, which will be comprised in the matrix $\boldsymbol{U}$. To this end, it is considered that a discrete set of scenarios $\mathcal{S} = \{\boldsymbol{U}^k\}_{k\in\mathcal{K}}$, with $\mathcal{K}$ the scenario indexes set, and $\boldsymbol{U}^k$ the k–th scenario values for every parameter subject to uncertainty, is an available approximation for the real multidimensional probability distribution function of $\boldsymbol{U}$, where each value $\boldsymbol{U}^k$ is considered having the same probability of occurrence. In that way, the resultant formulation is the following:

$$\min_{\boldsymbol{X}\in\mathcal{X}_{3bin}} \quad \sum_{h\in\mathcal{H}} (\boldsymbol{c}^{fix})^T \boldsymbol{x}_h + \rho_\lambda\big(\{z(\boldsymbol{X},\boldsymbol{U}^k)\}_{k\in\mathcal{K}}\big) \tag{3.6}$$

where $\boldsymbol{c}^{fix}$ is the cost vector associated with the generators statuses, containing the start–up, shut–down and no–load costs, $\boldsymbol{x}_h$ is first stage vector of binary variables, representing the status of the generators at the hour h, with $\mathcal{H}$ the set of hours of planning, and $\boldsymbol{X}$ is the matrix that groups the $\boldsymbol{x}_h$ vectors, i.e., $\boldsymbol{X} = [\boldsymbol{x}_1\ \boldsymbol{x}_2\ \ldots\ \boldsymbol{x}_{|\mathcal{H}|}]$, belonging to the feasible region $\mathcal{X}_{3bin}$, defined by generators start–up costs definition and minimum up/down times constraints formulated in [3] (see sections II-A2 and II-B2). On the other hand, $z(\boldsymbol{X},\boldsymbol{U}^k)$ corresponds to the second stage costs, depending on both the generator statuses vectors $\boldsymbol{X}$ and the scenario values $\boldsymbol{U}^k$. Therefore, the value of $z(\boldsymbol{X},\boldsymbol{U}^k)$ is defined by the following second stage problem:

$$z(\boldsymbol{X}, \boldsymbol{U}^k) = \min \sum_{h \in \mathcal{H}} \left\{ (\mathbf{c}^g)^T \mathbf{p}_h^k + (\mathbf{c}^+)^T \mathbf{sp}_h^k + (\mathbf{c}^-)^T \mathbf{sm}_h^k \right\} \tag{3.7}$$

s.t.

$$\mathbf{A}\mathbf{f}_h^k + \mathbf{B}\mathbf{p}_h^k + \mathbf{P}\mathbf{w}_h^k = \mathbf{d}_h^k + \mathbf{sp}_h^k - \mathbf{sm}_h^k \qquad \forall h \in \mathcal{H} \tag{3.8}$$

$$\mathbf{1}^T \boldsymbol{r}_h^k \geq R_h \qquad \forall h \in \mathcal{H} \tag{3.9}$$

$$\mathbf{p}_h^k + \mathbf{r}_h^k \leq \mathbf{p}^{\max} \mathbf{x}_h \qquad \forall h \in \mathcal{H} \tag{3.10}$$

$$\mathbf{p}_h^k \geq \mathbf{p}^{\min} \mathbf{x}_h \qquad \forall h \in \mathcal{H} \tag{3.11}$$

$$\mathbf{f}_h^k = \mathbf{S}\boldsymbol{\theta}_h^k \qquad \forall h \in \mathcal{H} \tag{3.12}$$

$$-\bar{\mathbf{f}_h} \leq \mathbf{f}_h \leq \bar{\mathbf{f}_h} \qquad \forall h \in \mathcal{H} \tag{3.13}$$

$$\mathbf{p}_h^k + \mathbf{r}_h^k - \mathbf{p}_{h-1}^k \leq \mathbf{r}_{\text{up}}(\mathbf{x}_h, \mathbf{x}_{h-1}) \qquad \forall h \in \mathcal{H} \tag{3.14}$$

$$\mathbf{p}_h^k - \mathbf{p}_{h-1}^k \leq -\mathbf{r}_{\text{down}}(\mathbf{x}_h, \mathbf{x}_{h-1}) \qquad \forall h \in \mathcal{H} \tag{3.15}$$

$$0 \leq \mathbf{sp}_h^k \leq \mathbf{d}_h^k \qquad \forall h \in \mathcal{H} \tag{3.16}$$

$$0 \leq \mathbf{sm}_h^k \leq \mathbf{P}\mathbf{w}_h^k \qquad \forall h \in \mathcal{H} \tag{3.17}$$

The above problem defines the optimal generators dispatches given the statuses $\boldsymbol{X}$ and the uncertainty realization $\boldsymbol{U}^k$. In particular, the matrix $\boldsymbol{U}^k$ comprises the values of available REG $\mathbf{w}_h^k$ and demand levels $\mathbf{d}_h^k$ scenarios. All of the second stage variables are indexed by period, $h \in \mathcal{H}$, and by scenario, $k \in \mathcal{K}$. The vector $\boldsymbol{p}_h^k$ represents the dispatches levels of conventional generators, $\boldsymbol{f}_h^k$ is the vector of lines fluxes, $\boldsymbol{sp}_h^k$ and $\boldsymbol{sm}_h^k$ denote the non–served demand and the renewable generation spillage vectors, respectively, $\boldsymbol{r}_h^k$ is the vector of conventional generators reserves levels, and $\boldsymbol{\theta}_h^k$ the vector of bus voltage angles. The second stage costs (3.7) consider the dispatch costs and the costs related with the non–served demand and power spillage. Power and reserves requirements are accounted in equations (3.8) and (3.9), respectively, where matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{P}$ contains the appropriate parameters accordingly with the connection of every element into the network, and the vector $\mathbf{1}$ is the column vector with dimension equal to the number of conventional generators, on which every entry is equal to 1. Generation limits are stated in (3.10) and (3.11). Power fluxes definition and its limits are present in (3.12) and (3.13), respectively, where the matrix $\boldsymbol{S}$ contains the constants that define the linear approximation of DC power flow equations [7]. Ramp constraints are present in (3.14) and (3.15). Finally, non–served demand and power spillage limits are defined in (3.16) and (3.17), respectively. Although existing approaches optimize reserves levels as first–stage variables [20], our proposal is focused on highlighting the adaptation layer of the UC problem, which is why we apply typical procedures used by different ISOs, where a fixed value of total reserve is defined by an external method.

The above two–stage formulation determines the generator statuses in the first–stage, and generators dispatches and reserves levels are subsequently calculated for each $k \in \mathcal{K}$ in the second–stage. The risk measure $\rho_\lambda(.)$ accounts for the impact of the second stage costs, and therefore, the set of scenarios, into the first stage variables, in which the parameter λ define the robustness of the risk–measure. Is worth note that the same scenario set $\mathcal{S}$ could have a different impact on the commitment solution $\boldsymbol{X}^*$, depending on the definition of $\rho_\lambda(.)$. In this sense, many approaches fit in the above formulation, particularly every approach examined in section (2.4.1). For example, it can be considered the stochastic case by defining $\rho_\lambda(.) = \mathbb{E}_{k \in K}[.]$, and even use a distributionally robust optimization approach by imposing some conditions over the scenario set $\mathcal{S}$. The conditional–value–at–risk measure also fits, by simply defining $\rho_\lambda(.) = \text{CVaR}_\lambda(.)$, where the variation of λ allows to transit from the stochastic case with $\lambda = 0$, to the robust case with $\lambda = 1$, according to the definition made in [149]. The robust case also can be stated in this framework, i.e., $\rho_\lambda(.) = \max_{\boldsymbol{U} \in \mathcal{U}_\lambda}(.)$, if $\mathcal{U}_\lambda = \mathcal{U}_\lambda(\{\boldsymbol{U}_k\}_{k \in \mathcal{K}})$, i.e., if the corresponding uncertainty set is constructed from $\mathcal{S}$, in which λ could have, as an example, the same interpretation as the used in section (3.2). In this way, previous works have developed scenario–based uncertainty sets that are suitable in this formulation [19, 20, 23]. However, the present proposal uses modified version of the above–mentioned uncertainty, characterized by adding a degree of freedom λ, designed to control the robustness of the solution, according to the definitions made in the following section.

3.3.2 Uncertainty modeling

In the context of generation scheduling in systems with a high level of renewable-based generation penetration, the parameters subject to uncertainty usually considered in the short-term planning stage are renewable generation levels and buses' demands level. The present proposal assumes the availability of historical data for each of the parameters mentioned above. Concretely, the existence of time series with an hourly resolution for each bus's demanded active and reactive power is considered, besides the time series of the hourly historical power availability for each renewable generator connected to the system. Hence, the methodology applied for the scenario generation is the same as the one used in [20], where time series are divided by hours to construct the scenario set s. Thus, let $\mathcal{H}$ the set of hours of planning considered in the UC model, $\mathcal{B}$ the set of buses, and $\mathcal{R}$ the set of renewable generators. Each scenario k will be represented as an $(|\mathcal{B}|+|\mathcal{R}|)\times|\mathcal{H}|$ matrix, namely $\boldsymbol{U}_k$, with $k \in \mathcal{K}$, containing the historical values for $\mathcal{H}$ consecutive hours, for both the demanded power and maximum availability for renewable generators. Note that for $|\mathcal{H}| = 24$, each scenario contains the daily profile for each uncertainty parameter, as can be seen in Fig. 3.4.

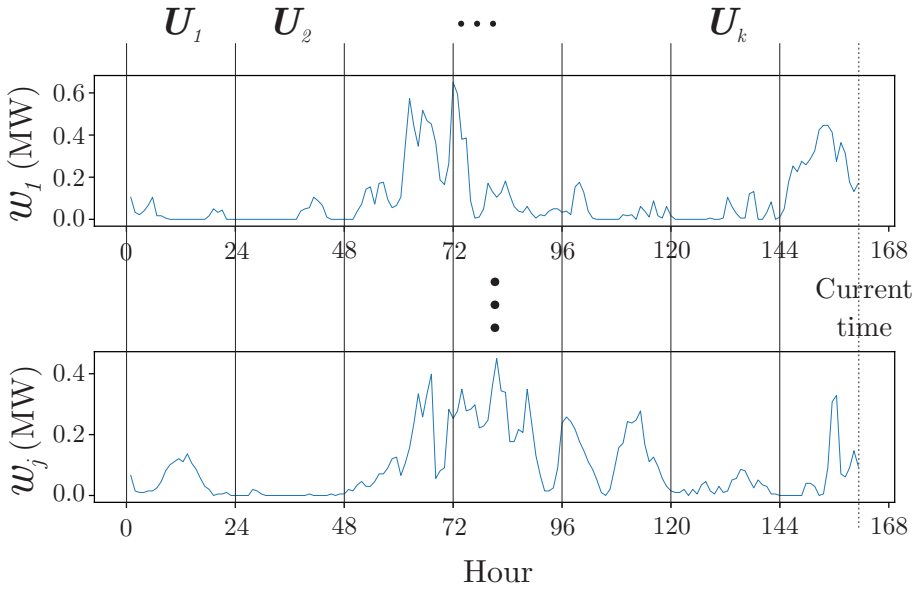


Figure 3.4: Scenario construction example

Consequently, if the problem (3.6)-(3.17) is solved at time t, observations from the hour $t-|\mathcal{K}||\mathcal{H}|$ until the hour t are equally divided to form the $|\mathcal{K}|$ scenarios. This results in a moving window, on which recent observation updates the samples used in the planning stage, as is depicted in the closed–loop I of the Fig. (3.3). Therefore, the extracted samples from the time series are ordered to construct the scenario matrix $\boldsymbol{U}_k$ as is presented below:

$$\boldsymbol{U}_k = \begin{bmatrix} d_1^1 & d_1^2 & \dots & d_1^{|\mathcal{B}|} & w_1^1 & w_1^2 & \dots & w_1^{|\mathcal{R}|} \\ d_2^1 & d_2^2 & \dots & d_2^{|\mathcal{B}|} & w_2^1 & w_2^2 & \dots & w_2^{|\mathcal{R}|} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{|\mathcal{H}|}^1 & d_{|\mathcal{H}|}^2 & \dots & d_{|\mathcal{H}|}^{|\mathcal{B}|} & w_{|\mathcal{H}|}^1 & w_{|\mathcal{H}|}^2 & \dots & w_{|\mathcal{H}|}^{|\mathcal{R}|} \end{bmatrix} \tag{3.18}$$

One of the greatest advantages of this scenario definition is the capacity to capture the underlying correlations between the parameters on each multivariate point. Fig. (3.4) shows this feature graphically. Every profile bounded by two vertical bars is compacted in one matrix scenario. In the case of RUC models, uncertainty sets constructed as the convex hull of s have been previously used in [19, 20], taking advantage of the simple definition that allows ease implementations while maintaining the data–driven spirit of construction. In those implementations, the robustness is completely determined by the number of scenarios considered $|\mathcal{K}|$. Therefore, a modified version of the above-defined uncertainty set is made to incorporate another measure of the risk-averse level. Then, consider the multivariate mean point of the scenario set $\mathcal{S}$, namely $\overline{\boldsymbol{U}}$. Let $\tilde{\boldsymbol{U}}_k(\lambda)$ the convex

combination between the original point $\boldsymbol{U}_k$ and $\overline{\boldsymbol{U}}$:

$$\tilde{\boldsymbol{U}}_k(\lambda) = \lambda \boldsymbol{U}_k + (1-\lambda)\overline{\boldsymbol{U}}, \quad \lambda \in [0,1] \tag{3.19}$$

With this, let $\mathcal{U}_\lambda$ the modified uncertainty set constructed as the convex hull of the modified points $\tilde{\boldsymbol{U}}(\lambda)$.

$$\mathcal{U}_\lambda = \left\{ \boldsymbol{U} \in \mathbb{R}^{(|\mathcal{B}|+|\mathcal{R}|)\times|\ \mathcal{H}|} \middle| \boldsymbol{U} = \sum_{k\in\mathcal{K}} \alpha_k \tilde{\boldsymbol{U}}_k(\lambda),\ \sum_{k\in\mathcal{K}} \alpha_k = 1,\ \alpha_k \geq 0 \right\} \tag{3.20}$$

Hence, for $\lambda = 1$, the scenario–based uncertainty set $\mathcal{U}_\lambda$ defines a robust approach, and for $\lambda = 0$, the above formulation is equivalent to the deterministic approach, where the parameters subject to uncertainty take their expected values.

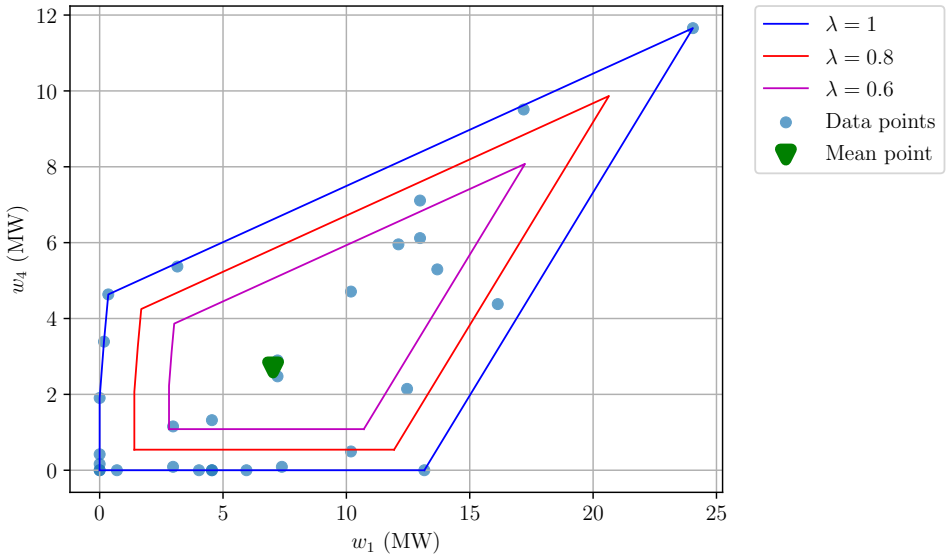


Figure 3.5: Two–dimensional proposed uncertainty sets for the renewable generation of two wind generators at hour 5, using 30 samples extracted from [2].

