Pontifícia Universidade Católica
do Rio de Janeiro

Andrew David Werner Rosemberg

A Framework for Assessing the Impacts of Network Formulations in the Operation of Hydrothermal Power Systems

Dissertação de Mestrado

Dissertation presented to the Programa de Pós-Graduação em Engenharia Elétrica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Elétrica.

Advisor : Prof. Alexandre Street de Aguiar
Co-Advisor: Prof. Davi Michel Valladão

Rio de Janeiro
January 2020

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

Andrew David Werner Rosemberg

A Framework for Assessing the Impacts of Network Formulations in the Operation of Hydrothermal Power Systems

Dissertation presented to the Programa de Pós-Graduação em Engenharia Elétrica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Elétrica. Approved by the Examination Committee.

Prof. Alexandre Street de Aguiar
Advisor
Departamento de Engenharia Elétrica – PUC-Rio

Prof. Davi Michel Valladão
Co-Advisor
Departamento de Engenharia Elétrica – PUC-Rio

Prof. Arild Helseth
Department of Electrical Power Engineering – SINTEF

Prof. Andy Philpott
Department of Engineering Science – University of Auckland,
New Zealand

Prof. Bruno Fanzeres
Departamento de Engenharia Industrial – PUC-Rio

Rio de Janeiro, January 30th, 2020

Andrew David Werner Rosemberg

Andrew Rosemberg received his B.Sc. degree in Control Engineering in 2017 from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Brazil. He did a double degree at Ecole Centrale Marseille, recieving a B.Sc. and a M.Sc. in General Engineering. Since 2017, he has been actively participating in research projects at the Laboratory of Applied Mathematical Programming and Statistics (LAMPS) in the Department of Electrical Engineering at PUC-Rio.

Bibliographic data

Rosemberg, Andrew David Werner

A Framework for Assessing the Impacts of Network Formulations in the Operation of Hydrothermal Power Systems / Andrew David Werner Rosemberg; advisor: Alexandre Street de Aguiar; co-advisor: Davi Michel Valladão. – 2020.

68 f. : il. color. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica, Rio de Janeiro, 2019.

Inclui bibliografia.

1. Engenharia Elétrica – Teses. 2. Hydrothermal Economic Dispatch. 3. Optimal Power Flow. 4. Stochastic Dual Dynamic Programming. 5. Otimização. I. Aguiar, Alexandre Street de. II. Michel Valladão, Davi. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título.

CDD: 621.3

Acknowledgments

First, I would like to thank my parents, Abraham Lincoln Rosemberg and Cristina Rosemberg, for all their love, support and sacrifices during my upbringing. They and my sister Clara Rosemberg have provided me with a happy household and have helped me in every way through the most difficult times.

I'd like to thank my advisor Alexandre Street for his excellent teachings inside and outside the classroom. From him, I have learned the importance of rigorous work and of aiming to produce something valuable to society. He's been there for me when I needed and encouraged me when necessary.

I'd like to thank my co-advisor Davi Valladão for his great orientation since my undergraduate program and during my masters. I am also thankful for his patience with my unending questions posed through all means available to me. His always pertinent suggestions have raised the quality of my work.

I'd like to thank Joaquim Garcia and Thuener Silva for their counsels through the development of my projects. Their high understanding of both theoretical and applied programming knowledge has guided me in challenges I thought impossible to overcome. They have helped me become a better researcher and programmer, besides becoming great friends.

I am thankful to Oscar Dowson for the vast help in my project before I could even ask. He has revised my code and my written work besides attending my every demand faster than I could deliver them. I hope I can continue to collaborate with him.

I wish to thank my colleagues at LAMPS for helping to create a warm and productive environment to work in. I'd also like to thank professors Cristiano Fernandes, Bruno Fanzeres, Carlos S. Kubrusly, Álvaro Veiga and Marco Molinaro for the excellent lectures and talks throughout this program.

I'd like to thank my girlfriend Natália Lamas and her family for all their love and affection in the past years. Natália has taught me to be patient and to have a balanced life, without which I wouldn't have achieved my goals.

This study was supported and partially financed by FGV Energia's project: P&D PD-09344-1703/2017.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Werner Rosemberg, Andrew David; Street de Aguiar, Alexandre (Advisor); Michel Valladão, Davi (Co-Advisor). **A Framework for Assessing the Impacts of Network Formulations in the Operation of Hydrothermal Power Systems**. Rio de Janeiro, 2020. 68p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

One of the most efficient algorithms for solving hydrothermal operation planning problems, which are large-scale multi-stage stochastic models, is the so-called stochastic dual dynamic programming (SDDP) algorithm. Operation planning of power systems aims to assess the value of the scarce resources (e.g. water) to feed short-term dispatch models used in the actual implementation of the decisions. When the planning model significantly deviates from the reality of the implemented operation, decision policies are said to be time-inconsistent. Recent literature has explored different sources of inconsistency such as time-inconsistent dynamic risk measures, inaccurate representation of the information process and simplifications in the network planning model. This work addresses the time-inconsistency due to simplifications in the network representation in the planning model extending the existing literature.
The objective of this work is to propose a framework, comprised of a methodology and an open-source computational package, for testing the operative and economic impact of modeling simplifications over the network power-flow in hydrothermal power systems. Among the myriad of formulations available in the package, we focused on assessing the cost and operative performance of the following model approximations: the transportation network-flow model (NFA), currently in use by the Brazilian system operator; the second-order cone relaxation (SOC); the semidefinite programming relaxation (SDP); the DC power-flow approximation (DC); and the DC with line-loss power-flow approximation (DCLL). All the previously mentioned formulations are tested as approximations for the network model in the planning stage, where the cost-to-go function is built. Then, we evaluate each approximation by simulating the system's operation using an implementation model, which minimizes the immediate cost under AC power-flow constraints and the respective cost-to-go function. The comparison is made for two systems, one composed of a cycle and the other approximately radial.

Keywords

Hydrothermal Economic Dispatch; Optimal Power Flow; Stochastic Dual Dynamic Programming; Power System Operation.

Resumo

Werner Rosemberg, Andrew David; Street de Aguiar, Alexandre; Michel Valladão, Davi. **Um Framework para Avaliar os Impactos das Formulações de Rede na Operação de Sistemas de Energia Hidrotérmica**. Rio de Janeiro, 2020. 68p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Um dos algoritmos mais eficientes para resolver problemas de planejamento de operações hidrotérmicas, que são modelos estocásticos multiestágio de larga escala, é o chamado algoritmo de programação dinâmica dupla estocástica (SDDP). O planejamento da operação dos sistemas de energia visa avaliar o valor dos recursos escassos (por exemplo, água) para alimentar os modelos de despacho de curto prazo usados na implementação real das decisões. Quando o modelo de planejamento se desvia significativamente da realidade da operação implementada, as políticas de decisão são consideradas inconsistentes no tempo. A literatura recente explorou diferentes fontes de inconsistência, como medidas de risco dinâmico inconsistentes no tempo, representação imprecisa do processo de informação e simplificações no modelo de planejamento de rede. Este trabalho aborda a inconsistência no tempo devido a simplificações na representação da rede no modelo de planejamento que estende a literatura existente.

O objetivo deste trabalho é propor uma estrutura, composta por uma metodologia e um pacote computacional de código aberto, para testar o impacto operacional e econômico das simplificações da modelagem sobre o fluxo de energia da rede em sistemas de energia hidrotérmica. Entre as inúmeras formulações disponíveis no pacote, nos concentramos em avaliar o custo e o desempenho operacional das seguintes aproximações de modelos: o modelo de rede de transporte (NFA), atualmente em uso pelo operador de sistema brasileiro; o relaxamento de cone de segunda ordem (SOC); o relaxamento de programação semidefinida (SDP); a aproximação do fluxo de energia de corente continua (DC); e o DC com aproximação de fluxo de potência com perda de linha (DCLL). Todas as formulações mencionadas anteriormente são testadas como aproximações para o modelo de rede na fase de planejamento, onde é construída a função de custo futuro. Em seguida, avaliamos cada aproximação simulando a operação do sistema usando um modelo de implementação que minimiza o custo imediato sob as restrições de fluxo de energia AC e a respectiva função de custo futuro. A comparação é feita para dois sistemas, um composto por um ciclo e o outro aproximadamente radial.

Palavras-chave

Despacho Econômico Hidrotérmico; Fluxo de Potência Ótimo; Programação Dinâmica Estocástica Dual; Operação de Systema de Potência.

Table of contents

List of figures

List of tables

1
Introduction

Assessing the value of systems' scarce resources is a key activity of power system operators, especially those in charge of hydrothermal power systems [2,3]. Such assessment aims at evaluating implicit opportunity costs of relevant systems' resources, such as water, which can be stored and used in future periods to prevent expensive thermal generation or load shedding. An accurate assessment is crucial for practical purposes, as solving the true multistage stochastic problem with all details used in short-term operational models is deemed intractable. Therefore, such an assessment allows us to decouple detailed short-term operational models from the long-term horizon problem yet keeping the coherence between the decisions of the former with the objective of the latter.

The assessment of relevant opportunity costs of systems' resources, such as water in hydrothermal power systems, is generally performed through long-term operational planning studies [4]. In these studies, the future operation of the system is simulated for many different inflow scenarios and the future economic benefit of water storage is passed for the short-term models as a function of the reservoir levels[1]. However, as planning studies comprise the solution of the long-run problem to simulate the optimal future operation of the system, it is inherent to planning activities the simplification of many features of the true problem. Therefore, the actually implemented operative policy is, in general, defined by a detailed short-term model with some input information about the opportunity cost of the water brought from simplified long-term operative planning studies. In this context, as future planned operative decisions are obtained with a simplified view of the system, the actually implemented operative decisions may differ from those planned even if the same scenario takes place. This is the case of a time-inconsistent policy due to modeling simplifications, i.e., when the planned decisions differ from the actually implemented ones [6]. Such inconsistency can be understood as a modeling risk, which in the case of a hydrothermal economic dispatch

[1] Needless to say, uncertainty in renewable generation, demand, fuel costs, system components availability, among others, should also be considered in these studies to ensure an accurate representation of critical system states and the value of the water (see [3] for a brief discussion and [5]).

problem can lead to several side effects as demonstrated in [3].

The literature on time (in)consistency is vast (we refer to [3, 6–9] and references therein). Within the multistage stochastic programming-related literature, two main branches can be found: time inconsistency induced by dynamically inconsistent risk measures (we refer to [6] and references therein), and time inconsistency due to modeling simplifications in the planning step (see [3] and [10]). This work focuses on the latter. The time-inconsistency gap proposed in [6] measures the expected cost difference between the planning and implementation policies. As demonstrated in [3], statistically significant gaps can be induced when disregarding the Kirchhoff Voltage Law (KVL) and n-1 security criterion. In [10], statistically significant gaps are found due to the hazard-decision simplification, where the information-revelation process of the inflow uncertainty is simplified by a one-step-ahead anticipative model[2]. The present work aims to extend the results of [3] by studying the cost and distortions induced by different modeling simplifications of the network in the hydrothermal economic dispatch problem. To the best of our knowledge, the literature lacks a systematic study on the impact of network power-flow simplification as a source of time inconsistency for hydrothermal operation planning.

More specifically, independent system operators (ISO) of hydrothermal power systems aim to coordinate reservoirs levels to ensure power balance across the network at a minimum cost in the long-run. Therefore, the true hydrothermal economic dispatch problem is a highly complex non-convex multistage stochastic optimization problem, wherein all network and technological constraints must be accounted for. In practice, however, operation planning studies rely on computational models based on dynamic stochastic programming techniques [11]. The state-of-the-art solution method is the so-called Stochastic Dual Dynamic Programming (SDDP) proposed in [2]. The SDDP algorithm iteratively approximates the future cost of operation, also known as cost-to-go function, as a piecewise linear convex function of the amount of water stored in the reservoirs. The computational efficiency of SDDP notwithstanding, the algorithm relies on strong assumptions such as convexity of the dispatch problem defining the cost-to-go function. To attend the SDDP requirements, system operators rely on modeling simplifications, e.g., using convex approximations for the alternating current (AC) power flow model. For instance, the official Brazilian model (NEWAVE – [4]) uses a transportation network flow approximation (NFA) model that only considers the flow limits

[2] In this model, all decision variables of a given stage are considered as wait-and-see decisions.

between areas and disregards the KVL. As a consequence, the operative point achieved with such an approximation significantly differs from those implemented in reality. This discrepancy between the planning and implemented solutions induce to an implemented time-inconsistent policy as reported in [3].

The results published in [3] assume a DC power flow model as the reference model for the assessment of the time-inconsistency gap when planning the system operation using the transportation NFA model (hereinafter, referred to as NFA). However, as reported in the literature [12, 13], direct current (DC) power flow approximation still produces significant discrepancies and the actual operation can be better represented by an AC power-flow model of the network. Thus, this work extends previously reported results on the impact of network simplifications in two fronts: i) the more accurate AC power-flow model is considered as the reference model for the actually implemented decisions in the time-inconsistency gap assessment and ii) five network modeling simplifications are compared in terms of induced time-inconsistency gap, thermal generation, and spot-price distortions.

To achieve the aforementioned goals i) and ii), reliable and flexible implementations of the SDDP algorithm allowing multiple power flow approximations are needed. Unfortunately, such a tool is not available. Therefore, an open-source SDDP tool with this flexibility, namely, *HydroPowerModels* is developed and made available [14]. This tool enables not only the assessment of the results found in this work but also allows researchers in the electrical industry to test new ideas, leveraging state-of-the-art solution methods and mathematical formulations for the economic dispatch problem. It combines two other open-source packages developed in Julia Language [15], namely, *PowerModels.jl*, developed by Coffrin [16, 17] to solve steady-state power network optimization problems, and *SDDP.jl*, developed by Oscar Dowson [18, 19] to solve multistage convex stochastic optimization problems. This combination puts together two state-of-the-art open-source packages sharing the same layer for modeling mathematical optimization in Julia called *JuMP.jl* [20].

The objective of this work is to propose a framework, comprising a methodology and an open-source computational package, for testing the operative and economic impact of modeling simplifications over the network power-flow in hydrothermal power systems. Among the myriad of formulations available in the package, we focused on assessing the cost and operative performance of the following model approximations: the transportation network-flow model (NFA), currently in use by the Brazilian system operator [4]; the second-order cone relaxation (SOC) [21]; the semidefinite programming relaxation (SDP) [22,23]; the DC power-flow approximation (DC) [24], and the DC with line-loss power-flow approximation (DCLL) [25]. All the previously mentioned formulations will be tested as approximations for the network model in the planning stage, where the cost-to-go function is built. Then, we evaluate each approximation by simulating the system's operation using an implementation model, which minimizes the immediate cost under AC power-flow constraints [26, 27] and the respective cost-to-go function.

The main contributions of this work are:

1. Extension of the inconsistency gap measurement methodology proposed in [3], comparing the performance of five network formulations, namely, transportation network-flow model relaxation, second-order cone relaxation, semidefinite programming relaxation, direct current approximation, and direct current with line loss under a more accurate implementation model using the AC power-flow formulation. The various approximation models feature different aspects of the network flow as well as impose different computational burdens. Thus, by analyzing and comparing the results of policies that plan using these simplifications we can measure the tradeoff between computational burden and model accuracy.

2. Results show the detrimental effect of neglecting more complete power flow constraints and benefits are weighted against the computational burdens of choosing more sophisticated models. The SOC planning policy has a poor performance with high dispatch distortion when the system is full meshed and performs well when the system is full radial, under mild conditions, always presenting medium computational burden. The SDP planning policy has low inconsistency for all systems, but comes with a high computational burden. The DC planning policy performs reasonably well (but worst than SDP) with low distortions in prices and dispatches in all instances and with a reduced computational burden. The DCLL planning policy has the lowest system cost, presents low inconsistency gaps and its computational burden is not as reduced as in the DC case, but it is still low.

3. An open-source tool for solving hydrothermal economic dispatch problems with stochastic dynamic programming, namely *HydroPowerModels* [14]. Programmed as a package for the Julia language, it takes advantage of the productivity and speed that this language has to stimulate collaboration and relevance. The package provides an easy-to-use framework where the user is free to choose from a variety of network formulations and study parameters. Moreover, it allows the construction of shared da-

tabase of cases, helping discussions about solution methods and study choices.

The remainder of this work is organized as follows. Chapter 2.1 presents the theoretical background for multistage stochastic programming, hydrothermal economic dispatch, and SDDP algorithm. Chapter 3 explains the proposed framework, detailing the mathematical model of a single-stage sub-problem, the five network approximation models derived from the AC power-flow model, discusses time inconsistency, when it happens and its consequences, and presents open-source tool developed to support studies of hydrothermal dispatch. Chapter 4 presents case studies. Relevant conclusions are drawn in Chapter 5. The nomenclature utilized in Chapters 3 and 3.2 is presented in Appendix A. Finally, Appendix B gives a small usage example of the *HydroPowerModels* tool.

2
Theoretical Background

The goal of a hydrothermal economic dispatch problem is to find the optimal operation policy of an electrical system to meet demand throughout a long-run planning horizon. For this purpose, the policy must take into account the best use of water to ensure power balance across the network during all periods under the uncertainty of inflows, demand, fuel costs, etc. In this work, we will consider only the inflow uncertainty to avoid deviating from our main focus. The hydrothermal economic dispatch is modeled as a large-scale multistage stochastic optimization problem with recourse and generally solved through SDDP [2, 28].

2.1
Multistage Stochastic Programming

Multistage stochastic programming (MSP) is a subclass of optimization problems with multiple stages (periods) and decisions being made conditioned to the information revelation process of the uncertainties (nonanticipative decision process). This class of problem is a branch of optimization under uncertainty, where the stochastic process describing the uncertainties is well defined and probabilities are known [11, 29–31].

In SP problems [29], the realization of some random variables b(ω) influences the conditions of the problem and consequently the optimal decisions. Uncertainty relates to the probability distributions of parameters and may be incorporated into the problem in various manners. One framework for SP is the Recourse Model paradigm, where problems are defined in more than one stage. In this framework, a class of simple stochastic programs is known as a two-stage problem with recourse. It can be formulated as follows:

$$1^{\text{st}}\text{ Stage}\begin{cases}\min\limits_{\mathbf{x}} & C^T\mathbf{x} + \mathbb{E}[\mathcal{Q}(\mathbf{x},\omega)]\\ \text{s.t.} & \mathbf{x}\in\mathcal{X}\end{cases}$$

$$2^{\text{nd}}\text{ Stage}\left\{\mathcal{Q}(\mathbf{x},\omega) = \begin{cases}\min\limits_{\mathbf{y}} & g^T\mathbf{y}\\ \text{s.t.} & A\mathbf{y} = b(\omega) - H\mathbf{x}\\ & \mathbf{y}\in\mathcal{Y}\end{cases}\right\}.$$

The objective of the first stage is composed of an *immediate cost* term $C^T\mathbf{x}$ and a *cost-to-go function* (also known as recourse function) $\mathbb{E}[\mathcal{Q}(\mathbf{x},\omega)]$. The latter is a function of some decision variables of the first stage (also known as here-and-now decisions), $\mathbf{x}$, that fix the state of the second stage. First stage decisions are made under uncertainty, while the decision variables of the second stage, $\mathbf{y}$, are chosen after the realization of the variable ω and are called *recourse variables* (also known as wait-and-see variables). The recourse function, $\mathbb{E}[\mathcal{Q}(\mathbf{x},\omega)]$, is general enough to encompass many relevant *risk measures* [7, 8, 28].

