Stochastic Optimisation with Energy Storage for a One-Hundred Percent Renewable Power System for South Australia

Cameron Baulderstone

August 2022

Thesis submitted for the degree of
Master of Philosophy
in
Applied Mathematics
at The University of Adelaide
Faculty of Engineering, Computer and Mathematical Sciences
School of Mathematical Sciences

THE UNIVERSITY
of ADELAIDE

Dedication

To K.S.,

This has given us plenty to talk about.

Acknowledgements

I acknowledge both Dr Giang Nguyen and Dr Andrew Black, without you this project would not have begun. In particular I acknowledge Giang for the initial idea for this project and allowing me to take it on and Andrew for initial guidance particularly with coding.

There are many people whom without their help and support this thesis would not have been completed. Thank you to my supervisors Professor Matthew Roughan and Professor Langford White. Your enthusiasm, positivity, encouragement, guidance and professionalism have made this possible. I especially want to thank you both for taking the time to read my work and for the time you have taken to supervise me.

Thank you to Mike Liebelt, David Vowles, Martin Biel, Spyros Chatzivasileiadis and Aleksis Xenophon for your assistance. I appreciate the time you have taken to answer my questions and the support you gave.

Thank you to my friends in the postgraduate cohort; the discussions and camaraderie have been a highlight of this degree.

Thank you Kelli, you have encouraged and supported me through a challenging degree. Thank-you for your critical assessments throughout the thesis, particularly on formulating proofs.

Finally, thank you to my family for supporting me.

Signed Statement

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint award of this degree.

I give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

I acknowledge the support I have received for my research through the provision of an Australian Government Research Training Program Scholarship.

Signed: ..... Date: .....

Abstract

This thesis details a methodology that concerns both operation and planning of generators and energy storage using stochastic-optimal power flow (S-OPF). It formulates the S-OPF as a two-stage stochastic program with fixed recourse using the linearised DC-OPF formulation. This thesis details new results on the existence of solutions to such programs that apply in our application of interest. It forms the sampling average approximation of the stochastic program with a range of sample sizes and calculates confidence intervals for the true solution to understand the relationship between sample size and solution accuracy.

The S-OPF formulation accounts for both network constraints and the temporal uncertainty of renewable generation and demand. Time series models are fit to historic wind and demand data to model these uncertain parameters over time and are used in the S-OPF. Existing models for storage units are adapted and a new model for interconnectors is given, which are both applied in a novel way. The model is validated using test networks. In validation, the thesis also quantifies the cost benefit of this S-OPF with a comparable deterministic program.

The thesis applies the methodology to the South Australian power system, as it transitions to a 100% renewable power system. South Australia is of particular interest as it already has a high penetration of intermittent renewable generation, such as wind and solar, and is in the process of integrating storage to support these renewable generators. A model for the South Australian transmission system is obtained from [115] and line power flow limits from [13] are added. After validating the methodology against historical data as well as a copperplate model, the thesis uses the network model and simulated scenarios to apply the S-OPF methodology and investigates how new energy storage can be optimally located.

This thesis concludes that regions of high demand and high connectivity are locations where storage is most utilised, which are in close proximity to the Adelaide region. This validates that the methodology can be used to determine how energy storage should be optimally located and used in conjunction with current generation. The methodology proposed in this thesis may hold broad appeal as it can be applied in more general settings.

Contents

Chapter 1

Introduction

This thesis details a method to determine how generators and energy storage should be optimally located and operated in a power network. It accounts for the temporal uncertainty of renewable energy generation and demand, as well as network constraints such as transmission line limits. Our main network of interest is the South Australian electricity network, where we use this method to determine optimal locations of energy storage as South Australia transitions to a 100% renewable energy electricity network.

1.1 Motivation

Power systems in many developed nations are rapidly changing. This is driven by investment in renewable generation, which lowers production costs and addresses a growing need to reduce greenhouse gas emissions. Unfortunately, current electricity production from fossil fuel sources is one of the largest contributors to greenhouse gas emissions. In Australia in 2019 electricity generation for public use produced 32.9% of Australia's total carbon emissions [26]. With ambition to reduce greenhouse gas emissions contributing to climate change, Australia is one of many nations that has committed to net zero carbon emissions by the year 2050 [82].

In Australia fossil fuel generators have been historically relied on to provide the vast majority of power. However now this is not necessarily the case; in South Australia in the year 2020 over 60% of generation was from renewable sources [20]. South Australia is transitioning to a renewable power system and aims to be net 100% renewable by 2030 [84]. Yet this transition is not without its challenges. While fossil fuels such as coal and natural gas are easily stored and used when needed, renewable generators such as wind turbines are reliant on favourable weather conditions. Further, renewables require a range of short, medium and long range storage to be a viable supplier of power on their own.

Transitioning to renewables introduces additional challenges, as high proportion of renewable generators increase uncertainty in the supply of power. Other elements of

uncertainty have long been inherit in power systems, the most obvious example is demand for power which depends on human behaviour.

Due to a power system's size and the requirement to be continuously operational there is limited capacity for experimentation prior to development of new infrastructure. Mathematical modelling and simulations play a crucial role in understanding system requirements. Optimisation methods are used in power systems for daily operation as well as in system planning, as they can determine optimal power flows from generators so that demand can be met [31].

Given the increasing and compounding uncertainty introduced from a higher proportion of renewable generation, it is important to account for this uncertainty in modelling. Failing to account for extreme but unlikely circumstances can be catastrophic for the network, as was the case in South Australia in 2016 where a series of events led to a complete failure of the South Australian power system [87, 109]. In [31] the applicability of probabilistic optimisation methods for networks with a high amount of uncertainty is highlighted. These optimisation methods are founded on minimising the expected cost of uncertain variables, where Monte-Carlo sampling methods can be used.

In addition, it is also important to include network constraints when modelling power networks. This includes information such as the line limits of a power line, and the geographical locations of generators. These network parameters constrain the power flows of the network and can limit the ability of generators to meet demand [53]. As with other generators, energy storage is reliant on transmission infrastructure [62] and hence optimal locations are influenced by transmission line capacity. While these constraints are important, many existing studies have used a *copperplate* method where no network constraints are assumed, as we discuss further below.

The importance of power systems is reflected in the high volume of multidisciplinary research on the topic, including in Australia with studies on storage requirements in transitioning networks. The approaches to modelling storage and renewables are vast as documented in the review papers [44, 117, 51].

This thesis aims to contribute to this literature by developing a methodology for determining storage locations accounting for both network constraints and stochasticity in the system. Our approach is guided and motivated by the transition to renewables in Australian and in particular South Australia. Below we summarise relevant research and methodologies for this thesis, identifying how our research fills a gap in the literature.

Although we are interested in studies that have been applied to Australia and specifically South Australia, there are numerous studies on storage in the literature and many of these are applicable to the South Australian context. Multiple studies consider optimising the location and size of storage within transmission networks, including [96, 112, 116]. Summarising these papers, we see that

- Pandzic et al. [96] describes a three stage process that uses AC-OPF for siting and sizing of storage within a transmission network. Their method is applied to test networks where wind farms have been added to buses. Times series models are fit to wind speed data are used to produce a deterministic wind power forecast.

- Wogrin and Gayme [112] uses DC-OPF for siting and sizing for a portfolio of storage technologies with tests applied to a 14 bus network. While stochastic optimisation is not implemented in test cases it is outlined how stochastic methods could be used as an extension.

- Yacar, Tejada-Arango, and Wogrin [116] investigates the energy storage investment while considering increased renewable energy penetration, network constraints and transmission losses. DC-OPF is used together with a linear approximation for transmission losses. Network congestion is identified as a more important consideration over transmission line losses for storage allocation.

While these papers [96, 112, 116] consider optimising storage with varying assumptions they do not implement stochastic optimisation methods, which we do in this thesis. Other studies, in different ways, capture the stochasticity of power system operation within their modelling. For example

- Two-stage stochastic programming is used by Baker [30] for siting and sizing of storage based on DC-OPF. The first stage decisions are storage parameters while the second stage decisions include generation outputs which depend on the shortfall of wind power to meet demand. Observed historical data are used to form scenarios for the two stage problem. In an extension, chance constraints are used for wind power forecast error, where Gaussian distributions are fitted. These chance constraints must be satisfied with a pre-determined probability.

- Mirzaei et al. [86] takes a different approach and considers a whole energy network that contains a power network, energy storage and a gas network. Stochastic optimisation is used to optimise the network in the presence of forecast uncertainty for load and wind power. The optimisation is formed as a network constrained unit commitment problem. Simulations are performed on a number of test networks where a scenario reduction method is used to reduce 1000 scenarios to 10. Penalty costs are assigned to incentivise against wind curtailment and load shedding.

Both Baker [30] and Mirzaei et al. [86] have similarities to the approach taken in this thesis, however there are also differences. We do not use scenario reduction methods but rather use efficient solution algorithms that allow for a large number of scenarios to be considered in the stochastic optimisation. To generate our scenarios we use time series methods to fit stochastic models to real data for sampling scenarios. Another significant difference

is the structure of the first stage and second stage problems, Baker [30] incorporates the parameters of energy storage while [86] considered the day ahead and real time markets. In this thesis we consider the next dispatch (one step ahead) decision in the presence of uncertainty from wind power and demand.

Other studies use a range of stochastic optimisation methods in the context of optimising storage size and locations within power systems, this includes [33, 89] implementing chance constraints. While [65, 89] both use AC-power flow methods where accuracy and complexity are increased. In this thesis we do not consider chance constraints within our stochastic modelling further we do not use an AC power flow model as the non-linearity in not compatible with the efficient L-shaped method that we use.

Australia is relatively unique in its potential to generate large volumes of electricity from renewables. This, coupled with the relatively low population density, means the NEM and South Australia (within the NEM) are well positioned for transitioning to a 100% renewable electricity system. We now consider studies that have focus on energy storage requirements in Australia and South Australia.

Both [7, 77] consider storage requirements for 100% renewable power systems in Australia. In [7] the optimisation methods used is not clear, however storage is considered under different network scenarios. In [77] a copperplate network is assumed for the National Electricity Market (NEM) and their study is focused on extreme cases when low wind generation and low solar generation are observed. This is combined with a dispatch model that emulates an hourly competitive bidding process. It is assumed that biofuel and hydro-storage can be used to fill shortages in supply.

Other papers consider a particular type of storage and this guides their respective method. In [78] a geographic information system algorithm is used to allocate potential locations for hydro-storage. A case study considers South Australia, and identifies 190 sites with a total capacity of 276 gigawatt hours of energy storage. However, this study does not consider network structure and constraints when suggesting potential sites. In [75] a case study on the suitability of hydrogen for long-term and large-scale storage of energy in South Australia is undertaken. This approach includes the case of using battery storage and hydrogen as storage options. They find the use of hydrogen has multiple benefits and further reduces the cost of electricity generation. To find the optimal solution for storage the Hybrid Optimisation Model for Electric Renewables simulation software is used, while network constraints are not modelled.

The papers [49, 104] model the energy storage requirements for the South Australian power grid and use genetic algorithms for their optimisation. Similar to [75], this optimisation technique relies on total demand and generation and does not consider specific network constraints. Bullas [49] acknowledges that the findings are 'impractical and unrealistic' as they have not used real demand or generation data and have not considered the efficiency of energy storage. In [60] a genetic algorithm is used to find a least cost so-

lution for a projected generation mix of wind, photovoltaics, concentrating solar thermal with storage, hydroelectricity and bio-fuelled gas turbines. Improvements are made on a copperplate network by ensuring energy balance between regions of the NEM.

There have also been many studies that consider stochastic programming in the context of optimal power flow [73, 42, 74, 85, 98]. There has been limited research that focuses on stochastic programming in the context of modelling storage in interconnected power systems. When a system has some element of stochasticity it has been demonstrated that using stochastic programming methods to properly capture this with an optimisation problem can lead to better solutions [41].

Until 2018 there was no openly available model of the NEM in which the South Australia system operates. There has also been very limited research using power flow, optimal power flow and/or unit commitment on the NEM in publicly available research. However, in 2018 Xenophon and Hill [115] used openly available data to publish and make available a model of the NEM. This model contains enough detail to define optimal power flow and unit commitment problems within the NEM. This is one of the reasons there is a gap in the existing literature on using network constraints when studying storage locations for the South Australian network.

As we identified above, we know of no studies of the South Australian power network that use network constraints in determining the optimal location of storage. Most studies we have observed apply their optimisation and simulations to smaller test networks but not directly to real networks. In addition, while many studies such as Bai, Lee, and Lee [28] uses time series modelling to generate scenarios in their S-OPF, data for modelling renewable generation or demand is often not directly linked to the network studied.

We aim to contribute to the literature by using stochastic-optimal power flow programming methods to create a methodology to determine the optimal capacity and placement of storage within the South Australian power system. We will take into account both the network structure and stochasticity of the system when determining storage requirements. Our approach is motivated by accurately capturing the stochastic behaviour of wind power and demand within the optimisation constraints. Further we are guided by South Australia's unique circumstances with high renewable energy penetration in an interconnected network.

We form a two-stage stochastic program to make more optimal decisions, in the presence of uncertainty introduced from wind generation and demand. The stochastic information in the system is explicitly accounted for in the stochastic program. We optimise with respect to marginal cost of production and do not consider the bidding process in this work. While our focus is on the South Australian power system, this methodology is general enough to apply to other networks and we demonstrate the expected behaviour on test networks.

We take a different approach to the studies discussed above where we include renewable

generation as a decision variable within stochastic optimisation. That is, wind generation is a decision variable where it can be dispatched up to the varying capacity dependent on the weather. Further we devise our own framework for setting first stage and second stage variables in transitioning networks.

In our methodology, we optimise power dispatch by defining a DC optimal power flow (DC-OPF) program. We then formulate this as a stochastic-optimal power flow (S-OPF), which has the form of a two-stage stochastic program with fixed recourse. We apply this to several test networks, as well as the South Australian portion of the open NEM model from Xenophon and Hill [115]. At the time of the model construction in Xenophon and Hill [115] there was no grid scale solar photovoltaic generation connected to the South Australian grid, so we do not include this in this thesis. These characteristics of Xenophon and Hill [115] guide our approach, where our focus in on capturing the uncertainty introduced from wind generation capacity and power demand only.

We model the uncertainty from demand and renewable generation over time using time series methods. Using linear programming methods, such as the efficient L-shaped method, we solve the S-OPF to determine the optimal power flows at each time step. Our method allows us to optimise for each time step over long time periods, which captures extremes in uncertain variables. We consider the South Australian network in detail, where we use a *greedy approach* adding additional storage and renewable generation until the network is net 100% renewable and then removing less utilised storage. We then analyse the locations of the storage, discussing how they depend on network constraints, which validates our methodology.

Overall, we find that locations of high demand and high connectivity are the optimally used storage locations in our S-OPF. We also observe that optimal storage locations do depend on season, and both summer and winter seasons are important to analyse when placing storage. Including a model of the planned SA-NSW interconnector also makes a difference to where storage is optimally placed. These optimal locations for storage may be refined further considering practical limitations such as physical location and investment cost, which we leave for future work. This demonstrates that our methodology can be used to understand optimal locations for storage in power systems while accounting for network constraints and the stochasticity of the system.

Beyond the application to power systems, this thesis also studies the theory of the optimisation programs and solution methods used in our results. This includes new results on the existence of solutions to two-stage stochastic programs with fixed recourse, which apply to our application. This also includes detailing the L-shaped method convergence, as well as studying the accuracy of the sample average approximation (SAA) to the two-stage stochastic program with fixed recourse. In a test case our results show the confidence intervals for the optimal overall cost for the SAA are within $\pm 8\%$ of the true cost with only 10 samples, and this decreases to less than $\pm 4\%$ with 10,000 samples. This gives a measure of overall accuracy for our work. We now describe the structure of the thesis.

1.2 Structure of this thesis

This thesis is composed of five main chapters. As this thesis is an application of stochastic programming techniques, there are many multidiscplinary components. An expert in optimisation, power flow, and time series analysis may wish to focus on the implementation of methods which are covered in Chapter 4, Sections 5.2 to 5.5 and Chapter 6, whilst referring back to other sections as required. However, we note that we do present new material in all other chapters, as we detail below.

We begin in Chapter 2 which provides foundation for the optimisation techniques we require to form and solve stochastic-optimal power flow. We start by covering fundamental definitions and results in convex and linear programming. This leads to discussing stochastic programming where our focus is on two-stage stochastic programs. We define a two-stage stochastic program with fixed recourse along with relevant properties. We give new results detailing conditions that ensure that existence of solutions are guaranteed, which we use in our methodology. Importantly we define the sampling average approximation (SAA) of the two-stage stochastic program which has a decomposition structure. We study the accuracy of the SAA, where results in a test case show the confidence intervals for the optimal overall cost for the SAA are within $\pm 8\%$ of the true cost with only 10 samples, and this decreases to less than $\pm 4\%$ with 10,000 samples. The SAA program forms the basis of optimisation method in this thesis. We discuss the L-shaped method — an efficient solution algorithm that takes advantage of the SAA decomposition structure.

The background in power systems we require is covered in Chapter 3. We introduce the methods for modelling power flow within an electricity network, including the π-model for transmission lines. We give our own proof on solving for power flows on transmission lines. We use this to define an optimisation program called DC optimal power flow (DC-OPF) using the power transfer distribution factor (PTDF) formulation. Here generators are chosen to produce power in order to minimise cost subject to network constraints.

In Chapter 4 we develop our stochastic-optimal power flow (S-OPF) building on the background presented in Chapter 2 and Chapter 3. That is, we form a DC-OPF using the PTDF formulation of Chapter 3 as a two stage stochastic program with fixed recourse as in Chapter 2. Having adapted a model for generic storage and designed a model for interconnectors for the networks of interest, we devise a methodology for using the S-OPF to simulate optimal dispatch of power over time. We generate scenarios using time series models and implement our methodology on test networks with storage and interconnectors. We explain how the stochastic program choses generators to enforce feasibility, and how varying network constraints changes solutions. The cost benefits of the S-OPF are demonstrated through comparison with an equivalent deterministic optimal power flow. We also explore the relationship between solution accuracy and number of scenarios used.

In preparation for running S-OPF on the South Australian network in Chapter 6

we analyse South Australian data for power demand and wind generation capacity in Chapter 5. After covering relevant background on time series analysis, using real data we fit time series models for the wind capacity for each wind farm in South Australia and total power demand in South Australia. Our models show a strong fit with historical data. We use these models to generate scenarios for the random variables that we use in Chapter 6 to run S-OPF on the South Australian network.

Chapter 6 builds on all previous chapters to run S-OPF on the South Australian network. First we construct a model of the South Australian network that is suitable for implementing our S-OPF. Using the scenarios generated for uncertain variables in Chapter 5 we run S-OPF on the South Australian network. We compare to results from a copperplate network and also consider the effect of the new SA-NSW interconnector. From this, we analyse where storage is optimally placed within the South Australia network as the network transitions to 100% renewable generation.

In the appendix we give additional relevant details for the thesis. This includes additional results for power flows in Appendix A, the algorithms for implementing the storage model in Appendix B, further details on time series modelling Appendix C, and additional inputs for the South Australian network in Appendix D.

Chapter 2

Stochastic linear programming

This chapter provides an introduction to stochastic programming with a focus on two-stage stochastic programs with fixed recourse. After discussing notation, we begin in Section 2.2 with recapping fundamental definitions and results in convex and linear programming. Next in Section 2.3 we cover the required probability theory before in Section 2.4 introducing a two-stage stochastic program with fixed recourse. We then proceed to discuss existence of solutions in Section 2.4.3, followed by solution methods with a focus on the L-shaped method in Section 2.5. Finally in Section 2.6 we consider program performance measures and accuracy of the sampling average approximation.

While we generally follow the literature in this chapter, particularly Birge and Louveaux [41] as the main reference for stochastic programming and Boyd and Vandenberghe [46] for convex programming, we deviate away from the literature in Section 2.4.3 with our own results concerning the existence of solutions of stochastic programs.

2.1 Notation

Within these notes we use bold font to represent random variables and bold font with subscript to represent their realisations. We will use the notation that for vectors x, y then $x \geq y$ means that each element of x is greater than the corresponding element in y.

2.2 Convex programming

We begin with background on convex programming, which will form the foundation for our approach to stochastic programming. As discussed in Section 2.4.2, a two-stage stochastic program with fixed recourse is generally convex, and usually non-linear once the number of scenarios is greater than 1. Boyd and Vandenberghe [46] are the main reference for this section.

Definition 2.1. We define a *convex function* as a function $f : \mathbb{R}^n \to \mathbb{R}$ such that

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y),$$

for all $t \in [0, 1]$. Examples of convex functions include *linear functions* $f(x) = Ax : \mathbb{R}^m \to \mathbb{R}^n$ and the more general *affine functions* $f(x) = Ax + b : \mathbb{R}^m \to \mathbb{R}^n$, where A is an $n \times m$ real matrix and $b \in \mathbb{R}^n$.

Definition 2.2. A *convex set* $\mathcal{D}$ is any set where if $x, y \in \mathcal{D}$ then $tx + (1-t)y \in \mathcal{D}$ for all $t \in [0, 1]$.

This leads to defining a convex program as follows.

Definition 2.3. *Convex programs* are optimisation problems of the form

$$\begin{aligned} &\min_x f(x) \\ \text{such that} \quad &g(x) = b \\ &h(x) \leq d. \end{aligned} \tag{2.1}$$

We have $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and $d \in \mathbb{R}^\ell$. We require that the *objective function* f is a convex function of x, that $g = (g_1, \ldots, g_m) : \mathbb{R}^n \to \mathbb{R}^m$ is linear so that $g(x) = Ax$ for some real $m \times n$ matrix A, and that $h = (h_1, \ldots, h_\ell) : \mathbb{R}^n \to \mathbb{R}^\ell$ are such that each h_j is a convex function of x. The optimal decision is made with respect to x and hence x is referred to as the *decision* vector.

We denote the optimal objective value to Program 2.1 as f^* and we denote x^* as any corresponding x that lead to this minimum, which may not be unique. Values x that satisfy the constraints $g(x) = b, h(x) \leq d$ are called *feasible*. We call the set of all feasible solutions the *feasible set*

$$\mathcal{D} = \{x \mid g(x) = b, h(x) \leq d\}.$$

We will assume that there is a least one feasible solution $x \in \mathcal{D}$ so that Program 2.1 has either a finite solution or it has solution $-\infty$.

The feasible set $\mathcal{D}$ of Program 2.1 is a *convex* set. This is straightforward to prove using two results. Firstly, we use that intersections of convex sets are a convex set. Secondly, the inverse image under convex functions of intervals that are unbounded below are convex sets. For this latter statement, consider that if f is convex and $x_1, x_2 \in f^{-1}(-\infty, d]$ for $d \in \mathbb{R}$, then

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2) \leq td + (1-t)d = d$$

and so $tx_1 + (1-t)x_2 \in f^{-1}(-\infty, d]$ and hence $f^{-1}(-\infty, d]$ is convex. Similarly for $f^{-1}(-\infty, d)$.