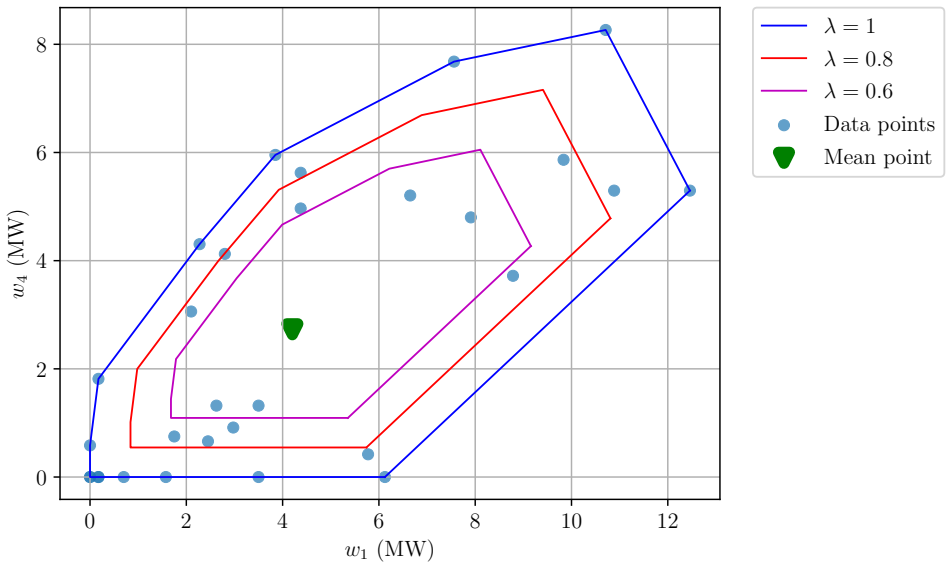


Figure 3.6: Two–dimensional proposed uncertainty sets for the renewable generation of two wind generators at hour 13, using 30 samples extracted from [2].

In Fig. 3.5 and 3.6, examples of two-dimensional uncertainty sets, i.e., considering two time series and one sample per each one, for two different hours of the day, are shown, with data extracted from [2]. That renewable generation of the past 30 days at the same hour is depicted with dots for two buses. It can be seen that for different hours, the shape of the sets can change drastically. The convex hull of the scenario points corresponds with the curve with $\lambda = 1$, representing the classical full robust approach. The modified uncertainty set (3.20) with variable shape is shown for $\lambda = 0.8$ and $\lambda = 0.6$.

As can be observed, these modified sets keep the shape of the original robust set ($\lambda = 1$) but are contracted, being closer to the average of the scenario points. The above definition is mainly inspired by simplicity, both in definition and implementation. Instead, the fulfillment of a specific list of constraints with statistical meaning is not pursued. In comparison with the CVaR–based uncertainty set, shown in Fig. (2.9), both sets regulate their robustness by changing a single parameter in the range $[0, 1]$. However, CVaR–based sets have statistical interpretation but require an expensive computational implementation when the number of scenarios grows [19]. Although considering the lack of statistical interpretation of the proposed set, the mere integration of a degree of flexibility into the set definition could improve the results of the fixed case ($\lambda = 1$). The basis for the dynamic determination of λ is described in the next section.

3.4 Sequential decision–making formulation

To formulate the problem of the adaptive determination of the robustness level λ for the two–stage UC, the formalism of DP is used, which was shown to be analogous to RL in section (2.5). Accordingly, is necessary to construct the equivalence between the closed–loop scheme (3.3) and a stochastic DP description, depicted in Fig. (2.11).

In this sense, consider that at a certain time t, the operator solves the two–stage UC problem defined by equations (3.6)-(3.17), getting optimal generator schedules $\boldsymbol{X}^\star = [\boldsymbol{x}_1^\star\ \boldsymbol{x}_2^\star\ ...\ \boldsymbol{x}_{|\mathcal{H}|}^\star]$. Even though $\boldsymbol{X}^\star$ contains the optimal generators' status for the next $|\mathcal{H}|$ periods, we consider the ISO uses a rolling horizon scheme and only applies solutions every $N\Delta$ hours, with Δ a minimum reference time–step, and $N\Delta \leq |\mathcal{H}|$. A previous work have showed the benefits of applying rolling horizon frameworks with small periodicities to the UC [65]. For the sake of the feasibility of the operation, is necessary to capture the previous operation with appropriate initial conditions. This is achieved by fixing the values of $\boldsymbol{x}_1$ and $\boldsymbol{p}_1^k$, $\forall k \in \mathcal{K}$ to those in the operation at time t, and by keeping track of generators' up/down times. On the other hand, ramp constraints for the initial period are naturally included by fixing the values of $\boldsymbol{p}_1^k$ as the ones obtained in the previous operation. The solutions obtained by this methodology correspond to the commitment solution $\boldsymbol{x}(\lambda)$ described in Fig. 3.3.

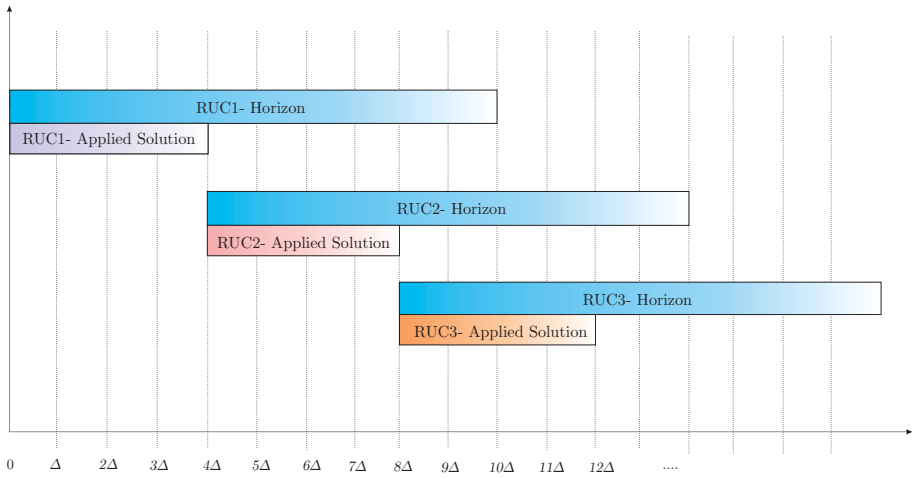


Figure 3.7: Calculation and implementation schedules for RUC solutions under the rolling horizon methodology. For this case $N = 4$, impliying that the implementation of UC solutions is made every 4Δ, while the horizon of the UC calculation is 10Δ.

In the Fig. 3.7 an schema of the calculation and application of UC solutions is given. In this

case, $N = 4$, resulting in a delay of 4Δ between the calculation and the application of the UC solution. Besides, it can be seen that the time of application of the solution i, and the beginning of the calculation of the solution $i+1$ are the same, where the last one includes as an initial condition the schedule according with the solution i.

Subsequently, the obtained solution is applied to the power system module of Fig. (3.3). In this module, dispatches levels for conventional generators are calculated, using a method that will be soon into the next section. Finally, given the realization of the uncertainty $\boldsymbol{\xi}_t$, corrective actions are taken. Note that $\boldsymbol{\xi}_t$ is referring to REG availability and demand values measured with a periodicity equal to Δ, whereas $\mathbf{w}_h^k/\mathbf{d}_h^k$ make reference to hourly measures, i.e., $\mathbf{w}_h^k/\mathbf{d}_h^k$ are obtained by the aggregation of $\boldsymbol{\xi}_t$. Operational indexes that account for these corrections are also evaluated here, allowing for the computation of the next value of λ. Consider the index $t(n) = \Delta N n$, accounting for the periodicity of the RUC calculation and, without loss of generality, let express us express the equation in terms of n. Given the system state at time n, $\boldsymbol{s}_n$, the function defined by the solution of (3.6)–(3.17) along with the uncertainty realization, define the next period state $\boldsymbol{s}_{n+1} = \boldsymbol{f}_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\Xi}_n)$, incurring in an operating cost $c_n = g_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\Xi}_n)$, with $\boldsymbol{\Xi}_n = [\boldsymbol{\xi}_{t(n-1)+1}, \boldsymbol{\xi}_{t(n-1)+2}, ..., \boldsymbol{\xi}_{t(n)}]$. Lastly, a mapping function μ_n defines the next robustness level $\lambda_{n+1} = \mu_{n+1}(s_{n+1})$.

In order to make a formal statement of the definitions made above, the elements of the dynamic programming formulation are presented below:

1. State variables $\boldsymbol{s}_n \in \mathcal{S}$: tuple composed of the concatenation of *(i)* the hour of the day, *(ii)* the sum of generator's status at time n, and *(iii)* an integer approximation of the sum of the differences between real and planning dispatch costs over the time interval $[t(n-1), t(n)]$.

2. Control variable $\lambda_n \in \Lambda$: robustness level of the risk measure $\rho_\lambda(\cdot)$ used in (3.6).

3. State transition function $\boldsymbol{f}_n$: sequential solution of the UC problem (3.6)–(3.17) for hour $h = N\Delta$, and simulation of the corresponding operation.

4. Transition cost function c_n: selected operational index, obtained as an output of the power system operation, which can be stated as $c_n = g_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\Xi}_n)$.

For a better understating of the proposed dynamic formulation for the UC problem, Fig. 3.8 shows how the DP variables are implemented in real–time operation for the case with $N = 4$. As can be observed, commitment solutions to be implemented $\boldsymbol{X}_n$ are calculated using the risk–level λ_n, calculated from a control policy $\mu_n(\boldsymbol{s}_n)$ that depends on the current state and the past experience. Then, while the system operates under the generators schedule defined by $\boldsymbol{X}_n$, the real values of REG availability and demand $\boldsymbol{\Xi}_n = \{\boldsymbol{\xi}_{t(n-1)+\Delta}, ..., \boldsymbol{\xi}_{t(n)}\}$ define the operation cost c_n, and the next state $\boldsymbol{s}_{n+1}$, which are subsequently stored for future calculations. Later, when $\boldsymbol{s}_{n+1}$ is reached, the process is repeated.

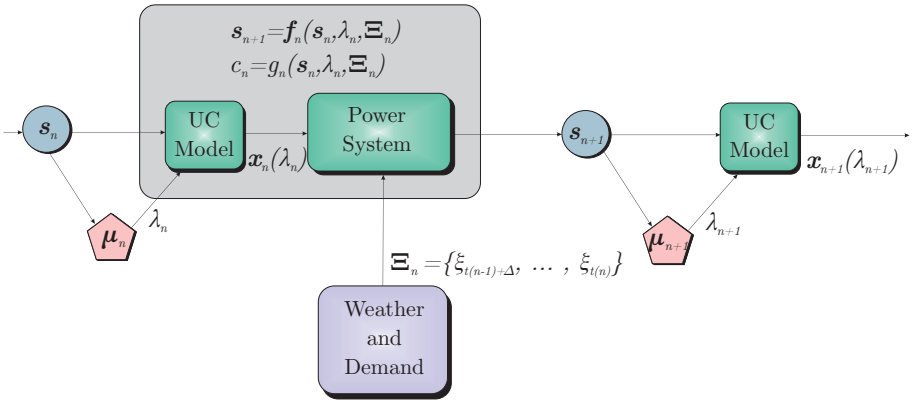


Figure 3.8: Schematic representation of the DP formulation for the adpative UC framework.

With those definitions being made, let $\boldsymbol{s}_0$ being the initial state, and consider the set of policies $\pi = \{\mu_0, \mu_1, ...\}$, from which the robustness level is determined by $\lambda_n = \mu_n(\boldsymbol{s}_n)$. Then, the optimal

expected total cost over an infinite horizon can be stated as:

$$J^{\star}(\mathbf{s}_0) = \min_{\pi} \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n g_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\Xi}_n)\right] \tag{3.21}$$

which is equivalent to optimize over $\lambda_n \in \Lambda$, where Λ is the domain of the robustness levels allowed by the operator. On that wise, the equation (3.21) addresses the principle of optimality of the tail subproblem [104], allowing the definition of Bellman's equation:

$$J^{\star}(\boldsymbol{s}_n) = \min_{\lambda_n \in \Lambda} \mathbb{E}\left[g_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\Xi}_n) + \gamma J^{*}(\boldsymbol{s}_{n+1})\right] \tag{3.22}$$

with $0 \leq \gamma \leq 1$ the discount factor. As was reviewed in section 2.5, the formulation presented above is relatively standard and has been extensively studied. However, formulating the resolution of the UC in this way allows the application of several techniques, particularly RL techniques, to dynamically determine approximations of the optimal control policies $\mu_n^{\star}(\cdot)$, which determine the values of λ. Nevertheless, since the complexity of transition function, and the behavior of random variable $\boldsymbol{\xi}_{t(n)}$, a series of approximations has to be made to obtain feasible and implementable solutions. In particular, the determination of operational indexes from system operation simulation and the construction of robustness levels requires a series of definitions developed in the next section.

3.5 Real–time operation simulation

This section presents the assumptions taken to model the power system's real-time operating conditions. UC decisions, dispatch decisions, and the evaluation of operational indicators are modeled to be taken sequentially. In this sense, the power system module of Fig. 3.3 is simulated by an optimal power flow (OPF) model, from which generators dispatches are obtained. Then, exact power flow equations are solved, simulating the real-time steady-state behavior of the system, accounting for the imbalances existing between planning and operation stages.

As illustrated in Fig. 3.9,RUC solutions are applied and fixed for a length period $N\Delta$. As the calculation and application of RUC solutions have a delay of $N\Delta$ hours, the state calculation at period t considers as initial condition the solution calculated at period $t - N\Delta$, as depicted on Fig. 3.7. Under this schedule, the OPF model is solved each Δ hours, determining the generator dispatches that will be applied at the next time step, with a delay of Δ between calculation and implementation. Finally, jointly with the uncertainty realization, dispatch levels previously calculated are used to solve exact power flow equations every Δ. The existing discrepancies between planning operation and real operation are supplied by the system flexibility resources, where the magnitude of their use, along with system operation costs, are used to compute the operational indicators of interest.

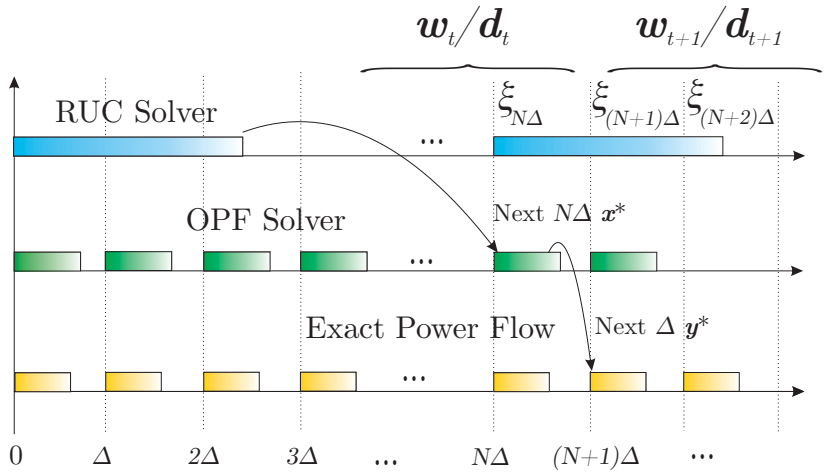


Figure 3.9: Simulation of real–time operation. $\boldsymbol{\xi}$ values are aggregated to obtain $\mathbf{w}_h^k/\mathbf{d}_h^k$ scenarios values. Rectangles lengths represent the calculation time of each model. Arrows indicate the times on which UC and OPF solutions are implemented, after being calculated.