In this work, there will be a slight abuse of notation: $b(\omega)$ will represent either the random variable or its realization. This should be clear from the context in which it is used.

This problem intends to find an optimal first stage decision $\mathbf{x}$ and an optimal second-stage decision u for each realization of ω conditioned on $\mathbf{x}$. The two-stage problem discussed above naturally extends to a *multistage* problem via recursion. A multistage stochastic program with T stages can be formulated as follows:

$$\min_{\mathbf{x}_1 \in \mathcal{X}_1} f_1(\mathbf{x}_1) + \mathbb{E}[\inf_{\mathbf{x}_2 \in \mathcal{X}_2(\mathbf{x}_1,\omega_2)} f_2(\mathbf{x}_2,\omega_2) + \mathbb{E}[... + \mathbb{E}[\inf_{x_T \in \mathcal{X}_T(\mathbf{x}_{T-1},\omega_T)} f_T(\mathbf{x}_T,\omega_T)]...]].$$

Assuming the problem is linear, we have:

$$\mathcal{X}_t(\mathbf{x}_{t-1},\omega) = \left\{\mathbf{x}_t \geq 0 : A_t\mathbf{x}_t = b_t(\omega) - H_t\mathbf{x}_{t-1}\right\}$$
$$f_t(\mathbf{x}_t,\omega) = C_t^T\mathbf{x}_t$$

In this framework, the scenarios, $\omega_t \in \Omega_t$, have given probability distribution and the sequence $b_1(\omega_1), ..., b_T(\omega_T)$ represents the realization of a multivariate stochastic process, which should follow a filtration dictating the information revelation process. In other words, the uncertain parameter is revealed gradually over time as in many real-world applications.

Just like in the two-stage problem, the state variables, $\mathbf{x}_{t-1}$, are the sufficient information from previous stages needed besides the realized value of uncertainties to define the optimal decisions of a stage t. The mapping from state and realization of the stochastic process to stage decisions is called a decision rule, and the set of decision rules from all stages is called a policy.

Multistage stochastic problems are a useful tool for modeling a diverse set of problems and discussing their aspects. However, solving a multistage stochastic program and finding the optimal policy is a challenging computational task. Exact solutions cannot be obtained in reasonable time and common approximations such as stochastic dynamic programming [32, 33] are intractable in the presence of high-dimensional state spaces.

Sampling-based decomposition algorithms, such as SDDP [2] and its variants [34–36], aim to overcome the curse of dimensionality. The idea behind it is to assume the decision rule in stage t depends only on the value of the state at the end of stage $t-1$ and the realization of the uncertain parameter ω_t. This allows the multistage stochastic problem to be decomposed in dynamic equations. Besides these assumptions, $b_t(\omega_t)$ should be an affine function for SDDP to work.

The dynamic equations constitute a set of recursive functions, $\{\mathcal{Q}_t\}_{t=1,\dots,T}$, representing the dynamics of optimal costs and decisions from stage t until the end of the horizon, T, also known as Bellman recursion. In every stage, $\mathcal{Q}_t$ is the optimal value of an optimization problem whose objective function is composed of an *immediate cost* term, say, $c_t^T\mathbf{x}_t$, and the next stage function $\mathcal{Q}_{t+1}$. $\mathcal{Q}_t$ relies on the decision variables of the current stage, $\mathbf{x}_t$, which plays the role of *state variables* of the following stage. This function is also affected by uncertain data, ω_t. Thus, to address the uncertainty in the optimization problem of a given period defining $\mathcal{Q}_t$, a *risk measure* μ, commonly assumed to be the expectation operator $\mathbb{E}$, is used [37]. Therefore, these functions are named *cost-to-go functions*, as they represent the optimal cost from a given stage until the end of the horizon, or *recourse functions*, as they can also be seen as the recourse function of the two-stage stochastic problem defining each stage dynamic equation.

The nested formulation for the aforementioned dynamic equations is represented by the following Bellman recursion:

$$\begin{aligned}
\mathcal{Q}_t(\mathbf{x}_{t-1}, \omega_t) = \min_{\mathbf{x}_t} \quad & c_t^T\mathbf{x}_t + \mathbb{E}[\mathcal{Q}_{t+1}(\mathbf{x}_t, \omega_{t+1})] && \text{(2-1)}\\
\text{s.t.} \quad & A_t\mathbf{x}_t = b_t(\omega_t) - H_t\mathbf{x}_{t-1} \quad [\pi_t(\omega_t)] && \text{(2-2)}\\
& \mathbf{x}_t \geq 0, && \text{(2-3)}
\end{aligned}$$

where for each stage $t \leq T$, and $\mathcal{Q}_{T+1}(\cdot,\cdot) = 0$. Here, $\pi_t(\omega_t)$ represents the dual variables of the problem that are conditioned on the uncertain data process.

In these equations, the optimal value at stage t depends on the previous decision $\mathbf{x}_{t-1}$ and the realized scenario ω_t. Furthermore, the optimal value of a stage represents the costs of all future stages and, finally, the first stage problem gives the optimal value of the corresponding multistage problem.

Since the Bellman recursion finds the best solution for every stage given previous states $\mathbf{x}$ and uncertainties ω, it determines an optimal policy.

Problems that can be decomposed in dynamic equations allow the separation of cost-to-go functions by stage, reducing the number of evaluations needed to solve them.

2.2 The SDDP solution Methodology

In multistage stochastic problems with recourse, the decision-maker faces a complicated problem structure with random variables and multiple stages. Due to the complexity of these problems, analytical solutions are limited, and conventional optimization methods for deterministic problems do not work. For a two-stage problem, the conventional solution methods for this problem approximate the future cost function with outer approximations and use a scenario tree to represent the stochastic process of the random parameters.

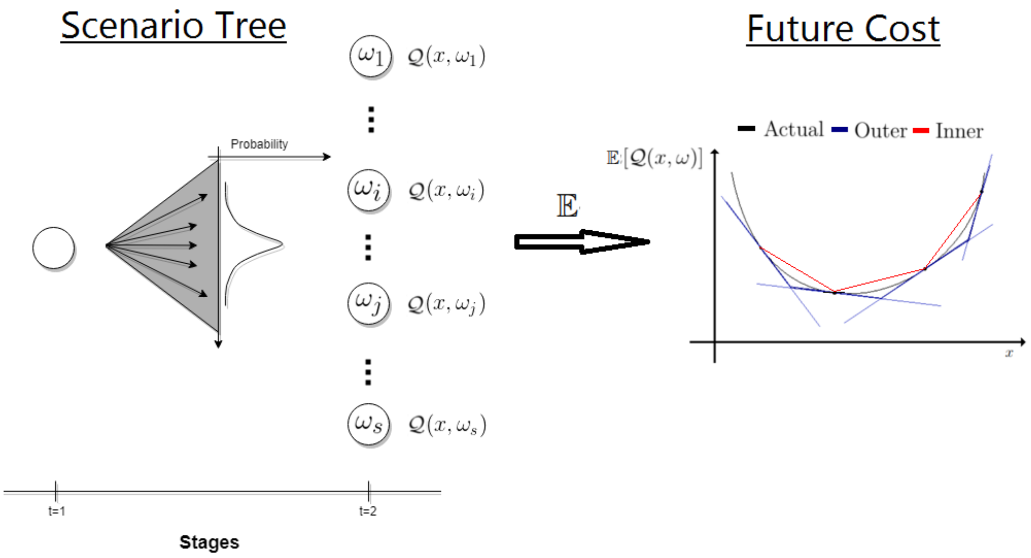
Scenario Tree
Probability
ω_1 $\mathcal{Q}(x,\omega_1)$
ω_i $\mathcal{Q}(x,\omega_i)$
ω_j $\mathcal{Q}(x,\omega_j)$
ω_s $\mathcal{Q}(x,\omega_s)$
t=1
t=2
Stages
$\mathbb{E}$
Future Cost
Actual Outer Inner
$\mathbb{E}[\mathcal{Q}(x,\omega)]$
x

Figure 2.1: Scenario Tree and Future Cost Function of a Two-Stage Problem.

Figure 2.1 shows the basic idea behind these methods. First, a set of values are sampled from the distributions and then used to evaluate the second stage cost for a state value $\mathbf{x}$. This evaluations are used to approximate the future cost function. The process is repeated for a number of different state values until a sufficient approximation is developed.

In multistage stochastic problems, we are faced with a cost-to-go function: $\mathbb{E}[\mathcal{Q}_{t+1}(\mathbf{x}_t, \omega_{t+1})]$. The issue is that $\mathbb{E}[\mathcal{Q}_{t+1}(\mathbf{x}_t, \omega_{t+1})]$ also depends on a cost-to-go $\mathbb{E}[\mathcal{Q}_{t+2}(\mathbf{x}_{t+1}, \omega_{t+2})]$, and the evaluation of those functions can be expensive.

Because of this cascade of recourse functions, the dimension size of an SP problem is exponentially proportional to the number of stages in a multistage stochastic program. Hence, most of the methods used for two-stage problems are not sufficient, and more efficient algorithms are required to overcome this issue.

The crucial step that facilitates the solution of these problems is to construct approximations of the cost-to-go functions, recursively, going backward in time. Thus, the optimal value of the first stage problem approximates the optimal value of the corresponding multistage problem.

As discussed previously, for the construction of this approximation a widely used method is dynamic programming, which evaluates the function in a range of discrete values of the state variable for further interpolation. However, this method becomes intractable with the growth of the state dimension (commonly referred to as the *curse of dimensionality* of dynamic programming). A solution to reduce the curse, but not alleviate it, was proposed by [2] with the SDDP method.

The methodology, simply posed, approximates the cost-to-go functions by the maximum of a set of linear hyper-planes, called *cuts*. This extends the cutting plane method (or Benders decomposition) from a two-stage setting to a multistage one.

The Cutting planes procedure constructs a polyhedral outer approximation for $\mathbb{E}[\mathcal{Q}_{t+1}(\mathbf{x}_t, \omega_{t+1})]$ (considering the expected value $\mathbb{E}$ as the risk measure) by alternating between two phases. At iteration k, first phase solves the stage t problem using a current approximation and outputs a state $\mathbf{x}_t^k$. The other evaluates the objective of stage $t+1$ at $\mathbf{x}_t^k$ and for different scenarios ω_{t+1}. Then, with the risk measure of those costs $\mathbb{E}[\tilde{\mathcal{Q}}_{t+1}(\mathbf{x}_t^k, \omega_{t+1})]$ and the dual variable $\bar{\pi}_{t+1,k}$ of the constraints where the state affects the right-hand-side, the second phase updates the approximation with a cut tangent to this measurement.

In practice, the method uses an auxiliary variable θ_{t+1}, with feasible region identical to the epigraph of the approximate function. This variable is, at the optimal, equivalent to this same function, as it is illustrated in figure 2.2.

Considering an arbitrary sub-problem $\mathcal{SP}_t(\mathbf{x}_{t-1}, \omega_t)$ of a multistage problem, as in equations (2-1)-(2-3), we identify the set of linear inequalities defining the cuts (K cuts):

$$\theta_{t+1} \geq \bar{g}_{t+1,k} + \bar{\pi}_{t+1,k}^T H_{t+1}(-\mathbf{x}_t) \qquad for \quad k = 1..K \tag{2-4}$$

Where:

$$\bar{g}_{t+1,k} = \mathbb{E}[\tilde{\mathcal{Q}}_{t+1}(\mathbf{x}_t^k, \omega_{t+1})] + \bar{\pi}_{t+1,k}^T H_{t+1}(\mathbf{x}_t^k) \tag{2-5}$$

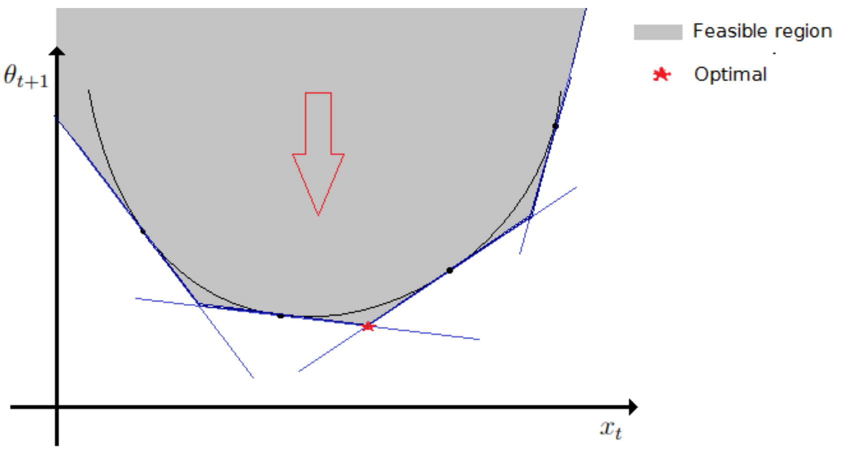

Figure 2.2: Feasible and opt. value θ_{t+1}

$$\bar{\pi}_{t+1,k} = \mathbb{E}[\pi_{t+1}(\omega_{t+1})] \tag{2-6}$$

In this setting, the sub-problem containing K cuts, $\mathcal{SP}_t^K(\mathbf{x}_{t-1}, \omega_t)$, is given by:

$$\tilde{\mathcal{Q}}_t^K(\mathbf{x}_{t-1}, \omega_t) = \underset{\mathbf{x}_t, \theta_{t+1}}{\text{Min}} \ C_t^T \mathbf{x}_t + \theta_{t+1} \tag{2-7}$$

$$\text{s.t.}$$

$$A_t \mathbf{x}_t = b_t(\omega_t) - H_t \mathbf{x}_{t-1} \qquad [\pi_t(\omega_t)] \tag{2-8}$$

$$\theta_{t+1} \geq \bar{g}_{t+1,k} + \bar{\pi}_{t+1,k}^T H_{t+1}(-\mathbf{x}_t) \qquad for \quad k = 1..K \tag{2-9}$$

$$\mathbf{x}_t \geq 0 \tag{2-10}$$

SDDP uses cutting planes for each pair of stages interactively constructing the outer approximation of the cost-to-go functions. The procedure is divided into alternating forward and backward passes (or steps). In the forward step, the algorithm uses the current approximation to obtain feasible trial solutions for a sampled stochastic process realization; this is an exploration phase where the algorithm samples trial points (state values) to enhance the cost-to-go function approximation. In the backward pass, the cost-to-go function is updated as the pointwise maximum of the current outer approximation and an additional first-order outer approximation around each trial point sampled in the forward step. Algorithm 1 summarizes the SDDP method.

Assuming stagewise independence, as it is done in most of the sampling-based methods, cuts can be shared among solutions of the same stage. This unique polyhedral outer approximation for the cost-to-go function of each stage reduces significantly the number of evaluations needed, allowing for a computationally tractable algorithm.

Algorithm 1: Stochastic Dual Dynamic Programming

```
Result: ${\mathcal{Q}_t}_{t=1}^{T}$
Set $K \leftarrow 0$
while not converged do
    Initialize $\mathbf{x}_0^K$ with $\nu_0$.
    Convergence Test from [3].
    /* Forward Pass                                                  */
    Sample an inflow time series $\{\omega_t\}_{t=1}^{T}$
    for $t = 1$ to $T$ do
        Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1}^K, \omega_t)$, using $\mathcal{Q}_{t+1}^K$
         and store $\mathbf{x}_t^K$.
    end
    /* Backward Pass                                                 */
    $\{\mathcal{Q}^{K+1}\}_{t=1}^{T} \leftarrow \{\mathcal{Q}^{K}\}_{t=1}^{T}$
    for $t = T$ to 2 do
        for $\omega_t \in \Omega_t$ do
            Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1}^K, \omega_t)$, using
             $\mathcal{Q}_{t+1}^{K+1}$ and store the objective value and dual vector for
             the backward scenario $\omega_t$.
        end
        Calculate new Bender's cut.
        Update the approximation of the cost-to-go function with the
         new Bender's cut.
    end
    $K \leftarrow K + 1$
end
.
```

For a detailed discussion of the SDDP algorithm and its convergence criterion, see [2, 18].

3
Proposed framework

In this chapter, we present the complete mathematical formulation of the hydrothermal economic dispatch sub-problem following the notation from Appendix A. It is defined as the function $\mathcal{Q}_t(\nu_{t-1}, \omega_t) =$

$$\begin{aligned}
\min_{\mathbf{x}_t}: \; & \sum_{i \in \mathcal{I}} C_{it}\, p_{it} + \sum_{n \in \mathcal{N}} C^{\delta}_{nt}\, \delta_{nt} + \sum_{j \in \mathcal{H}} C^{s}_{jt}\, s_j + \\
& \qquad \mathbb{E}_{t+1}[\mathcal{Q}_{t+1}(\nu_t, \omega_{t+1})] && \text{(3-1a)} \\
\text{s.t.:} \; & \\
& \sum_{i \in I_n} p_{it} + \sum_{j \in \mathcal{H}_n} u_{jt}\, \rho_{jt} - \sum_{m \in \mathcal{N}_n} f_{(n,m)t} - \ell_{nt} + \\
& \qquad = D_{nt} - \delta_{nt}, \;\; \forall n \in \mathcal{N} && \text{(3-1b)} \\
& \nu_{jt} + u_{jt} + s_{jt} = \nu_{j,t-1} + A_{j,t}(\omega_t) + \\
& \qquad \sum_{k \in \mathcal{H}^U_j} u_{kt} + \sum_{k \in \mathcal{H}^S_j} s_{kt}, \;\; \forall j \in \mathcal{H} && \text{(3-1c)} \\
& |f_{(n,m)t}| \le F_{(n,m)t} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-1d)} \\
& |f_{(m,n)t}| \le F_{(n,m)t} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-1e)} \\
& 0 \le p_{it} \le P_{it} \;\; \forall i \in I && \text{(3-1f)} \\
& 0 \le \nu_{jt} \le \overline{\nu}_{jt} \;\; \forall j \in \mathcal{H} && \text{(3-1g)} \\
& 0 \le u_{jt} \le U_{jt} \;\; \forall j \in \mathcal{H} && \text{(3-1h)} \\
& \ell_{nt} \ge 0 \;\; \forall n \in \mathcal{N} && \text{(3-1i)} \\
& \delta_{nt} \ge 0 \;\; \forall n \in \mathcal{N} && \text{(3-1j)} \\
& \mathbf{x}_t \in \mathcal{X}_t. && \text{(3-1k)}
\end{aligned}$$

In the above equation, $\mathbf{x}_t$ is the stacked vector of all decision variables, $\mathbf{x}_t = [p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t]^T$, and $\mathcal{X}_t$ represents different power-flow constraints related to the associated network formulation.