2.2.1 The Lagrangian

Definition 2.4. The *Lagrangian* of Program 2.1 is the function

$$\mathcal{L}(x, \lambda, \gamma) = f(x) + \lambda^\top (g(x) - b) + \gamma^\top (h(x) - d).$$

Here $\lambda \in \mathbb{R}^m$ and $\gamma \in \mathbb{R}^\ell$ are called *Lagrange multipliers*, which are also called *dual* or *simplex multipliers*.

Definition 2.5. The *Lagrange dual function* of Program 2.1 is defined as the function

$$\mathcal{L}(\lambda, \gamma) = \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \gamma).$$

Note that if we take any $x \in \mathbb{R}^n$ then $\mathcal{L}(x, \lambda, \gamma)$ is an affine function in λ and γ. As we know the infimum over a collection of affine functions is a concave function [46, pg. 80-81], then we have that $\mathcal{L}(\lambda, \gamma)$ is a *concave function*, that is, it is the negative of a convex function.

Proposition 2.6 (Weak duality)**.** *Whenever $\lambda \in \mathbb{R}^m$ and $\gamma \in \mathbb{R}^\ell, \gamma \geq 0$, then $\mathcal{L}(\lambda, \gamma)$ is a lower bound for the optimal value f^* of Program 2.1.*

Proof. Suppose $x \in \mathcal{D}$ then $h(x) - d \leq 0$ and as $\gamma \geq 0$ we have $\gamma^\top (h(x) - d) \leq 0$. Further, as x is feasible we have $g(x) - b = 0$ and hence $\lambda^\top (g(x) - b) = 0$. Then by definition of the Lagrangian for $x \in \mathcal{D}$ we have

$$\begin{aligned} \mathcal{L}(x, \lambda, \gamma) &= f(x) + \lambda^\top (g(x) - b) + \gamma^\top (h(x) - d) \\ &\leq f(x), \end{aligned}$$

for any $\lambda \in \mathbb{R}^m, \gamma \geq 0$. It follows that

$$\begin{aligned} \mathcal{L}(\lambda, \gamma) = \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \gamma) &\leq \inf_{x \in \mathcal{D}} \mathcal{L}(x, \lambda, \gamma) \\ &= \inf_{x \in \mathcal{D}} f(x) + \lambda^\top (g(x) - b) + \gamma^\top (h(x) - d) \\ &\leq \inf_{x \in \mathcal{D}} f(x) = f^*. \end{aligned}$$

Note that the first inequality follows as $\mathcal{D} \subseteq \mathbb{R}^n$. So we have $\mathcal{L}(\lambda, \gamma) \leq f^*$ as required. □

This means the Lagrangian dual function provides a lower bound for Program 2.1 parameterised by $\lambda \in \mathbb{R}^m$ and $\gamma \geq 0$. Naturally this leads to the question: What is the best lower bound for a convex program? More specifically, what is the best lower bound given by the Lagrange dual function? The best lower bound given by the Lagrange dual function is the solution to the *dual convex program*.

Definition 2.7. The *dual convex program* of Program 2.1 is

$$\begin{aligned} &\max_{\lambda,\gamma} \mathcal{L}(\lambda,\gamma) \\ &\text{such that } \gamma \geq 0. \end{aligned} \tag{2.2}$$

Note that while $\mathcal{L}(\lambda,\gamma)$ is a concave function, not convex, we still refer to Program 2.2 as a convex program, since maximising a concave function is equivalent to minimising the negative of this function, which is convex. Solving this dual convex program usually requires specific knowledge of f and h to write in the form of Program 2.1 or similar, which we will do in detail for linear programs below.

If we have that the optimal value of Program 2.2 is equal to the optimal value of Program 2.1 then we say that these programs have *strong duality*. We do not have strong duality in general for convex programs, however a condition such as *Slater's constraint qualification* can ensure strong duality for a large class of convex programs. This requires the existence of at least one feasible point x of Program 2.1 such that we have the strict inequality $h(x) < d$. See [46, §5.2.3] for details.

An important class of convex programs which do have strong duality are linear programs, which we consider in the next section.

2.2.2 Linear programming

We are particularly interested in a subclass of convex programs called *linear programs*. This is where we are optimising a linear objective function z subject to linear constraints. In this program, all inputs are deterministic. We will subsequently consider extending to non-deterministic inputs in the following sections.

Definition 2.8. We define a *linear program* as a optimisation program of the form

$$\begin{aligned} \min_{x} \quad & z = c^\top x \\ \text{subject to } \; & Ax = b, \\ & Cx \leq d. \end{aligned} \tag{2.3}$$

Here $x \in \mathbb{R}^n, c \in R^n, b \in \mathbb{R}^m, d \in \mathbb{R}^\ell$, C is a real $\ell \times n$ matrix and A is a real $m \times n$ matrix.

Note if we further require that $x \geq 0$, i.e. that all elements of x are non-negative, and all inequality constraints are converted to equality constraints then this is called *standard form*. All linear programs can be written in this form. For example, if we would like to write $Cx \leq d$ as an equality constraint, we can add slack variables $s \in \mathbb{R}^\ell$ to the linear program with $s \geq 0$ and modify the constraint $Cx \leq d$ to $Cx + s = d$. Alternatively, if we require $x \in \mathbb{R}^n$ not necessarily non-negative then we can use variables $y_1, y_2 \in \mathbb{R}^n$, $y_1, y_2 \geq 0$ and replace x by $y_1 - y_2$ wherever it appears in the linear program.

As with convex programs, we define z^* to be the optimal value of Program 2.3 and write x^* for any corresponding x that leads to this minimum, which may not be unique. All x that satisfy the constraints $Ax = b, Cx \leq d$ are called *feasible* solutions. We will assume that there is a least one feasible solution x so that Program 2.3 has either a finite solution or it has solution $-\infty$.

Linear programs have a specific *Lagrangian* function

$$\mathcal{L}(x, \lambda, \gamma) = c^\top x + \lambda^\top (Ax - b) + \gamma^\top (Cx - d),$$

where $\lambda \in \mathbb{R}^m$ and $\gamma \in \mathbb{R}^\ell$ are the Lagrange multipliers as in Definition 2.4.

The Lagrange dual function of Program 2.3 from Definition 2.5 is

$$\mathcal{L}(\lambda, \gamma) = \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \gamma).$$

We can reduce this as follows:

$$\begin{aligned}
\mathcal{L}(\lambda, \gamma) &= \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \gamma) \\
&= \inf_{x \in \mathbb{R}^n} (c^\top x + \lambda^\top (Ax - b) + \gamma^\top (Cx - d)) \\
&= \inf_{x \in \mathbb{R}^n} (c^\top + \lambda^\top A + \gamma^\top C)x - \lambda^\top b - \lambda^\top d.
\end{aligned}$$

Now if there is an $x \in \mathbb{R}^n$ such that for given λ, γ we have $(c^\top + \lambda^\top A + \gamma^\top C)x < 0$ (or $(c^\top + \lambda^\top A + \gamma^\top C)x > 0$), then multiplying this x by more positive numbers (or negative numbers respectively) will show that the infimum

$$\inf_{x \in \mathbb{R}^n} (c^\top + \lambda^\top A + \gamma^\top C)x$$

is $-\infty$. On the other hand, if we have that $(c^\top + \lambda^\top A + \gamma^\top C)x = 0$ for all $x \in \mathbb{R}^n$ and a given λ, γ, then $\inf_{x \in \mathbb{R}^n} (c^\top + \lambda^\top A + \gamma^\top C)x$ is 0, and the Lagrangian dual is equal to $-\lambda^\top b - \lambda^\top d$.

The requirement that $(c^\top + \lambda^\top A + \gamma^\top C)x = 0$ for all $x \in \mathbb{R}^n$ is equivalent to requiring $c^\top + \lambda^\top A + \gamma^\top C = 0$. This can be shown by considering x to be each of the standard basis vectors $e_i \in \mathbb{R}^n$, which are vectors of all zeros except for a 1 as the i-th element.

Therefore we have

$$\begin{aligned}
\mathcal{L}(\lambda, \gamma) &= \inf_{x \in \mathbb{R}^n} \mathcal{L}(x, \lambda, \gamma) \\
&= \inf_{x \in \mathbb{R}^n} \left(c^\top x + \lambda^\top (Ax - b) + \gamma^\top (Cx - d) \right) \\
&= \inf_{x \in \mathbb{R}^n} \left(c^\top + \lambda^\top A + \gamma^\top C \right) x - \lambda^\top b - \gamma^\top d \\
&= \begin{cases} -\lambda^\top b - \gamma^\top d & \text{if } c^\top + \lambda^\top A + \gamma^\top C = 0 \\ -\infty & \text{otherwise.} \end{cases}
\end{aligned} \tag{2.4}$$

Whenever $\lambda \in \mathbb{R}^m$ and $\gamma \geq 0$ we have weak duality by Proposition 2.6. This means $\mathcal{L}(\lambda, \gamma)$ is a lower bound for the optimal value z^* of Program 2.3 for such γ, λ. Note that if z^* is $-\infty$, then $\mathcal{L}(\lambda, \gamma)$ is as well and Proposition 2.6 still holds.

2.2.3 Duality of linear programs

We can formulate the best lower bound for z^* as finding the optimal solution to the following linear program

$$\begin{aligned} \max_{\lambda,\gamma} w = -\lambda^\top b - \gamma^\top d \\ \text{such that } \lambda^\top A + \gamma^\top C \geq -c^\top \\ \gamma \geq 0. \end{aligned}$$

However, we often rename γ and λ as their negatives to write this in standard form as follows

$$\begin{aligned} \max_{\lambda,\gamma} w = \lambda^\top b + \gamma^\top d \\ \text{such that } \lambda^\top A + \gamma^\top C \leq c^\top \\ \gamma \leq 0. \end{aligned} \tag{2.5}$$

This is called the *dual* linear program of Program 2.3, and we usually refer to Program 2.3 as the *primal* linear program. The dual linear program is simply the dual convex program of Program 2.3 were we are assuming a finite maximum is obtained. We label the optimal value to this linear program as w^* and any corresponding λ, γ that give this optimal value are labeled λ^*, γ^* and are often referred to as the *Lagrange* multipliers or *dual* variables associated to the optimal solution x^* of the primal. Note that taking the dual of the dual gives the primal linear program.

Theorem 2.9. *The optimal value w^* to dual program Program 2.5 is the same as the optimal value z^* to the primal program Program 2.3 where both are finite.*

This is a non-trivial result with many known proofs. A stronger version of this is the following theorem.

Theorem 2.10 (Strong duality)**.** *If either the dual or the primal have a finite solution, or there are feasible solutions to both the dual and the primal problem, then there is a finite optimal solution x^* to the primal and a corresponding finite optimal solution λ^*, γ^* to the dual such that $z^* = w^*$.*

We will not prove this here, however we note that one of the reasons the proof is non-trivial is that the constraints for both linear programs confine feasible solutions to

convex sets, which are not linear subspaces but rather intersections of affine subspaces. This aspect requires the proof to use facts beyond pure linear algebra. We refer the reader to Matousek and Gärtner [83, §6] where they describe two proofs, one using the Farkas Lemma, the other using the Simplex Algorithm, a well known algorithm for solving linear programs.

2.3 Introducing stochasticity

Before defining stochastic optimisation problems we first recall some definitions from probability theory.

Definition 2.11. Let Ω be a non-empty set, then a *σ-field* $\mathcal{F}$ is a collection of subsets of Ω such that:

- $\varnothing \in \mathcal{F}$,
- if $A \in \mathcal{F}$ then $\Omega \setminus A \in \mathcal{F}$,
- if $A_i \in \mathcal{F}$ for all $(A_i)_{i\in\mathbb{N}}$ then $\bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$.

Definition 2.12. Let Ω be a set and $\mathcal{F}$ be a σ-field of subsets of Ω. A *probability measure* on $(\Omega, \mathcal{F})$ is a function $\mathbb{P} : \mathcal{F} \to [0, 1]$ such that $\mathbb{P}(\Omega) = 1$ and $\mathbb{P}$ is countably additive, that is, for all countable collections $\{A_i\}$ of pairwise disjoint sets then $\mathbb{P}\big(\bigcup_{i=1}^{\infty} A_i\big) = \sum_i^{\infty} \mathbb{P}(A_i)$.

Definition 2.13. A *probability space* is triple $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω is a set, $\mathcal{F}$ is a σ-field of subsets of Ω, and $\mathbb{P}$ is a probability measure on $(\Omega, \mathcal{F})$. We call sets in $\mathcal{F}$ *measurable*.

Lemma 2.14. *If $\{\mathcal{F}_i : i \in I\}$ is a family of σ-fields of subsets of Ω then $\mathcal{F} = \bigcap_{i\in I} \mathcal{F}_i$ is also a σ-field.*

Definition 2.15. Take any family $\mathcal{G}$ of subsets of Ω, the σ-field *generated* by $\mathcal{G}$ denoted $\sigma(\mathcal{G})$ is given by the intersection of all σ-fields containing $\mathcal{G}$. By construction this is the smallest σ-field containing $\mathcal{G}$.

Definition 2.16. Let $\mathscr{J}$ be the collection of all open intervals in $\mathbb{R}$. We define the *Borel* σ-field as the σ-field generated by $\mathscr{J}$. We denote the *Borel* σ-field with $\mathcal{B}$ and call the sets in $\mathcal{B}$ *Borel sets*. We note that there are other equivalent definitions for the *Borel* σ-field; for example it can be generated by the collection of all intervals $(-\infty, x]$ for all $x \in \mathbb{R}$.

Definition 2.17. Let $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ be two measurable spaces. Then a *random variable* $\boldsymbol{\zeta} : \Omega_1 \to \Omega_2$ is a *measurable* function. That is, for every $B \in \mathcal{F}_2$ the pre-image $\boldsymbol{\zeta}^{-1}(B)$ is in $\mathcal{F}_1$. Further, if $\Omega_2 = \mathbb{R}$ and $\mathcal{F}_2 = \mathcal{B}$ we call $\boldsymbol{\zeta}$ a *real-valued* random variable.

2.4 Optimisation with uncertainty

When forming optimisation programs there may be parameters which have a random component. How this randomness is introduced into the program will vary based on what information is uncertain and what is being optimised.

We will consider solving an optimisation program where there is uncertainty within the constraints or uncertainty in the coefficients of the linear objective function. This can introduce significant issues, as some decisions may need to be made before the outcome of the uncertainty is known, and this may lead to a solution that is not optimal nor even feasible given the outcome of the uncertainty. However, if the uncertainty can be represented by random variables with known probability distributions then there are methods for forming a well defined program, as we outline in the next section.

While this section focuses on theory, from a practitioners perspective we suggest focussing on the stochastic programs in Program 2.6 and Definition 2.22. Beyond this, it is important to understand that for stochastic programs of this form, while it is not guaranteed that solutions exist, there are conditions that can ensure their existence as we discuss in Section 2.4.3 using the properties in Section 2.4.2.

2.4.1 Two-stage stochastic program with fixed recourse

One approach for handling the issues introduced by the uncertainty in parameters is to allow for a *recourse action*. That is, initially a set of decisions is made subject to the deterministic constraints, then after the random variables have been realised, a second set of *recourse decisions* are made. To do this, we can divide the program into two stages: the *first stage* where a decision is made prior to the random variables being realised, and a *second stage* where a recourse action is made given the first stage decision and the realisations of random variables.

This recourse action involves making a *second stage* decision y given the realisation $\boldsymbol{\zeta}_\omega$ of the random variable $\boldsymbol{\zeta}$. However the second stage decision has an associated cost. Consequently the objective function is updated to account for this cost, adding the expected recourse cost. This is known as a *two-stage stochastic program with fixed recourse*. We thus define such a program formally as follows:

Definition 2.18. A *two-stage stochastic program with fixed recourse* is an optimisation program of the form

$$\begin{aligned} \min_{x} \quad & z = c^\top x + \mathbb{E}_{\boldsymbol{\zeta}}[Q(x, \boldsymbol{\zeta})] \\ \text{subject to} \quad & Ax = b, \\ & x \geq 0, \end{aligned} \tag{2.6}$$

where for each realisation $\boldsymbol{\zeta}_\omega$ of $\boldsymbol{\zeta}$ we have

$$\begin{aligned} Q(x, \boldsymbol{\zeta}_\omega) = \min_y \quad & \boldsymbol{q}_\omega{}^\top y \\ \text{subject to } & \boldsymbol{T}_\omega x + W y = \boldsymbol{h}_\omega, \\ & x, y \geq 0. \end{aligned} \tag{2.7}$$

Here

- $x \in \mathbb{R}^{n_1}$ contains the first stage decision variables and $c \in \mathbb{R}^{n_1}$ is the associated first stage marginal cost;
- A is an $m_1 \times n_1$ matrix and $b \in \mathbb{R}^{m_1}; y \in \mathbb{R}^{n_2}$ is the second stage decision variables and $\boldsymbol{q_\omega} \in \mathbb{R}^{n_2}$ is the associated second stage stage marginal cost; and
- $\boldsymbol{h}_\omega \in \mathbb{R}^{m_2}$ is a random vector and $\boldsymbol{T}_\omega$ is an $m_2 \times n_1$ random matrix.

By taking $\boldsymbol{T}_\omega^i$ to be the i-th row of $\boldsymbol{T}_\omega$ we form a realisation the random vector $\boldsymbol{\zeta}_\omega = (\boldsymbol{q}_\omega^\top, \boldsymbol{h}_\omega^\top, \boldsymbol{T}_\omega^1, \ldots, \boldsymbol{T}_\omega^{m_2})^\top$ with N components where $N = n_2 + m_2 + (m_2 \times n_1)$. Finally, W is called the *recourse matrix* and it has dimensions $m_2 \times n_2$. As W does not depend on $\boldsymbol{\zeta}$ we say that the program has *fixed recourse*, and we call Wy the *recourse action*.

Note that ω indexes each realisation of $\boldsymbol{\zeta}$, so we may have for example $\omega = 1, 2, 3, \ldots, ..$ if there are a discrete number of realisations of $\boldsymbol{\zeta}$.

We call this *standard form* for a two-stage stochastic program with fixed recourse and note that any inequality constraints and any requirement for the constraint $x \geq 0$ to be relaxed can be incorporated analogously to Definition 2.8. The minimisation in Equation (2.7) is called the *second stage* of the program and is a linear program. However the minimisation in Program 2.6 is usually a convex program, as the function $Q(x, \boldsymbol{\zeta}_\omega)$ and the feasible region are often convex. We will discuss this further in Section 2.4.2.

We will define the following sets that will be useful for describing existence of solution for a two-stage stochastic program with fixed recourse.

Definition 2.19. Let K_1 be the feasible region with respect to the first stage decision x, so that

$$K_1 = \{x \mid Ax = b, x \geq 0\}.$$

We define $K_2(\boldsymbol{\zeta}_\omega)$ to be the *elementary feasibility set* given by

$$K_2(\boldsymbol{\zeta}_\omega) = \{x \mid \text{ there exists } y \geq 0 \text{ s.t. } Wy = \boldsymbol{h}_\omega - \boldsymbol{T}_\omega x\}.$$

Here $K_2(\boldsymbol{\zeta}_\omega)$ is a function of $\boldsymbol{\zeta}_\omega$, which is a realisation of the random variable $\boldsymbol{\zeta}$.

We define the *second stage feasibility* set as

$$K_2 = \left\{ x \;\middle|\; \mathbb{E}\left(Q(x, \boldsymbol{\zeta})\right) < \infty \right\}.$$

We also define

$$K_2^P = \bigcap_{\omega} K_2(\boldsymbol{\zeta}_\omega)$$

and note that K_2 and K_2^P often coincide, as we will discuss in Section 2.4.2.

We see that the feasible region for finite solutions to Program 2.6 is

$$\mathcal{D} = K_1 \cap K_2, \tag{2.8}$$

as explained further in Section 2.4.2. First we discuss alternative forms of Program 2.6 that will be useful in the coming analysis.

Definition 2.20. An alternative form of Program 2.6 is

$$\begin{aligned} \min_{x,\theta} z &= c^\top x + \theta \qquad (2.9)\\ \text{such that } Ax &= b,\\ \mathscr{Q}(x) &\leq \theta,\\ x &\in \mathcal{D},\\ \mathscr{Q}(x) &= \mathbb{E}_{\boldsymbol{\zeta}}[Q(x, \boldsymbol{\zeta})]. \end{aligned}$$

This is equivalent to Program 2.6 as minimising with respect to x and θ will set $\theta^* = \mathscr{Q}(x^*)$, and requiring $x \in K_1 \cap K_2$ enforces the same constraints as Program 2.6.

If the random vector $\boldsymbol{\zeta}$ is discrete with a finite sample space Ω then the two-stage stochastic program with fixed recourse can be written as a *deterministic equivalent program*. This is often referred to as *finite extensive form*. We do this by considering the explicit form of the expected recourse as follows.

$$\begin{aligned} \mathbb{E}_\zeta[Q(x, \boldsymbol{\zeta})] &= \sum_{k=1}^{s} p_k Q(x, \boldsymbol{\zeta}_k)\\ &= \sum_{k=1}^{s} p_k \min_{y_k} \boldsymbol{q}_k^\top y_k\\ &= \min_{y_1,\ldots,y_s} \sum_{k=1}^{s} p_k \boldsymbol{q}_k^\top y_k \qquad (2.10) \end{aligned}$$

where we are also minimising over $\boldsymbol{T}_k x + W y_k = \boldsymbol{h}_k$ for each y_k in the second and third lines. Here s is the number of outcomes for $\boldsymbol{\zeta}$ in Ω and p_k is the probability of outcome $\boldsymbol{\zeta}_k$. Note that y_k is the second stage decision corresponding to outcome $\boldsymbol{\zeta}_k$.

Definition 2.21. Suppose $\boldsymbol{\zeta} = [\boldsymbol{q}^\top, \boldsymbol{h}^\top, \boldsymbol{T}_1^\top, \ldots, \boldsymbol{T}_{m_2}^\top]^\top$ is a discrete random vector with a finite sample space taking s realisations. A *deterministic-equivalent program* to Program 2.6 is the optimisation program

$$\begin{aligned} \min_{x, y_1, \ldots, y_s} \quad & c^\top x + \sum_{k=1}^{s} p_k \boldsymbol{q}_k^\top y_k \\ \text{subject to :} & \\ Ax &= b, \\ \boldsymbol{T}_k x + W y_k &= \boldsymbol{h}_k \quad \text{for} \quad k = 1, \ldots, s \\ x, y_k &\geq 0, \quad \text{for} \quad k = 1, \ldots, s. \end{aligned}$$

Note that there is a separate second stage decision vector y_k for each realisation of the random variable. The decisions y_k are referred to as the second stage decision for the k-th *scenario*. This is a linear program as the minimisation over the y_k occurs at the same time as minimising over the x, so the constraints and the objective function are all linear or affine.

In general we can form the *sampling average approximation* drawing r samples $\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \ldots, \boldsymbol{\zeta}_r$ using Monte-Carlo methods. This is often used when the random variable $\boldsymbol{\zeta}$ is countable or continuous. This looks similar to above, however here we are taking samples of the random variable with each sample forming a scenario with equal probability of $1/r$.