For the OPF model resolution, renewable generation $\boldsymbol{w}$ and demand $\boldsymbol{d}$ values are set equal to the last observed realization, i.e., the value available at the beginning of the calculation process. Hence, there is a delay of Δ between the real-time value and the one used for the calculation.

In this way, the system's operation is divided into three levels of decisions, focusing on the adaptation process of commitment solutions. Their quality is measured through the performance of the third layer, i.e., out–of–sample results obtained by solving exact power flow equations, using out–of–sample data. Moreover, this 3-level schema, as a simulation framework of real-time operation, could be used to test either OPF or UC models using the same adaptation strategy. Then, the general framework shown in Fig. 3.3is now specified on the schema illustrated in Fig. 3.10, where the closed–loop structure is kept.

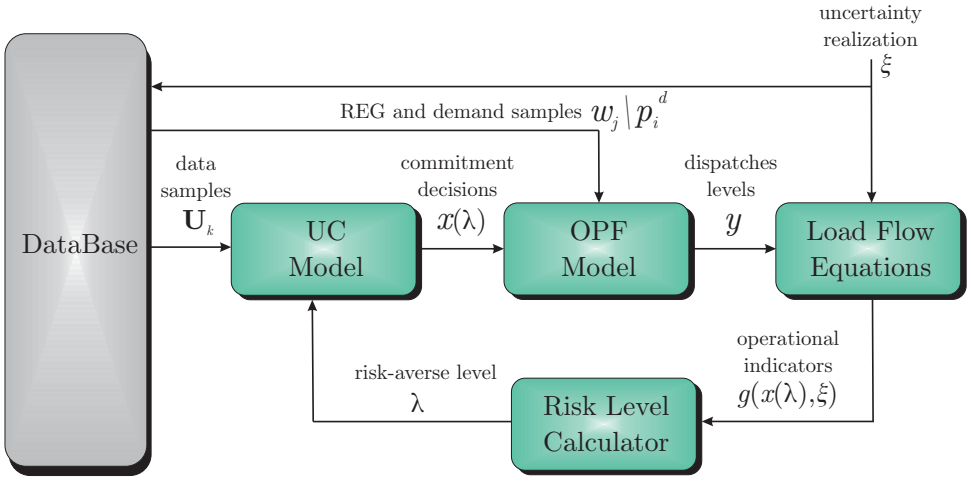


Figure 3.10: Proposed closed–loop scheme and experimental framework. Arrows represent the flow of information between modules.

Concerning the OPF model, the proposed framework uses a second-order cone programming (SOCP) relaxed model [150], including slack variables for non–served demand and renewable power spillage, which are appropriately penalized as in the cost function of the UC problem (3.7). The specific OPF model used is formulated below:

$$\min \quad \sum_{i\in\mathcal{G}} c_i^g p_i^g + \sum_{b\in\mathcal{B}} \left(c^+(sp_b^p + sp_b^q) + c^-(sm_b^p + sm_b^q) \right) \tag{3.23}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{G}_i} p_j^g + \sum_{j\in\mathcal{R}_i} w_j - p_i^d + sm_i^p - sp_i^p = G_{ii}c_{ii} + \sum_{j\in\delta(i)} (G_{ij}c_{ij} - B_{ij}s_{ij}) \quad \forall i \in \mathcal{B} \tag{3.24}$$

$$\sum_{j\in\mathcal{G}_i} q_j^g - q_i^d + sm_i^q - sp_i^q = -B_{ii}c_{ii} + \sum_{j\in\delta(i)} (-B_{ij}c_{ij} - G_{ij}s_{ij}) \quad \forall i \in \mathcal{B} \tag{3.25}$$

$$\underline{V}_i^2 \le c_{ii} \le \bar{V}_i^2 \quad \forall i \in \mathcal{B} \tag{3.26}$$

$$c_{ij} = c_{ji},\ s_{ij} = -s_{ji} \quad \forall (i,j) \in \mathcal{L} \tag{3.27}$$

$$c_{ij}^2 + s_{ji}^2 = c_{ii}c_{jj} \quad \forall (i,j) \in \mathcal{L} \tag{3.28}$$

$$p_i^{min} \le p_i^g \le p_i^{max} \quad \forall i \in \mathcal{G} \tag{3.29}$$

$$q_i^{min} \le q_i^g \le q_i^{max} \quad \forall i \in \mathcal{G} \tag{3.30}$$

where $\mathcal{G}_i$ and $\mathcal{R}_i$ are the sets of conventional and renewable generators connected to bus i respectively. The set $\delta(i)$ contains the indexes of buses connected to bus i. The cost function (3.23) contains linear generation costs for active power dispatches of conventional generators, and imbalances costs accounted for positive slack variables for active and reactive power on each bus b, sp_b^p and sp_b^q, respectively, and negative slack variables for the same quantities, sp_b^q and sm_b^q. Equations (3.24) - (3.25) represent balances in both active and reactive power on each bus, with p_i^g and q_i^g the active and reactive power dispatches of the conventional generator i, respectively, w_j the active power supplied by the renewable generator j, with p_i^d and p_i^q the demanded active and

reactive power of bus i, respectively. The parameters G_{ij} and B_{ij} denote the real and imaginary components of the system admittance matrix, whereas $e_i = |V_i|\cos(\theta_i)$ and $f_i = |V_i|\sin(\theta_i)$ are the real and imaginary components of the voltage phasor of bus i, respectively. In this model, the parameters w_j, p_i^d and q_i^d are set equal to the last realization.

It is known that using relaxations of the OPF model could produce optimal solutions that could be infeasible in the real power flow manifold. Therefore exact power flow equation could not have any solutions using this solution dispatch. However, in this proposal, equations of the third layer are formulated with enough flexibility to absorb both model and data inconsistencies.These power flow equations include extended characteristics both for generators and loads, modeling system local controllers, and ensuring a solution's existence simultaneously. Specifically, the equations include power–frequency droop characteristics for generators and impedance–voltage dependence for loads, which has been already implemented in simulation softwares, like *OpenDSS* [151]. Besides, voltage–dependent loads have been studied in the context of demand response, where consumers can react to power prices variations [152]. The above–mentioned characteristics are respectively shown hereunder:

$$P_g(w) = \begin{cases} P^g_{\max} & w \leq w^g_{\min} \\ P^g_{nom} - \frac{1}{d_c}(w - w_n) & w^g_{\min} \leq w \leq w^g_{\max} \\ P^g_{\min} & w^g_{\max} \leq w \end{cases} \tag{3.31}$$

$$S_d(V_b) = \begin{cases} \frac{V_b^2}{Z_1} & V_b \leq V_{\min} \\ S_{nom} & V_{\min} \leq V_b \leq V_{\max} \\ \frac{V_b^2}{Z_2} & V_{\max} \leq V_b \end{cases} \tag{3.32}$$

where P_g is the active power generated by the conventional generator g, w is the system angular frequency, $P^g_{\max}/P^g_{\min}$ are maximum and minimum generation levels for the generator g, respectively, P^g_{nom} is the dispatch solution of the OPF model, d_c the droop constant, w_n is the system nominal frequency, and $w^g_{\max}/w^g_{\min}$ are the maximum and minimum frequency limits for the generator g, respectively. In addition, S_d represents the apparent power consumed by the load d, V_b is the voltage magnitude of bus b, Z_1 and Z_2 are constant impedance values, and $V_{\min}/V_{\max}$ are the minimum and maximum admitted bus voltage magnitudes, respectively. The characteristic defined in (3.31) states that each generator posses a window on which can deliver power, limited by generation limits and ramps rates. On the other hand, equation (3.32) imposes that the power consumed by loads is constant between admissible values of bus voltage magnitudes. Conversely, operating out of this range means a constant impedance behavior. Graphically, modified characteristics of both generators and loads are described by piece–wise linear functions, as is shown in Fig. 3.11 and 3.12, respectively.

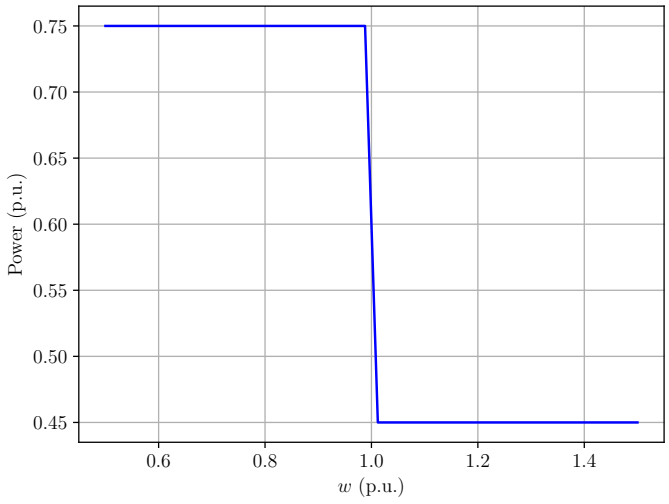


Figure 3.11: Power–frequency curve for generators.

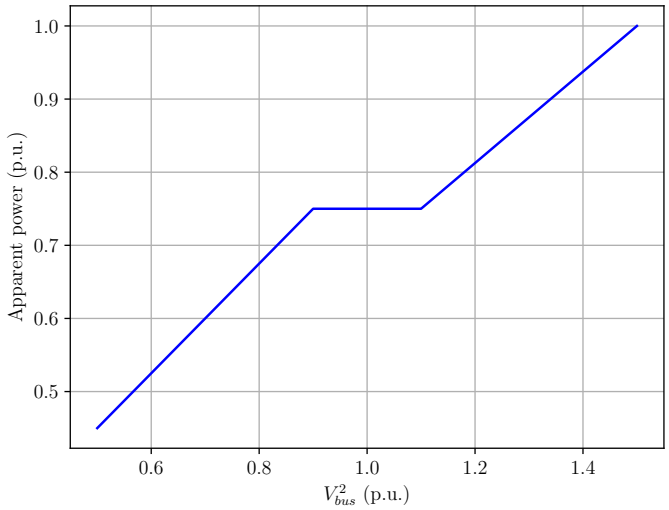


Figure 3.12: Power–voltage curve for loads.

As result, the modified load flow equations used for the operation evaluation are stated as follows:

$$\sum_{j\in\mathcal{G}(i)} p_j^g(w) + \sum_{j\in\mathcal{R}(i)} w_j - p_b^d(V_i) = \sum_{i\in\delta(i)} V_i V_j (G_{ij}\cos(\theta_i-\theta_j) + B_{ij}\sin(\theta_i-\theta_j)) \qquad \forall i \in \mathcal{B} \tag{3.33}$$

$$\sum_{j\in\mathcal{G}(i)} q_j^g - q_i^d(V_i) = \sum_{j\in\delta(i)} V_i V_j (G_{ij}\sin(\theta_i-\theta_j) - B_{ij}\cos(\theta_i-\theta_j)) \qquad \forall i \in \mathcal{B} \tag{3.34}$$

(3.31) $\forall i \in \mathcal{G}$ (3.35)

(3.32) $\forall b \in \mathcal{B}$ (3.36)

$$q_j^g = \begin{cases} Q_{slack} & j = j_{slack} \\ \bar{q}_j^g & j \neq j_{slack} \end{cases} \qquad \forall j \in \mathcal{G} \tag{3.37}$$

$$V_{i_{slack}} = 1 \tag{3.38}$$

$$\theta_{i_{slack}} = 0 \tag{3.39}$$

where w_j is the value of renewable generation extracted from real–time data, and j_{slack} is the index of the slack generator that supplies the necessary amount of reactive power Q_{slack}. The other generators are defined to dispatch the reactive power $\bar{q}_j^g$ calculated in the OPF. Finally, i_{slack} is the index of the slack bus, which has a fixed value of voltage, equal to 1(p.u.), and defines the reference bus angle.

3.6 Solution methodology

In this work, the adaptation is solely evaluated for the robust risk measure, i.e., solution methods for problem (3.6)-(3.17) are only analyzed for $\rho_\lambda(.) = \max_{\boldsymbol{U}\in\mathcal{U}_\lambda}(.)$.

3.6.1 Two–stage UC model resolution

Solution methods for the problem (3.6)-(3.17) heavily depend on the function $\rho_\lambda(.)$, leading to the need to analyze strategies case by case. In the present work, two cases are analyzed, namely the robust case, with $\rho_\lambda(.) = \max_{\boldsymbol{U}\in\mathcal{U}_\lambda}(.)$, using the scenario–based uncertainty set (3.20), and the conditional value–at–risk case, with $\rho_\lambda(.) = \mathrm{CVaR}_\lambda(.)$.

In this case, we will use the C&CG algorithm, leading to a master–subproblem structure. Since the second stage problem (3.7)-(3.17) is linear, the optimal solution of $\max_{\boldsymbol{U}\in\mathcal{U}_\lambda}(z_k(\boldsymbol{X},\boldsymbol{U}))$ lies in an extreme point of $\mathcal{U}_\lambda$ [153] and, considering that $\mathcal{U}_\lambda$ is constructed as the convex hull of the scenarios $\{\boldsymbol{U}^k\}_{k\in\mathcal{K}}$, optimizing over $\mathcal{U}_\lambda$ is equivalent to optimize over $\{\boldsymbol{U}^k\}_{k\in\mathcal{K}}$. Therefore, let $\tilde{z}_k(\boldsymbol{p}_h^k, \boldsymbol{sp}_h^k, \boldsymbol{sm}_h^k) = \sum_{h\in\mathcal{H}}((\mathbf{c}^g)^T\mathbf{p}_h^k + (\mathbf{c}^+)^T\mathbf{sp}_h^k + (\mathbf{c}^-)^T\mathbf{sm}_h^k)$ the second stage cost of the scenario k, and the master problem defined as follows:

$$\min_{\boldsymbol{X}\in\mathcal{X}_{3bin}} \sum_{h\in\mathcal{H}} (\boldsymbol{c}^{fix})^T\boldsymbol{x}_h + \eta \tag{3.40}$$

s.t.

$$\eta \geq \tilde{z}_k(\boldsymbol{p}_h^k, \boldsymbol{sp}_h^k, \boldsymbol{sm}_h^k) \qquad \forall k \in \mathcal{K}_j \tag{3.41}$$

$$\text{Constraints (3.8)-(3.17)} \quad \forall k \in \mathcal{K}_j \tag{3.42}$$

Problem **MP** is equivalent to (3.6)-(3.17) if $\mathcal{K}_j = \mathcal{K}$. In this way, the spirit of the C&CG algorithm is to sequentially add critical scenarios to solve a smaller problem with $\mathcal{K}_j \subseteq \mathcal{K}$. To this end, consider the following subproblem:

$$\textbf{SP}\text{: } Q(\boldsymbol{X}) := \max_{k\in\mathcal{K}} z_k(\boldsymbol{X}, \boldsymbol{U}^k) \tag{3.43}$$

Due to the existence of slack variables $\boldsymbol{sp}_h^k$ and $\boldsymbol{sm}_h^k$, the original problem has a complete resource in the second stage [147], i.e., $\forall \boldsymbol{X} \in \mathcal{X}_{3bin}$, the problem **SP** has a feasible solution. Thus, if $\tilde{\boldsymbol{X}}$ is a feasible solution of (3.6)-(3.17), then $\tilde{\boldsymbol{X}} + Q(\tilde{\boldsymbol{X}})$ is an upper bound for the original problem (3.7)-(3.17). Conversely, if $\mathcal{K}_j \subseteq \mathcal{K}$, **MP** represents a relaxation of the original problem. Accordingly, a sequence of every time tighter bounds can be obtained by following Algorithm 2 presented below.