The objective function of the sub-problem (3-1a) is to minimize the sum of immediate costs represented by the costs of active power generation and the cost of energy supply deficit, and future costs represented by the cost-to-go function $\mathbb{E}_{t+1}[\mathcal{Q}_{t+1}(\nu_t, \omega_{t+1})]$. Some other cost terms can be and were added to the objective function, for instance, cost of spillage, C^s - a useful tool to avoid degenerate intermediary solutions and slow convergence of SDDP. Constraints (3-1f) bound the power generation, representing the physical limitation of generators and fuel source availability. The power flow is bounded in (3-1d) through its absolute value. These power limits of the lines represent thermal limits and stability limits. Constraints (3-1b) implements Kirchhoff's Current Law (KCL), which refers to power balance at each node. Deficit variables guarantee feasibility in case of a lack of power availability. An important variable in an economic dispatch problem is the marginal cost of energy at each bus, which is determined by the optimal value of the dual variable associated with constraint (3-1b). This value is also referred to as a shadow price, local marginal price (LMP), or nodal price: the cost of an extra unit of energy in a bus. The water mass balance equation is implemented in (3-1c), where the water stored at a reservoir should equal the water previously stored plus the incoming inflows and water discharged from upstream reservoirs, minus the portion used to generate energy and the one spilled away. Constraints (3-1g) and (3-1h) bound respectively the volume of water stored and the amount of water used in generation. These limits are defined by the capacity limit of the storage facility and the equipment installed. Finally, expression (3-1k) represents a feasibility set used to model all network models considered in this work. The characterization of this set to represent the different network models is made in the following sections.

3.1 AC network power-flow constraints

Problem (3-1) implements the active power flow constraints of a system ignoring the constraint (3-1k). More complete power-flow equations may be drawn from the pi-section equivalent model that represents electric elements which interfere in line energy flow. This model is shown in Figure 3.1. The complex parameters $G_{(n,m)t} + \Im B_{(n,m)t}$ are related to line impedance, while $G^c_{(n,m)t} + \Im B^c_{(n,m)t}$ relate to the capacitive or inductive characteristic of the line. These parameters appear in the second Kirchhoff law constraints, defining the relationship between busses' voltage and line flow. The other parameters depicted in this picture, Y^s_{nt} and Y^s_{mt}, relate to the shunt aspects of each bus and define energy losses at these busses, namely ℓ_{nt} and ℓ_{mt}. Moreover, Observe that the flow variables are always outgoing from each bus, so $f_{(n,m)t}$ and $f_{(m,n)t}$ will have opposite signs and might have different magnitudes in the presence of power-flow losses. The loss value in a given line (n, m) is given by $f_{(n,m)t} + f_{(m,n)t}$. A more detailed explanation of this equivalent model may be found in [38, Chapter 6].

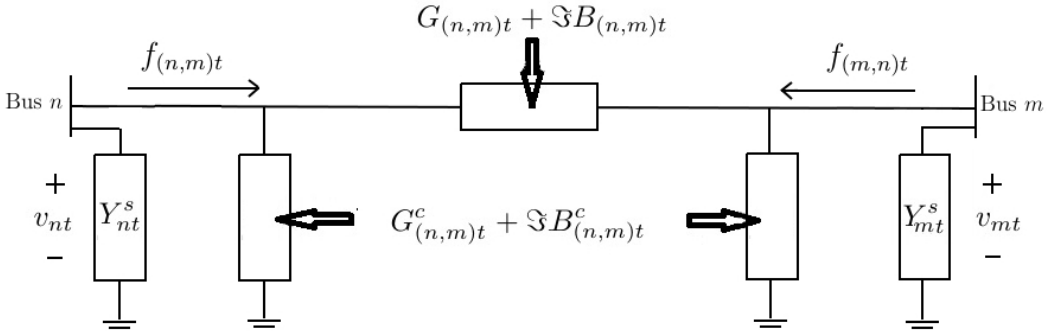

Figure 3.1: Pi-Section Model

Therefore, in order to fully represent the power flow of a realistic system, the alternating current constraints in (3-2) must be imposed [26]: $\mathcal{X}_t \leftarrow \mathcal{X}_{AC,t}$, where

$$\mathcal{X}_{AC,t} = \left\{ p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \,\middle|\, \exists\ q_t, v_t, \theta_t, f^q_t : \right.$$

$$\angle v_{nt} = \theta_{nt} \ \ \forall n \in \mathcal{N} \tag{3-2a}$$

$$\theta_{nt} = 0 \ \ \forall n \in \mathcal{N}_0 \tag{3-2b}$$

$$\sum_{i \in I_n} q_{it} - Y^{sq}_{nt}|v_{nt}|^2 - \sum_{m \in \mathcal{N}_n} f^q_{(n,m)t} = 0 \ \ \forall n \in \mathcal{N} \tag{3-2c}$$

$$(f_{(n,m)t})^2 + (f^q_{(n,m)t})^2 \leq F^2_{(n,m)t} \ \ \forall (n,m) \in \mathcal{L} \tag{3-2d}$$

$$(f_{(m,n)t})^2 + (f^q_{(m,n)t})^2 \leq F^2_{(n,m)t} \ \ \forall (n,m) \in \mathcal{L} \tag{3-2e}$$

$$\underline{V}_{nt} \leq |v_{nt}| \leq \overline{V}_{nt} \ \ \forall n \in \mathcal{N} \tag{3-2f}$$

$$\ell_{nt} = Y^s_{nt}|v_{nt}|^2 \ \ \forall n \in \mathcal{N} \tag{3-2g}$$

$$\underline{\Delta}^\theta_{(n,m)t} \leq \angle(v_{nt}v^*_{mt}) \leq \overline{\Delta}^\theta_{(n,m)t} \ \ \forall (n,m) \in \mathcal{L} \tag{3-2h}$$

$$\begin{aligned} f_{(n,m)t} = & \left(G_{(n,m)t} + G^c_{(n,m)t}\right)|v_{nt}|^2 + \\ & \left(-G_{(n,m)t} + B_{(n,m)t}\right)|v_{nt}||v_{mt}|cos(\theta_{nt} - \theta_{mt}) + \\ & \left(-B_{(n,m)t} - G_{(n,m)t}\right)|v_{nt}||v_{mt}|sin(\theta_{nt} - \theta_{mt}) \\ & \forall (n,m) \in \mathcal{L} \end{aligned} \tag{3-2i}$$

$$\begin{aligned} f_{(m,n)t} = & \left(G_{(m,n)t} + G^c_{(m,n)t}\right)|v_{mt}|^2 + \\ & \left(-G_{(m,n)t} + B_{(m,n)t}\right)|v_{nt}||v_{mt}|cos(\theta_{mt} - \theta_{nt}) + \\ & \left(-B_{(m,n)t} - G_{(m,n)t}\right)|v_{nt}||v_{mt}|sin(\theta_{mt} - \theta_{nt}) \\ & \forall (n,m) \in \mathcal{L} \end{aligned} \tag{3-2j}$$

$$f^q_{(n,m)t} = -\left(B_{(n,m)t} + B^c_{(n,m)t}\right)|v_{nt}|^2 -$$

$$\left(-G_{(n,m)t} - B_{(n,m)t}\right) |v_{nt}||v_{mt}| cos(\theta_i - \theta_j) +$$
$$\left(B_{(n,m)t} - G_{(n,m)t}\right) |v_{nt}||v_{mt}| sin(\theta_i - \theta_j)$$
$$\forall (n,m) \in \mathcal{L} \tag{3-2k}$$
$$f^q_{(m,n)t} = -\left(B_{(m,n)t} + B^c_{(m,n)t}\right) |v_{mt}|^2 -$$
$$\left(-G_{(m,n)t} - B_{(m,n)t}\right) |v_{nt}||v_{mt}| cos(\theta_n - \theta_m) +$$
$$\left(B_{(m,n)t} - G_{(m,n)t}\right) |v_{nt}||v_{mt}| sin(\theta_m - \theta_n)$$
$$\forall (n,m) \in \mathcal{L} \tag{3-2l}$$
$$- Q_{it} \leq q_{it} \leq Q_{it} \;\; \forall i \in I \Big\}. \tag{3-2m}$$

In this set of constraints, we use the following usual transformation of the line parameters:

$$\frac{1}{R_{(n,m)t} + \Im X_{(n,m)t}} = G_{(n,m)t} + \Im B_{(n,m)t}. \tag{3-3}$$

Constraint (3-2b) fixes reference buses complex voltage angles to zero, as the remaining angles will be defined accordingly. The magnitude of the complex voltage is bounded in constraint (3-2f). The upper limit defines a circular feasible region for each voltage, while the lower limit reshapes the region as a ring, producing a non-convex feasible region. The branch complex power flow is formulated in (3-2i), (3-2j), (3-2k) and (3-2l), that are dependent on the voltage at each end and implement elements of line charging. Angle differences between buses are bounded in (3-2h). The reason for these limits is to approximate the transient stability constraints of power flowing in branches. These restrictions refer to the synchronism among machines at each end of a line. The limits depend on the equipment installed and the system configuration. Note that problem induced by (3-2), that is when $\mathcal{X}_t \leftarrow \mathcal{X}_{AC,t}$ in (3-1), better represents reality, but is a nonlinear and non-convex optimization problem. This issue will have important implications for our solution approach.

3.2 Network power-flow modeling simplifications

The AC-OPF model is a non-convex non-linear problem (NLP), not suitable for the classical SDDP algorithm. Thus, in many applications, convex approximations and relaxations are used to meet the SDDP convexity hypothesis. At the same time, other reasons have led to the development of general approximations and relaxations, converting the non-trivial NLP into tractable convex or linear problems [39]. Such reasons range from the lack of global-optimality guarantees for local solutions to the need for fast solution methods in time-sensitive applications. In general, the proposed formulations are simplifications of the full AC model, and each one of them focuses on some particularities of the original problem. As a result, it is relevant to understand the tradeoff between each approximation quality and model tractability.

The models with *relaxations* of the nonlinear power flow constraints, when solved to optimality, provide lower bounds to the original problem because their feasible sets include all the solutions of the original problem. Convex relaxations are especially useful because their solutions are globally optimal for the relaxed problem, and the cuts generated by these relaxations are valid outer approximations for the real problem. Although there exist many convex relaxations for the optimal power flow problem, we focus on the most relevant ones. One simple linear relaxation used in the Brazilian official dispatch tools (see [4]) is the *transportation NFA model.* NFA implements power flow limits for each line but ignores buses power voltage. Consequently, constraints in (3-2) are not considered, and only the real part of the power flow equations are modeled. Moreover, in this lossless model, all the energy that is injected from an arbitrary bus n into a line (n, m), outputs at the receiving bus m. In other words, we make $\mathcal{X}_t \leftarrow \mathcal{X}_{NFA,t}$ in (3-1) where:

$$\mathcal{X}_{NFA,t} = \Big\{ p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \;\Big|\; f_{(n,m)t} = -f_{(m,n)t} \;\; \forall (n,m) \in \mathcal{L} \Big\}, \tag{3-4a}$$

In the attempt to provide better relaxations, researchers have used an equivalent model for (3-2) ([21,22,40]). This equivalent formulation, presented in (3-5), uses an auxiliary variable w_{nm} to represent the product of the voltage from buses n and m, i.e., $\mathrm{w}_{nmt} = v_{nt} v_{mt}$. Therefore, an equivalent description of the AC power-flow model is as follows:

$$\mathcal{X}_{AC,t} = \Big\{ \; p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \;\Big|\; \exists \; q_t, v_t, \mathbf{W}_t, f^q_t \; :$$

$$\mathrm{w}_{nmt} = v_{nt} v_{mt} \;\; \forall (n,m) \in \mathcal{L} \tag{3-5a}$$

$$\mathbf{W}_t \succeq 0 \tag{3-5b}$$

$$|\mathrm{w}_{nmt}|^2 \le \mathrm{w}_{nnt} \mathrm{w}_{mmt} \;\; \forall (n,m) \in \mathcal{L} \tag{3-5c}$$

$$\underline{V}^2_{nt} \le \mathrm{w}_{nnt} \le \overline{V}^2_{nt} \;\; \forall n \in \mathcal{N} \tag{3-5d}$$

$$(f_{(n,m)t})^2 + (f^q_{(n,m)t})^2 \le F^2_{(n,m)t} \;\; \forall (n,m) \in \mathcal{L} \tag{3-5e}$$

$$\underline{\Delta}^\theta_{(n,m)t} \le \angle(\mathrm{w}_{nmt}) \le \overline{\Delta}^\theta_{(n,m)t} \;\; \forall (n,m) \in \mathcal{L} \tag{3-5f}$$

$$\sum_{i \in I_n} q_{it} - Y^{sq}_{nt} \mathrm{w}_{nnt} - \sum_{m \in \mathcal{N}_n} f^q_{(n,m)t} = 0 \;\; \forall n \in \mathcal{N} \tag{3-5g}$$

$$\ell_{nt} = Y^s_{nt} \mathrm{w}_{nnt} \;\; \forall n \in \mathcal{N} \tag{3-5h}$$

$$\begin{aligned}
&f_{(n,m)t} = \left(G_{(n,m)t} + G^{c}_{(n,m)t}\right)\mathrm{w}_{nnt} + \\
&\left(-G_{(n,m)t} + B_{(n,m)t}\right)\mathrm{w}^{\Re}_{nmt} + \\
&\left(-B_{(n,m)t} - G_{(n,m)t}\right)\mathrm{w}^{\Im}_{nmt} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-5i)}\\
&f_{(m,n)t} = \left(G_{(m,n)t} + G^{c}_{(m,n)t}\right)\mathrm{w}_{mmt} + \\
&\left(-G_{(m,n)t} + B_{(m,n)t}\right)\mathrm{w}^{\Re}_{mnt} \\
&\left(-B_{(m,n)t} - G_{(n,m)t}\right)\mathrm{w}^{\Im}_{mnt} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-5j)}\\
&f^{q}_{(n,m)t} = -\left(B_{(n,m)t} + B^{c}_{(n,m)t}\right)\mathrm{w}_{nnt} - \\
&\left(-G_{(n,m)t} - B_{(n,m)t}\right)\mathrm{w}^{\Re}_{nmt} + \\
&\left(B_{(n,m)t} - G_{(n,m)t}\right)\mathrm{w}^{\Im}_{nmt} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-5k)}\\
&f^{q}_{(m,n)t} = -\left(B_{(m,n)t} + B^{c}_{(m,n)t}\right)\mathrm{w}_{mmt} - \\
&\left(-G_{(m,n)t} - B_{(m,n)t}\right)\mathrm{w}^{\Re}_{mnt} + \\
&\left(B_{(m,n)t} - G_{(m,n)t}\right)\mathrm{w}^{\Im}_{mnt} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-5l)}\\
&-Q_{it} \le q_{it} \le Q_{it} \;\; \forall i \in I \Big\}. && \text{(3-5m)}
\end{aligned}$$

Where $\mathrm{w}^{\Re}_{nmt}$ represents the real part of the w_{nmt} variable, and, $\mathrm{w}^{\Im}_{nmt}$, the imaginary part.

One of these convex relaxations is the *semidefinite programming relaxation* (SDP) that deals with the non-convex constraint (3-5a), composed of the bilinear product of the voltage variables, by using the fact that it defines a positive semidefinite matrix with rank 1: $\mathrm{w}_{nmt} = v_{nt}v_{mt} \leftrightarrow \mathbf{W}_t \succeq 0 \wedge rank(\mathbf{W}_t) = 1$. Proposed in [22], the relaxation comes from removing the rank 1 restriction resulting in (3-5b).

$$\mathcal{X}_{SDP,t} = \left\{ p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \;\middle|\; \exists\; q_t, \mathbf{W}_t, f^{q}_t : \right.$$

$$\begin{aligned}
&\mathbf{W}_t \succeq 0 && \text{(3-6a)}\\
&|\mathrm{w}_{nmt}|^2 \le \mathrm{w}_{nnt}\mathrm{w}_{mmt} \;\; \forall (n,m) \in \mathcal{L} && \text{(3-6b)}\\
&\text{(3-5d)} - \text{(3-5m)}\Big\}. && \text{(3-6c)}
\end{aligned}$$

Another relaxation is the *Second-Order Cone relaxation* (SOC), a non-linear convex relaxation that is tighter than the linear versions, i.e., its feasible region is strictly contained inside them. Proposed in [21], this formulation relaxes (3-5b), and so even further (3-5a), by neglecting the phases of the voltages and saving only their branch-wise difference and magnitudes. The result is the following inequality: $|\mathrm{w}_{nmt}|^2 \le |v_{nt}|^2|v_{mt}|^2 = \mathrm{w}_{nnt}\mathrm{w}_{mmt}$. The resulting problem may be specified as a second-order cone formulation, shown in (3-7).

$$\mathcal{X}_{SOC,t} = \Bigg\{ p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \Bigg| \exists\ q_t, \mathbf{W}_t, f^q_t :$$

$$|\mathrm{w}_{nmt}|^2 \leq \mathrm{w}_{nnt}\mathrm{w}_{mmt} \ \ \forall (n,m) \in \mathcal{L} \tag{3-7a}$$

$$\text{(3-5d)} - \text{(3-5m)}\Bigg\}. \tag{3-7b}$$

In this regard, the feasible region of the AC formulation, shown in (3-5), is contained within the feasible region of the SDP relaxation (3-6), which is contained in the SOC set (3-7) [12,41]. To demonstrate this inclusion order, we have first reformulated the AC power flow set, $\mathcal{X}_{AC,t}$, defined in (3-2), as (3-5). In the latter, we considered two additional loose constraints, namely, (3-5c) and (3-5b), which are always attended in the presence of (3-5a). Then, we defined the set $\mathcal{X}_{SDP,t}$ by dropping expression (3-5a), and defined the set $\mathcal{X}_{SOC,t}$ by subsequently dropping expression (3-5b). Notice that, since SDP defines a tighter relaxation than SOC and NFA, it provides a better bound to the original problem. These and other convex relaxations (see [40]) provide alternatives to approximately solve the original problem in the planning stage. However, they may give optimistic estimates of the operation and deficit costs. Thus, assessing the bias produced by these simplifications is of foremost importance.