Definition 2.22. Let $\boldsymbol{\zeta}$ be a continuous random variable where r samples have been taken, then the *sampling average approximation* program (SAA) of Program 2.6 is the linear program

$$\begin{aligned} \min_{x, y_1, \ldots, y_r} \quad & c^\top x + \frac{1}{r} \sum_{k=1}^{r} \boldsymbol{q}_k^\top y_k \\ \text{subject to :} & \\ Ax &= b, \\ \boldsymbol{T}_k x + W y_k &= \boldsymbol{h}_k \quad \text{for} \quad k = 1, \ldots, r \\ x, y_k &\geq 0, \quad \text{for} \quad k = 1, \ldots, r. \end{aligned}$$

.

It is important to note that the SAA is only an approximation of Program 2.6 and hence any solution for the SAA may not be an optimal solution for Program 2.6. However under certain conditions the sampling average approximation converges to the solution of Program 2.6 as the number of samples increases and the error can be quantified. This convergence is discussed in further in Section 2.7.

2.4.2 Properties of Program 2.6

We will first discuss properties of Program 2.6, considering the cases when $\boldsymbol{\zeta}$ is discrete separately. We start by discussing convexity, recalling from Definition 2.19 that K_1 is the feasible region with respect to the first stage decision x, and K_2 is the set of all x such that the expected recourse is finite. Note that K_1 is automatically convex and closed, as the constraints $Ax = b$ and $x \geq 0$ are affine. To discuss properties of K_2 we first define convex polyhedrons,

Definition 2.23. A *convex polyhedron* $\mathcal{P}$ is any convex subset of $\mathbb{R}^n$ that is the intersection of finitely many *half-spaces* $\{x \in \mathbb{R}^n \mid a_1^T x \leq b\}$, that is

$$\mathcal{P} = \bigcap_{i=1}^{m} \{x \mid a_i^T x \leq b_i\} = \{x \mid Ax \leq b\},$$

where the vectors $a_i \in R^n$ form the rows of the $m \times n$ matrix A and the elements $b_i \in \mathbb{R}$ form the vector $b \in \mathbb{R}^m$. Then $\mathcal{P}$ is also called *polyhedral*.

We now state two theorems without proof from Birge and Louveaux [41].

Theorem 2.24. *If $\boldsymbol{\zeta}$ is a discrete random variable then from Birge and Louveaux [41, §3.1b, Thm. 1&2] we have that*

1. $K_2 = K_2^P = \bigcap_\omega K_2(\boldsymbol{\zeta}_\omega)$
2. *For a given ω, $K_2(\boldsymbol{\zeta}_\omega)$ is a convex polyhedron, and therefore K_2 is a convex polyhedron.*
3. *For a given ω, then $Q(x, \boldsymbol{\zeta}_\omega)$ is piecewise linear convex function in $\boldsymbol{h}_\omega, \boldsymbol{T}_\omega$ and all $x \in K_2$, and is piecewise linear concave function in $\boldsymbol{q}_\omega$.*
4. *If $\boldsymbol{\zeta}$ is finite then $\mathscr{Q}(x) = \mathbb{E}[Q(x, \boldsymbol{\zeta})]$ is a piecewise linear convex function of x on K_2.*

We can extend Theorem 2.24 beyond the requirements that $\boldsymbol{\zeta}$ is discrete or finite in the next theorem, provided we instead assume that $\boldsymbol{\zeta}$ has finite second moments. Note that finite second moments imply finite first moments (so finite mean) by Jensen's inequality (see for example Durret [57, pg. 29]) and many distributions have this property, such as the normal distribution, and the exponential distribution.

Before this next theorem, recall that *Lipschitzian* is a strong form of continuity where there is a given constant $s > 0$ such that for all x_1, x_2,

$$|\mathscr{Q}(x_1) - \mathscr{Q}(x_2)| \leq s|x_1 - x_2|.$$

For a convex set K, we also define the *relative interior* of K to be the set

$$\text{reint}(K) = \{x \in K \mid \forall y \in K, \exists \lambda > 1 \text{ such that } \lambda x + (1-\lambda)y \in K\}.$$

Note that the relative interior of K_2 is similar to the interior of a set in $\mathbb{R}^n$ however it is more useful for convex optimisation. It coincides with the interior in many cases, however gives more information when K_2 does not contain a basis that spans $\mathbb{R}^n$. See Hiriart-Urruty and Lemaréchal [67, pg. 103] or Boyd and Vandenberghe [46, pg. 23] for more details.

Theorem 2.25. *If $\boldsymbol{\zeta}$ is discrete or continuous and it has finite second moments then from Birge and Louveaux [41, §3.1c, Thm. 4-6] we have that*

1. $K_2 = K_2^P = \bigcap_\omega K_2(\boldsymbol{\zeta}_\omega)$
2. *K_2 is closed and convex*
3. *If T does not depend on $\boldsymbol{\zeta}$ then K_2 is polyhedral*
4. *If the support of $\boldsymbol{\zeta}$ is polyhedral and $\boldsymbol{T}_\omega$ and $\boldsymbol{h}_\omega$ are independent random variables then K_2 is polyhedral*
5. *$\mathscr{Q}(x)$ is Lipschitzian, convex and finite on K_2*

Note that a distribution with finite support has finite moments of all orders, as all finite sums converge, so Theorem 2.25 applies whenever $\boldsymbol{\zeta}$ has finite support.

Theorem 2.24 and Theorem 2.25 are important as they explain when Program 2.6 is a convex optimisation program, and therefore when convex optimisation programming techniques can be applied to find the optimal solutions.

2.4.3 Existence of solutions for two-stage stochastic programs with fixed recourse

We now discuss how properties of K_1 and K_2 imply whether we can find optimal solutions to Program 2.6. This section offers a different perspective to Birge and Louveaux [41] and the results in Lemma 2.26, Lemma 2.27, Proposition 2.28, Proposition 2.29, and Proposition 2.36 are, as far as we are aware, novel.

We begin by writing the objective function of Program 2.6 as the function

$$z : \mathcal{D} \to \mathbb{R},\ z(x) = c^\top x + \mathscr{Q}(x).$$

Then the optimal value of Program 2.6, if it exists, is the minimum value of the image of $\mathcal{D}$ under z. The corresponding x values x^* are the inverse image of the optimal value z^* under z. We will show that $z(\mathcal{D})$ is connected, and therefore is an interval. Then we discuss conditions on K_1, K_2 and $\boldsymbol{\zeta}$ that ensure an optimal value is found, which will require that $z(\mathcal{D})$ is non-empty, bounded below and closed from below.

Lemma 2.26. *The image of $\mathcal{D}$ under z is an interval whenever $\boldsymbol{\zeta}$ has finite second moments.*

Proof. The conditions that $Ax = b$, $x \geq 0$, $\boldsymbol{T}_\omega x + Wy = \boldsymbol{h}_\omega$ and $y \geq 0$ from Program 2.6 give affine or half spaces. These are all connected, and as intersections of connected spaces are connected, then $\mathcal{D}$ is connected. By Theorem 2.25, point 3, whenever $\boldsymbol{\zeta}$ has finite second moments, then $\mathscr{Q}(x)$ is continuous and so is $z(x)$. As connectedness is preserved under continuous maps then, $z(\mathcal{D})$ is connected whenever $\boldsymbol{\zeta}$ has finite second moments. □

Now, suppose that $\mathscr{Q}(x) = \infty$ for all $x \in K_1$. This would imply the minimum over all realisations of $\boldsymbol{\zeta}$ is ∞. This would then mean the optimum value is ∞, and there is no need to solve anything.

Instead, we would like to assume that there exist $x \in K_1$ such that $\mathscr{Q}(x) < \infty$, or equivalently that the feasible set $\mathcal{D} = K_1 \cap K_2$ is non-empty. This means there is at least one feasible x that has non-infinite objective function z, and as the objective is to minimise over all such z, then the two-stage stochastic program with fixed recourse (if it has a solution) must have a solution less than ∞.

To ensure that the solution (if it exists) is finite we require the interval $z(\mathcal{D})$ to be bounded below. This occurs, for example, when $c \geq 0$ and $\mathscr{Q}$ is bounded below on $\mathcal{D}$. Notably, if $\boldsymbol{q} \geq 0$ for all realisations $\boldsymbol{q}_\zeta$ then $\mathscr{Q}(\mathcal{D}) \geq 0$ is bounded below.

We also require that $z(\mathcal{D})$ is closed from below, so that the minimum exists and is equal to the infimum. This occurs for example when $z(\mathcal{D})$ is a closed set. If we have that $z(\mathcal{D})$ is non-empty, closed and bounded below, then a finite solution will be attained for Program 2.6.

We note that we have the following lemma.

Lemma 2.27. *The set K_1 is closed.*

Proof. As the conditions $Ax = b$ and $x \geq 0$ give rise to closed sets and intersections of closed sets are closed, then K_1 is closed. □

We therefore have the following proposition.

Proposition 2.28. *If $\boldsymbol{\zeta}$ has a finite distribution; $\mathcal{D}$ is non-empty; $c \geq 0$, and $\boldsymbol{q} \geq 0$, then Program 2.6 has a finite optimal value.*

Proof. As $\boldsymbol{q} \geq 0$, then $\mathscr{Q}$ is bounded below and then $c \geq 0$ implies z is bounded below on $\mathcal{D}$. Now Theorem 2.25 means $\mathscr{Q}$ is piecewise linear and K_2 closed, and we also know that K_1 is closed by Lemma 2.27. Then $\mathcal{D} = K_1 \cap K_2$ is the intersection of closed sets and so is also closed. As images of closed sets under piecewise linear functions are closed, then the image of $\mathcal{D}$ under z is non-empty, closed and bounded below, so must attain its minimum. Hence Program 2.6 has a finite optimal value. □

More generally we have the following proposition.

Proposition 2.29. *If $\mathcal{D}$ is nonempty, closed and bounded, and $\mathscr{Q}$ is continuous, then a finite optimal value is obtained for Program 2.6.*

Proof. If $\mathcal{D}$ is closed and bounded, then as it is a subset of $\mathbb{R}^n$ it is compact as in Munkres [88, Thm. 27.5]. Also, in general we know that images of compact sets are compact under continuous functions, as in Munkres [88, Thm. 26.5]. As $c^T x$ is linear it is continuous, and then z is continuous if and only if $\mathscr{Q}$ is continuous. If $\mathcal{D}$ is compact and non-empty, and $\mathscr{Q}$ is continuous, then the image of z is compact and non-empty, so a finite solution is obtained. This implies that $\mathscr{Q}$ is bounded on $\mathcal{D}$ and indeed bounded below, so this is stronger than the previous condition. □

Note that Theorem 2.25 says that if $\boldsymbol{\zeta}$ has finite second moments then $\mathscr{Q}$ is continuous. Also note that K_1 is closed as in Lemma 2.27, however, K_2 may not be closed as the condition on x that $\mathscr{Q}(x) < \infty$ gives rise to an open set which is not necessarily closed. In Theorem 2.25 we saw that if $\boldsymbol{\zeta}$ has finite second moments, this is enough to guarantee that K_2 is closed.

If we have that $K_1 \subseteq K_2$ then $K_1 \cap K_2 = K_1$ is closed. This motivates the following definition

Definition 2.30. We say that Program 2.6 has *relatively complete recourse* if $K_1 \subseteq K_2$.

A stronger condition that is potentially easier to check is the following.

Definition 2.31. We have *complete recourse* if $\{s \mid s = Wy, y \geq 0\} = \mathbb{R}^{m_2}$, with m_2 as in Program 2.6.

A necessary condition for complete recourse is that $\operatorname{rank}(W) = m_2$, and this can be a quick check to see if this condition may apply. Complete recourse means that $K_2 = \mathbb{R}^n$ and therefore K_2 is both open and closed and $K_1 \subseteq K_2$.

Further a special case of complete recourse is simple recourse.

Definition 2.32. We say we have *simple recourse* when $W = [I, -I]$.

Simple recourse can often reduce the complexity of many results, although we do not consider this case in detail here. See Birge and Louveaux [41, §3.1d] for further details.

We now have a corollary to Proposition 2.29.

Corollary 2.33. *If $\mathcal{D}$ is non-empty and bounded; we have complete, relatively complete or simple recourse; and $\mathscr{Q}$ is continuous on $\mathcal{D}$; then a finite optimal value is obtained for Program 2.6.*

Birge and Louveaux [41, §3.1e, Thm. 8] give two conditions that ensure attainment of an optimal solution to Program 2.6 provided it is finite. These conditions essentially reduce to $\mathscr{Q}$ being continuous, and $\mathcal{D}$ being compact or $\mathscr{Q}$ being bounded below. We require the following definition before introducing their theorem.

Definition 2.34. A *recession direction* of a convex set S is a direction vector $d \in \mathbb{R}^n$ such that for all $\lambda > 0$ and all $u \in S$ we have that $u + \lambda d \in S$. That is, starting at any point $u \in S$, if we head in direction d we always remains in S. For a function f on S to be *eventually linear in all recession directions* means that starting at a point $u \in S$ and taking any recession direction d then there is an increment $\bar{\lambda} \geq 0$ after which f "looks linear", that is for all $\lambda \geq \bar{\lambda}$ there is a value $v \in \mathbb{R}$ depending on d only such that

$$f(u + \lambda d) - f(u + \bar{\lambda} d) = (\lambda - \bar{\lambda})v.$$

We call v the *recession value* of f at u, and it can be shown that

$$v = \sup_{y \in \mathrm{dom}(f)} \big(f(y + d) - f(y)\big)$$

which is the difference that d can make to f. It can also known as the "slope at infinity", and is related to the asymptotic or recession cone of S. See Hiriart-Urruty and Lemaréchal [67, pg. 109, 179] for more details.

Theorem 2.35. *From Birge and Louveaux [41, §3.1e, Thm. 8], if $\boldsymbol{\zeta}$ has finite second moments and either of the following holds*

1. *$\mathcal{D}$ is bounded, or*
2. *$\mathscr{Q}$ is eventually linear in all recession directions of $\mathcal{D}$,*

then if Program 2.6 has finite optimal value, it is attained for some $x \in \mathbb{R}^n$.

Proof. In both cases, $\boldsymbol{\zeta}$ has finite second moments so we can apply Theorem 2.25 to see that K_2 is closed, and so therefore $\mathcal{D}$ is closed. Theorem 2.25 also implies that $\mathscr{Q}$ is continuous, so we are almost in the case of Proposition 2.29.

For 1. we see that $\mathcal{D}$ is bounded by assumption. If $\mathcal{D}$ is empty, then there is no finite solution. If $\mathcal{D}$ is not empty then z has a finite solution which must be attained by Proposition 2.29.

For 2. as $\mathscr{Q}$ is eventually linear in all recession directions then if the minimum is finite, we see that for a recession direction d the corresponding recession value v must be non-negative. (If it were negative, we would be able to send λ to ∞ and show that the infinum of f is negative infinity). This then means the "slope at infinity" of $\mathscr{Q}$ in any direction along $\mathcal{D}$ is non-negative, so it is bounded below. Our previous discussion shows that this means that if K_2 is non-empty, we have a finite solution for z and we must attain it, and that otherwise we have solution ∞. □

We can combine Theorem 2.25 and the previous reasoning to obtain the following proposition, which has similar reasoning to Theorem 2.35.

Proposition 2.36. *If $\boldsymbol{\zeta}$ has finite second moments, $\mathcal{D}$ is non-empty, and z is bounded below on $\mathcal{D}$, then a finite optimal value is obtained for Program 2.6.*

Proof. As $\mathcal{D}$ is non-empty, the solution must be less than ∞. As z is bounded below, then it must be greater than $-\infty$, so a solution (if attained) must be finite. As the conditions of Theorem 2.35 hold, then $\boldsymbol{\zeta}$ has finite second moments, so Theorem 2.25 implies K_2 is closed, and then $\mathcal{D} = K_1 \cap K_2$ is closed. Therefore the solution is finite and is attained at some $x \in \mathbb{R}^n$ by Theorem 2.35. □

Proposition 2.28 can be seen as a corollary to Proposition 2.36, as finite distributions have finite second moments.

Birge and Louveaux [41, §3.1] go on to describe further explicit conditions for optimality using the Karush-Kuhn-Tucker (KKT) conditions. These use subgradients of $\mathscr{Q}$ and Q, and the normal cone of K_2. We do not do this here as we will not use them later, although we suggest Hiriart-Urruty and Lemaréchal [67, pg. 136, 239, 306] as a good reference for the KKT conditions, subgradients and normal cones.

2.5 Solution methods

The deterministic equivalent program and the sample average approximation program are linear programs, so there are many solution methods such as the simplex algorithm, the ellipsoid method and the interior point method Matousek and Gärtner [83, §5&7]. However, as the number of scenarios increases it can be useful to take advantage of the programs *block structure* within the constraint matrix as shown below.

$$\begin{bmatrix} A & 0 & \dots & \dots & 0 \\ T_1 & W & \ddots & & \vdots \\ T_2 & 0 & W & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ T_r & 0 & \dots & 0 & W \end{bmatrix} \begin{bmatrix} x \\ y_1 \\ y_2 \\ \vdots \\ y_r \end{bmatrix} = \begin{bmatrix} b \\ \boldsymbol{h}_1 \\ \boldsymbol{h}_2 \\ \vdots \\ \boldsymbol{h}_r \end{bmatrix} \tag{2.11}$$

This block structure comes from writing the constraints for the programs in Definition 2.21 and Definition 2.22 in standard form. This is done by forming one decision vector, by stacking $x, y_1, y_2, \dots, y_r$ into one vector.

Solving programs with this block structure can take advantage of decomposition algorithms where the initial problem can be decomposed into subproblems and solved sequentially until converging to a solution for the initial program. One of these decomposition

methods is *Benders Algorithm* from Benders [35], and another is *Dantzig-Wolf decomposition* described in Dantzig and Wolfe [55] which solves the dual to that of Benders algorithm.

These methods are used to solve a wide range of programs which have block structure, where the deterministic equivalent program in Definition 2.21 and the sampling average approximation in Definition 2.22 are specific examples. In the context of stochastic programming, Benders decomposition has been adapted to the *L-Shaped Method* originally described in Van Slyke and Wets [107], where the name is derived from shape of the block structure within the constraint matrix.

There are many other methods for solving stochastic programs including variations of the L-Shaped method such as *the Multicut L-shaped Method* and algorithms that use the Stochastic Program Lagrangian. See Birge and Louveaux [41, §5] for further details. In this work we focus on the L-shaped method described in [107] as a *cutting plane algorithm*. Cutting plane methods are used to find piecewise linear (or affine) approximations of convex functions, and construct a sequence of points using these approximations that converge to the optimal solution. The linear (or affine) functions can be visualised as planes that cut through the region $\mathcal{D} \times \mathbb{R}$ to bound the graph $\{(x, f(x)) \mid x \in \mathcal{D}\} \subseteq \mathcal{D} \times \mathbb{R}$ of the real valued convex function f with domain (or feasible region) $\mathcal{D}$, giving rise to the name 'cutting plane' method. See Ruszczyński [101, §7.2] for further details.

We now proceed to describe the L-shaped method in detail. While we broadly follow the results of Birge and Louveaux [41], we also expand on details for the explanations and proofs.

2.5.1 L-Shaped method

We seek to solve the two-stage stochastic program with fixed recourse as given in Program 2.6 which has the following form

$$\begin{aligned} \min_{x} \quad & z = c^\top x + \mathbb{E}_{\boldsymbol{\zeta}}[Q(x, \boldsymbol{\zeta})] \\ \text{subject to} \quad & Ax = b, \\ & x \geq 0, \end{aligned}$$

where for each realisation $\boldsymbol{\zeta}_\omega$ of $\boldsymbol{\zeta}$ we have

$$\begin{aligned} Q(x, \boldsymbol{\zeta}_\omega) = \min_{y} \quad & \boldsymbol{q}_\omega{}^\top y \\ \text{subject to} \quad & \boldsymbol{T}_\omega x + W y = \boldsymbol{h}_\omega, \\ & y \geq 0. \end{aligned}$$

This is a convex program provided that $\boldsymbol{\zeta}$ has finite second moments since this guarantees that $\mathscr{Q}(x) = \mathbb{E}_{\boldsymbol{\zeta}}[Q(x, \boldsymbol{\zeta})]$ is convex from Theorem 2.25. If $\boldsymbol{\zeta}$ is a finite distribution we

can form the deterministic equivalent program, while in general the sampling average approximation program can be formed. Both of these programs can be solved efficiently using methods such as the L-shaped method.

2.5.2 High level explanation of the L-shaped method

The L-shaped method is a solution method for solving two-stage stochastic programs in deterministic equivalent form as in Program 2.6. We can represent $\mathbb{E}_{\boldsymbol{\zeta}}[Q(x, \boldsymbol{\zeta})]$ by an additional variable θ as in Definition 2.20, and solve instead the simpler linear program

$$\begin{aligned} \min_{x} &= c^\top x + \theta \\ \text{such that } & Ax = b, \\ & x \geq 0, \end{aligned} \tag{2.12}$$

with additional constraints on x and θ that ensure the solution is both feasible and optimal with respect to the original problem. Hence in Step 1, the program initially solves with $\theta = 0$ and no other constraints. It then uses this solution to define certain 'cuts' (i.e. affine constraints/cutting planes) that restrict x and θ in the subsequent iterations.

Step 2 considers the feasibility cuts, which check whether there is feasibility for y variables, given the x solution in Step 1, and if not adds cuts to Equation (2.12) and returns to Step 1. Step 3 considers the optimality cuts by checking whether the optimal value has been achieved, and if not adding cuts to Equation (2.12) and returning to Step 1. Eventually there are no longer any cuts needed and the final iteration of the solution at Step 1 of Equation (2.12) with all of the cuts as constraints gives the optimal solution of the original deterministic equivalent program.

Summarising, the L-shaped method proceeds as follows:

Step 0: Initialise.

Step 1: Solve the LP in Equation (2.12). Proceed to Step 2.

Step 2: Check if a feasibility cut is required. If so add to Equation (2.12) and return to Step 1. If not go to Step 3.

Step 3: Check if an optimality cut is required. If so, add to Equation (2.12) and return to Step 1. If not then the solution to the most recent Step 1 is optimal and the algorithm terminates.

2.5.3 L-shaped method in full

Here we expand in detail the steps of the L-shaped method.

Step 0. Initialisation.

Set $s = t = u = D_0 = d_0 = E_0 = e_0 := 0$. Set r to be the number of realisations for the random variable $\boldsymbol{\xi}$

Step 1.

Set $u = u + 1$. Consider the linear program:

$$\begin{aligned} \min z &= c^\top x + \theta \\ \text{such that } Ax &= b, \\ D_l x &\geq d_l \quad \text{for } \ell = 1, \ldots, s, \\ E_l x + \theta &\geq e_l \quad \text{for } \ell = 1, \ldots, t, \\ x &\geq 0, \\ \theta &\in \mathbb{R}. \end{aligned} \tag{2.13}$$

If $t = 0$, set $\theta = 0$ and solve LP (2.13) for x, and set $x^u = x$. Set $\theta^u = -\infty$.

Else solve LP (2.13) for x and θ, and denote the solution at iteration u as (x^u, θ^u). If no feasible solution can be found (that is, the feasible region is empty), terminate algorithm and return 'Program is infeasible'. Otherwise proceed to **Step 2**.

Step 2. Feasibility cuts.

Set $w = 0$, $k = 0$.