Algorithm 2: C&CG algorithm for the two–stage RUC

```
Initialization: j ← 0, K_j ← ∅, UB ← +∞, LB ← −∞, and select a tolerance value δ ;
Solve MP and get the optimal solution (X*, η*) ;
Update LB = Σ_{h∈H} (c^fix)^T x_h^* + η* ;
Solve SP and get Q(X*) and k* ;
Update UB = Σ_{h∈H} (c^fix)^T x_h^* + Q(X*) ;
if UB − LB ≤ δ then
    Return X* ;
else
    Update j ← j + 1 and K_{j+1} ← K_j ∪ {k*} ;
    Go to step 2 ;
end
```

3.6.2 Robustness level determination via double Q–learning

According to [104], many strategies can be selected to approximately solve Bellman's equation (3.22). In this work, is opt to use the double Q–Learning algorithm [110]. Accordingly, consider the Q-function defined as $Q(\boldsymbol{s}_n, \lambda_n) = \mathbb{E}\left[g_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\xi}_n) + \gamma J(\boldsymbol{s}_{n+1})\right]$, which evaluates the future expected cost of applying the control λ_n given the current state $\boldsymbol{s}_n$. Then, the optimal value of Q also satisfies the Bellman's equation:

$$Q^\star(\boldsymbol{s}_n) = \min_{\lambda_n\in\Lambda} \mathbb{E}\left[g_n(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\xi}_n) + \gamma \min_{\nu\in\Lambda} Q(\boldsymbol{s}_{n+1}, \nu)\right] \tag{3.44}$$

where the optimal policy (control law) is obtained as

$$\mu_n^\star(\boldsymbol{s}_n) \in \arg\min_{\lambda_n\in\Lambda} Q(\boldsymbol{s}_n, \lambda_n) \tag{3.45}$$

To solve (3.44) approximately and generate a recursive method to update the Q–function, consider a discretization of the robustness level domain $\Lambda = \{\lambda^j\}_{j\in\mathcal{J}}$, with $\mathcal{J} = \{1, 2, ..., M\}$. Given

the definition of $\boldsymbol{s}_n$ in Section 3.4, the set $\mathcal{S} = \{\boldsymbol{s}^i\}_{i\in\mathcal{I}}$ is already discrete, with $\mathcal{I} = \{1, 2, ..., N\}$. Hereby, the Q–function can be interpreted as an $N \times M$ matrix or table, $Q(i, j)$, on which every entry (i, j) contains the Q–value of the pair $(\boldsymbol{s}^i, \lambda^j)$. Without loss of generality, let us express the DP equation in terms of states and control variables indexes, $(\boldsymbol{s}_n, \lambda_n) \to (i_n, j_n) \in \mathcal{I} \times \mathcal{J}$. In this way, consider that at certain time n, the system is in the state i_n, and the estimation of the Q–table is $\tilde{Q}_n(i, j)$, which generates an estimate of the optimal robustness level index $j_n \in \arg\min_{j\in\mathcal{J}} \tilde{Q}_n(i_n, j)$. The Q–learning algorithm uses the recursive definition (3.44) to update the Q–table as follows:

$$\tilde{Q}_{n+1}(i, j) = (1 - \alpha_n)\tilde{Q}_n(i, j) + \alpha_n(F_n(\tilde{Q}_n, v))(i, j) \tag{3.46}$$

where $0 < \alpha_n < 1$ is the learning rate at time n and $F_n(\tilde{Q}_n, v)$ is the updating operator defined as:

$$(F_n(\tilde{Q}_n, v))(i, j) = \begin{cases} g_n(i_n, j_n, \boldsymbol{\xi}_n) + \gamma\tilde{Q}_n(i_{n+1}, v) & (i, j) = (i_n, j_n) \\ \tilde{Q}_n(i, j) & (i, j) \neq (i_n, j_n) \end{cases} \tag{3.47}$$

with $v \in \arg\min_{j\in\mathcal{J}} \tilde{Q}_n(i_n, j)$. Equation (3.47) states that the Q–value of the pair (i_n, j_n) is updated using a step–size α_n, remaining all others Q–values unchanged.

Since at the beginning of the algorithm few historical information has been captured in $\tilde{Q}_t$, at the first periods is necessary to explore many values of λ rather than select the value induced by the estimator. Oppositely, as the algorithm evolves, it becomes desirable to choose values induced by $\tilde{Q}_t$ rather than exploring. To this end, an ϵ–greedy selection policy is often used:

$$S_\epsilon(\tilde{Q}_n) = \begin{cases} \text{Select } j \text{ w. prob. } \frac{1-\epsilon_n}{|\mathcal{J}_{\tilde{Q}}|} & \forall j \in \mathcal{J}_{\tilde{Q}} \\ \text{Select } j \text{ w. prob. } \frac{\epsilon_n}{|\Lambda^d/\mathcal{J}_{\tilde{Q}}|} & \forall j \in \Lambda^d/\mathcal{J}_{\tilde{Q}} \end{cases} \tag{3.48}$$

where $\mathcal{J}_{\tilde{Q}} := \arg\min_{j\in\mathcal{J}} \tilde{Q}_n$, and $0 \leq \epsilon_n \leq 1$ is the exploring probability, which is expected to decrease as iterations progress. In this way, the double Q–Learning algorithm to the robustness level determination is described in Algorithm 3.

Algorithm 3: Double Q–learning algorithm for robustness level determination.

Initialization: $n = 0$, $\tilde{Q}_0^1(i, j) = \tilde{Q}_0^2(i, j) = 0$, $\forall(i, j)$, $\boldsymbol{s}_0 = \bar{\boldsymbol{s}}$, $\alpha_0 = \bar{\alpha}$, $\epsilon_0 = \bar{\epsilon}$;
while $n \leq |\mathcal{T}|$ **do**
  Obtain j_n from $S_\epsilon(\tilde{Q}_n^1 + \tilde{Q}_n^2)$;
  Solve (3.6)-(3.17) with $\lambda = \lambda_{j_n}$ to obtain $\boldsymbol{x}^*$;
  Run the simulation model to obtain $\boldsymbol{s}_{n+1} = \boldsymbol{f}(\boldsymbol{s}_n, \lambda_n, \boldsymbol{\Xi}_n)$ and $g_n(i_n, j_n, \boldsymbol{\Xi}_n)$;
  if *event with probability* 0.5 **then**
    Select $v \in \arg\min_{j\in\mathcal{J}} \tilde{Q}_n^1(i_n, j)$;
    Update $\tilde{Q}_{n+1}^1(i, j) = (1 - \alpha_n)\tilde{Q}_n^1(i, j) + \alpha_n(F_n(\tilde{Q}_n^2, v))(i, j)$;
  else
    Select $v \in \arg\min_{j\in\mathcal{J}} \tilde{Q}_n^2(i_n, j)$;
    Update $\tilde{Q}_{n+1}^2(i, j) = (1 - \alpha_n)\tilde{Q}_n^2(i, j) + \alpha_n(F_n(\tilde{Q}_n^1, v))(i, j)$;
  end
  $n \leftarrow n + 1$;
end

Chapter 4

Computational experiments

The present chapter comprises the methodologies, data, and results corresponding to the developed computational experiments. In the first place, an overall description of the instances used for evaluating the proposal is presented. References of the data used are included for the sake of reproducibility of the experiments. Then, the tested methods used for comparison are introduced, and the evaluation methodology is described. The specifications of the measured indicators and the formulation of the used Q–learning cost functions are also incorporated. Lastly, the two tested cases are presented, and the specific parameters configuration, measured indicators, and performance comparison analyses are shown.

4.1 Preliminaries

Numerical experiments were performed to test the performance of the proposed scheme against previously reported approaches. For the sake of reproducibility, the instances and the data utilized are the same as the ones used in [20], namely an illustrative 4–bus system and the IEEE 118–bus system. Additionally, wind generators were considered for the inclusion of the uncertainty, using the data of the Global Energy Competition [2].

The study compares the performance of the closed-loop scheme using the robust approach for the UC problem, in the future referred to as ARUC, against the open-loop version of the same model using fixed values of robustness, from now on referred to as FRUC, and which has previously used in [20, 23]. The value of the robustness level used in the FRUC method was defined as $\lambda = 1$, as in [20]. This calculation was made by exhaustive enumeration of λ values from 0 to 1, using a step of 0.05. The value of $\lambda = 1$ results in the best performance in the indicators described in the next section. For further evaluation, we also include results for a two–stage stochastic model and a model considering CVAR for the second–stage cost measure, which will be referred to as STO and CVAR, respectively. All methods were tested according to the procedure described in section 3.5.

The two–stage STO model, considering equiprobable scenarios, can be formulated as follows:

$$\min_{\boldsymbol{x} \in \mathcal{X}_{3bin}} \sum_{h \in \mathcal{H}} (\boldsymbol{c}^{fix})^T \boldsymbol{x}_h + \frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} \tilde{z}_k(\boldsymbol{p}_h^k, \boldsymbol{sp}_h^k, \boldsymbol{sm}_h^k) \tag{4.1}$$

s.t.

$$\text{Constraints (3.8)-(3.17)} \tag{4.2}$$

On the other hand, according to [149], the conditional value–at–risk is defined as $\text{CVaR}_\lambda(z) = \min_\eta \left(\eta + \frac{1}{1-\lambda}\mathbb{E}([z-\eta]_+)\right)$, where the function $[a]_+ = \max\{0, a\}$. Then, considering discrete and

equiprobable scenarios, the CVAR formulation applied to the problem (3.6)-(3.17) is equivalent to:

$$\min_{\boldsymbol{X}\in\mathcal{X}_{3bin}} \sum_{h\in\mathcal{H}} (\boldsymbol{c}^{fix})^T \boldsymbol{x}_h + \eta + \frac{1}{|\mathcal{K}|(1-\lambda)} \sum_{k\in\mathcal{K}} \psi_k \tag{4.3}$$

s.t.

$$\tilde{z}_k(\boldsymbol{p}_h^k, \boldsymbol{sp}_h^k, \boldsymbol{sm}_h^k) - \eta - \psi_k \leq 0 \quad \forall k \in \mathcal{K} \tag{4.4}$$

$$\eta \geq 0 \tag{4.5}$$

$$\text{Constraints (3.8)-(3.17)} \tag{4.6}$$

where the value of λ was selected as 0.05, capturing the average of the 5% of worst cases.

Concerning to the number of scenarios considered for the UC resolution, $|\mathcal{K}| = 30$ was selected for every method, derived from the numerical studies made in [20]. In this way, daily profiles of the latest 30 days for every REG source and load are synthesized. Finally, different REG penetration levels were analyzed for the comparison of the FRUC and ARUC methods, where the original data of wind availability was consistently scaled.

4.2 Evaluation methodology

For the use of the experimentation framework described in section 3.5, Δ was selected as 15 minutes, and $N\Delta$ as one hour, resulting in $t = n$. Therefore, following Fig. 3.9, at every simulation period t the UC model (3.6)–(3.17) is solved, and the solution of the hour 2, $\boldsymbol{x}_2^\star$, is implemented in the period $t+1$. Thereafter, every 15 minutes of simulation time, the OPF model is solved using data of the past 15 minutes, defining generators' dispatch levels. Finally, 15 minutes later, exact power flow equations (3.33)-(3.39) are solved using out–of–sample data. Hence for every UC solution, four evaluations are made, where data with 15–minute granularity was generated from the original dataset using cubic splines.

Respecting the double Q–learning algorithm, two cost functions $g(i_t, j_t, \boldsymbol{\xi}_t)$ were selected. The first one corresponds to the differences between the real generation cost and the planning generation cost, $g(i_t, j_t, \boldsymbol{\xi}_t) = (\mathbf{c}^g)^T(\mathbf{p}^{Real} - \mathbf{p}^{k^*})$, where $\boldsymbol{p}^{Real}$ is the vector of real generated power, obtained after the corrections made by droop–frequency characteristic, and $\mathbf{p}^{k^*}$ is the vector of the dispatches levels on the scenario k^*, with $k^* = \text{argmax}_k\{z(\boldsymbol{X}, \boldsymbol{U}^k)\}_{k\in\mathcal{K}}$. The second cost function considered was $g(i_t, j_t, \boldsymbol{\xi}_t) = \sum_{i=1}^4 ||\boldsymbol{x}^* - \boldsymbol{x}_i^{eval}||$, where $\boldsymbol{x}^*$ is the optimal commitment solution calculated by the planning model, at the corresponding hour, and $\boldsymbol{x}_i^{eval}$ is the optimal commitment solution of a single–period UC problem with a second order cone relaxation on the power flow manifold, using the data of the *i–th* quarter of the hour t, i.e., with the out–of–sample data used in the evaluation. Specifically, the model used for the calculation of the second learning cost function is presented below:

$$\min \quad \sum_{i\in\mathcal{G}} (c_i^{NL} x_i + c_i^{SU} v_i + c_i^{SD} w_i + c_i^g p_i^g) + \sum_{b\in\mathcal{B}} \left(c^+(sp_b^p + sp_b^q) + c^-(sm_b^p + sm_b^q) \right) \tag{4.7}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{G}_i} p_j^g + \sum_{j\in\mathcal{R}_i} w_j - p_i^d + sm_i^p - sp_i^p = G_{ii}c_{ii} + \sum_{j\in\delta(i)} (G_{ij}c_{ij} - B_{ij}s_{ij}) \quad \forall i \in \mathcal{B} \tag{4.8}$$

$$\sum_{j\in\mathcal{G}_i} q_j^g - q_i^d + sm_i^q - sp_i^q = -B_{ii}c_{ii} + \sum_{j\in\delta(i)} (-B_{ij}c_{ij} - G_{ij}s_{ij}) \quad \forall i \in \mathcal{B} \tag{4.9}$$

$$(-G_{ij}c_{ii} + G_{ij}c_{ij} - B_{ij}s_{ij})^2 + (B_{ij}c_{ii} - B_{ij}c_{ij} - G_{ij}s_{ij})^2 \leq (f_{ij}^{max})^2 \quad \forall (i,j) \in \mathcal{L} \tag{4.10}$$

$$\underline{V}_i^2 \leq c_{ii} \leq \bar{V}_i^2 \quad \forall i \in \mathcal{B} \tag{4.11}$$

$$c_{ij} = c_{ji},\ s_{ij} = -s_{ji} \quad \forall (i,j) \in \mathcal{L} \tag{4.12}$$

$$c_{ij}^2 + c_{ij}^2 \leq c_{ii}c_{jj} \quad \forall (i,j) \in \mathcal{L} \tag{4.13}$$

$$p_i^{min} u_i \leq p_i^g \leq p_i^{max} u_i \quad \forall i \in \mathcal{G} \tag{4.14}$$

$$q_i^{min} u_i \leq q_i^g \leq q_i^{max} u_i \quad \forall i \in \mathcal{G} \tag{4.15}$$

$$(2.6) - (2.8),\ (2.12) - (2.13) \tag{4.16}$$

where the binary variables u_i, v_i, w_i are defined according to the *3–bin* model (section 2.2.2), as well the fixed costs c_i^{NL}, c_i^{SU}, c_i^{SD}, and the variables $c_{ii} = e_i^2 + f_i^2$, $c_{ij} = e_i e_j + f_i f_j$, $s_{ij} = e_i f_j - e_j f_i$, with e_i, f_i, according to the definition of the OPF model (3.23)-(3.29).

After this, these ARUC methods will be mentioned as ARUC1, and ARUC2, respectively, according to the cost functions used. The idea behind evaluating two different cost functions is to appreciate the effects of two different criteria of qualifying a solution as good. The first function considers that the better out-of-sample solution is reached if planning and real dispatch costs are equal. The second function tries to capture in $\boldsymbol{x}_i^{eval}$ the commitment solution that achieves null values for voltage violations and use of reserves while minimizing generation costs, assigning the distance between the actual commitment solution and $\boldsymbol{x}_i^{eval}$ as a measure of cost.