Instead of relaxing, it is possible to approximate some of the nonlinear power flow constraints. The *linear DC approximation* (DC) [24] is a linear formulation, which partially represents power voltages. To achieve such formulation some assumptions are made: voltage magnitudes are sufficiently close to the nominal value one, angle differences are close to zero, and line power losses are negligible. When applied to (3-2i) and (3-2k), these assumptions make $f^q_{ij} = f^q_{ji} = 0$ and $f_{ij} = -f_{ji} = -b_{ij}(\angle(V_i) - \angle(V_j))$. Within these conditions, the model boils down to a set of linear constraints and active and reactive power are decoupled. In this context, since the reactive power does not restrict the feasible set and does not affect the objective function of (3-1), reactive power variables can be dropped. Therefore, the only left parameter from the pi-equivalent model is a modifeied $B_{(n,m)t}$ defined in (3-9). The DC power-flow set $\mathcal{X}_{DC,t}$ can be defined as follows:

$$\mathcal{X}_{DC,t} = \Bigg\{ p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \Bigg| \exists\ \theta_t :$$

$$f_{(n,m)t} = -\left(B_{(n,m)t}\right)(\theta_{nt} - \theta_{mt}) \ \ \forall (n,m) \in \mathcal{L} \tag{3-8a}$$

$$f_{(m,n)t} = -\left(B_{(n,m)t}\right)(\theta_{mt} - \theta_{nt}) \ \ \forall (n,m) \in \mathcal{L} \tag{3-8b}$$

$$\ell_{nt} = 0 \ \ \forall n \in \mathcal{N}\Bigg\}, \tag{3-8c}$$

where we define

$$B_{(n,m)t} = \frac{1}{X_{(n,m)t}}, \tag{3-9}$$

which is a usual notation used in the literature.

Thus, the DC optimal power-flow model is derived by making $\mathcal{X}_t \leftarrow \mathcal{X}_{DC,t}$ in (3-1). Still, it is important to notice that, while this model approximates the feasible region of the AC power flow, it might not include the entire feasible region. Hence, the global optimum of the AC optimal power flow problem can be lost. It is worth mentioning that this approximation provides neither an optimistic view of the problem (like the convex relaxations) nor a pessimistic one. Empirically, it is a useful model when trying to estimate the operating costs of well balanced systems complying with the aforementioned assumptions that give rise to this model.

In order to better approximate the power flow equations, some extensions to the DC formulation were developed to approximate line losses [42]. One approximation is the *quadratic DC line loss approximation*, hereinafter referred to as DCLL. This approximation can be constructed by first adding an equivalent constraint to (3-8b): $f_{(n,m)t} + f_{(m,n)t} = 0$. This constraint, which defines power-flow loss to be equal to zero, is then changed to comprise positive losses. To do that, the equality is replaced with an inequality, i.e., $f_{(n,m)t} + f_{(m,n)t} \geq$ line losses[1]. Since power-flow losses can be defined as proportional to the quadratic value of the current on a line, and current is not available in this formulation, the idea behind this approximation is to use the power-flow as a proxy for the current. The resulting formulation is as follows:

$$\mathcal{X}_{DCLL,t} = \left\{ p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t \,\middle|\, \exists\, \theta_t : \right.$$

$$f_{(n,m)t} = -\left(B_{(n,m)t}\right)(\theta_{nt} - \theta_{mt}) \;\; \forall (n,m) \in \mathcal{L} \tag{3-10a}$$

$$\ell_{nt} = 0 \;\; \forall n \in \mathcal{N} \tag{3-10b}$$

$$\left. f_{(n,m)t} + f_{(m,n)t} \geq \frac{G_{(n,m)t}}{G^2_{(n,m)t} + B^2_{(n,m)t}} f^2_{(n,m)t} \;\; \forall (n,m) \in \mathcal{L} \right\}, \tag{3-10c}$$

A better explanation of the quadratic approximation of the line loss for

[1] This modeling choice creates the possibility of free disposal of energy, as the losses are not equal to the flow difference, but only smaller than or equal to. However, in general, the least-cost dispatch found by deterministic single-period models without ramp constraints are such that losses are minimized. In the literature, some approaches are proposed to avoid this issue. This and other aspects of the DCLL approximation are discussed in [43]. However, it is beyond the scope of this work to explore these variants. Notwithstanding, it is relevant to mention that this is a network approximation that may be infeasible as all other approximations studied in this work. The quality of different policies based on the studied approximations will be tested in the implementation step based on the accurate AC power flow model.

the DCLL approximation can be found in [42].

The *PowerModels.jl* package, a framework for steady-state power network optimization, comprises all these and other mathematical programming formulations of the OPF problem. Since *PowerModels.jl* uses *JuMP.jl* to build these formulations, they can be passed to a variety of solvers and inherit all the features of this mathematical language. *PowerModel.jl* allows the user to easily choose an approximation or relaxation, solve it, and consequently discuss and compare the impacts of using different relaxations and approximations in the economic dispatch problem.

3.3 Assessing the time inconsistency gap

Simplifying the OPF network formulation is still a necessary step to achieve a tractable formulation for the hydrothermal dispatch problem compatible with efficient solution methods such as SDDP. Planning agents use simplified models to find the best reservoir levels in medium- and long-term horizons by computing the cost-to-go functions. For implementing operating decisions, however, ISOs seek feasible dispatches complying with all the details of the network. This is done by coupling the simplified view of the future operation of the system, implicitly considered in the cost-to-go function, with OPF models with more realistic descriptions of the real network. Unfortunately, in practice, the simplifications considered in the network model to evaluate cost-to-go functions are significantly optimistic in comparison to the representation needed to ensure feasibility. This optimistic bias leads to the implementation of expensive (sub-optimal) time-inconsistent policies [3].

More specifically, in the case of Brazil and Chile, for instance, ISOs converge the SDDP algorithm using the transportation NFA model for the network and couple the second-stage cost-to-go function into a more detailed model with a more accurate representation of the system. Then, in the next period, the state is updated with the actual reservoir levels and the same process is repeated. This rolling-horizon operating scheme produces a hybrid time-inconsistent policy in which implemented decisions deviate from those obtained in the planning stage embedded in the cost-to-go function. Hence, this hybrid and inconsistent implemented policy delivers decisions that are not optimal, either for the planning problem or for the true optimal policy fully based on the detailed model. Additionally, this sub-optimal policy has detrimental effects on the operation of the system. As reported in [3, 10], high operating costs as well as highly volatile thermal generation and distorted spot prices are observed.

To evaluate the performance of the time-inconsistent policies induced by the five approximations for the AC power-flow model presented in Chapter 3, we follow the framework proposed in [3]. According to the previously described decision process, two different models for the network are used, one for the planning stage (used to obtain the recourse function) and another for implementing decisions. The difference between the planning model and the implementation model in this work is concentrated in $\mathcal{X}_t$. The set used in the planning phase is denoted as $\mathcal{X}_t^{plan}$, while the one used in the implementation step is denoted as $\mathcal{X}_t^{imp}$. However, the only result of the planning stage that is actually used in the implementation phase of period t is the second-stage cost-to-go function, $\mathcal{Q}_{t+1}^{plan}$.

The process that simulates M scenarios (complete temporal paths), $\{\mathbf{x}_{t,s}^{(plan,imp)}\}_{t=1,s=1}^{T,M}$, of the implemented inconsistent policy $\mathcal{P}\Big(\{\mathcal{Q}_t^{plan}\}_{t=1}^{T}, \{\mathcal{X}_t^{imp}\}_{t=1}^{T}, \{\omega_{t,s}\}_{t=1,s=1}^{T,M}\Big)$ is summarised in Algorithm 2.

Algorithm 2: Simulation of the rolling horizon process

```
Result: ${\mathbf{x}_{t,s}^{(plan,imp)}}_{t=1,s=1}^{T,M}$
for $s = 1$ to $M$ do
    Set initial conditions to $\mathbf{x}_{0,s}^{(plan,imp)}$ with $\nu_0$
    Sample inflow time series $\{\omega_{t,s}\}_{t=1}^{T}$
    $t \leftarrow 1$
    while $t \leq T$ do
        Converge SDDP with initial condition $\nu_{t-1,s}$ of $\mathbf{x}_{t-1,s}^{(plan,imp)}$,
          $\mathcal{X}_t \leftarrow \mathcal{X}_t^{plan}$, and store $\mathcal{Q}_{t+1}^{plan}$
        Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1,s}^{(plan,imp)}, \omega_{t,s})$, using
          $\mathcal{Q}_{t+1}^{plan}$ and $\mathcal{X}_t^{imp}$, and store $\mathbf{x}_{t,s}^{(plan,imp)}$.
        Update $\mathcal{P}$ with $\mathbf{x}_{t,s}^{(plan,imp)}$ and initial conditions for the next
          stage, $\nu_{t,s}$.
        $t \leftarrow t + 1$
    end
end
```

Thus, we can estimate the expected cost of a policy using set $\mathcal{X}_t^{plan}$ in the planning phase and set $\mathcal{X}_t^{imp}$ in the implementation model by means of the following expression:

$$\bar{C}^{(plan,imp)} = \frac{1}{M} \sum_{s=1}^{M} \sum_{t=1}^{T} \Big(\sum_{i \in \mathcal{I}} C_{it}\, p_{its}^{(plan,imp)} + \sum_{n \in \mathcal{N}} C_{nt}^{\delta}\, \delta_{nts}^{(plan,imp)} + \sum_{j \in \mathcal{H}} C_{jt}^{s}\, s_{jts}^{(plan,imp)} \Big). \tag{3-11}$$

Recall that $\mathbf{x}_t = [p_t, f_t, u_t, s_t, \nu_t, \delta_t, \ell_t]^T$ and in (3-11) we identify the components of $\mathbf{x}_{t,s}^{(plan,imp)}$ with the same superscript, $(plan, imp)$.

Notwithstanding, the procedure summarised in Algorithm 2 is very time consuming and imposes a significant computational burden through the series of planning and implementation steps in this rolling-horizon fashion. To avoid that while simulating inconsistent policies, we use the *Fast Algorithm* also proposed in [3]. This algorithm alternates the network representations in the forward and backward passes of a single SDDP run. While the backward pass improves the cost-to-go functions that would be obtained in the planning phases (with $\mathcal{X}_t \leftarrow \mathcal{X}_t^{plan}$), the forward pass emulates the implementation process and samples new states considering the more detailed implementation model (with $\mathcal{X}_t \leftarrow \mathcal{X}_t^{imp}$). The result of this fast algorithm is a set of cost-to-go functions well approximated on the relevant points that would be visited in the implementation phase. Therefore, a final out-of-sample simulation to obtain a representative sample of decisions, $\{\mathbf{x}_{t,s}^{(plan,imp)}\}_{t=1,s=1}^{T,M}$, of policy $\mathcal{P}\Big(\{\mathcal{Q}_t^{plan}\}_{t=1}^{T}, \{\mathcal{X}_t^{imp}\}_{t=1}^{T}, \{\omega_{t,s}\}_{t=1,s=1}^{T,M}\Big)$ can be carried out using the cost-to-go functions of this fast algorithm.

It is important to mention that the aforementioned fast algorithm is a modified version of the SDDP algorithm that can not rely on the standard stopping criterion based on upper and lower bound convergence [2]. This is because the lower bound is obtained with a different, in general, relaxed model of the system in comparison to the model used in the upper bound assessment. Therefore, in [3], a new stopping criterion, based on the lower bound and upper-bound moment stabilization is proposed. In this algorithm, an evaluation process is carried out for specific iterations, namely, $\mathcal{K}$, which are specified by the user. For instance, in [3,5] and [10], $\mathcal{K} = \{1000, 1100, 1200, ...\}$. The fast algorithm proposed in [3] is summarised in this work as Algorithm 3. To ease the notation, the Kth approximation of the cost-to-go function is denoted as $\{\mathcal{Q}_t^{plan,K}\}_{t=1}^{T}$. We also denote $K^{prev} = max\{k \in \mathcal{K} \mid k < K\}$ to refer to the iteration wherein the previous evaluation process was performed.

Algorithms 2 and 3 are general algorithms to simulate a policy based on different network models, $\mathcal{X}_t^{plan}$ for the planning phase and $\mathcal{X}_t^{imp}$ for implementation step, while expression (3-11) provides an estimation of the cost of this policy. To quantify the time-inconsistency gap however, we compare the cost of the inconsistent policy, $\mathcal{P}\Big(\{\mathcal{Q}_t^{plan}\}_{t=1}^{T}, \{\mathcal{X}_t^{imp}\}_{t=1}^{T}, \{\omega_{t,s}\}_{t=1,s=1}^{T,M}\Big)$, with the cost of the respective reference planning policy[2], $\mathcal{P}\Big(\{\mathcal{Q}_t^{plan}\}_{t=1}^{T}, \{\mathcal{X}_t^{plan}\}_{t=1}^{T}, \{\omega_{t,s}\}_{t=1,s=1}^{T,M}\Big)$. Hence, the

[2] It is worth emphasizing that the results of planning policies might be infeasible in practice. Therefore, the planning policies relying on strong simplification assumptions are only useful for allowing ISOs and planners to estimate the opportunity cost of the water (or other scarce resources) and for comparison purposes. In this sense, one should not use planning results in practical studies for regulatory or investment assessments. The

Algorithm 3: Approximating rolling horizon process with a fast algorithm

```
Result: $\{\mathbf{x}_{t,s}^{(plan,imp)}\}_{t=1,s=1}^{T,M}$
$K \leftarrow 0$, $converged \leftarrow False$, $\mathcal{P} \leftarrow \emptyset$
while not converged do
    Initialize $\mathbf{x}_0^K$ with $\nu_0$.
    /* Evaluation step and Convergence Test                    */
    if $K \in \mathcal{K}$ then
        Initialize $\mathbf{x}_0^{eval,K}$ with $\nu_0$.
        Sample $M$ inflow time series $\{\omega_{t,s}^{eval}\}_{t=1}^{T}$
        for $t = 1$ to $T$ and $s = 1$ to $M$ do
            Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1}^{eval,K}, \omega_{t,s}^{(eval)})$, using
              $\mathcal{Q}_{t+1}^{plan,K}$ and $\mathcal{X}_t^{imp}$, and store $\mathbf{x}_{t,s}^{eval,K}$.
        end
        Calculate $\bar{C}_K^{(plan,imp)}$ defined in (3-11) with $\mathbf{x}_{t,s}^{eval,K}$.
        if $K > \min\{\mathcal{K}\}$ then
            if $\frac{LB^K}{LB^{K^{prev}}} \leq 1.01$ & T-test($\bar{C}_K^{(plan,imp)}$,$\bar{C}_{K^{prev}}^{(plan,imp)}$) then
                converged = True
            end
        end
    end
    /* Forward Pass                                             */
    Sample an inflow time series $\{\omega_t\}_{t=1}^{T}$
    for $t = 1$ to $T$ do
        Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1}^K, \omega_t)$, using $\mathcal{Q}_{t+1}^{plan,K}$
          and $\mathcal{X}_t^{imp}$, and store $\mathbf{x}_t^K$.
    end
    /* Backward Pass                                            */
    $\{\mathcal{Q}^{plan,K+1}\}_{t=1}^{T} \leftarrow \{\mathcal{Q}^{plan,K}\}_{t=1}^{T}$
    for $t = T$ to $2$ do
        for $\omega_t \in \Omega_t$ do
            Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1}^K, \omega_t)$, using
              $\mathcal{Q}_{t+1}^{plan,K+1}$ and $\mathcal{X}_t^{plan}$
        end
        , and store the objective value and dual vector for the
          backward scenario $\omega_t$.
    end
    Calculate new Bender's cut.
    Update the approximation of the cost-to-go function with the
      new Bender's cut.
end
$LB^K \leftarrow$ objective value of the first stage problem.
$K \leftarrow K + 1$
/* Final Simulation Step                                        */
Sample $M$ inflow time series $\{\omega_{t,s}\}_{t=1}^{T}$
for $t = 1$ to $T$ and $s = 1$ to $M$ do
    Solve sub-problem for period $t$ with $(\mathbf{x}_{t-1}^{(plan,imp)}, \omega_{t,s})$, using
      $\mathcal{Q}_{t+1}^{plan,K}$ and $\mathcal{X}_t^{imp}$, and update $\mathcal{P}$ with $\mathbf{x}_{t,s}^{(plan,imp)}$.
end
```

time-inconsistency gap measures the hidden cost of neglecting constraints of the implementation problem in the planning phase. It can be seen as the operator's regret with respect to its planning expectations. More importantly, it allows ISOs to detect and quantify the impact of the inconsistencies induced by a given simplification without the need of simulating the full, and possibly currently intractable, policy based on the more complex network model. The time-inconsistency gap proposed in [3] is adapted to the notation of this work as follows:

$$GAP = \bar{C}^{(plan,imp)} - \bar{C}^{(plan,plan)} \tag{3-12}$$

3.4 Open-source Julia package: HydroPowerModels.jl

Solving a hydrothermal dispatch problem depends on the complex and not always easily available SDDP algorithm. A number of programs implementing SDDP are in-use around the world, ranging from unpublished implementations in academic institutions to professional software such as the product developed by PSR, a software and consulting company, also called *SDDP*[3] [44]. However, until recently, there was no fast, reliable, and open-source implementation of the SDDP algorithm. Without such a tool, researchers and practitioners have not had a common ground for the discussion and analysis of different hydrothermal dispatch formulations and their solutions.

One of the contributions of this work is an open-source tool, called *HydroPowerModels.jl*, that can be this common ground. *HydroPowerModels.jl* can be used to assess the impact of modeling choices during the planning of a hydrothermal power system. These choices include the usage of different network formulations, the consideration of different risk measures, and the planning horizons for uncertain future costs. Addressing these issues provides the research community and the energy industry with a powerful tool for the efficient design of hydrothermal power systems.

To develop a tool that can be used by both researchers and industry professionals, we take advantage of the Julia language [15] and two main packages: *PowerModels.jl* [16], which implements power flow models for electrical dispatch, *SDDP.jl* [18], which implements the stochastic dual dynamic programming algorithm. Both *PowerModels.jl* and *SDDP.jl* handle their respective

inconsistent policy, derived as a more realistic *chained planning-and-implementation process* (as per Algorithm 2 or approximation schemes such as Algorithm 3) offers more accurate estimates for future decisions than the related planning policy.