While $w \leq 0$ and $k < r$, set $k = k + 1$ and solve the linear program

$$\begin{aligned} \min_{v^+, v^-, y_k^u} \ & \widetilde{\omega}_k = e^\top v^+ + e^\top v^- \\ \text{such that } \ & W y_k^u + v^+ - v^- = \boldsymbol{h}_k - \boldsymbol{T}_k x^u \\ & y_k^u \geq 0, \\ & v^+ \geq 0, \\ & v^- \geq 0. \end{aligned} \tag{2.14}$$

where $e^\top = (1, \ldots, 1)$ is a vector of ones of the appropriate dimension, and variables v^+ and v^- have the same dimension as $\boldsymbol{h}_k$.

Set $w = \widetilde{\omega}_k^*$.

End while.

If $w > 0$, then let σ_k^u be the associated vector of Lagrange multipliers associated with the dual optimal solution of (2.14). We then define

$$\begin{aligned} D_{s+1} &= (\sigma_k^u)^\top \boldsymbol{T}_k \\ d_{s+1} &= (\sigma_k^u)^\top \boldsymbol{h}_k, \end{aligned}$$

which will be the new linear constraints at **Step 1**, called the *feasibility cuts*. Also, set $s = s + 1$, and go back to **Step 1**.

Otherwise, $\widetilde{\omega} = 0$ for all $k = 1, 2, \ldots, r$, and proceed to **Step 3**.

Step 3. Optimality cuts.

For $k = 1, 2, \ldots, r$ solve the linear program

$$\begin{aligned} \min \hat{\omega} &= \boldsymbol{q}_k^\top y_k^u \\ \text{such that } W y_k^u &= \boldsymbol{h}_k - \boldsymbol{T}_k x^u \\ y_k^u &\geq 0. \end{aligned} \tag{2.15}$$

Let π_k^u be the vector of Lagrange multipliers associated with a dual optimal solution of LP (2.15) for a given k. Then define

$$\begin{aligned} E_{t+1} &= \sum_{k=1}^{r} p_k (\pi_k^u)^\top \boldsymbol{T}_k \\ e_{t+1} &= \sum_{k=1}^{r} p_k (\pi_k^u)^\top \boldsymbol{h}_k \end{aligned}$$

and set

$$J^u = e_{t+1} - E_{t+1} x^u.$$

If $\theta^u \geq J^u$, stop the iteration procedure, as $\boldsymbol{x}^u$ is optimal. Otherwise, add the linear constraint $E_{t+1} x + \theta \geq e_{t+1}$ called an *optimality cut* to 2.13, set $t = t + 1$, and return to **Step 1**.

2.5.4 Details on feasibility cuts

Say we have r realisations of the random variable $\boldsymbol{\zeta}$ and for $k \leq r$ we solve the linear program

$$\begin{aligned} \min_{y_k, v^+, v^-} \ & \tilde{w}_k = e^\top v^+ + e^\top v^- \\ \text{such that } & W y_k^u + v^+ - v^- = \boldsymbol{h}_k - \boldsymbol{T}_k x^u \\ & y_k \geq 0, \\ & v^+ \geq 0, \\ & v^- \geq 0, \end{aligned} \tag{2.16}$$

where x^u is the solution at **Step 1**. Here e is the vector of all entries equal to 1. If the optimal value $\tilde{w}_k^* = 0$ then the corresponding y_k are feasible, as this means $v^+ - v^- = 0$ and $W y_k + \boldsymbol{T}_k x^u = \boldsymbol{h}_k$. If this is not the case then we need to add a cut to Equation (2.13) and then return to Step 1. We do this as follows, by first taking the dual program to Equation (2.16)

$$\begin{aligned} \max_{\sigma_k} \ & (\boldsymbol{h}_k - \boldsymbol{T}_k x^u)^\top \sigma_k \\ \text{such that } & W^\top \sigma_k \leq 0 \\ & -e \leq \sigma_k \leq e. \end{aligned} \tag{2.17}$$

As $\tilde{w}_k^* \neq 0$ for this k then we must have $\tilde{w}_k^* > 0$. By strong duality of linear programs from Theorem 2.10 we must have optimal σ_k^* for Equation (2.17) such that

$$0 < \tilde{w}_k^* = (\boldsymbol{h}_k - \boldsymbol{T}_k x^u)^\top \sigma_k^* = \boldsymbol{h}_k^\top \sigma_k^* - x^{u\top} \boldsymbol{T}_k^\top \sigma_k^*.$$

To make sure that y_k is feasible for the next iteration, we want to ensure that the solution x^u to Equation (2.13) is such that $\boldsymbol{h}_k^\top \sigma_k^* - x^{u\top} \boldsymbol{T}_k^\top \sigma_k^* \leq 0$, so we write

$$D_{s+1} = (\sigma_k^*)^\top \boldsymbol{T}_k, \quad d_{s+1} = \boldsymbol{h}_k^\top \sigma_k^*.$$

Here s indexes the number of such feasibility constraints added so far.

Then adding the constraint $D_{s+1} x \geq d_{s+1}$ to Equation (2.13) will ensure that this occurs. We can see this by noting that the constraints to Equation (2.17) do not depend on x^u, only the maximum does. Then if we have solved Equation (2.13) for x^u with the constraints $D_{s'} x^u \geq d_{s'}$, for each $s' = 1, \ldots, s+1$ then $D_{s'} x^u \geq d_{s'}$ and therefore we have

$$\boldsymbol{h}_{k_{s'}}^\top \sigma_{k_{s'}}^* - x^{u\top} \boldsymbol{T}_{k_{s'}}^\top \sigma_{k_{s'}}^* \leq 0,$$

for the $k_{s'}$ corresponding to the s' contraint. The $\sigma_{k_{s'}}^*$ is a feasible solution to Equation (2.17) for this x^u, and therefore the optimal value of Equation (2.17) is at most 0.

However, by weak duality, we know the optimal value is at least zero, as the dual of Equation (2.17) is Equation (2.16) which non-negative. Therefore both Equation (2.16) and Equation (2.17) have optimal value zero by strong duality, and no feasibility cut is required for each of the realisations $k_1, \ldots, k_s$ corresponding to these $s' = 1, \ldots, s$.

This means the number of cuts required is at most r, one for each of the realisations of $\boldsymbol{\zeta}$. In many cases, less feasibility cuts are required, as in the following lemma.

Lemma 2.37. *If Program 2.6 has relatively complete, complete, or simple recourse, then no feasibility cuts are required.*

Proof. In each of these cases, $K_1 \subseteq K_2$. This means that any x^u that is a solution to Equation (2.13) has corresponding feasible y_k for each realisation of $\boldsymbol{\zeta}$. Then Equation (2.16) has optimal value $\tilde{w}_k^* = 0$ for all k and no feasibility cuts are required. □

2.5.5 Details on optimality cuts

At this stage, there are now feasible y_k for each realisation $\boldsymbol{\zeta}_k$. We can then solve for each $k = 1, \ldots, r$ the linear program

$$\begin{aligned} \min_{y_k} \hat{w}_k &= \boldsymbol{q}_k^\top y_k \\ \text{such that } W y_k &= \boldsymbol{h}_k - \boldsymbol{T}_k x^u \\ y_k &\geq 0. \end{aligned}$$

The solution we label $\hat{w}_k^*$, which is equal to $Q(x^u, \boldsymbol{\zeta}_k)$. We also solve its dual program for each k

$$\begin{aligned} \max_{\pi_k} (\boldsymbol{h}_k - \boldsymbol{T}_k x^u)^\top \pi_k \\ \text{such that } W^\top \pi_k \leq \boldsymbol{q}_k. \end{aligned} \tag{2.18}$$

Note that from strong duality we have

$$Q(x^u, \boldsymbol{\zeta}_k) = \hat{w}_k^* = \boldsymbol{q}_k^\top y_k^* = (\boldsymbol{h}_k - \boldsymbol{T}_k x^u)^\top \pi_k^* = \max_{\pi_k} (\boldsymbol{h}_k - \boldsymbol{T}_k x^u)^\top \pi_k$$

for this x^u, where the maximum is subject to $W^\top \pi_k \leq \boldsymbol{q}_k$. Note also that this constraint is independent of the value of x^u, so that this π_k^* is a feasible solution for the dual regardless of the value of x^u.

Then for a different value of x^u, say x' we must have

$$\max_{\pi_k} (\boldsymbol{h}_k - \boldsymbol{T}_k x')^\top \pi_k \geq (\boldsymbol{h}_k - \boldsymbol{T}_k x')^\top \pi_k^*,$$

where we are maximising over feasible π_k (that is, $W^\top \pi_k \leq \boldsymbol{q}_k$) and we know π_k^* is feasible for this maximisation over x' but is not necessarily optimal. This means we must have the corresponding $\min'_{y_k} \boldsymbol{q}_k^\top y'_k$ subject to $Wy'_k = \boldsymbol{h}'_k - \boldsymbol{T}_k x'$, $y'_k \geq 0$ lower bounded by $(\boldsymbol{h}_k - \boldsymbol{T}_k x')^\top \pi_k^*$ for each k and for any other x'. That is, we have

$$Q(x', \boldsymbol{\zeta}_k) = \boldsymbol{q}_k^\top {y'_k}^* = (\boldsymbol{h}_k - \boldsymbol{T}_k x')^\top {\pi'_k}^* = \max_{\pi'_k} (\boldsymbol{h}_k - \boldsymbol{T}_k x')^\top \pi'_k \geq (\boldsymbol{h}_k - \boldsymbol{T}_k x')^\top \pi_k^*,$$

where we are again maximising over π'_k such that $W^\top \pi'_k \leq \boldsymbol{q}_k$, and we have equality in this equation when $x' = x^u$.

If we now take the expected value over all k we must have that $\mathbb{E}(Q(x, \boldsymbol{\zeta}))$ is lower bounded by the affine hyperplane as follows

$$\mathscr{Q}(x) = \mathbb{E}_{\boldsymbol{\zeta}}(Q(x, \boldsymbol{\zeta})) = \sum_k p_k Q(x, \boldsymbol{\zeta}_k) \geq \sum_k p_k (\pi_k^*)^\top \boldsymbol{h}_k - \sum_k p_k (\pi_k^*)^\top \boldsymbol{T}_k^\top x = e_\ell - E_\ell x,$$

and we have equality when $x = x^u$. We must require that $\theta \geq e_\ell - E_\ell x$ for optimality of the solution for Program 2.6, as it is clear that $e_\ell - E_\ell x$ is a lower bound for $\mathscr{Q}(x)$.

If θ^u from Step 1 is such that $\theta^u < e_\ell - E_\ell x^u$ then

$$z = c^\top x^u + \theta^u < c^\top x^u + e_\ell - E_\ell x^u \leq c^t x^u + \mathscr{Q}(x^u)$$

is strictly lower than it should be (as it should be equal to $c^t x^* + \mathscr{Q}(x^*)$ for some optimal x^*), and we have not found the optimal solution to Program 2.6 at Step 1. We add the constraint that $\theta \geq e_\ell - E_\ell x$ to Equation (2.13) and return to Step 1.

If θ^u from Step 1 is such that $\theta^u \geq e_\ell - E_\ell x^u$, then as the pair x^u, θ^u solve Equation (2.13) then $\theta^u \geq e_{\ell'} - E_{\ell'} x^u$ for all $\ell' = 1, 2, \ldots, \ell$. As $e_{\ell'} - E_{\ell'} x^u$ is a lower bound for $\mathscr{Q}(x^u)$ for each ℓ', and $\mathscr{Q}(x^u) = e_\ell - E_\ell x^u$ then $\mathscr{Q}(x^u)$ is the smallest value of θ that satisfies Equation (2.13). As we are minimising over θ then $\theta^u = \mathscr{Q}(x^u) = e_\ell - E_\ell x^u$ and we have that x^u and the y_k solutions from Step 3 form the solution to Program 2.6, with optimal value $c^\top x^u + \theta^u$.

It remains to discuss how many optimality cuts can be added. Note that for each k Equation (2.18) has the feasible region $W^T \pi_k \leq \boldsymbol{q}_k$ which is polyhedral and independent of x^u. Then we know from linear programming theory that the optimal value of Equation (2.18) must occur at a vertex of this feasible region (that is, at a *basic feasible solution*), see Matousek and Gärtner [83, §4.2] for details on this. As the feasible region is polyhedral, there are only finitely many vertices/basic feasible solutions for each k. As there are only finitely realisations (at most r), then this means there is at most finitely many optimality cuts.

Note that there may be cases where the optimal value occurs at more than one vertex and therefore occurs at all convex combinations of these vertices. Then either this is always true for any value of x, and the algorithm will make the appropriate cut for any of

these values, or this changes for different values of x and then the optimality cut will be able to distinguish between the different vertices if required for optimality. Again there are only finitely many different vertices to distinguish from, so there will still be finitely many optimality cuts. A way to avoid this case is to use the Simplex Algorithm which only picks basic feasible solutions/vertices of the feasible region as solutions.

This discussion leads to the following theorem.

Theorem 2.38. *The L-shaped method terminates in finitely many steps with either the optimal solution to Program 2.6 or shows that Program 2.6 is infeasible.*

Proof. If Program 2.6 is infeasible, then adding the feasibility constraints as in Section 2.5.4 will mean that Equation (2.13) will eventually have no feasible solution for one of the finite possible feasibility cuts, and so the algorithm will terminate. Otherwise we have shown that the number of feasibility and optimality cuts is finite, so the algorithm terminates in finitely many steps with the solution. Note that if the solution is $-\infty$, then all feasibility cuts and optimality cuts will be able to be implemented as demonstrated however the optimal solution at **Step 1** will return $-\infty$ as a solution. □

2.6 Performance of stochastic programs

Stochastic programs can be difficult to solve and computationally demanding. This prompts the question: is it worth the effort? Often, programs which should be stochastic are instead solved with a deterministic substitutes, such as replacing a random variable with its mean. This results in a simpler program to solve. However, there is generally a trade off in taking this approach, potentially making a suboptimal decisions. Here we consider the benefit in solutions when using stochastic data within a program following Birge and Louveaux [41, §4].

In this section we use RP to denote the solution to the *two-stage stochastic program with fixed recourse* as given in Program 2.6. That is,

$$RP = \min_x \mathbb{E}[z(x, \boldsymbol{\zeta})].$$

We will also use

$$z(x, \boldsymbol{\zeta}_\omega) = c^\top x + Q(x, \boldsymbol{\zeta}_\omega),$$

where as previously defined we take the random variable $\boldsymbol{\zeta} = (\boldsymbol{q}^\top, \boldsymbol{h}^\top, \boldsymbol{T}^1, \ldots, \boldsymbol{T}^{m_2})^\top$ and a scenario for $\boldsymbol{\zeta}$ is denoted $\boldsymbol{\zeta}_\omega = (\boldsymbol{q}_\omega^\top, \boldsymbol{h}_\omega^\top, \boldsymbol{T}_\omega^1, \ldots, \boldsymbol{T}_\omega^{m_2})^\top$.

Consider the following program

$$\begin{aligned} \min_x \ & z(x, \boldsymbol{\zeta}_\omega) = c^\top x + Q(x, \boldsymbol{\zeta}_\omega) \\ \text{subject to} \quad & Ax = b, \\ & x \geq 0. \end{aligned}$$

This program is deterministic and a function of the scenario $\boldsymbol{\zeta}_\omega$. For a realisation $\boldsymbol{\zeta}_\omega$ of $\boldsymbol{\zeta}$ the solution is $x^*(\boldsymbol{\zeta}_\omega)$ and the associated objective value is $z(x^*(\boldsymbol{\zeta}_\omega), \boldsymbol{\zeta}_\omega)$. One can interpret this as being the optimal solution if it is known that the scenario $\boldsymbol{\zeta}_\omega$ occurs.

Taking the expectation of the objective function where the optimal decision $x^*(\boldsymbol{\zeta}_\omega)$ is made for each scenario gives the wait-and-see solution.

Definition 2.39. The *wait-and-see solution* (WS) is given by

$$WS = \mathbb{E}\left[z(x^*(\boldsymbol{\zeta}), \boldsymbol{\zeta})\right].$$

Unlike the two-stage stochastic program with fixed recourse, the WS cannot be used in real time for making a decision on x. However it can be used as a benchmark to understand the performance of a solution method ex-post.

Definition 2.40. The *Expected Value of Perfect Information (EVPI)* is defined as

$$EVPI = RP - WS.$$

The $EVPI$ is nicely described by Birge and Louveaux [41, pg. 163]. They say "the expected value of perfect information ($EVPI$) measures the maximum amount a decision maker would be ready to pay in return for complete (and accurate) information about the future." In some sense $EVPI$ provides a measure between the minimum cost from the RP and the mean cost with the benefit of hindsight from WS.

Computation can be demanding for a two stage stochastic program with recourse. This is especially the case for continuous random variables where a large number of scenarios may be used to form the SAA program. It is natural to ask questions such as: is it worth considering all scenarios for making the first stage decision? How costly is it to replace scenarios with their expectation? These questions motivate defining the *Value of the Stochastic Solution* (VSS), which compares solutions for two stage stochastic programs with fixed recourse, when the random vector $\boldsymbol{\zeta}$ is replaced with its expectation $\mathbb{E}[\boldsymbol{\zeta}]$. First we define the *expected value program* as follows.

Definition 2.41. The *expected value program* (EV) is defined as

$$EV = \min_x z(x, \mathbb{E}(\boldsymbol{\zeta}))$$

where the optimal decision is $x^*(\mathbb{E}(\boldsymbol{\zeta})) = \operatorname{argmin}_x z(x, \mathbb{E}(\boldsymbol{\zeta}))$. The expected value program is a deterministic program where the random variable ζ is replaced with the expectation $\mathbb{E}(\boldsymbol{\zeta})$.

Birge and Louveaux [41] give the follow relationship between EV and WS.

Proposition 2.42. *For stochastic programs with fixed objective coefficients and fixed W then,*

$$EV \leq WS.$$

This relationship makes it appear that EV is a better solution that WS. However, EV is the optimal solution to the stochastic program only if the expected value is the outcome for $\boldsymbol{\zeta}$. In general this does not occur, and instead we need to consider the *expected result using the EV solution.* That is, we take the first stage solutions to be $x^*(\mathbb{E}(\boldsymbol{\zeta}))$ and then solve for the expected recourse.

Definition 2.43. The *expected result of using the EV solution* is

$$EEV = \mathbb{E}_{\boldsymbol{\zeta}}\Big(z\big(x^*(\mathbb{E}(\boldsymbol{\zeta})), \boldsymbol{\zeta}\big)\Big).$$

The EEV solution can be used as a deterministic comparison with the RP and demonstrates the benefit of using stochastic data. Taking the difference between the EVV solution and the RP solution defines a method to measure this benefit, and is called the *Value of the Stochastic Solution (VSS)*

Definition 2.44. *Value of the Stochastic Solution (VSS)*

$$VSS = EEV - RP.$$

The VSS is of particular interest, and will be used to demonstrate how using stochastic data to make more optimal decisions can reduce costs.

We now describe the relationship between $RP, WS,$ and EEV as in Birge and Louveaux [41] and Madansky [79].

Proposition 2.45. *We have*

$$WS \leq RP \leq EEV.$$

This result can be interpreted as follows. If we knew for sure exactly which scenario would occur and could always pick optimally, then on average we would have cost WS. This is less (and therefore more optimal) than the cost RP that results if we do not know the scenario and instead pick our x using the optimal solution from Program 2.6. Finally, both WS and RP are also less (and therefore more optimal) than EEV which is the average cost of picking x by optimising with respect to the average of the scenarios.

It is possible to derive bounds for $EVPI$ and VSS. From Proposition 2.45 and their respective definitions it is clear that $EVPI \geq 0$ and $VSS \geq 0$. Cases when both $EVPI$ and VSS are both zero are not particularly interesting or common when considering typical stochastic programs. The relationship between $EVPI$ and VSS, and upper-bounds for each are discussed further in [41] using the so called *pairs subproblem* and *sum of pairs expected values* ($SPEV$).

While the expected value program approach is easier to implement and often used to find approximate solutions, in Section 4.6.2, we numerically examine the relationship between the expected value program using the sample mean and the result from the SAA which is an approximation to RP. In both cases, the same scenarios are used. We show that using the SAA results in a lower cost on average compared to the expected value program.

2.7 Accuracy of the sampling average approximation

The sampling average approximation (2.22) gives an approximate solution for the two stage stochastic program with fixed recourse (2.6). We want to quantify the error introduced form using the SAA. How can we measure the error of the SAA? At what rate does the SAA converge to the true solution? How many samples are required to obtain the desired accuracy of the SAA solution? These questions are important to answer so that results of the SAA can be used to understand our stochastic program. In this section we follow the work presented in Bayraksan and Morton [32].

Recalling the form of the objective function in Program 2.6 we now let

$$\begin{aligned} f = c^\top x + \min_y \ & \boldsymbol{q}_\omega^\top y \\ \text{such that } & \boldsymbol{T}_\omega x + W y = \boldsymbol{h}_\omega, \\ & y \geq 0. \end{aligned}$$

and

$$z^* = \min_{x \in \mathcal{D}} \mathbb{E}[f(x, \boldsymbol{\zeta})].$$

Here $\mathcal{D} = K_1 \cap K_2$ is the feasible region as defined in Equation (2.8). Recall that x^* and z^* are the optimal solution and cost respectively to (2.6). We also refer to x_r^* and z_r^* as the optimal solution and optimal cost respectively for the corresponding SAA with r samples of $\boldsymbol{\zeta}$.

Suppose we have a *candidate solution* denoted $\hat{x}$ and we want to understand the quality of this solution. One way of doing this is by considering *optimality gap* $\mu_{\hat{x}}$ defined as

$$\mu_{\hat{x}} = \mathbb{E}[f(\hat{x}, \boldsymbol{\zeta})] - z^*.$$

Generally, we cannot calculate the optimality gap $\mu_{\hat{x}}$ exactly. Here we focus on quantifying the optimality gap, and follow [32]. In particular we focus on the *single replication procedure* SRP, where the computation is relatively small. The SAA can be used to obtain an upper bound for the optimality gap.

First consider the following two theorems from [81].

Theorem 2.46. *Let $\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \ldots, \boldsymbol{\zeta}_r, \boldsymbol{\zeta}_{r+1}$ be i.i.d samples of $\boldsymbol{\zeta}$. Then*

$$\mathbb{E}[z^*_{r+1}] \geq \mathbb{E}[z^*_r].$$

Theorem 2.47. *Let $\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2, \ldots, \boldsymbol{\zeta}_r$ be i.i.d samples of $\boldsymbol{\zeta}$. Then*

$$\mathbb{E}[z^*_r] = \mathbb{E}\left[\min_{x\in\mathcal{D}} \frac{1}{r}\sum_{k=1}^{r} f(x, \boldsymbol{\zeta}_k)\right] \leq z^*$$

These two theorems mean that on average the optimal solution z^*_r to the SAA stochastic program with r scenarios is increasing and bounded above by the true optimal solution z^* of Program 2.6.