Hence, system indicators are obtained from power flow equations solutions for every benchmarked method. In our case, voltage violations, use of reserves, and generation costs are selected, from now on referred to as VVIOL, RES, and GCOST, respectively. As four values for each indicator are calculated per hour, hourly values are defined as the four abovementioned averages. It is important to highlight that our approach differs from the usual methodologies to account system violations. Classical approaches obtain out-of-sample values of non-served demand and renewable power spillage from solutions of OPF models that use the calculated commitment solutions and out-of-sample data. Since we are using exact power flow equations with extended characteristics for generators and loads (3.31) - (3.32), the system contains enough flexibility to deal with power imbalances, as is made in the primary frequency control, and demand response control schemes [152]. In this sense, VVIOL is considered an indicator of system violations, calculated as the sum of the absolute deviations from the minimum and maximum bus voltage limits, defined as 0.9 and 1.1 p.u., respectively. On the other hand, RES is calculated as the ratio of the sum of the absolute deviations of the generation levels from its nominal setpoints and the total generated power. Absolute deviations are obtained from exact power flow equations. Finally, GCOST is calculated as the sum of the fixed costs $(\boldsymbol{c}^{fix})^T \boldsymbol{x}^*$, and the dispatch costs resultant from the generated active power on the load flow equations (3.31). In this sense, the performance of methods can be compared in terms of the tradeoff between generation costs and voltage violations, which account for the typical comparison made between costs and robustness.

Finally, MIP (UC), SOCP (OPF), and MISOCP (second learning cost function) models were solved using Gurobi 9.0.2, whereas non–linear load flow system equations were solved using Knitro 12.0.0. Models were written on AMPL, and the double Q–learning algorithm was programmed on Python 3.6.2, on an Intel Core i7–10870H processor at 2.2 GHz and 16 GB of RAM.

4.3 Tests systems

4.3.1 Illustrative 4–bus System

Firstly, a 4–bus system, depicted in Fig. 4.1, was used for the application of the methods, in order to evidence and check the concept's effectiveness. The system is composed of 14 thermal generators, 4 buses, 4 lines, and 2 wind generators. Wind generation data were extracted from buses 2 and 5 from the Global Energy Competition data [2], and the specification of system parameters can be found in the references of [20]. All the UC models tested were solved monolithically due to the size of the system, using a MIP–gap of 10^{-6}. Fig. 4.1 shows the single line diagram of the system, and the maximum values of generation and loads values at each bus, for the case of 7% of REG penetration, while Fig. 4.2 shows the wind profiles for buses 1 and 4, for the same case of REG penetration, which were set as the values of buses 2 and 5 of [2], respectively.

Concerning to the double Q–learning parameters, α_t was selected as $\alpha_t = \frac{1}{t+1}$, and for the ϵ–greedy selection policy, $\epsilon_{t+1} = k\epsilon_t$, with $k = 0.99$, $\epsilon_0 = 1$, and a lower limit $\epsilon_{min} = 0.01$. For both ARUC methods, the domain of robustness levels were selected as $\Lambda = \{0, 0.05, 0.1, ..., 0.95, 1.0\}$. The evaluation was made for 210 days (5040 hrs.) of out–of–sample data for wind generation and active/reactive power loads, capturing daily average values.

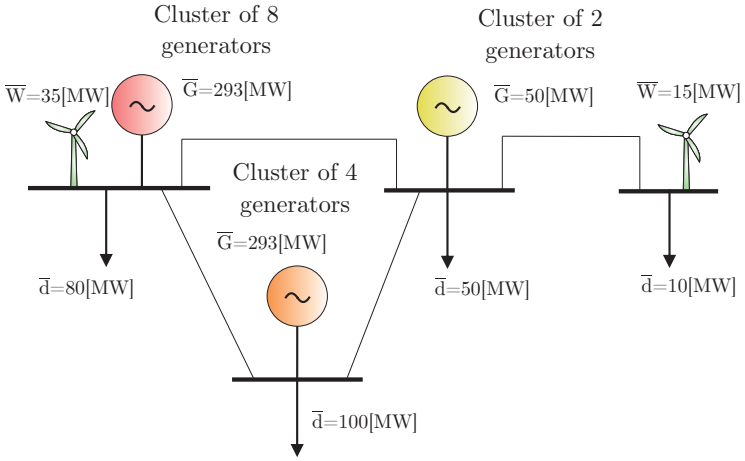


Figure 4.1: 4–bus test system single line diagram. The values shown for generators and loads represent maximum values.

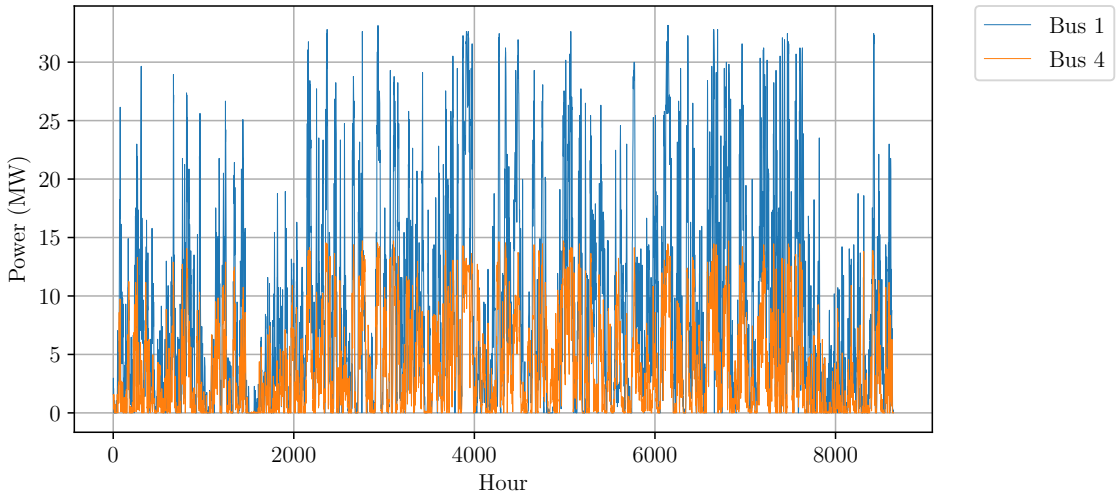


Figure 4.2: Wind profiles used in the 4–bus system. Profiles of buses 1 and 4 were set as the profiles of buses 2 and 5 of [2], respectively.

Firstly, the evolution of daily operational indicators are displayed for every method in Fig. 4.3. Similarly, for illustrative purposes, generators schedules for 168 consecutive hours (one week) are depicted in Fig. 4.4. Differences between ARUC2 and FRUC methods could be observed.

To obtain a more conclusive constrast, a comparison of daily average values between STO, CVAR, FROB, ARUC1 and ARUC2 methods was made for the case with 7% of REG penetration, where the results are shown in Table 4.1. As can be seen, the proposed closed–loop versions of the robust method achieve lower daily average values for every analyzed index, considering the two above–mentioned cost functions for the learning algorithm, overmatching the others analyzed methods. Since the similarity of the values obtained, statistical hypothesis tests were driven for the differences of each pair of method's samples, specifically the t–Student test and the Wilcoxon signed-rank test, both rejecting the null hypothesis of equal means and equal medians, respectively. Besides, standard deviation values of daily indicators are presented in Table 4.2. Results show that the dispersion on voltage violations is also lower for both ARUC methods, whereas dispersion of daily generation costs is lower for the FRUC method. However, it can be checked that the robust closed–loop methods achieve the lowest values for hourly values.

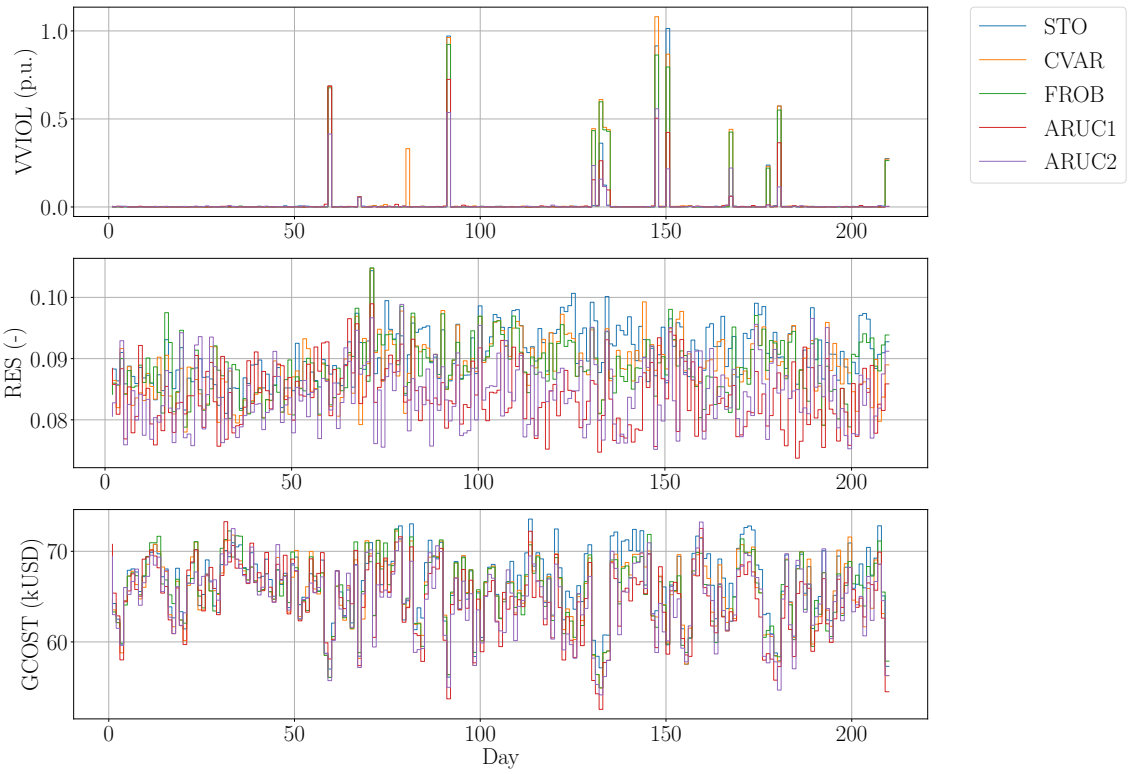


Figure 4.3: Daily operational indicators for 210 days of out–of–sample tests, for every analyzed method.

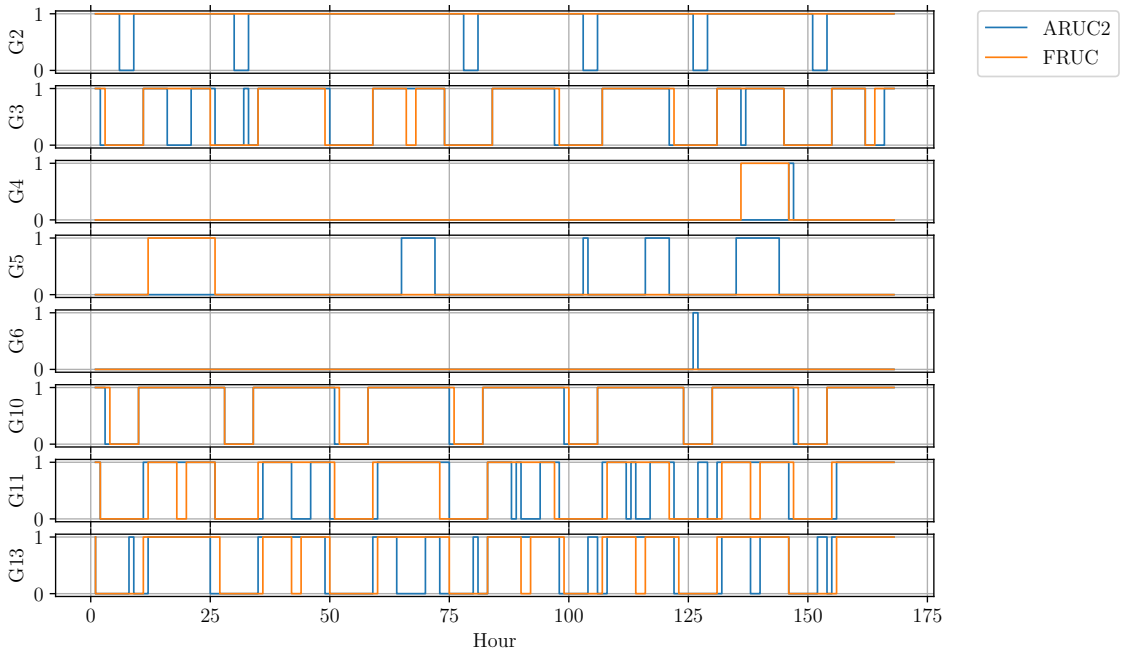


Figure 4.4: Differences on generators schedules for a week between ARUC2 and FRUC methods. Not plotted schedules were identical.

Metric	STO	CVAR	FRUC	ARUC1	ARUC2
VVIOL (p.u.)	0.027	0.037	0.032	0.018	0.014
RES (-)	0.091	0.088	0.088	0.084	0.084
GCOST (kUSD)	66.47	65.59	65.61	64.66	64.73

Table 4.1: Daily–average values of system indicators for the analyzed solution methods for the 4–bus case with 7% of renewable generation penetration.

Std. Deviation		STO	CVAR	FRUC	ARUC1	ARUC2
Daily	VVIOL (p.u.)	0.133	0.151	0.138	0.088	0.067
	GCOST (kUSD)	3.76	3.75	3.72	3.96	3.89
Hourly	VVIOL (p.u.)	0.014	0.017	0.015	0.009	0.008
	GCOST (kUSD)	0.824	0.825	0.824	0.809	0.814

Table 4.2: Standard deviation of demand violations and generation costs for the analyzed methods on the 4–bus case with 7% of REG penetration.

Also, in order to compare the trade–off between cost and robustness for each method, a Pareto front is constructed in Fig. 4.5 for the case with 7% of penetration. Daily average generation costs and daily average voltage violations for each method are plotted. Results show the effectiveness of both closed–loop versions of the robust method, which dominate, in a Pareto sense, every other method, by improving in average both daily voltage violations and generation costs.

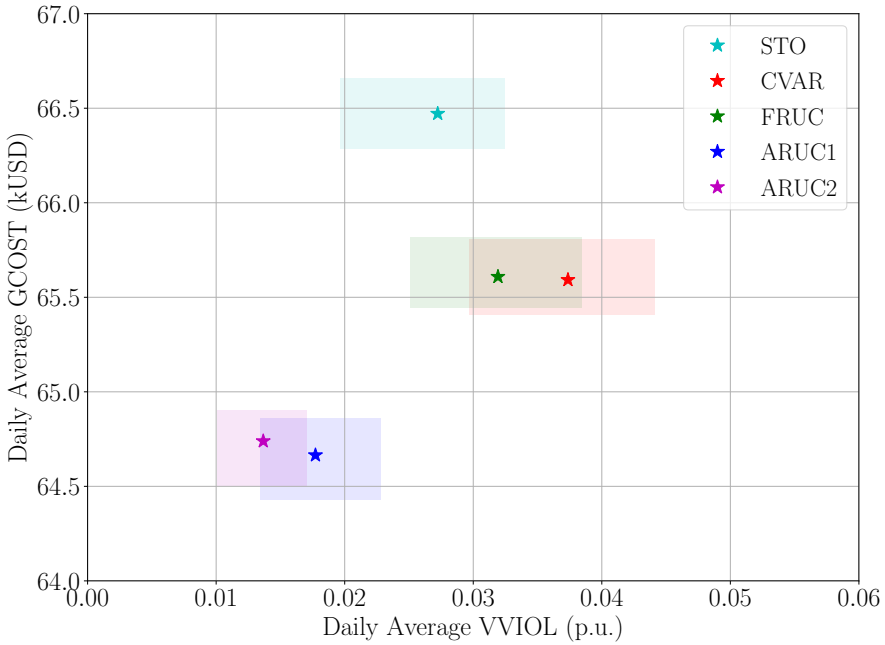


Figure 4.5: Pareto front of daily averages for metrics VVIOL and GCOST. Average values are depicted with an star, and the boxes surrounding them have edge lengths equal to their standard deviation.