[3] Note that the "SDDP" acronym is used to denote a software product, the original algorithm of [2], and a more general class of algorithms inspired by the original SDDP algorithm.

optimization models through *JuMP.jl* [20], a Julia package for mathematical optimization. *JuMP.jl* makes it simple to write optimization problems and solve them with numerous solvers. *HydroPowerModels.jl* takes advantage from the fact that *PowerModels.jl* and *SDDP.jl* were not only developed in Julia, but also deeply rely on *JuMP.jl* to build and solve mathematical optimization problems. *SDDP.jl* is used to specify the hydro storage dynamics and stochastics of inflows, renewables and loads. *PowerModels.jl* is used to provide multiple network dispatch formulations as a starting point for the *HydroPowerModels.jl* formulation that couples the electrical constraints with hydro constraints and uncertainty.

HydroPowerModels.jl uses the *PowerModels.jl* and *SDDP.jl* packages to implement and solve different hydrothermal dispatch formulations. It provides an interface to easily solve and simulate hydrothermal dispatch models and allows the creation of a collection of hydrothermal problems described in input files for the package (following the *PowerModels.jl* standard), thereby helping the discussion of methodology and the resulting policies for specific case instances.

In contrast to the previously available commercial software for hydrothermal dispatch models, the proposed package is part of an academic open-source effort. This helps to promote the continuous improvement of models and solution algorithms for the research community.

Other academic implementations of SDDP have been developed and may be applied to the hydrothermal dispatch problem. However, the advances of the Julia Language and *JuMP.jl* are a much more adequate framework than those of MATLAB [45] or Python [46, 47], with better memory management and better execution time. Moreover, a free and open-source tool can be of great help for the research community alternatively to commercial solvers as [48] and the renowned version from *PSR Inc.*

More implementations of SDDP are also available in Julia [49] [50], but *SDDP.jl* [18] has proven an easy to learn, efficient version of SDDP that is flexible enough for the purposes of the *HydroPowerModels.jl* package.

HydroPowerModels.jl is composed of different and useful functionalities, from compact case sharing to dispatch solution results visualizations. A workflow of simple usage of the package helps to give a basic overview:

- Load case data from input files describing: Power network data; Reservoir facilities details and water network data; Inflow scenarios.
- Set case parameters: Power network formulation; Number of stages; Number of hours in Stage and optimizer to solve the sub-problems.

- Build the multistage, hydrothermal steady-state power network optimization problem.
- Run SDDP method to approximate the cost-to-go functions.
- Simulate the policy.

A code example is presented in Appendix B to help clarify the usage of the package. Although, for a more extensive tutorial of the package, detailed documentation is made available in `https://andrewrosemberg.github.io/HydroPowerModels.jl/latest/`.

In the next chapter, we will simulate and assess the inconsistency cost for some case studies using the *HydroPowerModels.jl* we just presented.

4
Case Studies

In this chapter, the quality of the five approximations for the network constraints under study are compared. The time inconsistency gap, (3-12), and other relevant operative indexes such as reservoir levels, thermal generation, and spot prices are studied to provide a more in-depth understanding of the differences among them.

The inconsistent policies will be denoted according to the pair of network models used in the planning and implementation phases, respectively. In this work, all policies are evaluated under the same basis, with the most accurate network formulation in the implementation model, namely, the AC power-flow model (3-2). Thus, for instance, the policy that uses in the planning phase the transportation NFA model is named **NFA-AC inconsistent policy**. Following this idea, in this work we compare the following policies: **NFA-AC inconsistent policy**; **SOC-AC inconsistent policy**, **SDP-AC inconsistent policy**, **DC-AC inconsistent policy**, and **DCLL-AC inconsistent policy**. Finally, to estimate the time-inconsistency gap, we will also assess the cost for the planning policies, namely, the **NFA planning policy**, **SOC planning policy**, the **SDP planning policy**, the **DC planning policy**, and the **DCLL planning policy**, each of which relying solely on their respective relaxations.

4.0.1
Simple 3 Bus Case

As a case study, we use the three-bus system from [3] to illustrate the effects of the underlying policies. Figure 4.1 shows the system network of this case. Inflow scenarios are generated in the same manner as in [3]. The planning horizon is 48 periods and the number of hours at each stage is 730 (one month). We denote this case as "3-Bus Case".

In this system, there are two thermal generators and a hydro plant. Generators $G1$ and $G2$ have, respectively, maximum capacity of 100 MWh and 50 MWh, and costs \$20/MWh and \$100/MWh. The hydroelectric unit has no cost, maximum dispatch (U) of 80 MWh, maximum volume ($\overline{\nu}$) of 150 MWh and 50 MWh of initial volume (ν_0). The production factor ρ is 1 (MWh/

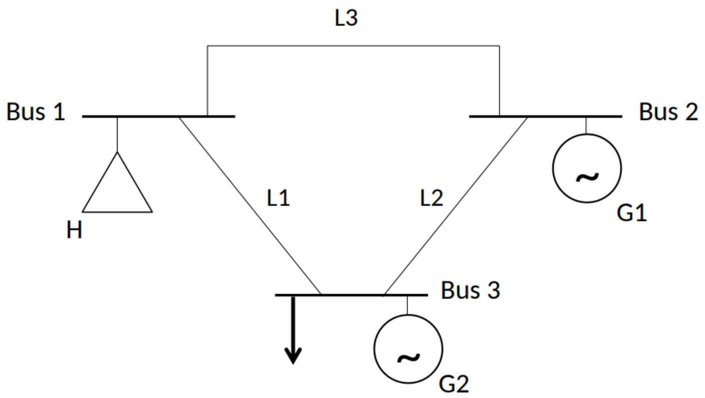

Figure 4.1: Network Grid graph

(m^3)) and there is no spillage cost (C^s). Consider a single constant demand (D) at bus 3 to be equal 100 MW, lower and upper voltage limits ($\underline{V}$ and $\overline{V}$) to be the same at all buses and equal to 0.9 p.u. and 1.1 p.u. respectively, and cost of deficit (C^δ) to be equal \$1000/MWh. As in the original case, we set reactances (X) of transmission lines $L1$, $L2$, and $L3$ to be equal to 1 p.u., 0.5 p.u. and 1 p.u., respectively, assuming nominal values of 100 MVA and 138 kV. Resistances (R) are set to 0.065 p.u., 0.025 p.u., and 0.042 p.u., respectively. The absolute angle limit ($\underline{\Delta}^\theta$ and $\overline{\Delta}^\theta$) for each line is 1.0472 (rad). Maximum power flows (F) are considered as 100 MW, 65 MW, and 25 MW for lines 1 to 3, respectively.

Table 4.1 shows the following information: column 1 depicts the names of the inconsistent policies as per their models (planning and implementation); column 2 presents the cost of the related planning policy; column 3 presents the inconsistent implemented policy; columns 4 and 5 present the time-inconsistency GAP in both absolute and relative units, respectively; column 6 provides the P-value for T-test for difference of means applied to the GAP (with null hypothesis that the $GAP = 0$); and finally, column 7 shows the computing times of Algorithm 3 disregarding the final simulation step time.

Policy	Planning	Implementation	GAP	GAP	P-value	Time
(plan,imp)	(10^6\$)	(10^6\$)	(10^6\$)	(%)		(min.)
NFA-AC	43.2317	54.7234	11.4917	26.4	$< 10^{-6}$	51.4
SOC-AC	45.0477	54.9197	9.87199	21.7	$< 10^{-6}$	93.8
SDP-AC	45.4524	48.2853	2.83291	6.1	$< 10^{-6}$	130.9
DC-AC	43.3940	48.8891	5.49508	12.5	$< 10^{-6}$	54.9
DCLL-AC	45.4158	48.0700	2.6500	5.68	$< 10^{-6}$	32.18

Table 4.1: 3-Bus Case: Summary of results for planning and implementation policies.

The first important remark from table 4.1 is that the GAP increases with the simplification (relaxation) level. This indicates that the more optimistic one is in the planning phase, the higher will be the regret, i.e., the deviation of the implementation cost with regard to the planning one. Note that as DC-based policies are not relaxations of neither the SOC nor the SDP based policies. For that end, the relaxations are the following: NFA is a relaxation of all the others, SOC is a relaxation of the SDP, and DC is a relaxation of the DCLL.

In this table, we also see that the **NFA-AC inconsistent policy**, having the simplest planning model, also features the lowest computational burden, except for the DCLL, which is the main reason why the NFA simplification is widely adopted in practical studies [4]. This simplification induces high operative costs in the implementation phase when a AC power flow is used to simulate the policy.

The **SOC-AC inconsistent policy** uses a nonlinear convex relaxation representing losses but features the second-highest GAP. Despite convex, this formulation is hard to solve and exhibits the second higher computational burden. Additionally, in this specific 3-Bus case, where the system is composed of a single cycle, the SOC relaxation is not tight and exhibits the lowest performance in the implementation phase. This happens because the SOC relaxation loses precision exactly on the presence of cycles[1]. Finally, it is relevant to note that the SOC-AC policy performs worse than the NFA-AC policy in the implementation phase, despite having a higher cost at the planning phase. This can be explained by the examination of the cost-to-go function produced by these two relaxations of the AC power flow model. Note that despite the difference between the cost-to-go functions produced by the NFA and SOC models (see Figure 4.2), which is due to the cost of losses not represented in the NFA model, the derivative of them are quite close to each other (see Figure 4.3). It is relevant to highlight that the NFA model provides slightly lower derivatives than the SOC, which produces slightly higher opportunity costs (water values) for the implementation model. This explains the lower cost of the NFA-AC policy in comparison to the SOC-AC in the implementation phase. It also suggests that the SOC approximation might exhibit very poor performances in the presence of meshed grids.

The **SDP-AC inconsistent policy** has the tightest convex relaxation and the second-lowest GAP and cost in the implementation phase. It is relevant to highlight that the implementation cost of this policy is not higher than 0.5% of the policy featuring the lowest implementation cost (DCLL-AC). This

[1] It is well-known that under mild conditions, this relaxation is tight for radial systems, which is not the case of this 3-Bus system [51].

benefit comes with the cost of the highest computational burden. As will be further shown, this model proves to be the most coherent and consistent in terms of operating results.

The **DC-AC inconsistent policy** performs better than the SOC-AC and worse than the SDP in terms of expected implementation cost. It is relevant to note that the DC-AC policy is a competitive policy as it produces an implementation cost less than 2% higher than the DCLL-AC policy, the cheapest one. Although it does not model transmission losses, as per Figure 4.3, it is capable of capturing the relevant operative constraints imposed by KVL constraints and passing opportunity costs to the implementation model similar to those passed by the SDP model, but under a lower computational burden. Therefore, this model is competitive and can be seen as an alternative to the nonlinear SOC and SDP formulations in the presence of meshed systems with low loss levels.

The **DCLL-AC inconsistent policy** results indicate the best performance in terms of implementation cost, GAP, and computing times. It puts together the DC model capability of approximating KVL constraints and a reasonable description of the transmission losses. These results demonstrate a quadratic approximation can perform better than the nonlinear SDP relaxation. Additionally, the DCLL quadratic formulation allows solving the problem much faster than the highly computationally intensive SDP formulation, thereby representing an interesting alternative for the SDP approximation in the presence of meshed grids with non-negligible losses.

To further analyze these results, the expected storage is depicted in Figure 4.4 and the expected spot prices in Figure 4.5 for all analyzed policies. This analysis is done in the light of what we observe in the cost-to-go functions in Figure 4.2 and its first derivatives in Figure 4.3. The solid lines in Figures 4.2 and 4.3 are the output of running the normal SDDP algorithm using the simplified model in both forward and backward step, and the dotted lines are the output of the Fast Algorithm where the detailed modeled is used to choose the trial values. In Figure 4.4, notice that the **NFA planning policy** is aggressive in the use of the water and has its reservoir drained when storage is not needed (within this modeling representation). In the first stage, the amount of water used from the reservoir is the difference between the dispatch capacity and the inflow. Its inconsistent counterpart, **NFA-AC inconsistent policy**, has the same simplified view of the future, as seen in 4.2 and 4.3, that ignores the KVL and would also like to dispatch all of its capacity. However, in the implementation phase, it is constrained by the electric constraint when operating the system. Thus, the remaining unused water from the inflow is

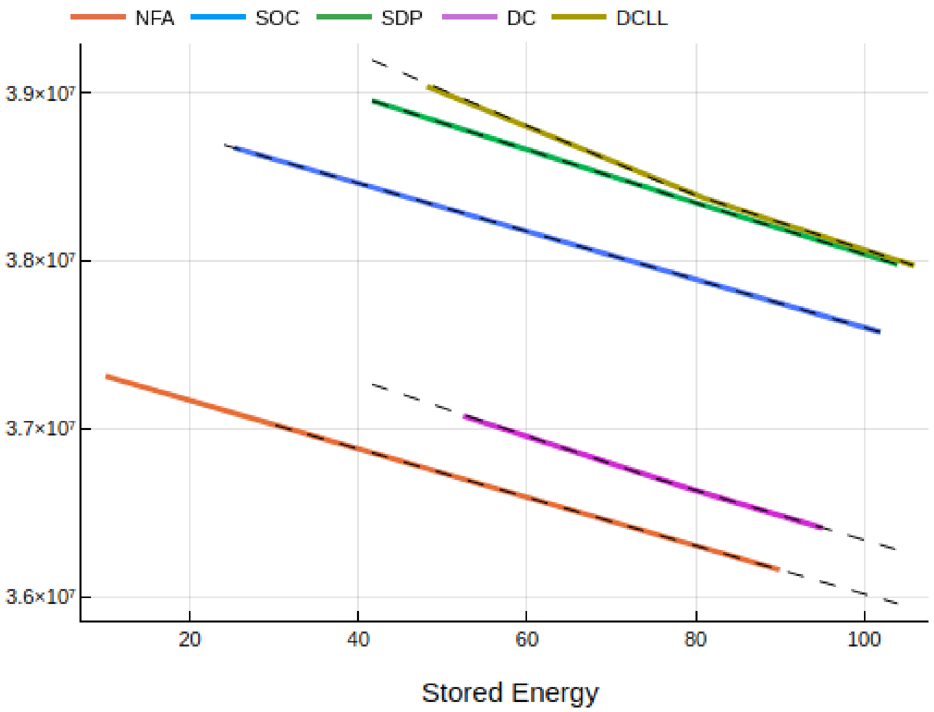

Figure 4.2: Stage one approximated cost-to-go functions for the different planning formulations.

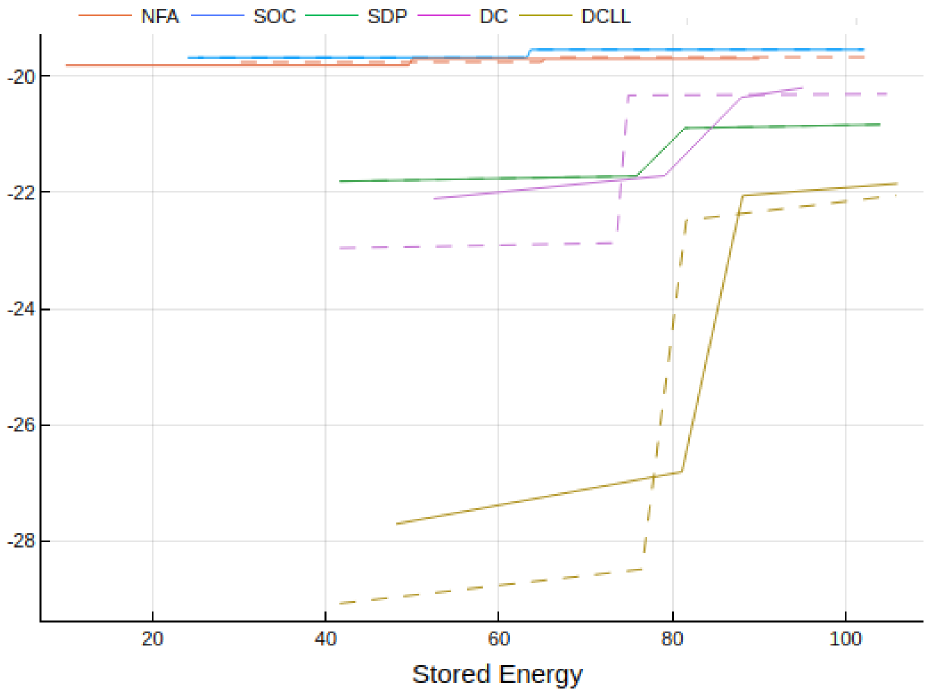

Figure 4.3: Stage one first derivative of the approximated cost-to-go functions.

stored, explaining why the reservoir level rises.

In 4.2 and 4.3, we also observe that the **DC planning policy** and the **SDP planning policy** are aware of the operative difficulties of future stages caused by electric constraints and, consequently, save the water as shown in 4.4. The **DCLL planning policy** is even more pessimistic and stores more energy for future stages. Similarly to the **NFA-AC inconsistent policy**, the **SOC planning policy** although representing some nonlinearities of the AC model such as transmission losses, also drains its reservoir in both planning and implementation phases. It is noticeable however that the discrepancy between the planning and implementation levels is significantly reduced in comparison to the NFA-based policies. As previously explained, in the presence of cycles, this formulation fails to represent the KVL constraints and therefore generates optimistic and infeasible dispatches that need to be compensated with expensive thermal dispatches in the implementation phase. As a consequence, both NFA-AC and SOC-AC policies exhibit highly volatile and distorted spot-price profiles with high peaks and valleys as depicted in Figure 4.5.

In summary, **SOC planning policy**, **DC planning policy**, **DCLL planning policy**, and **SDP planning policy** see an incrementally more accurate approximation of the AC electric constraints in future stages, thus saving water for the adverse future, as shown in Figure 4.4. Yet, the inconsistent implementation versions of these policies save less water since their AC reality is relatively harder than the simplified view provided by the approximations. Moreover, as we see in Figure 4.5, the marginal cost profiles, and thereby the dispatches, are still different in the implementation phase in comparison to the planning phase. This happens even for SDP-based policies, wherein the storage management is nearly the same on average as shown in Figure 4.4.