As z^* is the minimum over all $x \in \mathcal{D}$, then considering Theorem 2.47 and $\mathbb{E}[f(\hat{x}, \boldsymbol{\zeta})]$ with candidate solution $\hat{x} \in \mathcal{D}$ we can further write

$$\mathbb{E}[z^*_r] \leq z^* = \min_{x\in\mathcal{D}} \mathbb{E}[f(x, \boldsymbol{\zeta})] \leq \mathbb{E}[f(\hat{x}, \boldsymbol{\zeta})].$$

Hence we have an upper bound for $\mu_{\hat{x}}$ given by

$$\mu_{\hat{x}} = \mathbb{E}[f(\hat{x}, \boldsymbol{\zeta})] - z^* \leq \mathbb{E}[f(\hat{x}, \boldsymbol{\zeta})] - \mathbb{E}[z^*_r].$$

This upper bound is approximated using the SAA method and defined as

$$\begin{aligned} G_r(\hat{x}) &= \frac{1}{r}\sum_{j=1}^{r} f(\hat{x}, \zeta^j) - \min_{x\in\mathcal{D}} \frac{1}{r}\sum_{j=1}^{r} f(x, \zeta^j), \\ &= \bar{f}_r(\hat{x}) - z^*_r, \end{aligned}$$

where $\bar{f}_r(\hat{x}) = \frac{1}{r}\sum_{k=1}^{r} f(\hat{x}, \boldsymbol{\zeta}_k)$.

Single Replication Procedure

We now outline the *single replication procedure* SRP. This procedure calculates a $(1-\alpha)100\%$ confidence interval for the optimality gap $\mu_{\hat{x}}$. The steps are as follows.

1. Take r i.i.d. samples $\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_r$ from the distribution of $\boldsymbol{\zeta}$,
2. Solve the SAA program with samples to obtain x^*_r and z^*_r, and
3. Calculate $G_r(\hat{x})$ and $s^2_r(x^*_n)$.

Then the $(1-\alpha)100\%$ confidence interval for $\mu_{\hat{x}}$ is

$$\Big[0, G_r(\hat{x}) + \frac{t_{r-1,\alpha} s_r(x_r^*)}{\sqrt{r}}\Big],$$

where $s_r^2(x) = \frac{1}{r-1}\sum_{k=1}^{r}\Big[\big(f(\hat{x},\boldsymbol{\zeta}_k) - f(x,\boldsymbol{\zeta}_k)\big) - \big(\bar{f}_r(\hat{x}) - \bar{f}_r(x)\big)\Big]^2$, and $t_{r-1,\alpha}$ is $1-\alpha$ quantile for a t distribution with $r-1$ degrees of freedom.

The $(1-\alpha)100\%$ confidence interval can be used to guide the choice of sample size. The bound on the optimality gap decreases as the number of realisations r increases. We will use the SRP to guide our choice of sample size for a desired bound and confidence level.

From the SRP we have the $(1-\alpha)100\%$ confidence interval for $\mu_{\hat{x}} = \mathbb{E}[f(\hat{x},\boldsymbol{\zeta})] - z^*$ so

$$0 \leq \mathbb{E}[f(\hat{x},\boldsymbol{\zeta})] - z^* \leq G_r(\hat{x}) + \frac{t_{r-1,\alpha} s_r(x_r^*)}{\sqrt{r}}.$$

By rearranging this inequality we obtain a $(1-\alpha)100\%$ confidence interval for the true solution z^* given by

$$\mathbb{E}[f(\hat{x},\boldsymbol{\zeta})] - G_r(\hat{x}) - \frac{t_{r-1,\alpha} s_r(x_r^*)}{\sqrt{r}} \leq z^* \leq \mathbb{E}[f(\hat{x},\boldsymbol{\zeta})].$$

It is not always possible to evaluate $\mathbb{E}[f(\hat{x},\boldsymbol{\zeta})]$ explicitly and hence we may approximate this with $\bar{f}_r(\hat{x})$ giving an approximate confidence interval. This gives the approximate confidence interval

$$z_r^* - \frac{t_{r-1,\alpha} s_r(x_r^*)}{\sqrt{r}} = \bar{f}_r(\hat{x}) - G_r(\hat{x}) - \frac{t_{r-1,\alpha} s_r(x_r^*)}{\sqrt{r}} \leq z^* \leq \bar{f}_r(\hat{x}), \tag{2.19}$$

which we will consider in Section 4.6.8.

Finally, there is well understood limiting behaviour as r goes to infinity resulting in the following theorem. This guarantees that the confidence interval defined from the SRP has probability of $1-\alpha$ of containing the true optimal solution as the sample size increases. Importantly, [32, §5.2] show empirically that this confidence interval tends to contain the true optimal solution with more than a $1-\alpha$ probability when the sample size is around a thousand, although this depends on the problem. They also detail more complicated repeated sampling procedures to form confidence intervals that perform better with smaller sample sizes. However, the SRP will be able to give us an indication of the accuracy of the solutions we are generating in later sections.

Theorem 2.48. *Suppose $X \neq \varnothing$ and is compact, $\mathbb{E}[f(x,\zeta)]$ is lower semicontinuous on X, and $\mathbb{E}[\sup_{x\in X} f^2(x,\xi) < \infty]$. Let $\hat{x} \in X$ and take i.i.d realisation $\zeta^1, \zeta^2, \ldots, \zeta^n$ of $\boldsymbol{\zeta}$.*

Let $X^* = \operatorname{argmin}_{x\in X} \mathbb{E}[f(x,\boldsymbol{\zeta})], \sigma^2_{\hat{x}}(x) = \operatorname{var}[f(\hat{x},\boldsymbol{\zeta})]$ *and assume*

$$\inf_{x\in X^*} \sigma^2_{\hat{x}}(x) \leq \liminf_{n\to\infty} s_n^2(x_n^*) \text{ and}$$
$$\limsup_{n\to\infty} s_n^2(x_n^*) \leq \sup_{x\in X^*} \sigma^2_{\hat{x}}(x) \text{ w.p.1.}$$

Then given $0 < \alpha < 1$ *in the SRP,*

$$\liminf_{n\to\infty} \mathbb{P}\Big(\mu_x \leq G_n(\hat{x}) + \frac{z_\alpha s_n(x_n^*)}{\sqrt{n}}\Big) \geq 1-\alpha.$$

2.8 Conclusion

In this chapter we have defined a two-stage stochastic programs with fixed recourse, which has solution RP. To do this we first recall results from convex programming in Section 2.2, before describing the stochastic programs of interest. In particular, we also describe the sample average approximation program (SAA), which gives an approximation to RP and forms the basis for our stochastic-optimal power flow in Chapter 4 and Chapter 6. For the interested reader, we covered the properties of two stage stochastic program with fixed recourse in Section 2.4.2 and existence of solutions in Section 2.4.3. While we generally followed the existing literature on stochastic programs, we did deviate away from this in Section 2.4.3 with our own results.

In Section 2.5 we detail the L-shaped method, which is an efficient solution method for our stochastic programs. We add to the literature by giving further detailed explanations and proofs of convergence of the algorithm where relevant to this thesis. We use the L-shaped method for solving our stochastic programs in all subsequent chapters, which we implement using the JuMP package in Julia [37] as detailed in Section 4.6.

The final two sections are concerned with performance and accuracy of two-stage stochastic programs. With regards to performance, in Section 2.6 we define many alternative programs to which RP is compared. Of particular interest is the deterministic EV and EEV approaches. These are less computationally intensive however the stochastic scenarios are not accounted for, and so they do not perform as well as the RP solution in practice. We will go on to empirically compare them to RP in Section 4.6.2.

Finally in Section 2.7 we show the accuracy of a SAA program is dependent on the number of samples drawn from random variables. In Section 2.7 we present the SRP where a confidence interval is given for the difference between the true solution and proposed first stage solution. We describe how we can use this to understand the benefit of increasing the sample size for our stochastic programs. In Section 4.6.8 we will empirically compute these confidence intervals to determine the accuracy of our test networks.

This chapter focused on the background of stochastic optimisation programs, including defining a two-stage stochastic program with fixed recourse. Next, in Chapter 3 we recall

background on power systems modelling to set up the optimisation programs of interest. Following this, in Chapter 4 we combine this chapter and Chapter 3 where we describe stochastic-optimal power flow as a two-stage stochastic program with fixed recourse and use many of the results of this chapter.

Chapter 3

Modelling power systems for power flow analysis

In this chapter we introduce the necessary background on modelling power flow within an electricity network. We will use this to develop an optimisation program called optimal power flow. This enables us to choose generators that should produce power in order to meet the demand within the network, while also satisfying constraints posed by the network, in a cost efficient way.

We first introduce some general equations for describing power systems with an alternating current. We then introduce variables which are required to describe the physics of power flow along a transmission line. We describe the π-model, which gives an approximation for the behaviour of power flow on a medium length transmission line. In this chapter and the rest of this thesis, we assume no transformers although an extended π-model that includes transformers could be used for this application, which is future work. We then show how applying DC-approximation assumptions results in a linear approximation of power flow along a transmission line within the π-model. This allows us to form a linear optimal power flow known as DC-OPF. This chapter follows [39] and [53], although we have our own presentations for the proofs of Lemma 3.2. We also have some additional results in the appendix, including Lemma A.1 with our own proof.

We use the results in this chapter to develop stochastic-optimal power flow in §4.1 and then apply this to the South Australian transmission network in §6.1.

3.1 Modelling power systems

In order to determine how much storage is needed in a network we require a model of how the network operates. We use this to determine which generators optimally produce power with respect to cost. The simplest model for optimal generation is called the *economic dispatch* process which uses the assumption of a *copperplate* network, in which

all generators are assumed to be connected to all sources of demand without any physical constraints [53]. Effectively, the physics of the system is ignored other than the maximum and minimum operation limits of each generator.

Determining which generators to dispatch electricity in this model proceeds by ordering the generators by cost. Successively generators are chosen to turn on, with the lowest cost generators to turning on up to their maximum operation limit until enough electricity is dispatched to meet demand. Marginal costs are assumed to be fixed and hence total cost for each generator is proportional to power output. This is described further in for example [53, §2.1].

We do not want to completely ignore the physics of the system in this way. Instead, we will describe the physics of the system in the next section and then show how some assumptions lead to the formulation of an *optimal power flow* model called DC-OPF. This will use physical constraints such as line limits as well as how the generators connect to the different sources of demand through the network.

3.1.1 Dynamics of an electricity network

Mathematically, a power systems transmission network can be modelled as a graph where, the edges are the power lines, and the nodes, where power lines join, are called buses. Power generators and points of load — locations where there is demand for power — within the network are assigned to buses. If bus k is joined by a line to bus m, we refer to the joining line as line (k, m), where the order refers to the direction along the line. When referring to parameters or variables on a line, we will use subscript k, m to denote the quantity that belongs to the line (k, m). Similarly if a parameter is associated with a bus k we will use subscript k to denote the assignment.

Most transmission lines within the National Electricity Market (NEM) carry an alternating current (AC) induced by an alternating voltage [10]. The current varies in magnitude and direction in a sinusoidal way as time varies. The dynamics of AC electricity can be described with the following equations

$$\begin{aligned}
&\text{Voltage at bus } k \text{ at time } t: && v_k(t) := V_k^{\mathrm{Amp}} \cos(\omega t + \theta_k^V),\\
&\text{Current on along line } k, m \text{ at time } t: && i_{k,m}(t) := I_{k,m}^{\mathrm{Amp}} \cos(\omega t + \theta_{k,m}^I),\\
&\text{Instantaneous Power on line } k, m \text{ at time } t: && p_{k,m}(t) := v_k(t) i_{k,m}(t).
\end{aligned}$$

Here the quantities V_k^{Amp} and $I_{k,m}^{\mathrm{Amp}}$ are constants that determine the maximum amplitude of the voltage and current. The respective phase angles are θ_k^V and $\theta_{k,m}^I$, and ω is the angular frequency of oscillation, $\omega = 2\pi f$, where f the *frequency* within the network. In the NEM including South Australia the frequency is 50Hz [17].

The power $p_{k,m}(t)$ is rapidly varying with time t. We now consider the mean power along a line k, m denoted $p_{k,m}$ where the mean is taken over one period of time T, with

$T = \frac{2\pi}{\omega}$, and

$$p_{k,m} = \frac{1}{T}\int_0^T p_{k,m}(t)\,\mathrm{d}t = \frac{1}{2}V_k^{\mathrm{Amp}} I_{k,m}^{\mathrm{Amp}}\Big(\cos(\theta_k^V - \theta_{k,m}^I)\Big).$$

Alternatively we can consider the voltage V_k and current $I_{k,m}$ as complex numbers where

$$V_k = \frac{V_k^{\mathrm{Amp}}}{\sqrt{2}}e^{j\theta_k^V} \text{ and } \quad I_{k,m} = \frac{I_{k,m}^{\mathrm{Amp}}}{\sqrt{2}}e^{j\theta_{km}^I}, \text{ where } j := \sqrt{-1}.$$

This leads to defining the *complex power* or *apparent power* $S_{k,m}$ on line k, m as

$$S_{k,m} = V_k I_{k,m}^* \tag{3.1}$$

where we use * to denote the complex conjugate. Substituting in the expressions for V_k and $I_{k,m}^*$ and rearranging gives

$$S_{k,m} = \frac{V_k^{\mathrm{Amp}} I_{k,m}^{\mathrm{Amp}}}{2}\big(\cos(\theta_k^V - \theta_{km}) + j\sin(\theta_k^V - \theta_{km})\big) = p_{k,m} + jq_{k,m}$$

We have shown that $p_{k,m}$ (the *active* power) is the real component of $S_{k,m}$ and, we define $q_{k,m}$ (the *reactive* power) as the imaginary component of $S_{k,m}$.

3.1.2 The π-model for transmission lines

Describing electricity flow along a transmission line between buses k and m is complicated due to the ever-changing interacting affects of AC power. However we can use a series of assumptions to give a simplified model, which is known as the (nominal) π-model. The π-model assigns parameters of the line so that the physics of the line can be described and is appropriate for medium length transmission lines with approximate length between 80km - 240km as in [36, §4.5].

Here we assume there are no transformers in the network, although there are extensions of the π-model that do include transformers [39]. We now introduce parameters required for discussing the π-model and power flow.

resistance:	$r_{k,m}$,
reactance:	$x_{k,m}$,
series impedance:	$z_{k,m} = r_{k,m} + jx_{k,m}$,
series admittance:	$y_{k,m} = z_{k,m}^{-1} = g_{k,m} + jb_{k,m}$,
with $g_{k,m} = \frac{r_{k,m}}{r_{k,m}^2 + x_{k,m}^2}$, and	$b_{k,m} = \frac{-x_{k,m}}{r_{k,m}^2 + x_{k,m}^2}$,
shunt conductance:	$g_{k,m}^{sh}$,
shunt susceptance:	$b_{k,m}^{sh}$,
shunt admittance:	$y_{k,m}^{sh} = g_{k,m}^{sh} + jb_{k,m}^{sh}$.

Figure 3.1 shows a diagram representing the π-model for a transmission line between nodes k and m. Note that the π-model assumes that the shunt admittance for a line is distributed so that half of it appears at each node.

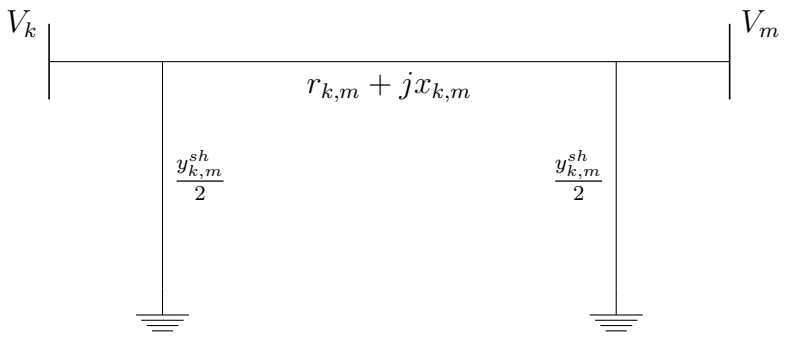


Figure 3.1: The π-model for a transmission line with no transformers.

For the π-model the complex current flowing along line (k, m) from bus k to bus m is

$$I_{k,m} = y_{k,m}(V_k - V_m) + y^{sh}_{k,m} V_k. \tag{3.2}$$

3.1.3 Per-unit system

The per-unit system is a scaling of variables such that their per-unit values are approximately one. Often the per-unit system is used for convenience within calculation. We define base quantities from which all other variables within the system can be deduced. The base quantities are P_{base}, the base power for the system and a base voltage $V_{\text{base},k}$, where k indicates base voltage for the bus. In this thesis we use $P_{\text{base}} = 100$ unless otherwise stated. The per unit quantities denoted p.u. are given by

$$P_{\text{p.u.}} = \frac{P}{P_{\text{base}}}, \quad V_{\text{p.u.}} = \frac{V}{V_{\text{base}}}.$$

3.1.4 DC-power flow

Using the π-model for transmission lines we can now form the linear approximation for active power flow on transmission lines, as in [39] and [53]. This approximation has a number of assumptions as outlined below and hold for lightly loaded and balanced power systems. We begin by considering the apparent power flow on a transmission line from bus k to bus m using the description for the current given in Equation (3.2).

$$S_{k,m} = V_k I^*_{k,m} = V_k \left(y^{sh*}_{k,m} V^*_k + y^*_{k,m} \big(V^*_k - V^*_m\big) \right) \tag{3.3}$$

1. Assume the shunt conductance for line k, m to be negligible. If $g^{sh}_{k,m} \simeq 0$ then we have $y^{sh}_{k,m} \simeq jb^{sh}_{k,m}$.
2. Next assume the series resistance to be negligible relative to the series reactance. Hence we have the impedence given by $z_{k,m} = jx_{k,m}$ and the series admittance given by $y_{k,m} = z^{-1}_{k,m} = -j\frac{1}{x_{k,m}}$.
3. For a per-unit system under normal conditions $|V_k| \simeq 1$ for all buses k.
4. Under normal conditions (normal meaning the system is approximately at equilibrium and not heavily loaded) $|\theta_k - \theta_m|$ is small for each line k, m and hence $\sin(\theta_k - \theta_m) \simeq \theta_k - \theta_m$.

Starting with Assumptions 1. and 2. we can write the conjugate current as

$$I^*_{ij} = -jb^{sh}_{k,m}V_k + j\frac{1}{x_{k,m}}(V^*_k - V^*_m). \tag{3.4}$$

Since we are only interested in the phase angle difference between buses k and m we can without loss of generality let the complex voltages be $V_k = |V_k|\, e^{j0} = |V_k|$ and let $V_m = |V_m|\, e^{j(\theta_m - \theta_k)}$. So we have

$$\begin{aligned} V^*_k &= V_k = |V_k|\,, \\ V^*_m &= |V_m|\, e^{-j(\theta_m - \theta_k)}. \end{aligned}$$

Substituting these voltages into Equation (3.4) gives

$$\begin{aligned} I^*_{k,m} &= -jb^{sh}_{k,m}|V_k| + j\frac{1}{x_{k,m}}\big(|V_k| - \big(|V_m|\cos(\theta_m - \theta_k) - j|V_m|\sin(\theta_m - \theta_k)\big)\big) \\ &= -jb^{sh}_{k,m}|V_k| + j\frac{1}{x_{k,m}}|V_k| - j\frac{1}{x_{k,m}}|V_m|\cos(\theta_m - \theta_k) - \frac{1}{x_{k,m}}|V_m|\sin(\theta_m - \theta_k), \end{aligned}$$

and

$$\begin{aligned} p_{k,m} &= \mathrm{Re}(S_{k,m}) \\ &= \mathrm{Re}(V_k I^*_{k,m}) \\ &= |V_k|\,\mathrm{Re}(I^*_{k,m}) \\ &= \frac{1}{x_{k,m}}|V_k||V_m|\sin(\theta_k - \theta_m). \end{aligned}$$

Now applying assumption 4 we obtain

$$p_{k,m} = \frac{\theta_k - \theta_m}{x_{k,m}}. \tag{3.5}$$

Note that $p_{k,m}$ is no longer dependent on $|V_k||V_m|$ and the trigonometric terms. Under this approximation, $p_{k,m}$ is *linearly* proportional to the voltage phase difference between buses k and m. This simplification allows us to derive the power flow decomposition, which is used within the DC-OPF optimisation problem.

3.1.5 Power flow decomposition for DC-power flow

Here we follow [39] and describe a method for calculating the active power flows within a *balanced* network, where total demand is equal to total generation. The net active power injected at bus k is

$$P_k = \sum_{m \in \ell_k} p_{k,m} = \sum_{m \in \ell_k} \frac{\theta_k - \theta_m}{x_{k,m}},$$

where

$$\ell_k = \{m \mid \text{there exists a line between nodes } k \text{ and } m\}.$$

Hence the power injected at bus k is the sum of active power for all lines flowing into bus k. Note if active power is flowing into bus k, it is negative by convention, which indicates that there is demand at bus k. To write this in matrix form for a network with n buses, we define B_{bus}, the $n \times n$ *bus susceptance* matrix, as follows

Definition 3.1. The bus susceptance matrix B_{bus} is defined as

$$B_{\text{bus}_{k,k}} = \sum_{m \in \ell_k} \frac{1}{x_{km}} \quad \text{for } k = 1, 2, \ldots, n,$$

$$B_{\text{bus}_{k,m}} = \begin{cases} \frac{-1}{x_{km}} & \text{if a line joins buses } k \text{ and } m \text{ with } k \neq m, \\ 0 & \text{if no line runs between buses } k \text{ and } m. \end{cases}$$

Now we can write the power injected at the buses P as

$$P = B_{\text{bus}}\theta$$

where P is a $n \times 1$ vector containing the net active power injected for each of the n buses, θ is a $n \times 1$ vector of voltage phase angles for the corresponding buses.

3.1.6 DC-OPF

Now that we have defined DC-power flow we can form the DC-Optimal Power Flow (DC-OPF) problem. Here we aim to minimise our objective function, the marginal electricity cost, by choosing generators to produce power. A generator being chosen to produce power is referred to has having been dispatched. This DC-OPF formulation only considers active

power generation, no line losses from resistance, and uses the linearised power flows and line limits. The DC-OPF problem is defined as follows

$$\begin{aligned} & \min_{P_G} \ c^T P_G \\ \text{subject to:} & \\ \text{Bus power:} \quad & B_{\text{bus}}\theta = P = P_G - P_D, \\ \text{Generation of Active Power:} \quad & P_G^{\min} \leq P_G \leq P_G^{\max}, \\ \text{Line Limit:} \quad & -p_{k,m}^{\max} \leq p_{k,m} = B_{\text{bus}_{k,m}}(\theta_m - \theta_k) \leq p_{k,m}^{\max}, \\ & \forall \text{ lines } (k,m). \end{aligned}$$

Here the decision variable P_G is the $n \times 1$ vector of active power generation to be dispatched, c is the $1 \times n$ vector containing the marginal cost for each generator, $P_G^{\min}$ and $P_G^{\max}$ are the respective $n \times 1$ vectors containing generator minimum and maximum active power limits, P_D is the $n \times 1$ active power demand, and $p_{k,m}^{\max}$ is the power flow line limit.