Conductive to the comparison of the differences between the open and closed–loop versions for the robust method, Table 4.3 shows the results of VVIOL, RES and GCOST for three levels of REG penetration studied. The percentage of improvement of ARUC1 and ARUC2 over FRUC are denoted by Δ_{FA1} and Δ_{FA2}, respectively. It can be seen ARUC schemes improve the indices at all

penetration levels. It is noted larger improvements in GCOST are observed for higher penetartion levels. For the case of 25% of penetration, ARUC1 and ARUC2 methods reach a 2.81% and a 6.02% of improvement in generation costs, which are known to be significant in the context of the UC [154]. Additionally, results on the use of reserves exhibit the capacity of ARUC methods of selecting more accurately the adequate set–points of generators, which produces lower corrections and lower use of reserves.

REG level	**Model**	**VVIOL (p.u.)**	**RES (-)**	**GCOST (kUSD)**
7%	FRUC	0.0319	0.0884	65.61
	ARUC1	0.0177	0.0842	64.66
	ARUC2	0.0137	0.0844	64.73
	$\Delta_{\text{FA1}}\%$	44.51	4.75	1.44
	$\Delta_{\text{FA2}}\%$	57.05	4.52	1.33
14%	FRUC	0.5638	0.0887	61.86
	ARUC1	0.4718	0.0853	61.02
	ARUC2	0.4039	0.0841	60.60
	$\Delta_{\text{FA1}}\%$	16.32	3.83	1.36
	$\Delta_{\text{FA2}}\%$	28.36	5.19	2.03
25%	FRUC	2.1558	0.0913	57.90
	ARUC1	1.8197	0.0883	56.27
	ARUC2	1.5528	0.0843	54.41
	$\Delta_{\text{FA1}}\%$	15.59	3.26	2.81
	$\Delta_{\text{FA2}}\%$	27.97	7.67	6.02

Table 4.3: Comparison of daily average results between open–loop and closed–loop methods for different values of renewable generation penetration for the 4–bus case.

Histograms of differences are drawn in Fig. 4.6 and 4.7 to graphically appreciate the differences between ARUC and FRUC methods. Those are plotted for every analyzed metric and the three levels of REG penetration. The differences of FRUC from ARUC1 and ARUC2 models are denoted by $|\Delta_{\text{FA1}}|$ and $|\Delta_{\text{FA2}}|$, respectively. As can be seen, on reserves use, and generation costs differences, a high mass of samples are right-shifted from zero. Moreover, on generation costs, the distribution shape is conserved and increasingly shifted to the right as the penetration level grows. On the other hand, voltage violations are mostly zero, showing that system flexibility is enough to supply imbalances due to the discrepancies between planning and real operation.

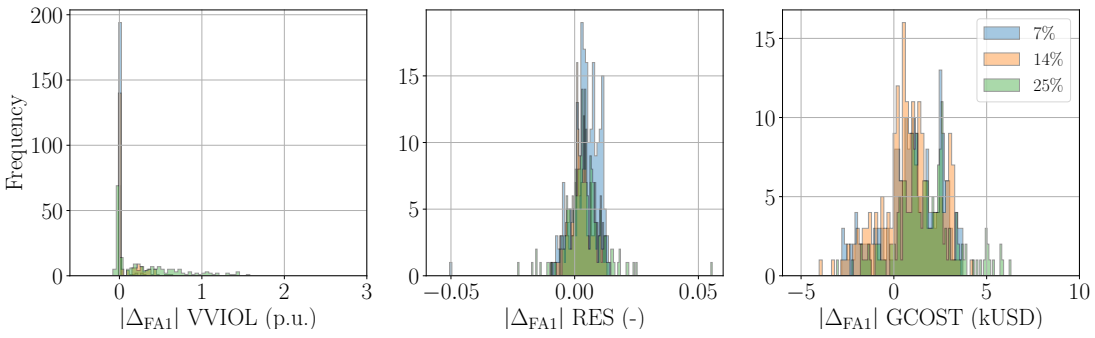


Figure 4.6: Histograms of the differences between FRUC and ARUC1 methods of VVIOL, RES and GCOST indexes, for the 4–bus system. Different REG penetration level are analyzed, where $|\Delta_{\text{FA1}}|I$ represents the difference on index I between FRUC and ARUC1 methods.

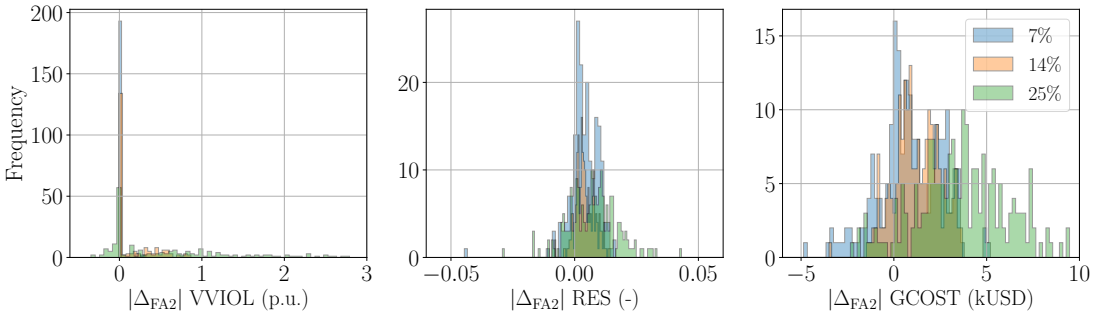


Figure 4.7: Histograms of the differences between FRUC and ARUC2 methods of VVIOL, RES and GCOST indexes, for the 4–bus system. Different REG penetration level are analyzed, where $|\Delta_{\text{FA2}}|I$ represents the difference on index I between FRUC and ARUC2 methods.

4.3.2 IEEE 118–bus system

The second system under analysis corresponds to a modified version of the IEEE 188–bus instance, showed in Fig. 4.8, which includes 10 wind farms, the same as was made in [20], where wind profiles are illustrated in 4.9. For this study case, only the robust case was tested, principally due to the poor computational performance of stochastic and CVaR–based models.

Two–stage robust models were solved using the C&CG algorithm, with a MIP gap of 0.005% for the master problem, and the subproblems solved to optimality. The methods were tested for 120 days of simulation (2880 hrs.). Finally, tests were run using real–time data, extracted from [2], whereas network data can be found in [20]. Respecting the settings to the double Q–learning algorithm, learning parameters, and sets were defined as the same as the 4–bus case.

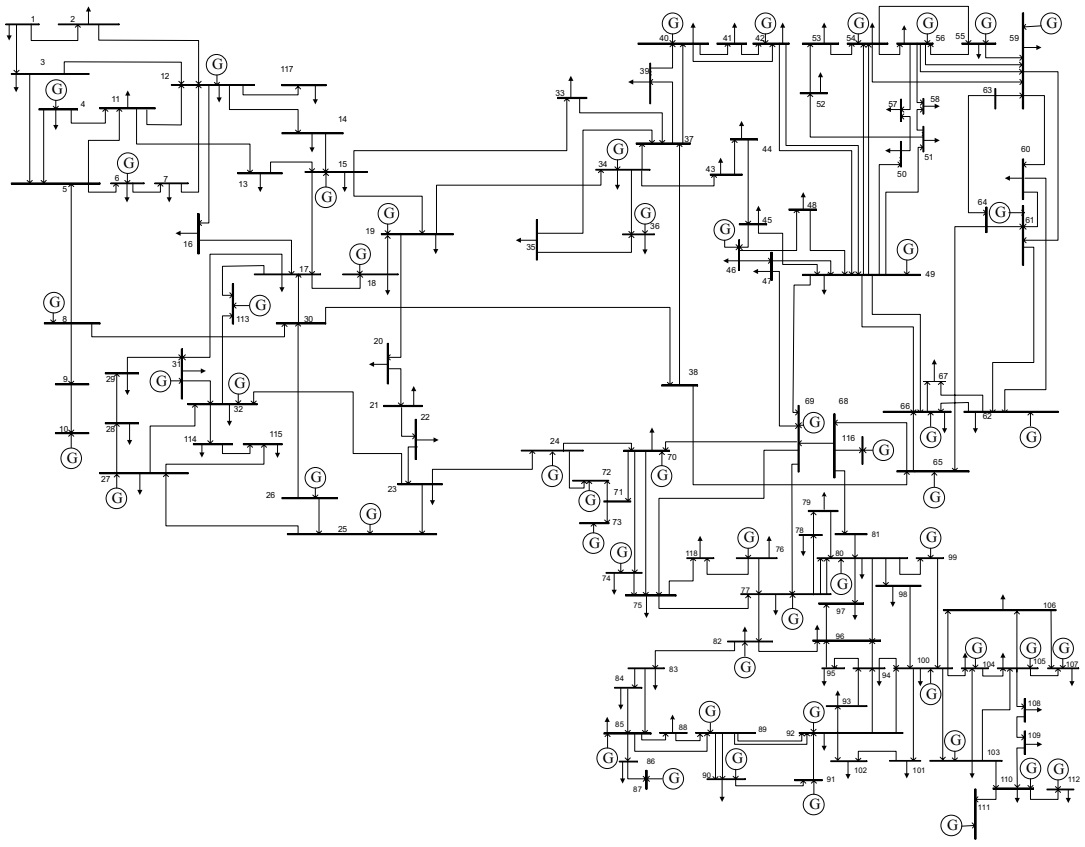


Figure 4.8: IEEE 118–bus system, composed by 118 buses, 186 lines, 91 loads and 54 generators.

Daily–average values of the operational metrics are presented in Table 4.4, for REG penetration levels of 14 and 25%. As can be seen, VVIOL is very small which means no voltage violations are encountered. In contrast with 4–bus case, the 118–bus instances possess much more conventional generators, with each one having a power–frequency droop characteristic. This results in such flexibility that allows to completely compensate power discrepancies between planned and real renewable generation values. On the other hand, significant improvements observed on the GCOST metric are noticed, which increases with REG penetration level. Similar to the 4–bus system case study, the use of reserves is also reduced by the ARUC models, showing the better capability of the proposal on predicting real renewable generation values.

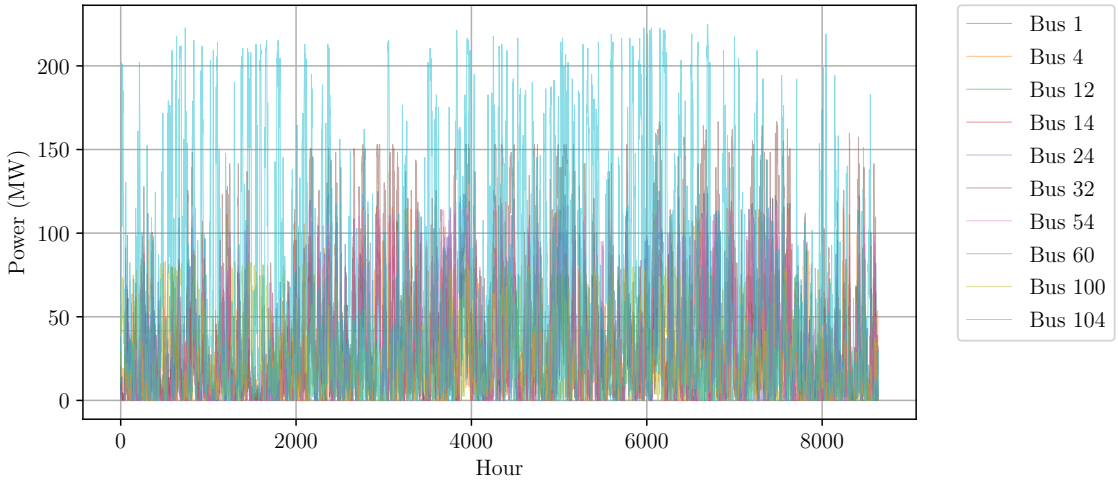


Figure 4.9: Wind profiles of the 10 wind generators included in the IEEE 118–bus system, considering a 7% of REG penetration.

REG level	Method	VVIOL (p.u.)	RES (-)	GCOST (USD)
14%	FRUC	$\sim 10^{-5}$	0.0362	1129.62
	ARUC1	$\sim 10^{-5}$	0.0355	1124.81
	ARUC2	$\sim 10^{-5}$	0.0356	1124.74
	$\Delta_{\text{FA1}}\%$	~ 0	1.93	0.43
	$\Delta_{\text{FA2}}\%$	~ 0	1.66	0.43
25%	FRUC	$\sim 10^{-5}$	0.0327	985.88
	ARUC1	$\sim 10^{-5}$	0.0316	971.73
	ARUC2	$\sim 10^{-5}$	0.0319	958.43
	$\Delta_{\text{FA1}}\%$	~ 0	3.63	1.43
	$\Delta_{\text{FA2}}\%$	~ 0	2.45	2.78

Table 4.4: Comparison of daily average results between open–loop and closed–loop methods for different values of renewable generation penetration for the IEEE 118–bus case.

It is essential to highlight that, by definition, the ARUC1 model adds no computational complexity to the solution algorithm since the calculation of the double Q-learning cost function is read from the Q-table directly. On the other hand, the ARUC2 model requires solving a mixed-integer SOCP (MISOCP) problem for the UC model for one period with deterministic REG and load levels values.

Table 4.5 shows different measures of the computational time needed to solve one-step of every involved model for the case with 25% of REG level penetration. As already discussed and shown in [20], the FRUC model possesses relatively short computation times compared with the maximum

limit that ensures the applicability of the proposed framework with hourly periodicity, i.e., 3600(s). In this sense, the present results confirm the scalability of FRUC under the proposed methodology. At the same time, a step of the ARUC1 model takes, on average, the same time as FRUC, where the rest of the measures keep very similar. Finally, ARUC2 model time measures are slightly greater than the reported on FRUC and ARUC1, with an average increment of 4.10% concerning FRUC, which results negligible in the light of the improvement in operational results. In this sense, both ARUC models keep the order of the FRUC computational time, ensuring the scalability of the proposal.

Measured Time	**FRUC**	**ARUC1**	**ARUC2**
Average (s)	536	536	558
Std. Deviation (s)	567	566	577
Min. (s)	24	11	25
Max. (s)	3352	3342	3362

Table 4.5: Computational times for the 118–bus system with a 25% of REG penetration level.

Another interesting fact is the better performance of model ARUC2 against ARUC1 at high levels of penetration. From both learning cost functions, including information about a more detailed plant's model helps estimate the impact of λ on the out-of-sample solution. That cost function captures the best possible operation of the system, knowing the values of the uncertain data and given a previous system state. Since the detailed UC model is executed for a single period, the approach can be classified as greedy, as it does not measure the effects on the subsequent periods. However, this definition intrinsically pursues a solution that minimizes all the analyzed system metrics.

Chapter 5

Conclusions and future work

The present work has described a novel and comprehensive approach to address the UC problem considering uncertainty management. The proposal is based on the design of a sequential decision–making formulation, representing a real-time operation. Specifically, the decision process from the generator schedule determination to real-time, steady-state network behavior is modeled.

Taking advantage of previous approaches that deal with uncertainty in the UC, the present proposal defines a robustness level variable that controls the conservativeness of the commitment solution. To this end, a scenario-based uncertainty set with a robustness control parameter is proposed, allowing to account for REG variability, controlling the size of the set at the same time while maintaining an efficient computational implementation.