According to Figure 4.5, significant structural differences are found in the average dispatch between the respective planning and implementation policies. This stems from the fact that additional and expensive dispatches are needed to compensate the optimistic view of the approximations in critical states (low reservoir levels). In the same Figure 4.5, the **NFA planning policy** presents the lowest prices, as expected for an infeasible relaxed problem, which uses water resources as if no electrical constraints exist. The **SOC planning policy** has the second-lowest price given that it does not provide accurate representations of the network in the presence of cycles. In the sequel, the **DC planning policy**, **SDP planning policy**, and **DCLL planning policy** still provide a simplified version of the true network, albeit their representation are capable of considerably reducing the spot-price spikes. This better behaved spot-price profiles stems from the better representation of the electrical constraints, which although more expensive in the planning phase, allows the system to achieve better states in the implementation phase. Furthermore, the more inconsistent is the policy, the higher are the spot-price

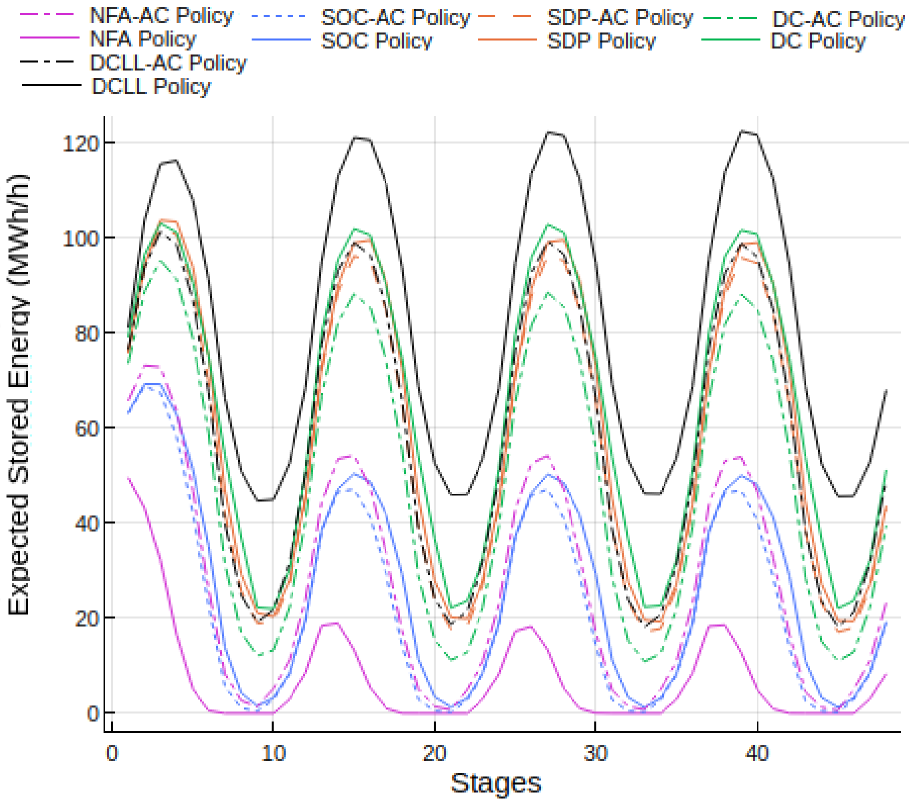

Figure 4.4: Expected Reservoir levels with time inconsistency due to simplifications in transmission-line modeling.

spikes. This can be seen by comparing the **SDP-AC inconsistent policy**, the **SOC-AC inconsistent policy**, and the **NFA-AC inconsistent policy** which, in this order, incrementally relax the electric constraints in the planning stage. Notice that the **SOC planning policy** and the **SOC-AC inconsistent policy** differ here since the electrical operation provided by the planning policy is infeasible even though it has found an implementable storage management schedule in the average. In this case, the risk of infeasible states is high and when the quantiles are analysed, significant differences are found.

4.0.2
5 Bus Case: Brazil With a Reduced Equivalent Network Grid and Reservoirs

In order to further analyse the impacts of time inconsistency due to network formulations we now use a reduced model equivalent to the Brazilian national grid and reservoir system. This case is based on [1, 3, 52].

In this model, the water reservoirs have been aggregated into four separate storage units located in different subsystems. These subsystems are called Southeast (SE), South (S), Northeast (NE), and North (N) subsystems and are represented as nodes in our equivalent model.

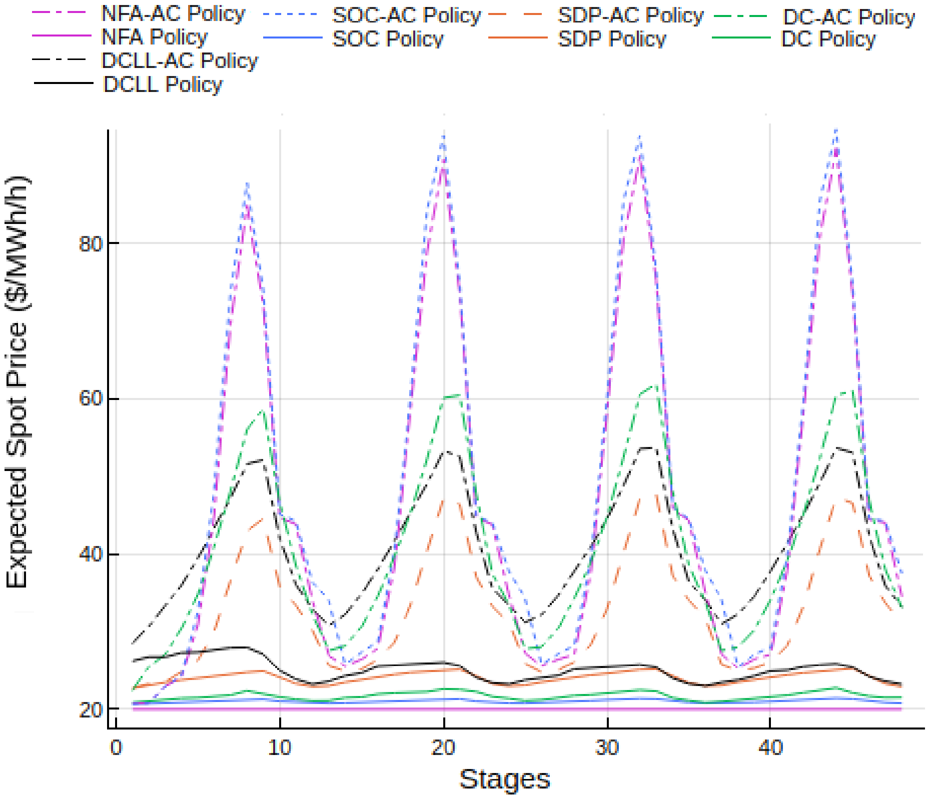

Figure 4.5: Expected Spot Price with time inconsistency due to simplifications in transmission-line modeling.

We use 25 inflow scenarios derived from monthly historical data for 25 years between 1931 and 2013 in Brazil, each having equal probability. The planning horizon and length of scenarios spans 84 months (7 years).

Besides this hydro-generation capacity, there are 95 thermal generators distributed across the system. Moreover, the subsystems are interconnected by multiple branches which were aggregated into five interfaces and a transshipment node (called "Imperatriz"). The layout of the grid is presented in Figure 4.6.

Table 4.2 shows the following information: column 1 depicts the names of the inconsistent policies as per their models (planning and implementation); column 2 presents the cost of the related planning policy; column 3 presents the inconsistent implemented policy; columns 4 and 5 present the time-inconsistency GAP in both absolute and relative units, respectively; column 6 provides the P-value for T-test for difference of means applied to the GAP (with null hypothesis that the $GAP = 0$); and finally, column 7 shows the computing times of Algorithm 3 disregarding the final simulation step time.

The first important remark from 4.2 is that the **NFA-AC inconsistent policy**, having the simplest planning model, features the lowest computational burden for this case. It is clear that, as the system grows and its optimal power

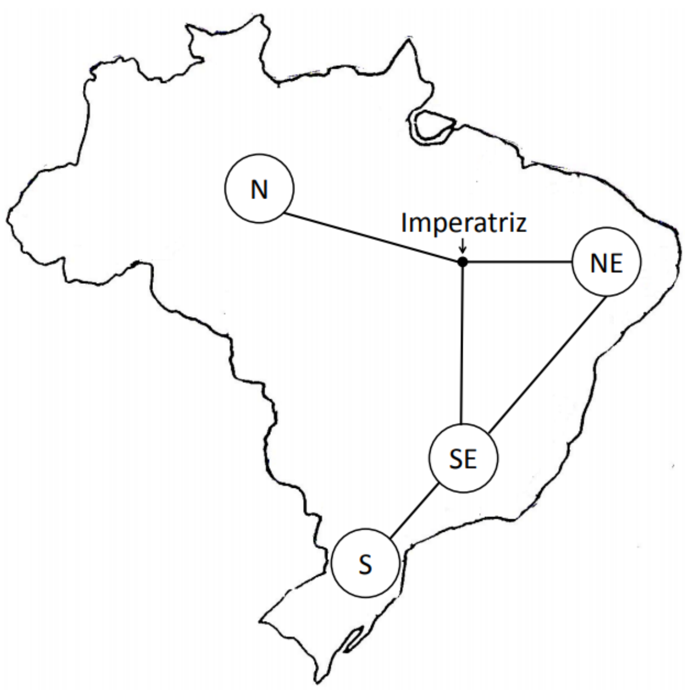

Figure 4.6: Brazilian Energy Subsystems Schematics. Source: [1]

Policy (plan,imp)	Planning (10^6\$)	Implementation (10^6\$)	GAP (10^6\$)	GAP (%)	P-value	Time (min.)
NFA-AC	$11,685.0$	$83,311.4$	$71,626.4$	870.952	$< 10^{-6}$	189.052
SOC-AC	$46,463.3$	$46,383.3$	-80.04	-0.17	0.999	477.415
SDP-AC	$46,427.0$	$46,374.0$	-52.98	-0.091	0.984	930.256
DC-AC	$45,221.1$	$46,545.1$	$1,324.02$	3.0	$< 10^{-6}$	328.282
DCLL-AC	$46,281.5$	$46,338.5$	56.98	0.23	0.0142	468.194

Table 4.2: Brazil: Inconsistent and Planning Policy Cost Comparison (10^6\$).

flow solution becomes more complex, no other simplification is able to compete with the NFA relaxation in terms of computational cost. This simplification induces high operative costs in the implementation phase when a AC power flow is used to simulate the policy and the **NFA-AC inconsistent policy** has an even worst result than in the small 3-bus case.

In this table, we also see that the convex relaxations (SOC and SDP) produce optimal policies and result in GAPs with mean indistinguishable from zero (as shown by the P-value from the t-test). These relaxations are tight for radial systems [51] and systems where cycles don't impact much on the dispatch, such as this Brazilian reduced system. Once again, the **SDP-AC inconsistent policy** presents the highest computational burden and, the **SOC-AC inconsistent policy** with almost half the computational effort to train the policy, the second highest.

The **DC-AC inconsistent policy** performs better than the NFA-AC and worse than the convex relaxations (SOC and SDP) in terms of GAP and expected implementation cost. It is relevant to note that the DC-AC policy is a competitive policy as it produces an implementation cost less than 1% higher than the DCLL-AC policy, the cheapest one, and a GAP of only 3%. Having significantly less solving effort than the convex relaxation policies, this simplification provides an acceptable choice for practical applications.

The **DCLL-AC inconsistent policy** results indicate the best performance in terms of implementation cost. This policy presents the 3rd lowest GAP and computing time. It puts together the DC model capability of approximating KVL constraints and a reasonable description of the transmission losses. These results demonstrate a quadratic approximation can have close performance to the nonlinear SDP relaxation even in more radial systems. Additionally, the DCLL quadratic formulation allows solving the problem much faster than the highly computationally intensive SDP formulation and slightly faster than the SOC formulation.

To further analyze these results, the expected storage is depicted in Figure 4.7, the expected thermal generation in Figure 4.8 and the expected spot prices in Figure 4.9 for all analyzed policies. In Figure 4.7, we see the **NFA planning policy** is capable of maintaining the highest reservoir volume even though it uses more water, but, when the electric constraints are imposed in **NFA-AC inconsistent policy**, this simplified planning results in lower levels than the other policies. The consequence is a state that incrementally forces higher thermal dispatches as we can see in Figure 4.8 and, thus, higher nodal prices shown in Figure 4.9.

In the same figures, we can see that the other simplifications have very similar dispatches between each other, both in the planning simulations and in the implementation ones. These formulations present sufficiently good electric constraints approximations, and sufficiently good power flow loss approximations for the low loss system without significant cycles in this case.

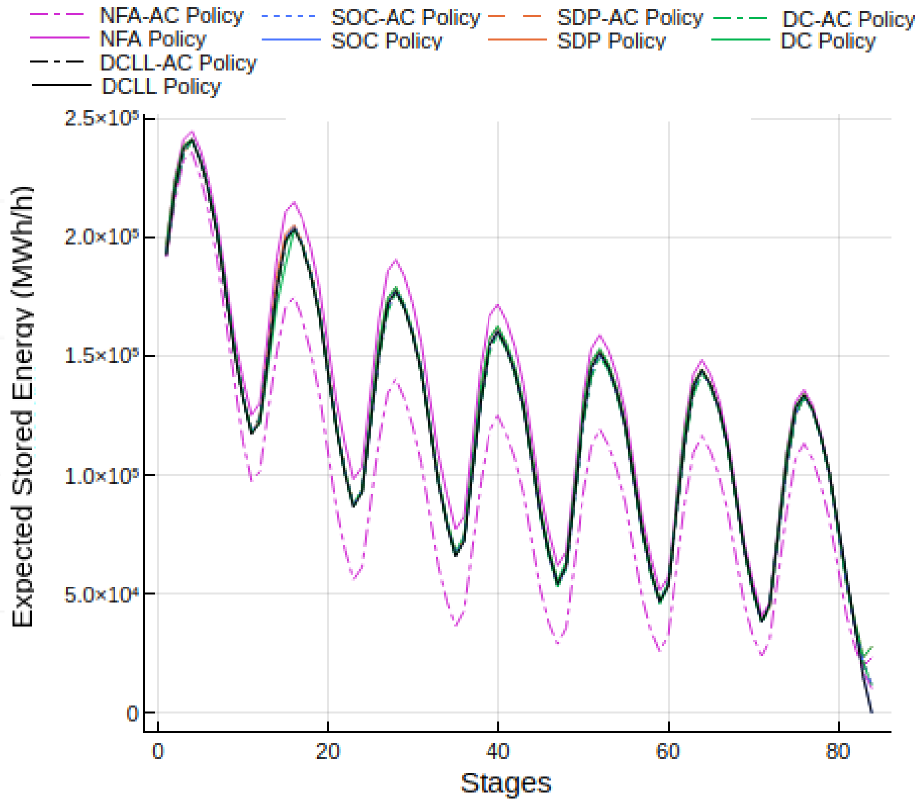

Figure 4.7: Brazil: Expected Reservoir levels with time inconsistency due to simplifications in transmission-line modeling.

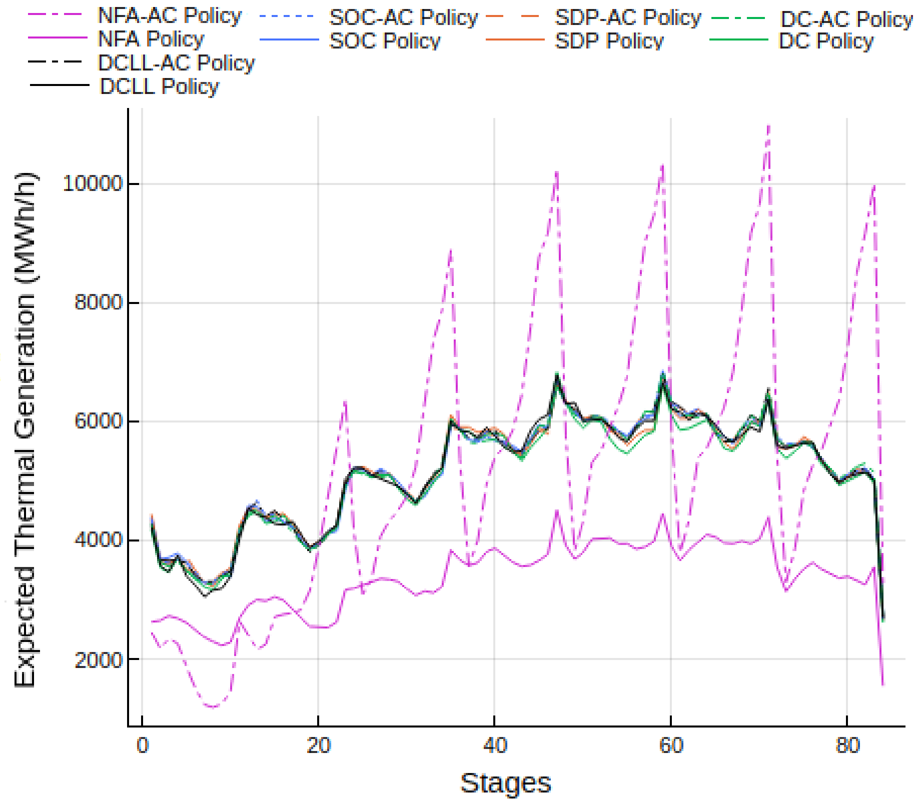

Figure 4.8: Brazil: Expected Thermal Generation with time inconsistency due to simplifications in transmission-line modeling.

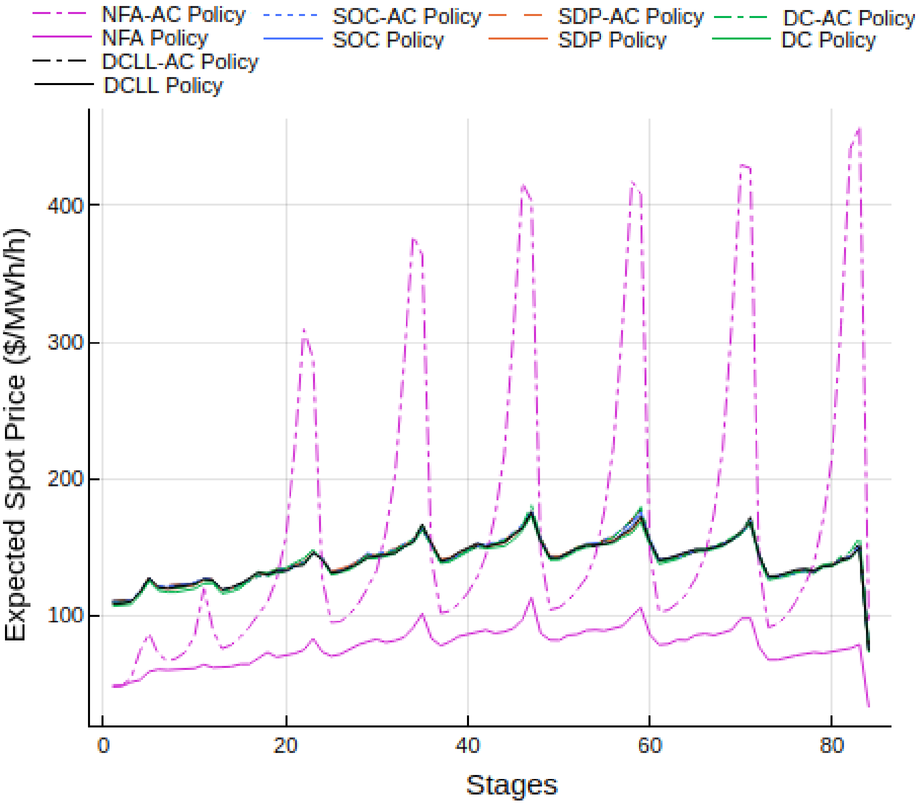

Figure 4.9: Brazil: Expected Spot Price with time inconsistency due to simplifications in transmission-line modeling.