As an aside, power systems often need to operate when a component of the system is not in use or fails. Suppose our system has N components (in our case lines, buses and generators) this would mean operating with $N-1$ components. Operating a system with $N-1$ *security* means that a system with N components can still operate with $N-1$ components safely. It is common to implement OPF under $N-1$ security, as in [52, §1.1.3]. We do not extend the DC-OPF formulation to have $N-1$ security in this thesis but the methods expressed here could be extended.

3.2 DC-OPF without dependence on phase angles

In this section we describe two equivalent approaches to write DC-OPF without dependance on phase angles. The first approach is entirely self contained and is based on [39] which uses the node-arc incidence matrix. The second approach uses the Power Transfer Distribution Factor (PTDF) matrix and is detailed in [53]. We briefly cover a third approach in Appendix A.1 based on the Moore-Penrose inverse, this approach is further described in [90].

3.2.1 Using the node-arc incidence matrix

The bus susceptance matrix can be further spit into components from which we can understand how the net power injections at the buses P affects the active power flow on the lines p. We define the $n \times \ell$ *node-arc incidence* matrix N where n is the number of buses and ℓ is the number of lines. Note this is a different N to the previous section on system security. In the incidence matrix, each row corresponds to a bus and each

column corresponds to a line. If a line runs from bus k to bus m we have $N_{k,(k,m)} = 1$ and $N_{m,(k,m)} = -1$. The remaining entries are set to zero. We can rewrite

$$B_{\text{bus}} = NYN^T, \text{ where } Y = \text{diag}\left\{\frac{1}{x_{km}}\right\}.$$

Now we can write the decomposition

$$P = B_{\text{bus}}\theta = NYN^T\theta = Np,$$

where p the vector of power flows is $p = YN^\top\theta$. As in Bienstock [39, pg. 15] we can obtain the power flows given that we have the power injections as follows.

Lemma 3.2. *For the network problem defined above, assume that the total demand is equal to the total generation, that is* $\sum_k P_k = 0$*, and that the power injections* P *are feasible for the system constraints. Then we can show that*

$$p = Y\widetilde{N}^\top(\widetilde{N}Y\widetilde{N}^\top)^{-1}\widetilde{P},$$

where $\widetilde{N}$ *is obtained by removing one of the linearly dependent rows from* N*, and* $\widetilde{P}$ *by removing the corresponding row from* P*. The row removed corresponds to a bus which we call the* slack bus.

For a proof of Lemma 3.2 see Appendix A.

3.2.2 Power Transfer Distribution Factor

An alternate notation is introduced by [53], which allows us to define the DC-OPF without explicitly including the voltage phase angles within constraints. This is done by defining the Power Transfer Distribution Factor (PTDF) Matrix. The power transfer distribution factors are referred to as a "linear sensitivity" as they give a linear relationship between the change in power injection at the buses and change in active power flow on a line.

This method is slightly different yet equivalent to the methods outlined in Section 3.1.5 and Appendix A.1. We begin by introducing alternate notation used by Chatzivasileiadis [53].

Definition 3.3. We define the *line susceptance matrix* B_{line} to be an $\ell \times n$ matrix where each row corresponds to a line and each column corresponds to a bus and defined as follows

$$B_{\text{line}_{k,m}} = \begin{cases} \frac{1}{x_{k,m}} \text{ if a line runs from bus } k \text{ to } m, \\ \frac{-1}{x_{k,m}} \text{ if a line runs from bus } k \text{ to } m, \\ 0 \quad \text{if there is no line between buses } k \text{ and } m. \end{cases}$$

Equivalently, using the notation from the previous section we can write $B_{\text{line}} = YN^\top$ and hence the power flow along the lines p is given by

$$p = B_{\text{line}}\theta. \tag{3.6}$$

To form the *bus reactance matrix* we would like to invert the bus susceptance matrix B_{bus}, but we know the bus susceptance matrix is singular and non-invertible. However (as in Equation (A.1)) the power flows are only dependent on a difference in phase angle θ between buses, not the absolute phase angles. Since the system is also balanced, we can identify a *slack bus* as a reference where the phase angle is set to zero. The process for forming the bus reactance matrix is summarised as follows

1. Take the the bus susceptance matrix B_{bus} and remove the row and column corresponding with the slack bus to form $\widetilde{B}_{bus}$.

2. As $\widetilde{B}_{bus}$ is now non-singular so can be inverted, so invert it to form $\widetilde{B}_{bus}^{-1}$.

3. In the row and column corresponding with the slack bus add zeros. This forms the the bus reactance matrix X_{bus}.

The bus reactance matrix X_{bus} is a sudo-inverse for the bus susceptance matrix and gives an alternative yet equivalent method to using $\widetilde{B}_{bus}^{-1}$ for calculating the power flows p. The voltage phase angles are given by

$$\theta = X_{\text{bus}}P,$$

and we can write down the vector of active power flows p as

$$\begin{aligned} p &= B_{\text{line}}\theta \\ &= B_{\text{line}}X_{\text{bus}}P. \end{aligned}$$

This motivates defining the $PTDF$ matrix as

$$PTDF = B_{\text{line}}X_{\text{bus}}.$$

3.2.3 DC-OPF with PTDF formulation

We can now use one of the methods outlines above to describe DC-OPF without any dependence on phase angles. We chose the PTDF matrix formulation and we can write

the DC-OPF model using the PTDF matrix within constraints as follows.

$$\begin{aligned} \min_{P_G} \; & c^T P_G \\ \text{subject to :} \quad & \sum_i P_{G_i}^{\min} \leq P_{G_i} \leq \sum_i P_{G_i}^{\max}, \\ & \sum_i P_{G_i} - \sum_i P_{D_i} = 0, \\ & -p_{\text{line}}^{\max} \leq [PTDF](P_G - P_D) \leq p_{\text{line}}^{\max} \end{aligned}$$

Here the decision variable P_G is the $n \times 1$ vector of active power generation to be dispatch, c is the $1 \times n$ vector containing the cost for each generator, $P_G^{\min}$ and $P_G^{\max}$ are the respective $n \times 1$ vectors containing generator minimum and maximum active power limits, P_D is the $n \times 1$ active power demand, and $p_{km}^{\max}$ is the power flow line limit.

3.3 Conclusion

In this chapter we have presented the key concepts that we require for modelling the South Australian electricity network and construction of DC-OPF. We note that this chapter is a small summary of a large topic, and we have covered the foundation for our approach. We recommend the reader refer to our main references [39, 52] or a text book in the area such as [113] for further details.

There are many alternate forms of optimal power flow where the assumptions and network components considered influence the program. A particular example is AC optimal power flow (AC-OPF) where the optimal power flow program is formed using the full AC power flow equations. These equations are described in Section 3.1.1 and use both reactive power and active power, unlike the power flow equations that arise from applying the DC approximation on active power only.

Although use of AC-OPF is considered best practice and presents a more accurate representation of a power system [29] the increased complexity is problematic for our approach. In particular, the AC-OPF problem is usually a non-linear and non-convex program requiring more difficult solution methods that are not covered in Chapter 2. Often *convex relaxations* are used to create convex programs as in [52, §4] so that convex programming methods can be used. Our approach relies on the linear constraints in DC-OPF to form and a two stage stochastic program with fixed recourse and solve with the L-shaped method. Further the Xenophon and Hill [115] model of the NEM (used to develop the South Australian network model in Section 6.1) does not contain all parameters required to form AC-OPF constraints. For example reactive power constraints are often included within AC-OPF and the reactive power limits for generators and not provided in the NEM model [115].

We have not covered modelling of power flow in networks with other components such as transformers or High Voltage DC (HVDC) lines. Details on power flow with transformers and HVDC connections can be found in [113, §6]. We also note that in general optimal power flow can be improved using a *unit commitment* approach, where binary variables are introduced that indicate whether a generator is available to be dispatched or not. We will leave this for future work.

In summary the power flow analysis presented in this chapter will be used in Chapter 4, where we use the DC-OPF with PTDF formulation as a basis for stochastic-optimal power flow. The South Australian Network in Chapter 6 will be defined using the conventions given in this chapter.

Chapter 4

Stochastic-optimal power flow

In this chapter we build on the background of previous chapters to demonstrate how optimal power flow can be formed as a two stage stochastic program called *stochastic-optimal power flow* (S-OPF). That is, we form a DC-OPF using the PTDF formulation of Chapter 3 as a two stage stochastic program with fixed recourse as in Chapter 2. We use this formulation to simulate dispatching power from generators over time.

To define our S-OPF we need to first model interconnectors in Section 4.2, which is new, and storage in Section 4.3 which is adapted from [62]. We have further details on the storage model in Appendix B. Our models are constructed in such a way that if a network used 100% renewable generation, then the cost would be zero and we further detail the relationship between costs and renewable generation in Section 4.4. We do not explicitly account for stability requirements, such as where minimum amounts of gas power are required to be available.

We describe how first stage and second stage generators are chosen in Section 4.1.1, keeping the SA network in mind for our approach. Using this, we introduce two test networks, Network 1 and Network 2, and apply S-OPF to them under varying conditions.

On Network 1 we demonstrate in practice how S-OPF is implemented and show that simulation results demonstrate expected behaviour with and without storage. This includes showing the benefit of S-OPF over deterministic methods for optimal power dispatch and showing how line limits affect optimal decisions.

Network 2 is designed to have features analogous to the South Australian network including an interconnector, storage and a mixture of generators. We use S-OPF to understand the optimal storage location in the network. Finally, we use the single replication procedure from Section 2.7 to understand how sample size impacts the accuracy of the SAA method.

We begin by providing context for our approach before defining our S-OPF.

4.1 Formulation of stochastic-optimal power flow

As described in Chapter 3, DC-OPF is one method of optimising the dispatch of power within a network while satisfying the constraints of the network. Within a power system generators are dispatched continuously at set intervals of time. Prior to October 2021 [8], generators would bid to supply energy to the NEM for each half hour interval. AEMO then compiles these bids and choses the generators to dispatch in five minutes blocks. The price is the averaged out over the half hour interval [5]. For simulation of dispatch over time in our model, we will assume an interval between dispatch times of half hour to align with this. We also take this approach because the demand data we use is available at half-hour increments.

In DC-OPF the power outputs for each generator are the decision variables. Recall from Section 2.4.1 that in a two-stage stochastic program with fixed recourse there are first stage and second stage decision variables. First stage decisions need to be made prior to the realisation of random variables, whereas second stage decision depend directly or indirectly on the outcome of random variables.

To form the S-OPF we need to first decide what parameters are to be modelled as random variables. Next we can understand how decision variables (power outputs for the generators) depend on these stochastic parameters and hence choose the first stage and second stage power generators. This process will vary depending on the components of the particular network under consideration. Here we initially present the general formulation before proceeding to consider specific network examples.

We assume that a network is transitioning to 100% renewable generation and hence will have a mixture of generation types. These generators are incorporated within the first stage/second stage structure.

The two categories of random variables that we incorporate within our S-OPF are:

- Power demand at each node, and
- Wind farm generation capacity (which we define as the maximum wind power available given the weather conditions).

Definition 4.1. Our *stochastic-optimal power flow* (S-OPF) (presented in SAA form as

in Definition 2.22) is an optimisation program of the form

$$\begin{aligned}
&\min_{P_{G_1}, P^1_{G_2}, \ldots, P^r_{G_2}} \quad c^\top P_{G_1} + \frac{1}{r}\sum_{k=1}^{r} q^\top P^k_{G_2} \\
&\text{subject to :} \\
&\qquad P^{\min}_{G_1} \leq P_{G_1} \leq P^{\max}_{G_1}, \\
&\qquad P^{\min}_{G_2} \leq P^k_{G_2} \leq \boldsymbol{P}^{\max_k}_{G_2} \ \text{ for } \ k = 1, \ldots, r., \\
&\qquad \sum P_G - \sum \boldsymbol{P}^k_D = 0, \ \text{ for } \ k = 1, \ldots, r., \\
&-p^{\max}_{\text{line}} \leq [PTDF](P_G - \boldsymbol{P}^k_D) \leq p^{\max}_{\text{line}} \ \ k = 1, \ldots, r.
\end{aligned}$$

Here we denote the vector of first stage power outputs of the generators with $P_{G_1} \in \mathbb{R}^{n_1}$ where $c \in \mathbb{R}^{n_1}$ is the associated first stage marginal cost $P^{\max}_{G_1}$ and $P^{\min}_{G_1}$ are the first stage generation upper and lower bounds respectively; $P^k_{G_2} \in \mathbb{R}^{n_2}$ is the vector of second stage power outputs of the generators for the k^{th} scenario where $q \in \mathbb{R}^{n_2}$ is the second stage stage marginal cost and $\boldsymbol{P}^{\max_k}_{G_2}$ and $P^{\min}_{G_2}$ are the second stage generation upper and lower bounds respectively; and $P_G \in \mathbb{R}^n$ vector generation measured from the n buses with $\boldsymbol{P}^k_D \in \mathbb{R}^n$ the vector of demand measured from the n buses. We also have that $p^{\max}_{\text{line}} \in \mathbb{R}^\ell$ is the power flow line limit, and $[PTDF]$ is the $\ell \times n$ power transfer distribution factor matrix defined in Chapter 3.

We use the same convention as in the two-stage stochastic program section (Section 2.4.1), where bold face font is used to represent a vector of random variables. The superscript k for random variables represents the k^{th} sample of the random variable out of the r total realisations in the SAA. The k superscript on $P^k_{G_2}$ represents the second stage decision variable corresponding with the k^{th} sample. We also note that the second stage cost q is not a random variable since we are assuming the cost is fixed regardless of the realisation of the random variables. We make note that not all elements of $\boldsymbol{P}^{\max_k}_{G_2}$ and $\boldsymbol{P}^k_D$ are necessarily random variables. If an element is fixed and deterministic it can be incorporated within the stochastic framework by assigning its value with probability one. The S-OPF is not written in standard form for a two-stage stochastic program with fixed recourse, rather it is written in a form that corresponds to the formulation of DC-OPF as in Section 3.2.3.

4.1.1 Choosing first stage and second stage generators

In our analysis we consider networks that may include a range of generation sources such as gas power stations, wind generation stations, diesel generation stations, storage and

connections to adjacent networks via an interconnector. These generators have varying behaviour and play different roles within network; and they are the types of generators within the South Australian network as modelled in Chapter 6. In future work other generation sources with similar behaviour could be included in this framework.

We assume that all generators can be dispatched within their respective generation limits and hence all generators are decision variables. Sometimes within optimal power flow it is assumed that renewable generation is always dispatched at its full capacity. This is not the approach we take as we are interested in using the S-OPF results for deciding the dispatch for generators. Consider for example a wind farm where the wind farm generation capacity is a random variable and a wind generator is a second stage decision variable. A realisation of the random variable determines the upper bound of wind power under that scenario, but the amount chosen to be dispatched may be less.

Generators often have complicated cost and production profiles. For example, they may have a non-zero lower limit of energy production, or shutting down may require considerable amounts of time, planning and additional costs. We simplify the models of generators in this thesis. For first stage and second stage generation we are mostly concerned with response times. We consider first stage variables to be too slow for an immediate response after the realisation of random variables and system conditions and hence need to be decided prior realisation of any random variables. This will depend on the particular characteristic of the power station and is the approach we take when considering categorising first stage and second stage generators for South Australia in Section 6.1. However in the test networks that follow in Section 4.6.2 and Section 4.6.6 we make some assumptions regarding this choice.

Storage can be either first stage or a second stage variable or a mixture which may depend on the type of storage. Storage units such as the Hornsdale Power Reserve in South Australia have a proportion of their dispatch allocated to system security services [3]. This capability is often utilised for balancing of the systems generation and demand when an imbalance contributes to system instability and must be reserved. Hence to model these capabilities we can split some storage devices into first stage and second stage variables to better represent the system.

Unlike traditional generators such as gas power where we assume the fuel is readily available, the power output for a storage device is dependent on the capacity and the charge as well as the physical power limit. Further, a storage device can draw power from the network when charging and in this way be perceived as demand within the network. We now define a model for generic storage devices the we use within the S-OPF.

4.2 Continuous linear model for storage

The way in which storage can be modelled will depend on the optimal power flow formulation. A review of storage models is presented in Eyisi et al. [62]. We base our model for

charging and discharging on the *Continuous Linear Model* [62]. Unlike other models the continuous linear model does not include binary variables that govern the state of storage (charging/discharging) and does not model any power losses. In addition, we use this model as it does not depend on the type of storage and it maintains linear constraints. The linear constraints allow it to be used within the two-stage stochastic program with fixed recourse to be solved with the L-shaped method. Extensions of the storage model are not covered in this thesis.

We use subscript notation ES, n, t as in [62] were ES indicates the source of power is associated with an energy storage unit, n is an index assigned to the storage unit and t is the time. The power dispatched $P^{net}_{ES,n,t}$ from the storage unit is limited by the maximum and minimum power outputs $P^{max}_{ES,n}$ and $-P^{\min}_{ES,n}$ respectively. If $P^{net}_{ES,n,t}$ is negative the storage unit is charging, and if positive the storage unit is discharging. The storage units have a maximum and minimum capacity $E^{\max}_{ES,n}$ and $E^{\min}_{ES,n}$ respectively. Often $E^{\min}_{ES,n}$ is taken to be zero, which we use here.

For each time t the following constraints govern the charging and discharging ($P^{net}_{ES,n,t}$) from all energy storage units:

$$P^{\min}_{ES,n} \leq P^{net}_{ES,n,t} \leq P^{max}_{ES,n},$$
$$E^{\min}_{ES,n} \leq E_{ES,n,t-1} + \delta t P^{net}_{ES,n,t} \leq E^{max}_{ES,n},$$

Note that the power dispatch for energy storage units is further constrained in the S-OPF with other constraints such as line limits. See Appendix B for how the *Continuous Linear Model* constraints are equivalently formulated and applied in practice within the S-OPF over time using the algorithm for *initialising available power output and charge for each storage unit* detailed in Algorithm 1 and the algorithm for *updating available power output and charge for each storage unit* detailed in Algorithm 2.

4.3 Modelling an interconnector

The NEM is composed of five regions that can exchange power through large capacity lines know as interconnectors. South Australia is connected to another region, Victoria, with two separate interconnectors [19]. The power flows on an interconnector are dependent on the state of the two networks connected by the interconnector.

We would like to account for interconnectors in our network while not explicitly considering the state of the other network. Here we model an interconnector in a manner that takes into account circumstances of South Australia's transition to renewables. That is, South Australia is aiming to be a net 100% renewable power system. Hence we treat the interconnector as a 'last resort' and, instead of modelling it as a line, it is modelled as a storage unit (exporting) and a generator (importing), at the appropriate bus. With this in mind we assume the following behaviour

- Excess power should first be used for charging storage devices (such as the Hornsdale Power Reserve) before being exported via an interconnector.
- Power from the interconnector should only be utilised when other sources of generation cannot satisfy the demand within the network.

To achieve this behaviour the storage unit for exporting for the interconnector will have the following properties

- infinite storage capacity,
- charging power limit $P_{ES,n}^{\min}$ set to the line limit of the interconnector,
- discharging power limit $P_{ES,n}^{\max}$ set to zero (since the storage unit is not used for modelling importing).
- a fixed marginal cost of a measured in \$/MWh, where the marginal cost of exporting for the interconnector is slightly lower than that of all other storage units within the network to incentivise charging storage devices within the network first.

The generator for importing for the interconnector will have the following properties

- maximum power limit set to the interconnector line limit (MW).
- minimum power limit set to zero (MW).
- a fixed marginal cost of b measured in \$/MWh, where this cost is the highest of all generators such that it is only used when needed for feasibility.

We will use this model for an interconnector on a test network to demonstrate the resulting behaviour in Section 4.6.6. Note that while the interconnector has effectively infinite capacity, its use comes at a high cost so it is not prioritised with respect to all other generators and only used for feasibility of the network. In addition, any line limits connecting the interconnector to the network also limit its generation.

4.4 Relationship between costs and renewable generation

In this chapter, and for the rest of this thesis, we will consider the important concept of transitioning to a 100% renewable electricity network. Note that the South Australian electricity network is connected to Victoria through interconnectors, and that the states in the NEM often export or import electricity through these. Therefore we will define 100% renewable as any network that uses only renewable or storage generators and exports more electricity than it imports. To that end, we describe how our choice of marginal

costs for the different generators, interconnectors and storage will result in the network using 100% renewable generation when the cost is zero, and how the cost is an indicator of what proportion of renewables is being used.

The models for wind dispatch in this section, as in the SA network in Chapter 6, have a marginal cost of zero. This means renewable generation is always used whenever possible and that if 100% renewable generation is used then the total marginal cost is zero.

The cost of storage charging is set to be the same magnitude as the cost of storage discharging and assumed to be very low relative to non-renewable generation costs. This means that charging the battery will result in a slightly negative cost, which then implies that if it is possible to charge cheaply (e.g. with excess renewable generation) then this will occur. Also, if there is more demand than supply from renewables available, then storage capacity will be used first before non-renewable generation where possible. As storage units are charged and discharged with costs of the same magnitude, over time this contributes to an average cost of zero, but may be slightly higher or lower depending on the difference between the initial and final charges of the storage.

The cost of the importing power via the interconnector is set to be very high, higher than all other generation, so that this is only used as a last resort, given that this is likely to be a mixture of renewable and non-renewable generation. The cost of export is a smaller in magnitude than that of storage units in the network which ensures that storage units are charged first before exporting occurs. Note that exporting power is considered to be negative and hence the cost of exporting is negative. The relative costs of dispatch compared to export mean that if the interconnector is used at all, there must be significantly more generation exported along the interconnector for the interconnector for the cumulative costs over time to be zero.

The cost of non-renewable generation is chosen to be reflective of the marginal costs for generators in the South Australian network, which is set to be higher than the storage dispatch cost but lower than the interconnector dispatch cost. Any non-renewable generation therefore contributes to increasing the cost. Note that only the relative costs matter for the optimisation process to determine which generator is selected, however we will intend to use the actual costs for generators where possible in our SA network model so that the absolute costs should be realistic, as we discuss further in Section 6.1.

When we consider our test networks in the following sections, we will study the cumulative costs over time. Given our marginal costs, the lower these cumulative costs are, the less non-renewable energy is being dispatched and the closer the network is to using 100% renewable generation. We will consider that if the marginal cost is zero (or close to zero up to some tolerance depending on the initial and final charges of the batteries) then the network that it is likely to be using 100% renewable generation. Therefore one of our aims will be consider how to reduce costs, and hence drive up the use of renewables.

4.5 Simulating wind power and electricity demand

At a given dispatch time there is a distribution of scenarios the random variables (wind and demand) can take. Using time series ARIMA models we can model the distributions for these random variables from which we draw samples to form the scenarios, as we do for wind power and electricity demand in the SA network in Chapter 5. We draw enough samples to 'cover' the distribution of possible scenarios. The more samples we have the more accurate the SAA solution is to the true solution for the S-OPF. However, there is a trade off between accuracy and computational time. This is discussed in further detail in Section 4.6.8. Under the assumption that a suitable number of samples are drawn, then the samples will represent all possible scenarios and hence one of these samples must 'occur'. This sample is then fed back into the ARIMA model to generate samples at the next dispatch time.