Unlike conventional approaches, this work proposes defining this robustness level as a control variable and determining it as a function of real-time system indicators, constituting a closed–loop framework.

In conjunction with the framework, a simulation model of real–time operation is also proposed. It is composed of a non–linear OPF model that manages the determination of dispatches levels of generators, and a set of modified load flow equations, containing extended characteristics of generators and loads, representing the real–time actuation of fast controllers. Then, the evaluation methodology for the UC solutions uses a non-linear model of the real-time operation itself instead of constructing linear models that accomplish the infeasibility of the operation, as is reported in previous approaches.

The complete problem formulation is then suitable to be solved via optimal control methods. Particularly, this proposal solves it approximately by using the double Q–learning algorithm.

Computational experiments were performed to compare the effectiveness of the proposal against previous approaches. Specifically, a two-stage stochastic model and a two-stage robust model using scenario-based uncertainty sets were benchmarked. The mentioned methods were tested over two different size instances, showing the superior performance of the proposal, both in the robustness and cost of the solution. Besides, as the proposed uncertainty set can be constructed as a convex hull of a set of scenarios, for every value of the robustness level, the proposal can be efficiently solved by using the C&CG algorithm, allowing to be scalable to more complex systems.

Hereby, the principal conclusions and observations of the developed work are the following:

1. The dynamic determination of internal parameters that define the size and robustness of uncertainty sets, based on out-of-sample evaluation data and results, can positively impact robust models' performance. Results demonstrate that, for different hours and system conditions, robustness levels that achieve the lowest out-of-sample costs are generally differ as time evolves and new realizations are observed. Moreover, this thesis demonstrates the effectiveness of the concept by adapting one single parameter that can take few discrete values.

2. The statement of closed–loop frameworks for short-term planning problems can improve the performance of the used models. Results have shown that adapting solutions according to out–of–samples results and the real–world system's behavior can improve operational indexes. In that regard, constructing an appropriate simulation model of real operation is essential since it allows to approximate the impact of planning solutions on the objective indicators.

3. The addition of adaptation schemes for short–term planning problems is computationally cheap for some learning algorithms, as the double Q–learning algorithm. This thesis shows

that this concept applies to different policies for the account of uncertainty, as the robust and CVaR–based two–stage models. In this sense, this concept is generally applicable. The scalability depends only on the availability of efficient algorithms to solve optimization problems with uncertainty management, constituting a crucial element in the adaptation process, analogous to the plant model in the control language.

Finally, in the light of the results and the developed ideas, future research will consider:

1. Explore the possibilities of the present scheme with other uncertainty sets and other probability measures on the second stage costs. Many approaches use parameters that control robustness that are suitable to be dynamically calculated under this methodology. Moreover, the present proposal could be extended and consider the number of considered planning scenarios $|\mathcal{K}|$ as an additional parameter to be adapted. Indeed, any amount of parameters with different interpretations could be adapted under this methodology. Future research could consider a more general framework for adaptation in short–term planning problems.

2. Explore the performance of other algorithms for the resolution of the optimal control problem (3.22). This proposal uses the double Q–learning algorithm due to the ease of implementation and interpretation, more for testing the effectiveness of the concept of the closed–loop adaptation than to take advantage of the effectiveness of the double Q–learning algorithm. Even with simple definitions of states and cost functions, the proposal shows improvement concerning previous approaches. The use of more complex learning algorithms, or other approaches, as the approximation of the cost–to–go function $J(\boldsymbol{s}_t)$, could reach even better results.

3. Explore the possible benefits of applying this concept in other engineering contexts. System structures composed of a short–term planning model and a real–world plant are present in various problems in which the application of this concept could be worthy. As an example, this concept could be implemented in OPF problems with uncertainty management. As the dispatches levels calculation directly impacts real–time system indicators than the calculation of generator statuses, the improvement of the operational indexes could be even more significant.

4. Explore techniques of scenario generation to achieve good performance in out-of-sample evaluation models. One exciting fact observed along the development of the present thesis is that optimal dispatch points in the UC stage are generally non–optimal or even infeasible for other models using more detailed network equations. In this sense, scenario generation processes can produce commitment and dispatch solutions contained in the power flow manifold, and hopefully, optimal. This fact has a strong connection with the relationship between planning and plant models. The two significant differences between them are the difference of data and the difference between the equations. Even with perfect knowledge of future realizations, planning models with approximated equations could generate bad solutions in terms of the plant model. Therefore, an appropriate selection of scenarios could compensate for the models' differences to obtain better out–of–sample results.

Bibliography

[1] Xin Chen, Guannan Qu, Yujie Tang, Steven Low, and Na Li. Reinforcement learning for decision-making and control in power systems: Tutorial, review, and vision, 2021.

[2] T. Hong, P. Pinson, and S. Fan. Global energy forecasting competition 2012. *International Journal of Forecasting*, 30(2):357 – 363, 2014.

[3] G. Morales-Espa na, J. M. Latorre, and A. Ramos. Tight and compact milp formulation for the thermal unit commitment problem. *IEEE Transactions on Power Systems*, 28(4):4897–4908, Nov 2013.

[4] D. Bertsimas, E. Litvinov, X. A. Sun, J. Zhao, and T. Zheng. Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Transactions on Power Systems*, 28(1):52–63, Feb 2013.

[5] Á. Lorca and X. A. Sun. The adaptive robust multi-period alternating current optimal power flow problem. *IEEE Transactions on Power Systems*, PP(99):1–1, 2017.

[6] G. Morales-España, Á. Lorca, and M. M. de Weerdt. Robust unit commitment with dispatchable wind power. *Electric Power Systems Research*, 155(Supplement C):58 – 66, 2018.

[7] Allen J. Wood, Bruce F. Wollenberg, and Gerald B. Sheblé. *Power Generation, Operation, and Control*. John Wiley & Sons, 3 edition, 2012.

[8] A. Keyhani, M. N. Marwali, and M. Dai. *Integration of Green and Renewable Energy in Electric Power Systems*. Wiley, 1 edition, 2010.

[9] K. Pan and Y. Guan. Convex hulls for the unit commitment polytope, 2017.

[10] B. Knueven, J. Ostrowski, and J. Wang. The ramping polytope and cut generation for the unit commitment problem. *INFORMS Journal on Computing*, 30(4):739–749, 2018.

[11] Y. Feng and S. Ryan. Solution sensitivity-based scenario reduction for stochastic unit commitment. *Computational Management Science*, 13(1):29–62, January 2016.

[12] X. Zhu, Z. Yu, and X. Liu. Security constrained unit commitment with extreme wind scenarios. *Journal of Modern Power Systems and Clean Energy*, 8(3):464–472, 2020.

[13] Wim van Ackooij, Welington de Oliveira, and Yongjia Song. Adaptive partition-based level decomposition methods for solving two-stage stochastic programs with fixed recourse. *INFORMS Journal on Computing*, 30(1):57–70, 2018.

[14] J. Zou, S. Ahmed, and X. A. Sun. Multistage stochastic unit commitment using stochastic dual dynamic integer programming. *IEEE Transactions on Power Systems*, 34(3):1814–1823, 2019.

[15] E. Dall'Anese, K. Baker, and T. Summers. Chance-constrained ac optimal power flow for distribution systems with renewables. *IEEE Transactions on Power Systems*, 32(5):3427–3438, 2017.

[16] K. Baker and A. Bernstein. Joint chance constraints in ac optimal power flow: Improving bounds through learning. *IEEE Transactions on Smart Grid*, 10(6):6376–6385, 2019.

[17] K. Sundar, H. Nagarajan, L. Roald, S. Misra, R. Bent, and D. Bienstock. Chance-constrained unit commitment with n-1 security and wind uncertainty. *IEEE Transactions on Control of Network Systems*, 6(3):1062–1074, 2019.

[18] Bo Zeng and Long Zhao. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters*, 41(5):457 – 461, 2013.

[19] D. Bertsimas, V. Gupta, and N. Kallus. Data-driven robust optimization. *Mathematical Programming*, 167:235 – 292, 2018.

[20] A. Velloso, A. Street, D. Pozo, J. M. Arroyo, and N. G. Cobos. Two-stage robust unit commitment for co-optimized electricity markets: An adaptive data-driven approach for scenario-based uncertainty sets. *IEEE Transactions on Sustainable Energy*, 11(2):958–969, 2020.

[21] N. Kazemzadeh, S. Ryan, and M. Hamzeei. Robust optimization vs. stochastic programming incorporating risk measures for unit commitment with uncertain variable renewable generation. *Energy Systems*, 10:517 – 541, 2019.

[22] C. Ning and F. You. Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming. *Computers and Chemical Engineering*, 125:434 – 448, 2019.

[23] F. Mancilla-David, A. Angulo, and A. Street. Power management in active distribution systems penetrated by photovoltaic inverters: A data-driven robust approach. *IEEE Transactions on Smart Grid*, 11(3):2271–2280, 2020.

[24] G. B. Sheble and G. N. Fahd. Unit commitment literature synopsis. *IEEE Transactions on Power Systems*, 9(1):128–135, Feb 1994.

[25] N. P. Padhy. Unit commitment-a bibliographical survey. *IEEE Transactions on Power Systems*, 19(2):1196–1205, 2004.

[26] W. L. Snyder, H. D. Powell, and J. C. Rayburn. Dynamic programming approach to unit commitment. *IEEE Transactions on Power Systems*, 2(2):339–348, May 1987.

[27] D. P. de Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations Research*, 51(6):850–865, 2003.

[28] X. Guan, P.B. Luh, H. Yan, and J.A. Amalfi. An optimization-based method for unit commitment. *International Journal of Electrical Power & Energy Systems*, 14(1):9–17, feb 1992.

[29] Wei Fan, Xiaohong Guan, and Qiaozhu Zhai. A new method for unit commitment with ramping constraints. *Electric Power Systems Research*, 62(3):215 – 224, 2002.

[30] Antonio Frangioni and Claudio Gentile. Solving nonlinear single-unit commitment problems with ramping constraints. *Operations Research*, 54(4):767–775, 2006.

[31] Antonio Frangioni, Claudio Gentile, and Fabrizio Lacalandra. Solving unit commitment problems with general ramp constraints. *International Journal of Electrical Power & Energy Systems*, 30(5):316–326, jun 2008.

[32] E. C. Finardi and E. L. da Silva. Solving the hydro unit commitment problem via dual decomposition and sequential quadratic programming. *IEEE Transactions on Power Systems*, 21(2):835–844, May 2006.

[33] Erlon Cristian Finardi and Murilo Reolon Scuzziato. A comparative analysis of different dual problems in the lagrangian relaxation context for solving the hydro unit commitment problem. *Electric Power Systems Research*, 107:221 – 229, 2014.

[34] Lei Wu and Mohammad Shahidehpour. Accelerating the benders decomposition for network-constrained unit commitment problems. *Energy Systems*, 1:339 – 376, 2010.

[35] C. Liu, M. Shahidehpour, and L. Wu. Extended benders decomposition for two-stage scuc. *IEEE Transactions on Power Systems*, 25(2):1192–1194, 2010.

[36] L. Wu. An improved decomposition framework for accelerating lsf and bd based methods for network-constrained uc problems. *IEEE Transactions on Power Systems*, 28(4):3977–3986, 2013.

[37] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis. A genetic algorithm solution to the unit commitment problem. *IEEE Transactions on Power Systems*, 11(1):83–92, Feb 1996.

[38] Gwo-Ching Liao. Application meta-heuristics method for short-term unit commitment problem. In *IEEE PES Power Systems Conference and Exposition, 2004.*, pages 413–418 vol.1, Oct 2004.

[39] T. Senjyu, K. Shimabukuro, K. Uezato, and T. Funabashi. A fast technique for unit commitment problem by extended priority list. *IEEE Transactions on Power Systems*, 18(2):882–888, May 2003.

[40] S. H. Hosseini, A. Khodaei, and F. Aminifar. A novel straightforward unit commitment method for large-scale power systems. *IEEE Transactions on Power Systems*, 22(4):2134–2143, Nov 2007.

[41] Arne Løkketangen and David L. Woodruff. Progressive hedging and tabu search applied to mixed integer (0,1) multistage stochastic programming. *Journal of Heuristics*, 2:111–128, 1996.

[42] I.A.Farhat and M.E.El-Hawary. Optimization methods applied for solving the short-term hydrothermal coordination problem. *Electric Power Systems Research*, 79(9):1308 – 1320, 2009.

[43] Anupam Trivedi, D. Srinivasan, Subhodip Biswas, and Thomas Reindl. Hybridizing genetic algorithm with differential evolution for solving the unit commitment scheduling problem. *Swarm and Evolutionary Computation*, 05 2015.

[44] L. L. Garver. Power generation scheduling by integer programming-development of theory. *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems*, 81(3):730–734, 1962.

[45] J. A. Muckstadt and R. C. Wilson. An application of mixed-integer programming duality to scheduling thermal generating systems. *IEEE Transactions on Power Apparatus and Systems*, PAS-87(12):1968–1978, Dec 1968.

[46] F. N. Lee. A fuel-constrained unit commitment method. *IEEE Transactions on Power Systems*, 4(3):1208–1218, Aug 1989.

[47] K. Hara, M. Kimura, and N. Honda. A method for planning economic unit commitment and maintenance of thermal power systems. *IEEE Transactions on Power Apparatus and Systems*, PAS-85(5):427–436, May 1966.

[48] A. Borghetti, C. D'Ambrosio, A. Lodi, and S. Martello. An milp approach for short-term hydro scheduling and unit commitment with head-dependent reservoir. *IEEE Transactions on Power Systems*, 23(3):1115–1124, Aug 2008.

[49] J. M. Arroyo and A. J. Conejo. Modeling of start-up and shut-down power trajectories of thermal units. *IEEE Transactions on Power Systems*, 19(3):1562–1568, Aug 2004.

[50] G. Morales-España, J. M. Latorre, and A. Ramos. Tight and compact milp formulation of start-up and shut-down ramping in unit commitment. In *2013 IEEE Power Energy Society General Meeting*, pages 1–1, July 2013.

[51] George L. Nemhauser and Laurence A. Wolsey. *Integer and Combinatorial Optimization*. Wiley-Interscience, USA, 1988.

[52] Jon Lee, Janny Leung, and François Margot. Min-up/min-down polytopes. *Discrete Optimization*, 1(1):77–85, jun 2004.

[53] Deepak Rajan and Samer Takriti. Minimum up/down polytopes of the unit commitment problem with start-up costs. 2005.

[54] M. Carrion and J. M. Arroyo. A computationally efficient mixed–integer linear formulation for the thermal unit commitment problem. *IEEE Trans. Power Systems*, 21(3):1371–1378, Aug 2006.

[55] J. Ostrowski, M. F. Anjos, and A. Vannelli. Tight mixed integer linear programming formulations for the unit commitment problem. *IEEE Transactions on Power Systems*, 27(1):39–46, Feb 2012.

[56] Pascale Bendotti, Pierre Fouilhoux, and Rottner. The min-up/min-down unit commitment polytope. *Journal of Combinatorial Optimization*, 36(3):1024–1058, 2018.

[57] Pascale Bendotti, Pierre Fouilhoux, and Rottner. On the complexity of the unit commitment problem. *Annals of Operations Research*, 274:119–130, 2019.

[58] Ben Knueven, Jim Ostrowski, and Jianhui Wang. The ramping polytope and cut generation for the unit commitment problem. 2017.

[59] Kai Pan and Yongpei Guan. Convex Hulls for the Unit Commitment Polytope. jan 2017.

[60] Kai Pan and Yongpei Guan. A polyhedral study of the integrated minimum-up/-down time and ramping polytope, 2016.