5 Conclusion

In this work, we estimate and analyze the cost and impact of network simplifications in hydrothermal operation planning problems. The methodology proposed in [3] is used to evaluate the sub-optimality gap induced by time-inconsistent policies adopting network simplifications in the planning step. We extend its results by analyzing the policy planning in five different network formulations such as the NFA, the SOC and the SDP relaxations and the DC approximation. The policies optimized with this simplified network formulations are evaluated in the more constrained and realistic AC network model. To facilitate reproducibility, we have developed and made available an open-source tool called *HydroPowerModels.jl*.

Through case studies, our results show that time inconsistency may result in unpredictable behavior, extending the findings in [3]. We see the dangerous consequences of ignoring the electrical constraints as it is done in the Network Flow Approximation Model, such as high price spikes hidden from the planning scope. This adds hidden costs that accumulate over time, causing the actual cumulative costs incurred during operation to increasingly deviate from the costs expected during the planning phase.

Moreover, we found that using an approximation of the energy flow constraint during the planning stage (as done in the DC approximation and the SDP and SOC relaxations) may significantly reduce the inconsistency impacts.

In light of the results from the case studies, we may conclude the following:

- The NFA approximation presents on average the lowest computational burden, but performs poorly in all tested systems and exposes the system operator to high possibility of regrets due to time-inconsistency gaps, high costs in the actual implemented dispatches, and highly volatile spot-price profiles.
- The SOC provides poor approximations and performances in the presence of cycles (meshed grid), due to its inability to approximate KVL constraints. In this case, the DC and DCLL provide better approximations and performances than the SOC formulation. In radial systems however, SOC is tight to the AC model and thereby performs better

than the DC model, which disregards losses. Its computational burden is higher than linear formulations but significantly lower than the SDP relaxation.

- The SDP planning policy has low inconsistency in both meshed and radial cases and consequently low distortions in both prices and thermal generation dispatches. But this benefit comes with a high computational burden.
- The DC planning policy is a standard and fair approach, performing reasonably well (small gap and low distortions in prices and dispatches) in all instances and with a reduced computational burden.
- The DCLL planning policy has the best performance in terms of implementation cost. It consistently presents small inconsistency gaps with low distortions in prices and dispatches. The computational burden is not as reduced as in the DC case, but it is still low. The **DCLL-AC inconsistent policy** results indicate the best performance in terms of implementation cost. This policy presents the 3rd lowest GAP and computing time. It puts together the DC model capability of approximating KVL constraints and a reasonable description of the transmission losses. These results demonstrate a quadratic approximation can have close performance to the nonlinear SDP relaxation in meshed or even in more radial systems. Additionally, the DCLL quadratic formulation allows solving the problem much faster than the highly computationally intensive SDP formulation and slightly faster than the SOC formulation.

Those results show the detrimental effect of neglecting more complete power flow constraints and motivate further research and developments in the incorporation of detailed electric constraints in planning models.

The DCLL approximation seems a good alternative for the harder to solve convex relaxation and can bring significant benefits to operation planning in comparison to the widely used NFA model adopted by operators [4]. It is in our view that solution methods should be invested to reduce even further the computational burden of this quadratic approximation. Moreover, further studies should be done to see how well its assumptions hold for other systems.

Besides the findings of this study, we have delivered a flexible tool to assess those and other kinds of impacts of planning decisions in a hydrothermal economic dispatch problem. The open-source HydroPowerModels package creates a framework for complex analysis that is easy to use and to further develop. By integrating other high quality and popular packages it ensures continual upgrade and motivates discussions about this and other similar applications and packages integration.

6
References

[1] A. de Castro Brigatto, “Ensuring reserve deployment in hydrothermal power systems planning,” Ph.D. dissertation, Programa de Pós-Graduaçao em Engenharia Elétrica of the Departamento de . . . , 2016.

[2] M. V. Pereira and L. M. Pinto, “Multi-stage stochastic optimization applied to energy planning,” *Mathematical programming*, vol. 52, no. 1-3, pp. 359–375, 1991.

[3] A. Brigatto, A. Street, and D. M. Valladão, “Assessing the cost of time-inconsistent operation policies in hydrothermal power systems,” *IEEE Transactions on Power Systems*, vol. 32, no. 6, pp. 4541–4550, Nov 2017.

[4] M. Maceiral, D. Penna, A. Diniz, R. Pinto, A. Melo, C. Vasconcellos, and C. Cruz, “Twenty years of application of stochastic dual dynamic programming in official and agent studies in brazil-main features and improvements on the newave model,” in *2018 Power Systems Computation Conference (PSCC)*. IEEE, 2018, pp. 1–7.

[5] A. Street, A. Brigatto, and D. M. Valladão, “Co-optimization of energy and ancillary services for hydrothermal operation planning under a general security criterion,” *IEEE Transactions on Power Systems*, vol. 32, no. 6, pp. 4914–4923, Nov 2017.

[6] B. Rudloff, A. Street, and D. M. Valladão, “Time consistency and risk averse dynamic decision models: Definition, interpretation and practical consequences,” *European Journal of Operational Research*, vol. 234, no. 3, pp. 743–750, 2014.

[7] V. Kozmík and D. P. Morton, “Evaluating policies in risk-averse multi-stage stochastic programming,” *Mathematical Programming*, vol. 152, no. 1, pp. 275–300, Aug 2015. [Online]. Available: https://doi.org/10.1007/s10107-014-0787-8

[8] A. Shapiro and A. Pichler, “Time and dynamic consistency of risk averse stochastic programs,” *Published electronically in: Optimization Online*, 2016.

[9] J. Gönsch, "A survey on risk-averse and robust revenue management," *European Journal of Operational Research*, vol. 263, no. 2, pp. 337 – 348, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221717304800

[10] A. Street, A. Lawson, A. Velloso, and D. M. Valladão, "Assessing the hazard-decision simplification cost in multistage stochastic hydrothermal scheduling," 2018.

[11] W. B. Powell and H. Topaloglu, "Approximate dynamic programming for large-scale resource allocation problems," in *Models, Methods, and Applications for Innovative Decision Making*. INFORMS, 2006, pp. 123–147.

[12] S. H. Low, "Convex relaxation of optimal power flow—part i: Formulations and equivalence," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 15–27, 2014.

[13] K. Dvijotham and D. K. Molzahn, "Error bounds on the dc power flow approximation: A convex relaxation approach," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 2411–2418.

[14] D. V. J. G. T. S. Andrew Rosemberg, Alexandre Street and O. Dowson, "Hydropowermodels.jl juliacon 2019 proceedings," 2020. [Online]. Available: https://submissions.juliacon.org/papers/ad43bcbd43a6f904e60db8838c177520

[15] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," *SIAM review*, vol. 59, no. 1, pp. 65–98, 2017.

[16] C. Coffrin, R. Bent, K. Sundar, Y. Ng, and M. Lubin, "Powermodels.jl: An open-source framework for exploring power flow formulations," in *2018 Power Systems Computation Conference (PSCC)*, June 2018, pp. 1–8.

[17] ——, "lanl-ansi/powermodels.jl." [Online]. Available: https://github.com/lanl-ansi/PowerModels.jl

[18] O. Dowson and L. Kapelevich, "SDDP.jl: a Julia package for Stochastic Dual Dynamic Programming," *Optimization Online*, 2017. [Online]. Available: http://www.optimization-online.org/DB_HTML/2017/12/6388.html

[19] ——, "odow/sddp.jl." [Online]. Available: https://github.com/odow/SDDP.jl

[20] I. Dunning, J. Huchette, and M. Lubin, "Jump: A modeling language for mathematical optimization," *SIAM Review*, vol. 59, no. 2, pp. 295–320, 2017.

[21] R. A. Jabr, “Radial distribution load flow using conic programming,” *IEEE Transactions on Power Systems*, vol. 21, no. 3, pp. 1458–1459, Aug 2006.

[22] X. Bai, H. Wei, K. Fujisawa, and Y. Wang, “Semidefinite programming for optimal power flow problems,” *International Journal of Electrical Power & Energy Systems*, vol. 30, no. 6, pp. 383 – 392, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0142061507001378

[23] S. Sojoudi and J. Lavaei, “Physics of power networks makes hard optimization problems easy to solve,” in *2012 IEEE Power and Energy Society General Meeting*, July 2012, pp. 1–8.

[24] B. Stott, J. Jardim, and O. Alsac, “Dc power flow revisited,” *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1290–1300, Aug 2009.

[25] C. Coffrin and P. Van Hentenryck, “A linear-programming approximation of ac power flows,” *INModels Journal on Computing*, vol. 26, no. 4, pp. 718–734, 2014.

[26] J. Carpentier, “Contribution to the economic dispatch problem,” *Bulletin de la Societe Francoise des Electriciens*, vol. 3, no. 8, pp. 431–447, 1962.

[27] M. B. Cain, R. P. O’ Neill, and A. Castillo, “History of optimal power flow and Models,” Tech. Rep., 2012. [Online]. Available: https://www.ferc.gov/industries/electric/indus-act/market-planning/opf-papers/acopf-1-history-Model-testing.pdf

[28] A. Philpott, V. de Matos, and E. Finardi, “On solving multistage stochastic programs with coherent risk measures,” *Operations Research*, vol. 61, no. 4, pp. 957–970, 2013.

[29] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory*. SIAM, 2009.

[30] A. J. Conejo, M. Carrión, J. M. Morales *et al.*, *Decision making under uncertainty in electricity markets*. Springer, 2010, vol. 1.

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

[32] R. Bellman, “Dynamic programming,” *Science*, vol. 153, no. 3731, pp. 34–37, 1966.

[33] D. P. Bertsekas, *Dynamic programming and optimal control.* Athena scientific Belmont, MA, 1995, vol. 1, no. 2.

[34] C. J. Donohue and J. R. Birge, "The abridged nested decomposition method for multistage stochastic linear programs with relatively complete recourse," *Algorithmic Operations Research*, vol. 1, no. 1, 2006.

[35] Z.-L. Chen and W. B. Powell, "Convergent cutting-plane and partial-sampling algorithm for multistage stochastic linear programs with recourse," *Journal of Optimization Theory and Applications*, vol. 102, no. 3, pp. 497–524, 1999.

[36] M. Hindsberger and A. Philpott, "Resa: A method for solving multistage stochastic linear programs," *Journal of Applied Operational Research*, vol. 6, no. 1, pp. 2–15, 2014.

[37] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath, "Coherent measures of risk," *Mathematical finance*, vol. 9, no. 3, pp. 203–228, 1999.

[38] J. J. Grainger, W. D. Stevenson, W. D. Stevenson *et al.*, *Power system analysis*, 2003.

[39] D. K. Molzahn, I. A. Hiskens *et al.*, "A survey of relaxations and approximations of the power flow equations," *Foundations and Trends® in Electric Energy Systems*, vol. 4, no. 1-2, pp. 1–221, 2019.

[40] C. Coffrin, H. L. Hijazi, and P. Van Hentenryck, "The qc relaxation: A theoretical and computational study on optimal power flow," *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 3008–3018, 2015.

[41] S. H. Low, "Convex relaxation of optimal power flow—part ii: Exactness," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 2, pp. 177–189, 2014.

[42] C. Coffrin, P. Van Hentenryck, and R. Bent, "Approximating line losses and apparent power in ac power flow linearizations," in *2012 IEEE Power and Energy Society General Meeting*. IEEE, 2012, pp. 1–8.

[43] R. Palma-Benhke, A. Philpott, A. Jofré, and M. Cortés-Carmona, "Modelling network constrained economic dispatch problems," *Optimization and Engineering*, vol. 14, no. 3, pp. 417–430, 2013.

[44] PSR, "Software | PSR," 2019, [Online; accessed 2019-07-06]. [Online]. Available: http://www.psr-inc.com/softwares-en/

[45] L. Cambier and D. Scieur, "leopoldcambier/fast: v0.9.1b," Mar. 2018. [Online]. Available: https://github.com/leopoldcambier/FAST

[46] L. Raso, D. Dorchies, J. Kwakkel, and P. O. Malaterre, "Optimist: a python library for water system optimal operation and analysis using sddp," 2016.

[47] A. S. Lingquan Ding, Shabbir Ahmed, "A python package for multi-stage stochastic programming," May 2019. [Online]. Available: http://www.optimization-online.org/DB_HTML/2019/05/7199.html

[48] N. Löhndorf, "Quasar optimization software 2.4," 2018. [Online]. Available: http://www.quantego.com/

[49] V. Leclère, H. Gérard, F. Pacaud, and T. Rigaut, "Stochdynamicprogramming. jl a julia library for multistage stochastic optimization."

[50] B. Legat, "Juliastochopt/structdualdynprog.jl: v0.2.0," Oct. 2018. [Online]. Available: https://github.com/JuliaStochOpt/StructDualDynProg.jl

[51] 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*, pp. 1–1, 2019.

[52] M. P. Soares, A. Street, and D. M. Valladão, "On the solution variability reduction of stochastic dual dynamic programming applied to energy planning," *European Journal of Operational Research*, vol. 258, no. 2, pp. 743–760, 2017.

A
Nomenclature

The symbols used in Chapters 3 and 3.2 are defined in this section.

Sets and Indices

$\mathcal{N}$ — Set of bus indices n or m.

$\mathcal{N}_0$ — Set containing single reference bus index n or m.

$\mathcal{N}_n$ — Set of busses connected to bus n.

$\mathcal{L}$ — Set of directed transmission lines/branches, pairs of buses (n, m).

$\mathcal{I}, \mathcal{I}_n$ — Set of thermoelectric units i and subset of units at bus n, respectively.

$\mathcal{H}, \mathcal{H}_n$ — Set of hydroelectric units j and subset of units at bus n, respectively.

$\mathcal{H}_j^U, \mathcal{H}_j^S$ — Set of upstream hydroelectric units k out-flowing and spilling to the hydroelectric unit j, respectively.

Constants

P_{it}, Q_{it} — Real and reactive power bound, respectively, at generator i and period t.

C_{it} — Linear cost at generator i and period t.

$\overline{V}_{nt}$, $\underline{V}_{nt}$ — Upper and lower magnitude voltage limits, respectively, at bus n and period t.

D_{nt} — Real power demand (load) at bus n and period t.

Y_{nt}^s, Y_{nt}^{sq} — Shunt real and imaginary admittance at bus n and period t, respectively.

C_{nt}^{δ} — Deficit linear cost at bus n and period t.

$G^{c}_{(n,m)t}$ Real part of pi-section parameters at branch (n, m) and period t.

$B^{c}_{(n,m)t}$ Imaginary part of pi-section parameters at branch (n, m) and period t.

$F_{(n,m)t}$ Apparent power limit at branch (n, m) and period t.

$R_{(n,m)t}$ Resistance at branch (n, m) and period t.

$X_{(n,m)t}$ Reactance at branch (n, m) and period t.

$\overline{\Delta}^{\theta}_{(n,m)t}$ Voltage angle difference upper bound at branch (n, m) and period t.

$\underline{\Delta}^{\theta}_{(n,m)t}$ Voltage angle difference lower bound at branch (n, m) and period t.

$\overline{\nu}_{jt}$ Volume limit at hydroelectric units j period t.

$\overline{\nu}_{jt-1}$ Initial volume at hydroelectric units j period t.

A_{jt} Possible inflows at hydroelectric units j period t.

U_{jt} Outflow limit at hydroelectric units j period t.

ρ_{jt} Production factor at hydroelectric units j period t.

C^{s}_{jt} Spillage linear cost at hydroelectric units j period t.

Operators and Special Numbers

$\Im$ Imaginary unit.

$\angle(\cdot)$ Angle of the polar representation of a complex number.

$(\cdot)^*$ Complex conjugate.

$|\cdot|$ Absolute value.

Decision Variables

p_{it}, q_{it} Real and reactive power dispatch, respectively, at generator i and period t.

v_{nt} Complex voltage at bus n and period t.

θ_{nt} Phase angle at bus n and period t.

δ_{nt} Real power deficit at bus n and period t.

$\ell_{nt}^{(sh)}$ Shunt loss at bus n and period t.

$f_{(n,m)t}$ Real power flow at branch (n,m) and period t.

$f_{(n,m)t}^{q}$ Reactive power flow at branch (n,m) and period t.

u_{jt} Outflow at hydroelectric units j period t.

ν_{jt} Volume at hydroelectric units j period t.

s_{jt} Spillage at hydroelectric units j period t.

B
Usage of HydroPowerModels

The usage of *HydroPowerModels.jl* follow the paradigms of the Julia language and the structure of the dependent packages.

In order to access the available functionalities, first import *HydroPowerModels.jl* and an adequate solver:

```julia
using HydroPowerModels
using GLPK
```

Load a case by passing the folder containing the input files (`PowerModels.json`, `hydro.json`, and `inflows.csv`):

```julia
data = HydroPowerModels.parse_folder(
    "case3_folderpath")
```

Use `create_param` to create a set of problem parameters. For example, a 12-stage problem using the DC approximation can be specified as follows:

```julia
params = create_param(
 stages      = 12,
 stage_hours = 168.0,
 model_constructor_grid = DCPPowerModel,
 optimizer   = with_optimizer(GLPK.Optimizer))
```

Then, build the Model and execute the SDDP `train` method:

```julia
m = hydrothermaloperation(data, params);

HydroPowerModels.train(m)
```

Finally, simulate the performance of the policy with 1000 Monte Carlo scenarios:

```
# Simulate 1000 instances
results = HydroPowerModels.simulate(m, 1000);
```

```
 Dict{Any,Any} with 5 entries:
"simulations" => Dict[Dict{Any,Any}(Pair{Any,...
"data"        => Dict{Any,Any}[Dict{Any,Any}(...
"params"      => Dict{Any,Any}(Pair{Any,Any}(...
"machine"     => Dict("cpu"=>"Intel(R) Xeon(R)...
"solve_time"  => 205.31247
```

B.1
Case Study

For a case study, consider a the hydrothermal dispatch of a realistic system with the following specifications:

- Number of buses: 166
- Number of loads: 286
- Number of generators: 145
- Number of branches: 235

In order for a qualitative view of the system, the package disposes a graph illustration plot:

```
plot_grid(data, node_label=false)
```

Figure B.1 shows the installed power available in the network (grouped by bus) using a logarithmic scale. The red nodes represent the thermal generators, the blue represent the hydro generators. For comparison purposes, orange nodes have been added that are equivalent to average real power demands.