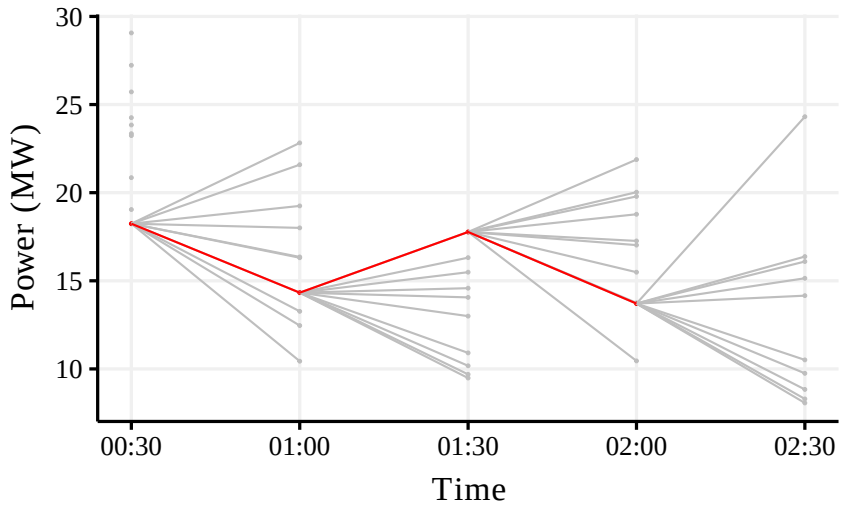


Figure 4.1: This plot shows the observed value a random variable takes over time (red) as well as the possible scenarios that could have occurred (grey). At the last time step (2:30) we see the ten possible scenarios that could occur. These scenarios are used to form constraints within the S-OPF

In Figure 4.1 we give a visualisation of the sampling procedure. More specifically we describe the procedure as follows. At a given point in time t and using the outcome of the random variables $x_1, x_2, \ldots, x_{t-1}$ at all previous times $1, 2, \ldots, t-1$, the ARIMA model is used to generate several samples (called scenarios) for the possible outcomes of the random variable x_t. These possible outcomes are plotted in grey in Figure 4.1. The SAA S-OPF will be run on these scenarios, using their values within the constraints. Then a

single scenario will be selected as the true outcome, x_t. This scenario is plotted in red in Figure 4.1. This process is then repeated until all time steps have been covered. Note that this is distinctly different from generating multiple sample paths.

4.6 Implementation of S-OPF in Julia

We now apply our stochastic-OPF formulation to test networks we have constructed to demonstrate how the S-OPF is implemented and the benefits of its use. To form and solve the S-OPF we use the language Julia [37] with the Stochastic Programs [38] package.

Recall that to simulate dispatch over time we will be solving the S-OPF at 30 minute intervals. The constraints within the stochastic program will be adjusted using the scenarios simulations for wind and demand. As previously mentioned the stochastic programs package uses the JuMP package [56] where optimisation programs can be defined using standard mathematical notation such as variables. This means we can define constraints that depend on the power network and random variables, and then easily update the constraints for each time interval by reassigning values to variables. We also make use of the Stochastic Programs L-shaped method solver. This is an efficient algorithm for solving programs with decomposition structure as presented in Chapter 2.

We first discuss the feasibility of the test networks considered.

4.6.1 Feasibility of S-OPF for test networks

In both test networks, we have set the parameters so that there is a very high chance that the corresponding S-OPF will have a feasible solution. We note that there is a very small chance that the demand will be too high and that network parameters prevent the generators from supplying adequate electricity. We did not observe this occurring in either network at any time step for any simulation except in Section 4.6.5 where we have reduced the number of generators. This suggests there is a very high probability of feasibility.

We can now consider whether our S-OPFs satisfy any of the results in Section 2.4.3 regarding existence of solutions. In particular Proposition 2.36 tells us that if the random vector $\boldsymbol{\zeta}$ has finite second moments, if there is a non-empty feasible set $\mathcal{D}$, and if the cost function z is bounded from below, then a finite optimal value is obtained. In our case, the $\boldsymbol{\zeta}$ are generated at each time step using an ARIMA time series model with normal error, which has finite moments. Each of the generators we use will have maximum and minimum limits, which means that the cost function is bounded from above and below. Therefore if we do have a feasible solution, then we have a finite optimal value that can be obtained for the S-OPF.

Finally Theorem 2.38 says that the L-shaped method applied to this S-OPF must converge in finitely many iterations to the optimal value. Hence using the L-shaped

method for our S-OPFs will obtain the optimal solution whenever there is at least one feasible solution.

4.6.2 Network 1

We begin with the test network with network diagram defined in Figure 4.2. We run the S-OPF over a 24 hour period and draw 100 realisations of the random variables to form scenarios at each time step. Using the same realisations for the random variables we compare two cases — with storage and without storage at bus 3. We also compare the optimal decision when the expected value program (EV) (as defined in Definition 2.41) is solved for the first stage solution.

To run our stochastic-OPF on this network we first need to make some assumptions and define some parameters. We define the generators in the network including storage in Table 4.1.

Table 4.1: Generators

ID	node -	stage -	min (MW)	max (MW)	marginal cost ($/MWh)	capacity MWh
gas	1	1	0.0	100.0	60.0	NA
wind	2	2	0.0	RV	0.0	NA
storage	3	2	−50.0	50.0	4.0	200.0
diesel	4	2	0.0	2000.0	300.0	NA

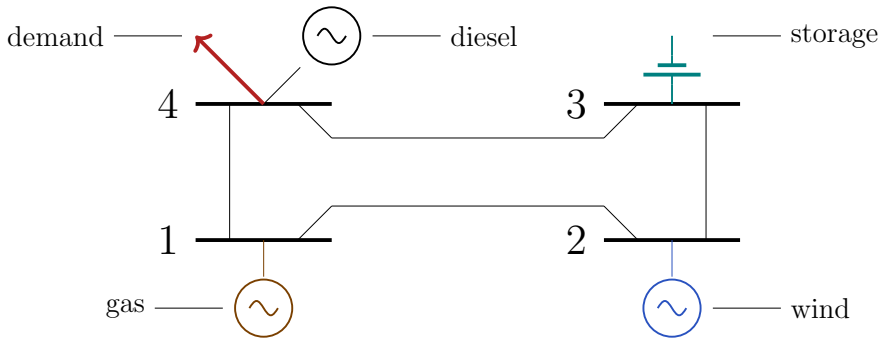


Figure 4.2: Network 1 - This simple four bus network with a mixture of generation types and first stage and second stage generation. Each line has a reactance of 0.1 (p.u.) and each assigned a relatively large power flow capacity such that there are effectively no line limits.

Recall that the random variables we consider are demand and wind power capacity.

We assume that the gas power output needs to be chosen prior to the realisation of any random variables and hence gas is a first stage variable. The second stage variables are wind, storage and diesel and are not chosen until after the random variables are realised. Storage and diesel have higher marginal cost than wind power and hence will effectively act as back up generators where they will only be dispatched when absolutely necessary.

The storage unit is modelled as described in Section 4.2. The initial storage at $t = 0$ is assumed to be half the total capacity, where we have $E_{ES,1,0} = \frac{1}{2}E^{max}_{ES,1}$. Note that different initial conditions for the initial storage charge we expect to have different results, particularly in the early time steps.

Both demand at bus 4 and wind power capacity scenarios are simulated using methods described in Section 4.5 however, to add some extra variability the ARMA model for wind is altered after the first 24 time steps (12 hr) such that the wind power in much lower on average. This allows us to see how the network behaves when wind power capacity is relatively high and low.

4.6.3 Discussion of results for Network 1

The optimal power dispatch and cost for the varying cases are plotted in Figure 4.3. The power dispatch for when there is no storage at bus 3 from the S-OPF is given in Figure 4.3a. Along with the power from each generator, the power demand and the wind power capacity are also plotted.

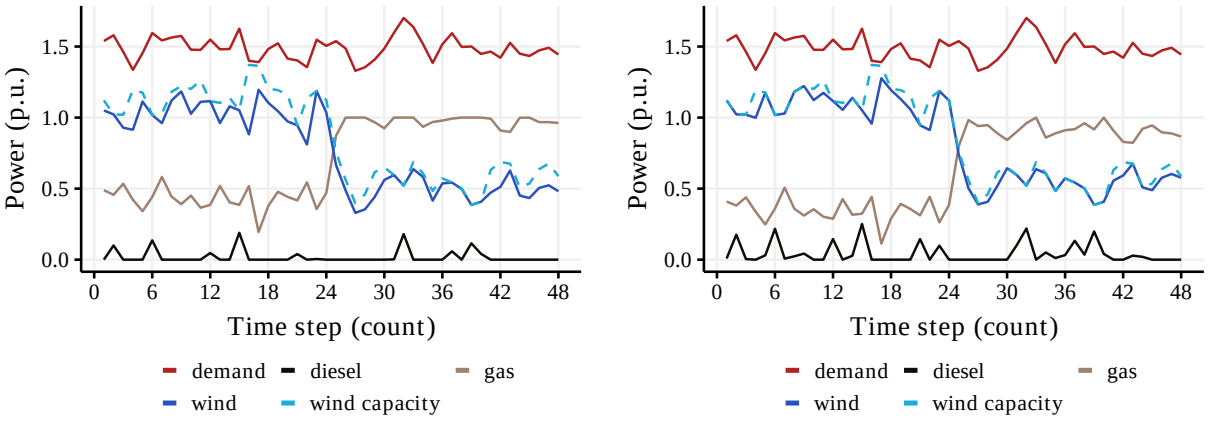
When wind is plentiful in the initial 24 time steps the power is dispatched in order of least cost. Alternatively, in the following 24 time steps when wind power decreases significantly more gas power is required to meet demand. Diesel power is dispatched in small amounts to meet the remaining short fall.

In Figure 4.3b where the first stage gas power decision is made with the expected value program as in Definition 2.41, the optimal decision slightly differs. On average less gas power is dispatched with the expected value program, although at times this means more wind power is used it also means that more diesel generation is needed to meet demand.

Figure 4.3c shows the power dispatched when storage is included at bus 3. Focusing on the other generators we see that all of the wind power capacity is used and on average less gas and diesel power is used when comparing to the case with storage at bus 3.

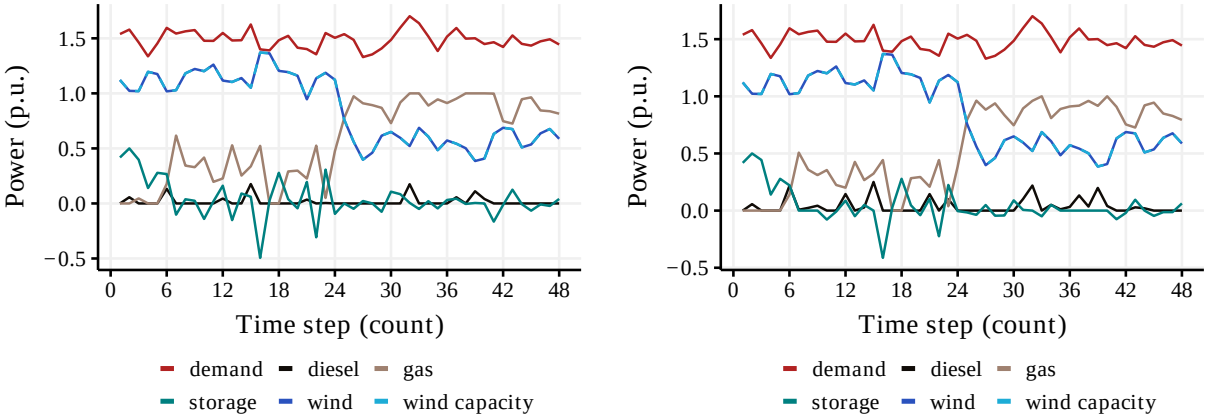
When storage is included in the network and the first stage decision is made using the expected value program the power dispatched is shown in Figure 4.3d. Again, all wind power capacity is used however no more diesel is used. A clear comparison of the resulting cost is made when comparing the cumulative cost of power for each case as shown in Figure 4.3e. Here we note that the no storage option shows around a 14% improvement in cumulative cost compared to the no storage EV option; the storage EV improves on this by a further 2%, and the storage option improves on this again by a further 10% compared to the non storage EV option. This test case clearly demonstrates how both

storage and using stochastic programming methods with optimal power flow contribute to making better decisions and a lower cost for power.



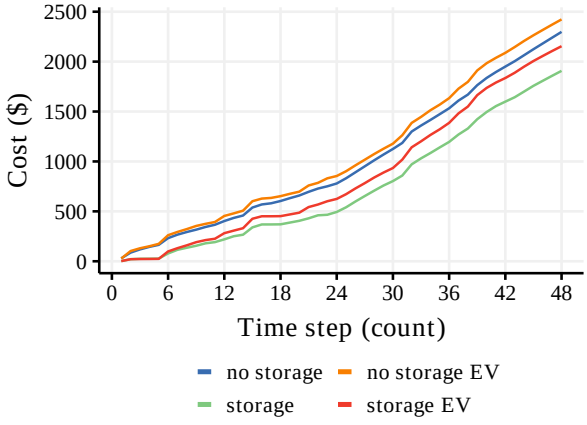
(a) No storage

(b) No storage with EV first stage



(c) Storage

(d) Storage with EV first stage



(e) Cumulative Cost

Figure 4.3: Power dispatch under different storage options for Network 1 Figure 4.2 and a comparison of cumulative cost. There are clear reductions in cost from using storage within the network and further reduction is cost from solving the two stage stochastic program rather than the EV method (Definition 2.41) when making the first stage decision. Note that at time step 24 the wind reduces on average power which subsequently increases the slope of the cumulative cost curve.

4.6.4 Network constraints

In this section we see how network constraints make a difference to the dispatch of power and the cost of electricity, improving on the methods of a copperplate network.

Here we consider Network 1 (Figure 4.2) with new scenarios drawn for random variables. Previously line power flow limits were effectively infinite, we now limit the power flow capacity on $\text{line}_{1,4}$ to be 0.55 p.u. while the remanding lines have a relatively large capacity. This affects the power generation mix is shown in Figure 4.4.

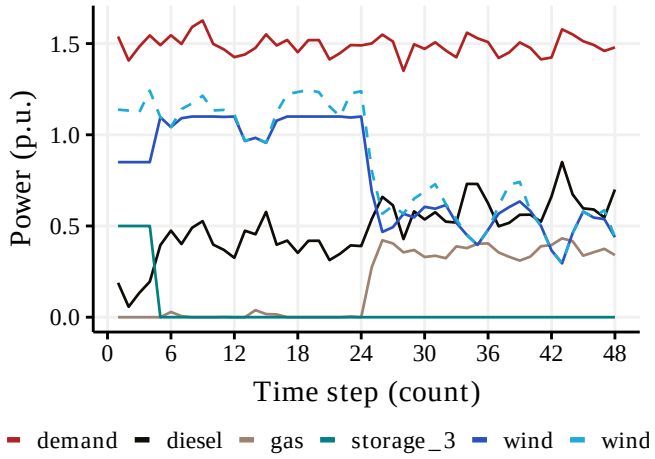


Figure 4.4: Power dispatch for Network 1 with $\text{line}_{1,4}$ capacity limited to 0.55 p.u. The change of capacity to one line within this network affects the power generation mix. This shows that it is important to consider the network topology and line capacities as it could determine how well a generator can be utilised. Hence this will be important for considering where to include storage within a network.

We see what looks like a strange result. Wind power is initially dispatched at a 0.85 p.u. before increasing to approximately 1.1 p.u. and is at times well below the varying maximum capacity for wind power, despite this energy source being the cheapest to generate. Further to this, more diesel is being dispatched rather than gas power despite diesel being more expensive than gas. We will show below that the reason this occurs is due to the line limits constraining the network.

Consider the constraints for the forward line limits as defined in Definition 4.1.

$$[PTDF]P_G \leq p_{\text{line}}^{\max} + [PTDF]\boldsymbol{P}_D^k$$

where $[PTDF]$ is a power transfer distribution factor matrix, P_G is the vector of power generation for the buses, $p_{\text{line}}^{\max}$ is the vector of power flow line limits, and $\boldsymbol{P}_D^k$ is the vector of demand at the buses.

Substituting the values for these parameters we see

$$\begin{bmatrix} 0.0 & -0.75 & -0.5 & -0.25 \\ 0.0 & -0.25 & -0.5 & -0.75 \\ 0.0 & 0.25 & -0.5 & -0.25 \\ 0.0 & 0.25 & 0.5 & -0.25 \end{bmatrix} \cdot \begin{bmatrix} \text{gas} \\ \text{wind} \\ \text{storage} \\ \text{diesel} \end{bmatrix} \leq \begin{bmatrix} 100.0 \\ 0.55 \\ 100.0 \\ 100.0 \end{bmatrix} + \begin{bmatrix} 0.0 & -0.75 & -0.5 & -0.25 \\ 0.0 & -0.25 & -0.5 & -0.75 \\ 0.0 & 0.25 & -0.5 & -0.25 \\ 0.0 & 0.25 & 0.5 & -0.25 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \boldsymbol{D}_4 \end{bmatrix}.$$

Here we use $\boldsymbol{D}_4$ to denote stochastic demand at bus 4. This matrix equations gives four separate constraints. The first, third and fourth constraints are irrelevant for wind as they are always easily satisfied. If we consider the second constraint where the adjusted line limit has an effect we have

$$-0.25(\text{wind}) - 0.5(\text{storage}) - 0.75(\text{diesel}) \leq 0.55 - 0.75(\boldsymbol{D}_4).$$

However we also know that the sum of generation must equal demand which implies

$$-0.25(\text{wind}) - 0.5(\text{storage}) - 0.75(\text{diesel}) \leq 0.55 - 0.75(\text{gas} + \text{wind} + \text{storage} + \text{diesel}),$$

which gives

$$0.5(\text{wind}) + 0.25(\text{storage}) + 0.75(\text{gas}) \leq 0.55. \tag{4.1}$$

Wind and storage have lower marginal cost than gas and hence if gas is not required we must at least have

$$0.5(\text{wind}) + 0.25(\text{storage}) \leq 0.55$$

or equivalently,

$$\text{wind} + 0.5(\text{storage}) \leq 1.1.$$

If storage is charged and dispatched at full capacity of 0.5 p.u. then we have wind $\leq$ 0.85. Alternatively if storage is not dispatched then wind $\leq$ 1.1. This is the behaviour we see when wind power capacity is high enough.

Further, after a few time steps storage capacity is zero and hence the output is zero, so from Equation (4.1) we must have

$$0.5(\text{wind}) + 0.75(\text{gas}) \leq 0.55. \tag{4.2}$$

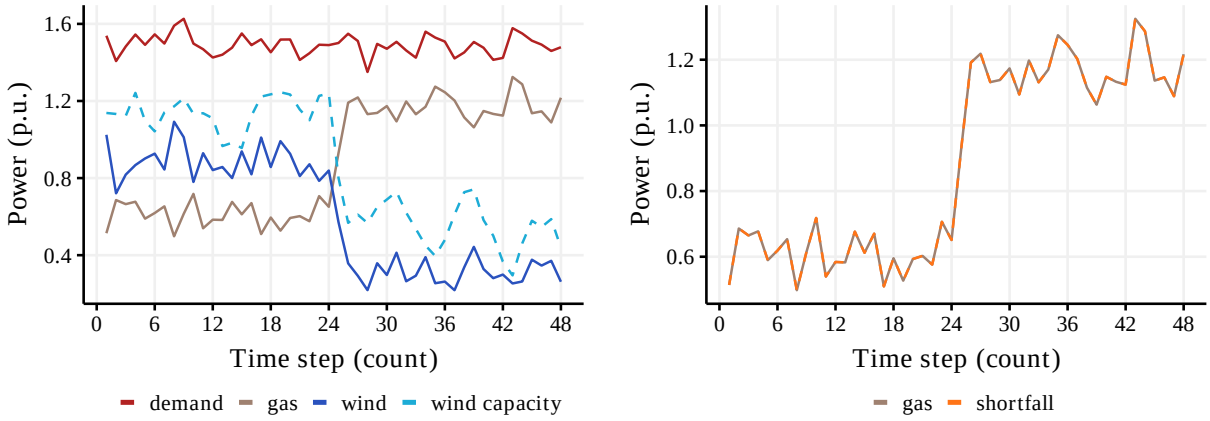
When wind power output is approximately 1.1 p.u. then gas dispatched must be approximately zero. In order to meet the demand diesel must be dispatched.

Having one line in Network 1 with limited capacity has compounding effects on the power that can be dispatched and results in more expensive generators being dispatched. This demonstrates why it is important to consider network constraints when solving optimal power flow problems and motivates our approach.

4.6.5 First stage decision is the maximum shortfall

In more simple cases with fewer generators active within a network it can be easier to see how the first stage decision is made. We demonstrate this in Figure 4.5 where the only generators we consider in Network 1 are wind and gas. When the wind capacity is high enough it provides the majority of the power to meet the demand. However, often available wind generation capacity is substantially higher than wind power that is dispatched. This is suboptimal, in the sense that using more wind power costs less than gas power.

However, the algorithm is optimising over possible outcomes of wind and demand, and the actual outcome for these is not known when gas needs to be chosen at the first stage. Gas is chosen so that even in the worst outcome for the wind, there is still enough total generation to meet all possible demand. Below in Figure 4.5b we plot the *maximum shortfall* (the maximum difference between wind capacity and demand over all scenarios at that time step) on the time, and also plot the gas power dispatch and we can see that they are identical as expected. This confirms the algorithm is performing as expected. This is the reason why wind dispatch is much lower than the wind capacity in Figure 4.5b.



(a) Power dispatch

(b) Maximum shortfall of wind power

Figure 4.5: In Figure 4.5a is the power dispatch for Network 1 (Figure 4.2) in the case of no diesel generation or storage. In this simple case the only generators are wind and gas, providing enough power to meet the demand at all times. As expected, with only these two generators, the wind power is rarely used at its full capacity. This is because enough gas must be dispatched in the first stage for all scenarios. This is clearly shown in Figure 4.5b where the gas power dispatched is identical to the maximum shortfall.

This network does not have guaranteed feasibility. Although all the scenarios drawn in this simulation have feasible solutions, there are other scenarios that can be generated

that do not have feasible solutions. Due to the stochasticity within the system it is not possible to guarantee feasibility however it is possible to have a high probability of feasibility. This is the case with Network 1 when all generators are active. In reality, energy market operators have other mechanisms available, such as demand response [27], to ensure network constraints are almost always satisfied.

The first stage decision variable, gas power, has been dispatched at the maximum shortfall as shown in Figure 4.5b. However, the obvious case where this would be an issue is if the maximum shortfall was greater than the maximum generation limit for gas and hence not enough gas could be dispatched to satisfy the maximum shortfall. A less obvious case occurs when there is a scenario such that the demand is less than the maximum shortfall. This would mean gas needs to be chosen to satisfy the maximum shortfall whilst also being less than the minimum demand, which is not possible and hence causes infeasibility. However if this network had an interconnector or storage that could be charged then when need they could draw the extra electricity produced from the gas power station and this problem would not occur. The next test network we consider has both storage and an interconnector. We seek to understand how these components interact with the network under S-OPF.