[61] A. Frangioni, C. Gentile, and F. Lacalandra. Tighter approximated milp formulations for unit commitment problems. *IEEE Transactions on Power Systems*, 24(1):105–113, Feb 2009.

[62] Aidan Tuohy, Eleanor Denny, and Mark O'Malley. Rolling unit commitment for systems with significant installed wind capacity. In *2007 IEEE Lausanne Power Tech*, pages 1380–1385, 2007.

[63] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Ljung Greta M. *Time Series Analysis: Forecasting and Control, 5th Edition.* Wiley Series in Probability and Statistics. Wiley, June 2015.

[64] Suresh Sethi and Gerhard Sorger. A theory of rolling horizon decision making. *Annals of Operations Research*, 29:387–415, 1991.

[65] Mitch Costley, Mohammad Javad Feizollahi, Shabbir Ahmed, and Santiago Grijalva. A rolling-horizon unit commitment framework with flexible periodicity. *International Journal of Electrical Power & Energy Systems*, 90:280–291, 2017.

[66] Min Zhou, Bo Wang, and Junzo Watada. Deep learning-based rolling horizon unit commitment under hybrid uncertainties. *Energy*, 186:115843, 2019.

[67] Warren B. Powell. *A Unified Framework for Optimization Under Uncertainty*, chapter 3, pages 45–83.

[68] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming.* Society for Industrial and Applied Mathematics, 2009.

[69] Ranjeet Kumar, Michael J. Wenzel, Matthew J. Ellis, Mohammad N. ElBsat, Kirk H. Drees, and Victor M. Zavala. A stochastic dual dynamic programming framework for multiscale mpc. *IFAC-PapersOnLine*, 51(20):493 – 498, 2018. 6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018.

[70] M. V. F. Pereira and L. M. V. G. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(20):359 – 375, 1991.

[71] Rafael Bruno S. Brandi, Tales Pulinho Ramos, Bruno Henriques Dias, André Luís Marques Marcato, and Ivo Chaves da Silva Junior. Improving stochastic dynamic programming on hydrothermal systems through an iterative process. *Electric Power Systems Research*, 123:147 – 153, 2015.

[72] W. van Ackooij and X. Warin. On conditional cuts for stochastic dual dynamic programming. *Mathematical Programming*, 8(20):173 – 199, 2020.

[73] F. Oliveira, V. Gupta, S. Hamacher, and I.E. Grossmann. A lagrangean decomposition approach for oil supply chain investment planning under uncertainty with risk considerations. *Computers & Chemical Engineering*, 50:184 – 195, 2013.

[74] Werner Römisch. Scenario reduction techniques in stochastic programming. In Osamu Watanabe and Thomas Zeugmann, editors, *Stochastic Algorithms: Foundations and Applications*, pages 1–14, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[75] A. Charnes and W. W. Cooper. Chance-constrained programming. *Management Science*, 6(1):73–79, 1959.

[76] A. Prékopa. *Stochastic programming.* Springer, Dordrecht, 1995.

[77] B. K. Pagnoncelli, S. Ahmed, and A. Shapiro. Sample average approximation method for chance constrained programming: Theory and applications. *Journal of Optimization Theory and Applications*, 142(2):399 – 416, 2009.

[78] James Luedtke and Shabbir Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization*, 19(2):674–699, 2008.

[79] Arkadi Nemirovski and Alexander Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2007.

[80] A. Ben-Tal, L. El Ghaoui, and A.S. Nemirovski. *Robust Optimization.* Princeton Series in Applied Mathematics. Princeton University Press, October 2009.

[81] A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operations Research*, 21(5):1154–1157, 1973.

[82] Dimitris Bertsimas and Melvyn Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[83] James E. Smith and Robert L. Winkler. The optimizer's curse: Skepticism and postdecision surprise in decision analysis. *Management Science*, 52(3):311–322, 2006.

[84] Anton J. Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2):479–502, 2002.

[85] Hongcheng Liu, Xue Wang, Tao Yao, Runze Li, and Yinyu Ye. Sample average approximation with sparsity-inducing penalty for high-dimensional stochastic programming. *Mathematical Programming*, 178:108 – 178, 2019.

[86] Pernille Seljom and Asgeir Tomasgard. Sample average approximation and stability tests applied to energy system design. *Energy Systems*, 12:107 – 131, 2021.

[87] Grani A. Hanasusanto, Vladimir Roitch, Daniel Kuhn, and Wolfram Wiesemann. A distributionally robust perspective on uncertainty quantification and chance constrained programming. *Mathematical Programming*, 151:35 – 62, 2015.

[88] Erick Delage and Yinyu Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612, 2010.

[89] Mohajerin Esfahani and Daniel Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: performance guarantees and tractable reformulations. *Mathematical Programming*, 171:115 – 166, 2018.

[90] R. Zhu, H. Wei, and X. Bai. Wasserstein metric based distributionally robust approximate framework for unit commitment. *IEEE Transactions on Power Systems*, 34(4):2991–3001, 2019.

[91] X. Zheng and H. Chen. Data-driven distributionally robust unit commitment with wasserstein metric: Tractable formulation and efficient solution method. *IEEE Transactions on Power Systems*, 35(6):4940–4943, 2020.

[92] Jianqiang Cheng, Céline Gicquel, and Abdel Lisser. Partial sample average approximation method for chance constrained problems. *Optimization Letters*, 13:657 – 672, 2019.

[93] Ashish R. Hota, Ashish Cherukuri, and John Lygeros. Data-driven chance constrained optimization under wasserstein ambiguity sets, 2018.

[94] Zhi Chen, Daniel Kuhn, and Wolfram Wiesemann. Data-driven chance constrained programs over wasserstein balls, 2018.

[95] Yi Zhang, Yiping Feng, and Gang Rong. Data-driven chance constrained and robust optimization under matrix uncertainty. *Industrial & Engineering Chemistry Research*, 55(21):6145–6160, 2016.

[96] Ruiwei Jiang and Yongpei Guan. Data-driven chance constrained stochastic program. *Mathematical Programming*, 158:291 – 327, 2016.

[97] Dimitris Bertsimas and David B. Brown. Constructing uncertainty sets for robust linear optimization. *Operations Research*, 57(6):1483–1495, 2009.

[98] Narges Kazemzadeh, Sarah M. Ryan, and Mahdi Hamzeei. Robust optimization vs. stochastic programming incorporating risk measures for unit commitment with uncertain variable renewable generation. *Energy Systems*, Dec 2017.

[99] Chao Ning and Fengqi You. Data-driven adaptive nested robust optimization: General modeling framework and efficient computational algorithm for decision making under uncertainty. *AIChE Journal*, 63(9):3790–3817, 2017.

[100] Chao Ning and Fengqi You. A data-driven multistage adaptive robust optimization framework for planning and scheduling under uncertainty. *AIChE Journal*, 63(10):4343–4369, 2017.

[101] Chao Ning and Fengqi You. Data-driven decision making under uncertainty integrating robust optimization with principal component analysis and kernel smoothing methods. *Computers & Chemical Engineering*, 112:190 – 210, 2018.

[102] Bo Zeng. Solving two-stage robust optimization problems by a constraint-and-column generation method, 2011.

[103] Tobias Sutter, Bart P. G. Van Parys, and Daniel Kuhn. A general framework for optimal data-driven optimization, 2021.

[104] Dimitri P. Bertsekas. *Reinforcement Learning and Optimal Control*. Athena Scientific, USA, 2019.

[105] George H. Polychronopoulos and John N. Tsitsiklis. Stochastic shortest path problems with recourse. *Networks*, 27(2):133–143, 1996.

[106] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[107] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning*. MIT Press, USA, 2018.

[108] Christopher J. C. H. Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8:279–292, 1992.

[109] Dimitri P. Bertsekas. *Neuro-dynamic programming*, pages 2555–2560. Springer US, Boston, MA, 2009.

[110] H. Hasselt. Double q-learning. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23, pages 2613–2621. Curran Associates, Inc., 2010.

[111] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Shane Wierstra, Daan andLegg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

[112] J. N. Tsitsiklis and B. Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE Transactions on Automatic Control*, 42(5):674–690, 1997.

[113] Z. Zhang, D. Zhang, and R. C. Qiu. Deep reinforcement learning for power system applications: An overview. *CSEE Journal of Power and Energy Systems*, 6(1):213–225, 2020.

[114] Z. Wan, H. Li, and H. He. Residential energy management with deep reinforcement learning. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, 2018.

[115] B. V. Mbuwir, M. Kaffash, and G. Deconinck. Battery scheduling in a residential multi-carrier energy system using reinforcement learning. In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–6, 2018.

[116] Z. Wan, H. Li, H. He, and D. Prokhorov. Model-free real-time ev charging scheduling based on deep reinforcement learning. *IEEE Transactions on Smart Grid*, 10(5):5246–5257, 2019.

[117] Yue Hu, Weimin Li, Kun Xu, Taimoor Zahid, Feiyan Qin, and Chenming Li. Energy management strategy for a hybrid electric vehicle based on deep reinforcement learning. *Applied Sciences*, 8(2), 2018.

[118] V. Francois-Lavet, D. Tarallaand D. Ernst, and R. Fonteneau. Deep reinforcement learning solutions for energy microgrids management. *Proceedings of European Workshop on Reinforcement Learning*, 2016.

[119] P. Kofinas, A.I. Dounis, and G.A. Vouros. Fuzzy q-learning for multi-agent decentralized energy management in microgrids. *Applied Energy*, 219:53–67, 2018.

[120] Haochen Hua, Yuchao Qin, Chuantong Hao, and Junwei Cao. Optimal energy management strategies for energy internet via deep reinforcement learning approach. *Applied Energy*, 239:598–609, 2019.

[121] José R. Vázquez-Canteli and Zoltán Nagy. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied Energy*, 235:1072–1089, 2019.

[122] Pierluigi Siano. Demand response and smart grids—a survey. *Renewable and Sustainable Energy Reviews*, 30:461–478, 2014.

[123] Damien Ernst, Mevludin Glavic, and Louis Wehenkel. Power systems stability control: reinforcement learning framework. *IEEE transactions on power systems*, 19(1):427–435, 2004.

[124] Damien Ernst, Mevludin Glavic, Florin Capitanescu, and Louis Wehenkel. Reinforcement learning versus model predictive control: a comparison on a power system problem. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):517–529, 2008.

[125] Da Wang, Mevludin Glavic, and Louis Wehenkel. Trajectory-based supplementary damping control for power system electromechanical oscillations. *IEEE transactions on power systems*, 29(6):2835–2845, 2014.

[126] Taoridi Ademoye and Ali Feliachi. Reinforcement learning tuned decentralized synergetic control of power systems. *Electric power systems research*, 86:34–40, 2012.

[127] Yinliang Xu, Wei Zhang, Wenxin Liu, and Frank Ferrese. Multiagent-based reinforcement learning for optimal reactive power dispatch. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1742–1751, 2012.

[128] Tao Yu, Bin Zhou, Ka Wing Chan, Liang Chen, and Bo Yang. Stochastic optimal relaxed automatic generation control in non-markov environment based on multi-step $Q(\lambda)$ learning. *IEEE Transactions on Power Systems*, 26(3):1272–1282, 2011.

[129] Tao Yu, B Zhou, KW Chan, Y Yuan, B Yang, and QH Wu. R (λ) imitation learning for automatic generation control of interconnected power grids. *Automatica*, 48(9):2130–2136, 2012.

[130] EA Jasmin, TP Imthias Ahamed, and VP Jagathy Raj. Reinforcement learning approaches to economic dispatch problem. *International Journal of Electrical Power & Energy Systems*, 33(4):836–845, 2011.

[131] T Yu, XS Zhang, B Zhou, and KW Chan. Hierarchical correlated q-learning for multi-layer optimal generation command dispatch. *International Journal of Electrical Power & Energy Systems*, 78:1–12, 2016.

[132] Reza Yousefian and Sukumar Kamalasadan. Design and real-time implementation of optimal power system wide-area system-centric controller based on temporal difference learning. *IEEE Transactions on Industry Applications*, 52(1):395–406, 2015.

[133] Ramtin Hadidi and Benjamin Jeyasurya. Reinforcement learning based real-time wide-area stabilizing control agents to enhance power system stability. *IEEE Transactions on Smart Grid*, 4(1):489–497, 2013.

[134] Chun Wei, Zhe Zhang, Wei Qiao, and Liyan Qu. Reinforcement-learning-based intelligent maximum power point tracking control for wind energy conversion systems. *IEEE Transactions on Industrial Electronics*, 62(10):6360–6370, 2015.

[135] Yufei Tang, Haibo He, Jinyu Wen, and Ju Liu. Power system stability control for a wind farm based on adaptive dynamic programming. *IEEE Transactions on Smart Grid*, 6(1):166–177, 2014.

[136] Daniel Görges. Relations between model predictive control and reinforcement learning. *IFAC-PapersOnLine*, 50(1):4920–4928, 2017.

[137] Dimitri P Bertsekas. Dynamic programming and suboptimal control: A survey from adp to mpc. *European Journal of Control*, 11(4-5):310–334, 2005.

[138] Lukas Beckenbach, Pavel Osinenko, and Stefan Streif. Addressing infinite-horizon optimization in mpc via q-learning. *IFAC-PapersOnLine*, 51(20):60–65, 2018. 6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018.

[139] Sébastien Gros and Mario Zanon. Data-driven economic nmpc using reinforcement learning. *IEEE Transactions on Automatic Control*, 2019.

[140] J. Shin and J. H. Lee. Multi-timescale, multi-period decision-making model development by combining reinforcement learning and mathematical programming. *Computers & Chemical Engineering*, 121:556 – 573, 2019.

[141] X. Zhu, B. Zeng, H. Dong, and J. Liu. An interval-prediction based robust optimization approach for energy-hub operation scheduling considering flexible ramping products. *Energy*, 194:116821, 2020.

[142] F. Li, J. Qin, and Y. Kang. Closed-loop hierarchical operation for optimal unit commitment and dispatch in microgrids: A hybrid system approach. *IEEE Transactions on Power Systems*, 35(1):516–526, 2020.

[143] A. Hauswirth, A. Zanardi, S. Bolognani, F. Dörfler, and G. Hug. Online optimization in closed loop on the power flow manifold. In *2017 IEEE Manchester PowerTech*, pages 1–6, 2017.

[144] S. S. Guggilam, C. Zhao, E. Dall'Anese, Y. C. Chen, and S. V. Dhople. Primary frequency response with aggregated ders. In *2017 American Control Conference (ACC)*, pages 3386–3393, 2017.

[145] E. Dall'Anese and A. Simonetto. Optimal power flow pursuit. *IEEE Transactions on Smart Grid*, 9(2):942–952, 2018.

[146] A. Lorca and A. Sun. Multistage adaptive robust optimization for the unit commitment problem. *Mathematical Optimization Society.*, pages 1–33, Sep 2015.

[147] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory, Second Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2014.

[148] X. Andy Sun and Álvaro Lorca. *Robust Optimization in Electric Power Systems Operations*, pages 227–258. Springer International Publishing, Cham, 2017.

[149] S. Arpón, T. Homem–de–Mello, and B. Pagnoncelli. Scenario reduction for stochastic programs with conditional value-at-risk. *Mathematical Programming*, 170:327–356, 2018.

[150] B. Kocuk, S. S. Dey, and X. A. Sun. Inexactness of sdp relaxation and valid inequalities for optimal power flow. *IEEE Transactions on Power Systems*, 31(1):642–651, 2016.

[151] Electric Power Research Institute. *OpenDSS*. EPRI, 2020.

[152] Farshid Habibi, Qobad Shafiee, and Hassan Bevrani. Online generalized droop-based demand response for frequency control in islanded microgrids. *Electrical Engineering*, 101(2):409–420, 2019.

[153] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[154] A. L. Ott. Evolution of computing requirements in the pjm market: Past and future. In *IEEE PES General Meeting*, pages 1–4, 2010.