As we can see from the plot of the grid B.1, this appears to be a well balanced case, with similar installed hydro and thermal power capacity and with a reasonable average demand. In addition, it is a well distributed network, without any evident critical sections susceptible to impacting power

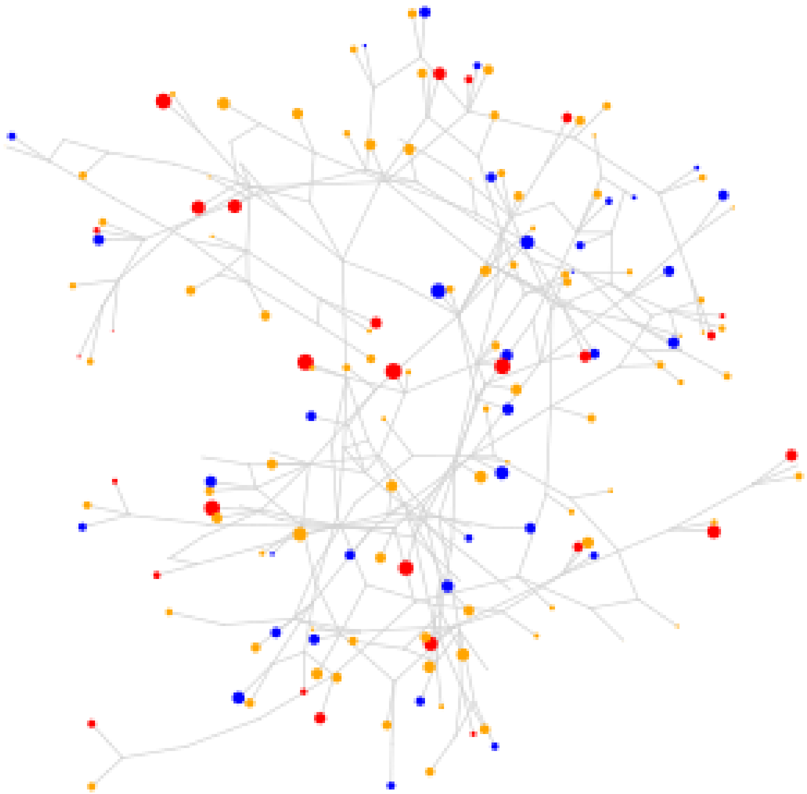

Figure B.1: Network Grid graph

flow problems. Those facts are indications of a significant hydro-generation optimal dispatch without large complications.

For this study, a 52 stage horizon planning and simulation have been executed using the following case parameters:

- Number of stages: 52
- Number of hours in stage: 168
- Network Formulation: Transportation model relaxation

B.2 Results

The simulate command returns a detailed dictionary of the execution. In order to plot those results returned by the simulate function, you may choose from a variety of methods, including the function `plot_aggregate_results()`. This function receives the dictionary results and generates the most common aggregated variable plots, which best summarize simulations of a hydrothermal dispatch:

```
plot_aggregated_results(results)
```

Figures 2 to B.9 show the output from the above command. As mentioned, the plots are of aggregated quantities, but the methods used to aggregate were chosen in order to help analysis. For example: the final nodal price is an average of nodal prices weighted by the contribution of local loads to the total demand; reservoir volume was grouped weighted by the amount of energy that could be produced by the stored water (as was the inflow of water).

As expected the optimal dispatch of the simulations uses more hydro-generators, however it needs thermal-generators to met all demands without deficit. On this hydro-dominated system, the uncertain inflow is a driving factor of optimal dispatch. As we can see in Figure B.9, the inflow has a strong seasonality component, resulting the significant seasonality trait observable in the variables of the policy simulations. 2-B.3, B.5-B.8.

Similar studies are possible for any case and formulation chosen, helping to analyze existing realistic cases and assess impacts of future system changes.

Measuring the impacts of possible simplifications is a needed step in discussing hydrothermal economic dispatch. *HydroPowerModels.jl* intends to provide a common ground for discussions and analysis and a easy to use tool for research and applications.

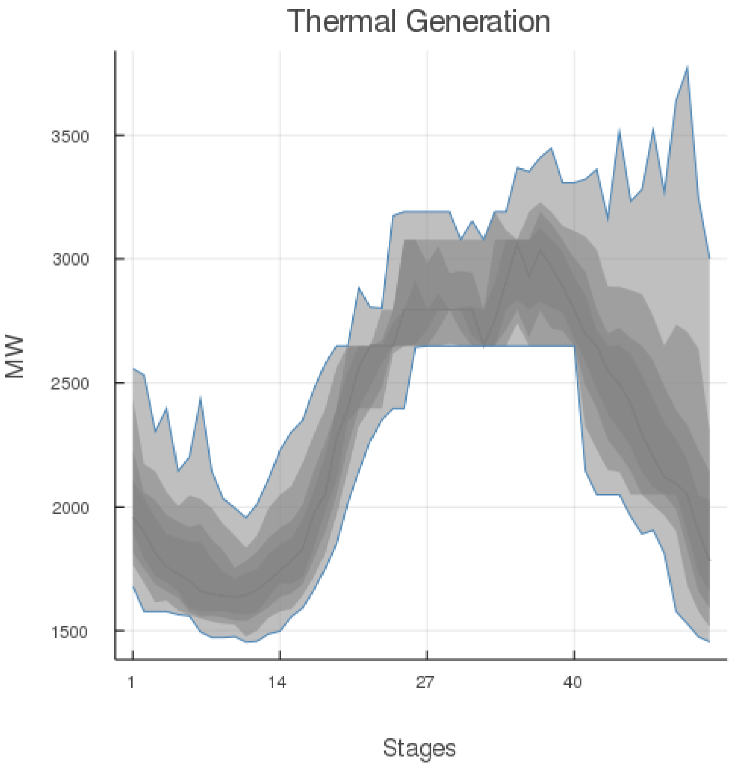

Figure B.2: Thermal Generation

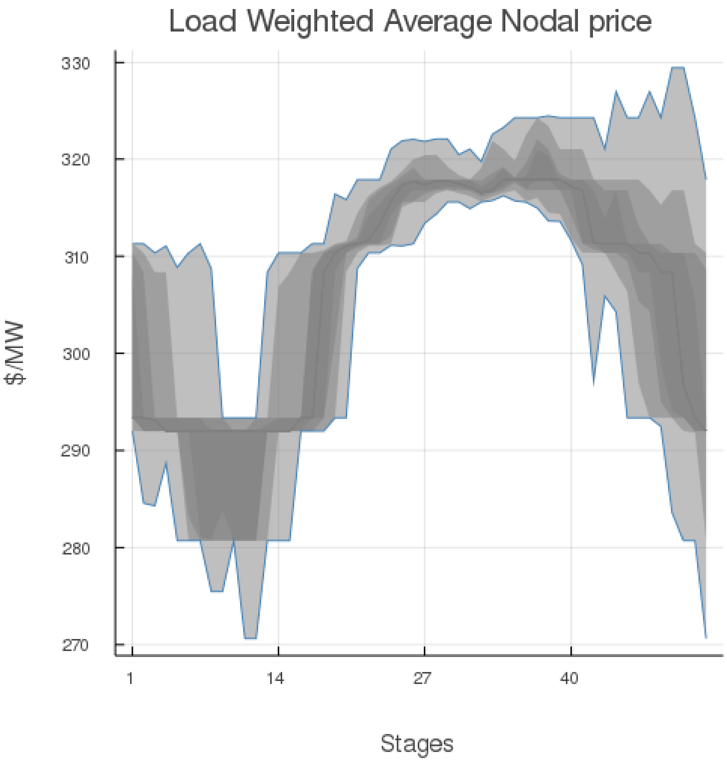

Figure B.3: Load Weighted Average Nodal Price

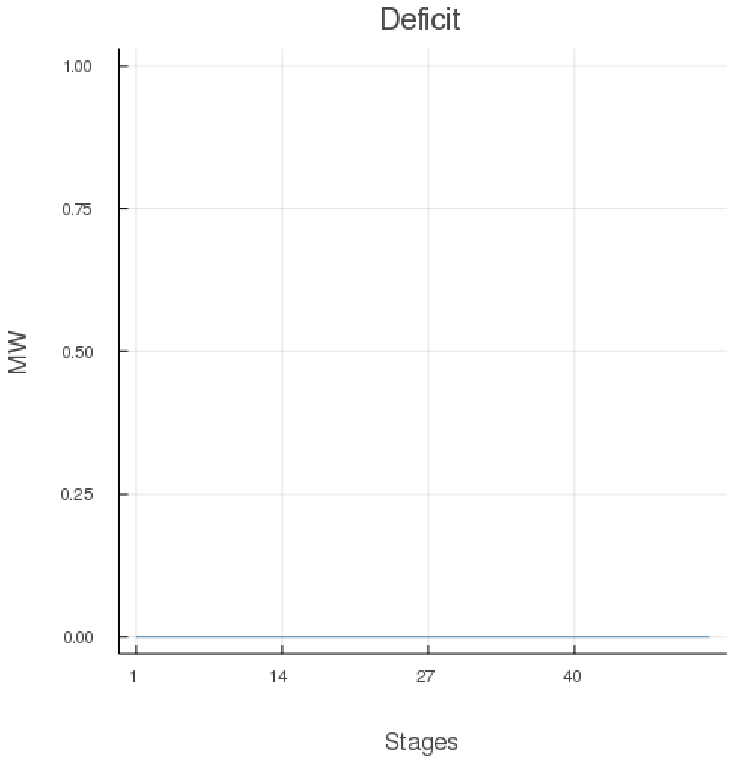

Figure B.4: Deficit

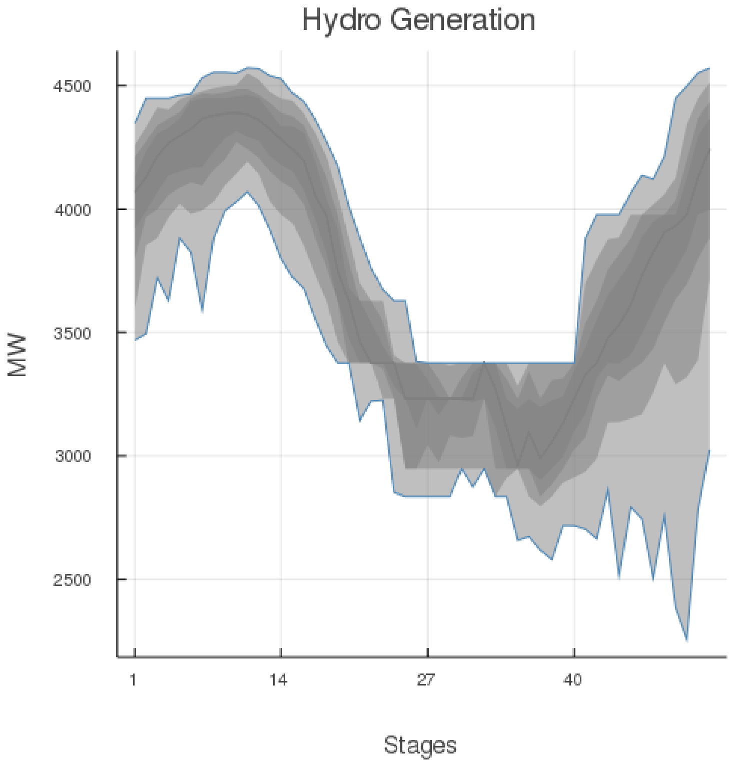

Figure B.5: Hydro Generation

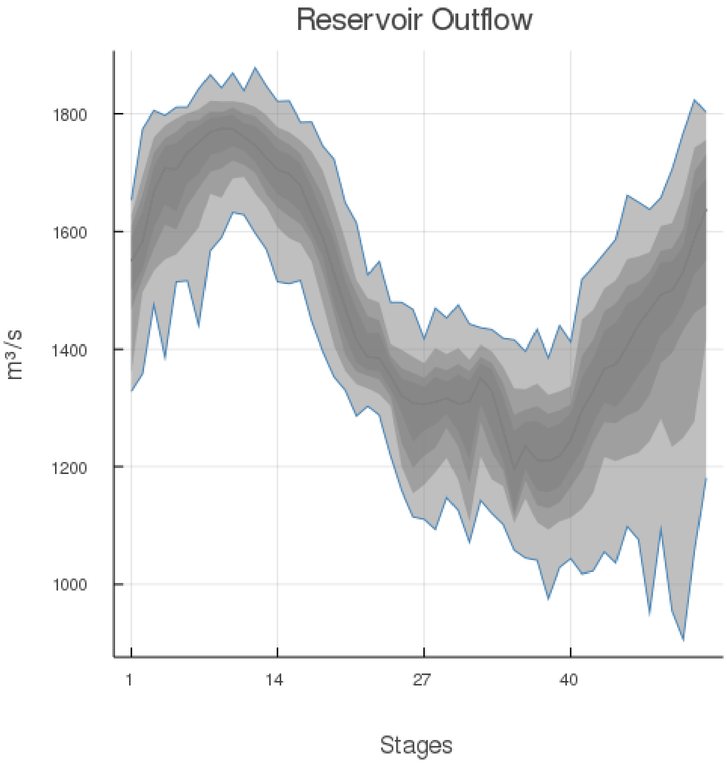

Figure B.6: Reservoir Outflow

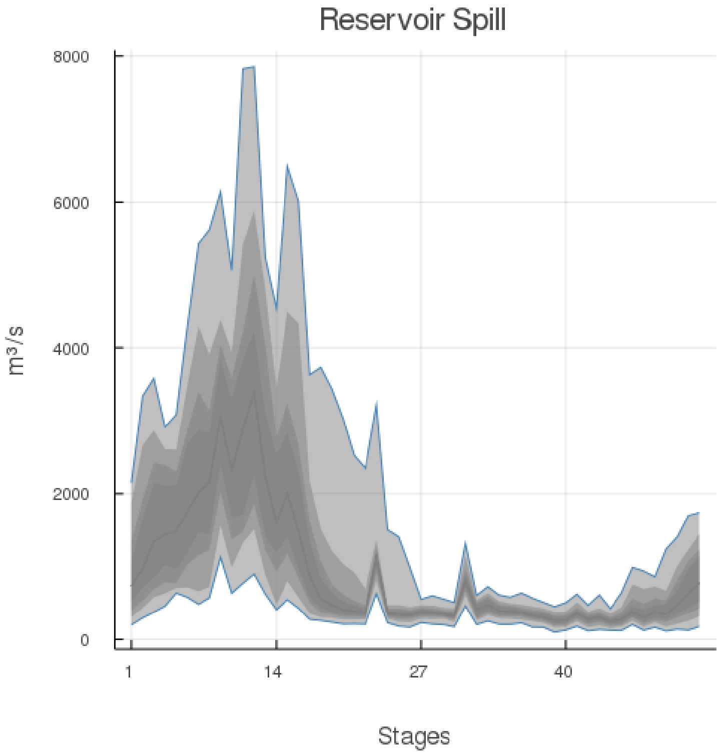

Figure B.7: Reservoir Spillage

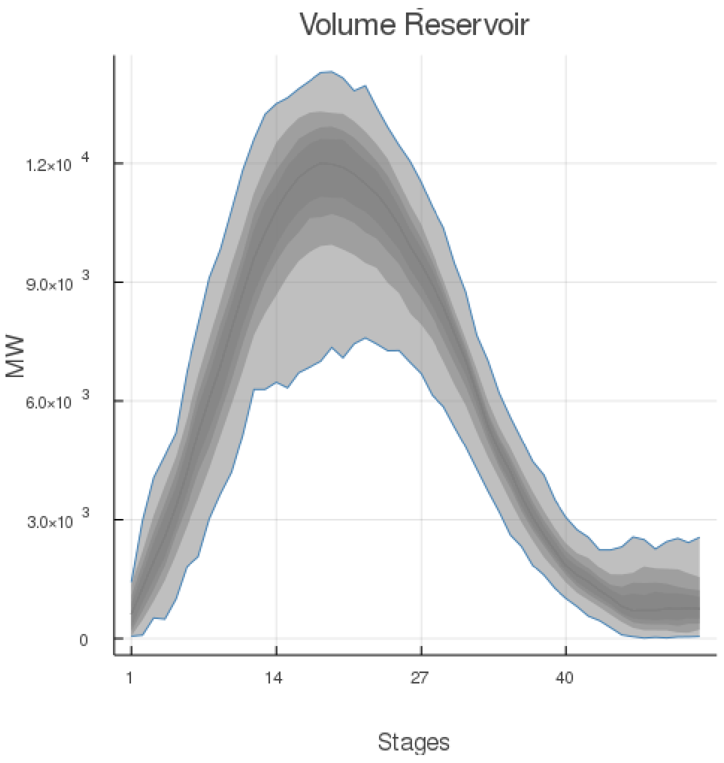

Figure B.8: Volume Reservoir

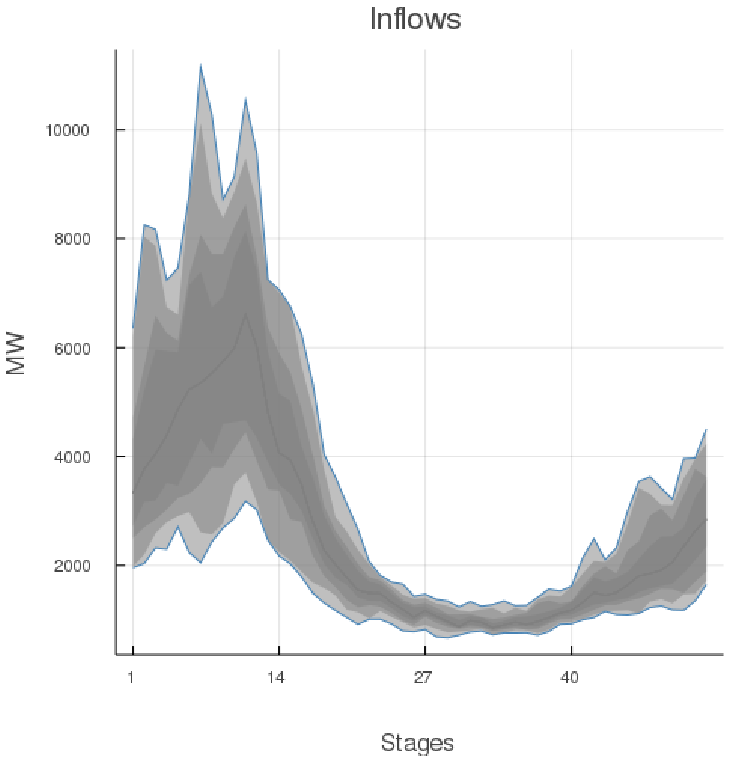

Figure B.9: Inflow