4.6.6 Network 2

We now run S-OPF on Network 2 as shown in Figure 4.6. We have designed this network to have some similar features to that of South Australia's transmission network or a network transitioning towards 100% renewable generation. This network includes an interconnector between our network of interest (Network 2) and another large network. The large system can supply electricity and can also use electricity. This is analogous to the situation in South Australia. Recall South Australia is connected to the National Electricity Market via two interconnectors between South Australia and Victoria. We use the methods described in Section 4.3 to model this interconnector and include it within the S-OPF.

Network 2 also contains storage co-located with renewable generation and storage co-located with demand. We do this to understand the relationship between locating storage at demand or renewable generation whilst having backup capacity from an interconnector. Gas generation is also co-located with demand as is the case with gas power generally being located near large populations.

We again run the S-OPF under different storage situations: no storage, storage only at bus 2, storage only at bus 3 and storage at both buses 2 and 3 where we seek to understand which location is most advantageous.

Table 4.2: Generators in Network 2

ID	node -	stage -	min (MW)	max (MW)	marginal cost ($/MWh)	capacity MWh
gas	1	1	0.0	100.0	60.0	NA
storage 2	2	2	−50.0	50.0	5.0	150.0
storage 3	3	2	−50.0	50.0	5.0	150.0
wind	3	2	0.0	RV	0.0	NA
interconnector export	4	2	0.0	100.0	4.0	NA
interconnector import	4	2	0.0	200.0	6000.0	NA

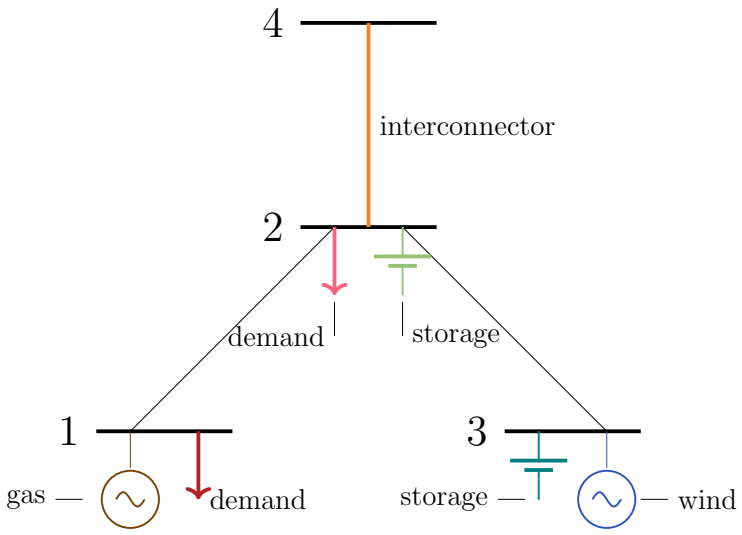
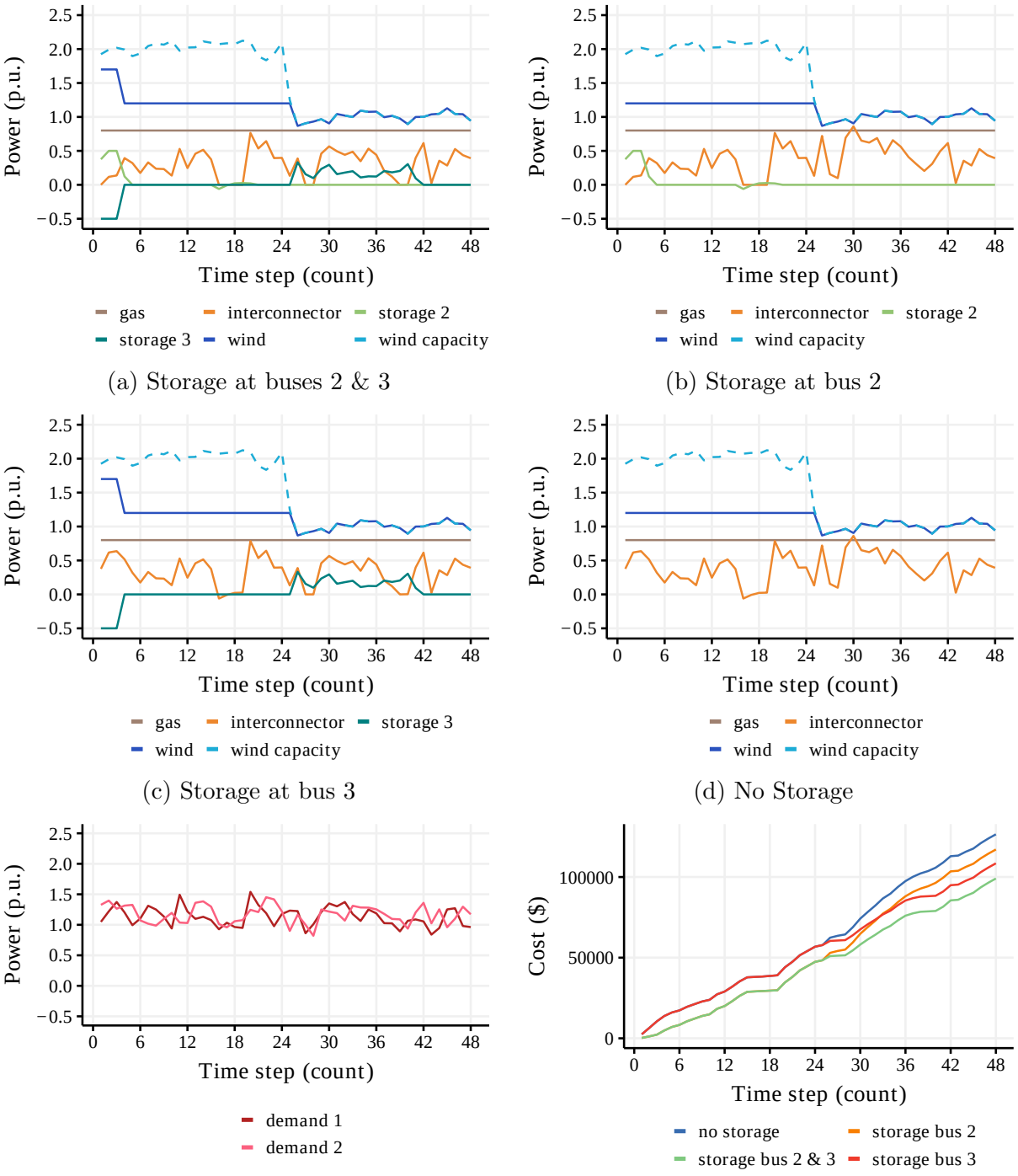


Figure 4.6: Network 2 - This network contains an interconnector that joins our network of concern (buses 1,2 and 3) with another large network represented by bus 4. Network 2 also contains a mixture of generators, demand and storage. Note here we assume $\text{line}_{1,2}$ and $\text{line}_{2,3}$ have power flow limits of 1.2 p.u. while the interconnector import and export limits are defined in Table 4.2. Note that a positive value for the interconnector indicates importing power and negative indicated exporting.



(a) Storage at buses 2 & 3

(b) Storage at bus 2

(c) Storage at bus 3

(d) No Storage

(e) Demand

(f) Cumulative Cost

Figure 4.7: Power dispatch under different storage locations for Network 2 (Figure 4.6). Having storage at both buses 2 and 3 is optimal. If storage can only be located at one bus, the optimal location depends on the relationship between the wind conditions and the storage capacity.

4.6.7 Discussion of results for Network 2

In Figure 4.7 we have plotted the power dispatched and cumulative cost for each storage case as well as power demand. Recall we have again simulated over 48 time steps and in each case the same stochastic scenarios have been used within the S-OPF. The maximum wind capacity has again been simulated to have a *change event* where capacity is significantly decreased from the 24^{th} time step. We also commence with both storage units having half their maximum capacity of 150MWh.

In all cases gas power is used at full capacity to meet the co-located demand at bus 1. Similarly, wind power is used at maximum network capacity when accounting for the combined factors of wind capacity, co-located storage and line limits of 1.2 p.u. The interconnector is relied on when demand cannot be satisfied by other generators. If storage can be utilised there is less reliance on the interconnector, hence lowering the cost of electricity. If network constraints allow, charging of storage is prioritised over exporting power on the interconnector due to the relative costs.

The cumulative cost is plotted for each storage case in Figure 4.7f, this gives an initial guide of the best performing storage case with respect to cost. As expected it is more cost effective to have storage at both buses 2 and 3. Further, it is least cost effective to have no storage. However, when only one storage unit can be chosen the behaviour is more complex.

Initially, in the first 24 time steps when maximum wind capacity is plentiful there is no difference in cost between having storage at bus 3 and having no storage at all. Similarly there is no difference in cost between having storage at bus 2 and having storage at both buses 2 and 3 during this time period. The reason for this is the line limit along the line from bus 3 to bus 2, which is capped at 1.2p.u. When wind capacity is higher than this, as it is for the first 24 time steps, exactly 1.2p.u. flows along this line, and the excess is used to charge the storage at bus 3 until it is fully charged. All of the wind generation is used to supply the demand, and the storage at bus 2 is discharged initially. If the demand was lower or the line limit was higher (e.g. if an additional line was installed), this wind generation may also have been used to charge the storage at bus 2. This would reduce the costs and may mean that having storage at bus 2 would be more optimal than bus 3. This is a demonstration of how network constraints can change the optimal location for storage.

After the change event at 24th time step, where the maximum wind capacity decreases, it becomes more advantageous to have storage at bus 3 rather than at bus 2. This is due to the wind capacity now being lower than the 1.2p.u. line limit of the line from bus 2 to bus 3, so the charged storage at bus 3 is now able to discharge along the line until its has no charge remaining.

It appears as though having storage at both buses 2 and 3 is optimal for reducing cumulative costs. If, however, only one storage location can be chosen then the optimal

choice depends on the wind conditions over time and the respective storage capacities. Also, from further inspection we see that increasing storage capacity is not the only method to reduce costs and move towards a 100% renewable grid, as in Section 4.4. Increasing transmission capacity and renewable generation are also important. This will be important for the SA network when considering how to chose optimal storage locations and capacities.

4.6.8 Sample size for SAA solution accuracy

In this section we seek to understand how the sample size will effect the quality of our solution when using the SAA for our particular problem. In Section 2.7 we described how the sampling average approximation solution relates to the true solution of a two-stage stochastic program with fixed recourse. On average the sampling average approximation will underestimate the true cost and will on average increase towards the true solution as the number of samples increases as shown in Theorem 2.46.

Here, we calculate the approximate $(1-\alpha)100\%$ confidence interval for z^* as in Equation (2.19). The accuracy of the solution for the SAA depends of the scenarios that give rise to the first stage solution x. We obtain a candidate first stage decision $\hat{x}$ by solving the S-OPF with $k = 10$, 100, 1,000, and 10,000 samples of $\boldsymbol{\zeta}$. Then for each k we calculate six approximate 95% confidence intervals using $r = 10,000$ samples of $\boldsymbol{\zeta}$ derived from the single repetition procedure Section 2.7. This calculation is to applied to Network 1 shown in Figure 4.2 in the case of no storage. This means the generators included are gas, wind and diesel with marginal costs of \$60/MWh, \$0/MWh and \$300/MWh respectively. All output is recorded in Figure 4.8.

In each case we have

$$0.77\text{p.u.} = 77\text{MW} \leq \text{gas } = \hat{x} \leq 0.92\text{p.u.} = 92\text{MW}$$

Overall, the average gas used was around 85MW, so gas is varying between at most 9% less and 8% more than its average value, and those percentages shrink as k increases.

Recall, the confidence interval is for the true cost z^* and we see here that regardless of the number of samples k used to calculate $\hat{x}$ the width of the confidence interval is relatively small. In general, we observe that the more scenarios used in the SAA method to find $\hat{x}$ then the smaller the width of the confidence interval, as expected. Here the average width was $9.220, 4.803, 4.861, 4.065$ for $k = 10$, 100, 1,000, and 10,000 respectively. The average midpoint of the confidence intervals was 59.2, 55.6, 54.6 and 56.6 respectively, so the width in as a percent of the average midpoint was around $\pm 7.8\%$,$\pm 4.3\%$, $\pm 4.5\%$ and $\pm 3.6\%$ respectively, which is smaller than the percentage differences in the gas output.

(a) $\hat{x}$ with $k = 10$ scenarios

$\hat{x}$	Lower	Upper	Width	z_r^*
0.827	52.19	52.29	0.092	52.23
0.848	61.99	86.42	24.434	66.89
0.779	54.27	79.66	25.382	61.25
0.900	52.85	54.92	2.066	53.41
0.919	52.37	55.67	3.297	53.19
0.830	53.66	53.71	0.054	53.68

(b) $\hat{x}$ with $k = 100$ scenarios

$\hat{x}$	Lower	Upper	Width	z_r^*
0.840	52.76	52.90	0.142	52.81
0.863	59.17	74.44	15.273	63.66
0.854	46.17	51.42	5.252	47.24
0.887	58.99	60.87	1.878	59.65
0.872	51.45	53.30	1.852	51.97
0.918	50.89	55.31	4.422	51.81

(c) $\hat{x}$ with $k = 1,000$ scenarios

$\hat{x}$	Lower	Upper	Width	z_r^*
0.830	55.02	55.94	0.925	55.33
0.912	57.75	57.76	0.012	57.75
0.879	50.96	53.47	2.494	51.66
0.782	49.59	58.44	8.846	55.26
0.879	49.78	53.48	3.706	51.71
0.808	49.83	63.02	13.189	57.75

(d) $\hat{x}$ with $k = 10,000$ scenarios

$\hat{x}$	Lower	Upper	Width	z_r^*
0.806	55.28	66.91	11.63	58.90
0.830	55.79	58.08	2.287	56.60
0.770	53.07	61.72	8.649	55.96
0.854	53.39	53.59	0.206	53.46
0.808	51.65	51.66	0.009	51.65
0.875	58.12	59.74	1.611	58.69

Figure 4.8: Approximate confidence intervals for z^* using $r = 10,000$ scenarios for the intervals as in Equation (2.19). Here z_r^* is the solution for the two-stage stochastic recourse problem with the SAA method with 10,000 scenarios in \$, $\hat{x}$ is the solution for gas in p.u. (where multiplying by 100 will give the MW value for $\hat{x}$), Lower is the lower bound of the confidence interval in \$, Upper is the corresponding upper bound in \$, and the Width is the difference in \$. See Section 2.7 for further details.

Note that these calculations are computationally expensive and further investigation would be needed to understand the average width for each value of k, however these results are indicative of the expected trend.

4.7 Conclusion

In this section we have used DC-OPF from Chapter 3 and stochastic optimisation methods from Chapter 2 to form our S-OPF. We then summarised our models for storage and interconnectors, and how we will model the random variables, wind power capacity and electricity demand. We demonstrate the S-OPF formulation on two test networks, detailing the results and understanding how network constraints affect the optimal solution. We also demonstrate in a specific case how feasibility relates to maximum shortfall.

Finally we calculate confidence intervals for the true solution using the single repetition procedure from Section 2.7. The methods described in this chapter form the foundation for our approach when we apply S-OPF to the South Australian network in Chapter 6 and Section 6.4.

Chapter 5

Modelling electricity demand and renewable generation

In this chapter we discuss modelling electricity demand and wind power generation for South Australia. Here electricity demand and wind generation are considered as random variables in our S-OPF formulation from Chapter 4. Recall that we intend to use the S-OPF at half-hour intervals to model the optimal dispatch of generation to meet demand, so this chapter covers modelling the distributions for electricity demand and wind generation to forecast a half-hour ahead. We will use these forecast samples to form scenarios within the S-OPF in Chapter 6. Our aim here is to show how historic data can be used to produce forecasts for required distributions.

While the time series modelling techniques used in this chapter are known in the literature, the aim of this chapter is to show how we applied these techniques, so that the models can be used in our S-OPF. We have also applied these modelling techniques to more recent 2017-2018 data for South Australia than observed in the current literature. Overall this chapter illustrates a framework to follow when forming scenarios to be used in S-OPF or more generally in stochastic optimisation.

In this chapter we begin in Section 5.1 with covering time series methods that underpin our modelling approach. Further preliminary background on time series analysis is covered separately in Appendix C. After describing our data in Section 5.2, we detail our modelling approach for demand in Section 5.3, and we detail our modelling approach for wind in Section 5.4.

5.1 Time series analysis

In Appendix C we recall basic definitions of time series following the references [45] and [70]. In this section we focus on ARMA models for time series and how we fit these models

to our data. We use the notation of a lower case to denote observations, for example, x, and an upper case to denote random variables, for example, X.

5.1.1 Stochastic time series models

We now define specific types of stochastic processes, leading up the ARMA process. Note that ARMA processes are very general, in the sense that for any autocovariance function of a stationary process, $\gamma(k)$ (as defined in Definition C.4), such that $\lim_{k\to\infty} \gamma(k) = 0$ there is an ARMA process with this autocovariance function [47, pg. 77].

To define an ARMA process, we first need the concept of a white noise process.

Definition 5.1. A *white noise* process $\{W_t : t \in T\}$ is a stochastic process where independent samples are drawn from a fixed distribution with mean zero and constant variance. Often white noise is taken to be normal distribution and unless otherwise state this will be the assumption in the sequel. White noise processes are necessarily stationary.

More general processes may include a linear dependence which we call autoregression as follows.

Definition 5.2. An *autoregressive* process $\{X_t : t \in T\}$ of order p is a stochastic process such that

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + W_t. \tag{5.1}$$

Here $\{W_t : t \in T\}$ is white noise and the ϕ_i coefficients are constant weights. This process is often denoted as AR(p) for convenience.

For convenience we introduce the *back shift operator* B where $BX_t = X_{t-1}$ and hence $B^i X_t = X_{t-i}$. Then using the back shift notation we can write equation 5.1 as follows

$$(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p) X_t = W_t.$$

Autoregressive processes may or may not be stationary. However, here the normality of W_t means that weakly stationary (Definition C.6) and stationary (Definition C.3) coincide for autoregressive processes [45, pg. 30].

The relationship between the X_t and W_t can be rewritten so that X_t is a linear function of possibly infinitely many of the $W_t, W_{t-1}, \ldots,$. In this case, if the coefficients of the W_i are such that they form a convergent sequence, then X_t is stationary [45, pg. 10].

A special case is where only finitely many W_i needed. Then the process is called a moving average process, which is stationary by definition.

Definition 5.3. A *moving average* process $\{X_t : t \in T\}$ of order q denoted MA(q) is a stochastic process such that

$$X_t = W_t - \theta_1 W_{t-1} - \theta_2 W_{t-2} - \cdots - \theta_q W_{t-q}, \tag{5.2}$$

where the θ_i are constant coefficients and $\{W_t : t \in T\}$ is a white noise process. Alternatively, using back shift notation we can equivalently write Equation (5.2) as

$$X_t = (1 - \theta_1 B - \theta_2 B^2 - \cdots - \phi_q B^q) W_t.$$

Again we can rewrite the relationship between X_t and W_t so that there is only one W_t term and possibly infinite X_t terms. If there is finitely many of them, then the process is also an autoregressive process. However, in general autoregressive processes are not usually a moving average process and vice-versa.

We now introduce the mixed autoregressive moving average (ARMA) model.

Definition 5.4. A mixed *autoregressive-moving average* process denoted ARMA(p, q) is a stochastic process $\{X_t : t \in T\}$ such that

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + W_t - \theta_1 W_{t-1} - \theta_2 W_{t-2} - \cdots - \theta_q W_{t-q}, \tag{5.3}$$

or equivalently using backshift notation we have

$$(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p) X_t = (1 - \theta_1 B - \theta_2 B^2 - \cdots - \phi_q B^q) W_t.$$

Here $\{W_t : t \in T\}$ is a white noise process.

If an ARMA model is stationary, there are many techniques that can be used to estimate both p and q, as well as the coefficients given a time series of observations $\{x_t : t \in T\}$. For an arbitrary time series, often a transformation or a decomposition is possible to reduce it to a stationary time series which can often be modelled by an ARMA model. We consider transformations in Section 5.1.3

5.1.2 Decomposition of time series

For a time series to be accurately modelled it is common to *decompose* the time series and model these components separately. Most simply a time series can be decomposed into a *deterministic* component d_t and a *stochastic* component Y_t and summed to create an additive model such that a stochastic process for a time series X_t can be written

$$X_t = d_t + Y_t,$$

where Y_t is stationary. Here the deterministic component can often be estimated using approaches such as regression, while time series methods can be used for the stochastic

component. Often it is useful for a time series to be further decomposed. The deterministic component may contain a *trend* and/or *seasonal* effects. The way in which these components are combined will depend on the time series. Most simply the components are summed such that for a time series X_t we have,

$$X_t = t_t + s_t + Y_t.$$

alternatively when seasonal effects vary as the trend varies we can form the following multiplicative model given by

$$X_t = t_t s_t + Y_t.$$

See for example [47, §1.4] for further details. We will explore which decompositions are appropriate for our data in Sections 5.3 and 5.4.

5.1.3 Transformations of time series

Often time series, even after decomposing into a stochastic and deterministic component, do not leave a stationary stochastic component. This means they can not be easily estimated by an ARMA model. This is the case for electricity demand and wind power time series as seen in the literature such as [80]. There exists numerous methods for overcoming this. We first describe the Box-Cox transformation that can transform data to have more constant variance as defined in Equation (C.2).

Definition 5.5. Suppose we have an observed time series $\{x_t\}$ where $x_t \geq 0$ for all t, then the Box-Cox transformation is defined by

$$f(x_t, \lambda) = \begin{cases} \frac{{x_t}^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ \log(x_t) & \text{if } \lambda = 0. \end{cases}$$

The Box-Cox transformation is a function of the parameter λ, where λ is usually chosen to give a transformed time series with the most stabilised variance. There are a range of methods for selecting an appropriate λ. We make use of the *guerrero* feature within the `feasts` package [92], where λ is computed by minimisation of the coefficient of variation for a sub-series of the time series [64].

Another simple transformation that can be useful for obtaining a stationary time series is *differencing*.

Definition 5.6. *Differencing*, usually represented by the difference operator Δ, applied to a time series $\{x_t : t \in T\}$ generates the time series $\{y_t : t \in T\}$ such that

$$\begin{aligned} y_t = \Delta x_t &= x_t - x_{t-1} \\ &= (1 - B)x_t. \end{aligned}$$

We refer to y_t defined above as a first order difference but can apply higher order differencing by reapplying the differencing operator. We indicate a differencing of order d with Δ^d, for example $y_t = \Delta^d x_t$.

Some time series may not be stationarity, however they may be stationary after differencing. It is common to include differencing as part of the ARMA model, which is called an ARIMA model.

Definition 5.7. An ARIMA(p, d, q) process is a stochastic process $\{X_t : t \in T\}$ such that

$$(1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p)\Delta^d X_t = (1 - \theta_1 B - \theta_2 B^2 - \cdots - \phi_q B^q)W_t \tag{5.4}$$

where $\{W_t : t \in T\}$ is a white noise process. It leads to the ARMA(p, q) process $\{\Delta^d X_t : t \in T\}$.

We discuss how we use differencing for our data using the automatic fitting in Section 5.1.4.

5.1.4 Automatic fitting of ARIMA

Here we introduce the ARIMA automatic fitting algorithm and describe the Corrected Akaike's Information Criterion which we later use for model fitting.

Given a time series of observations $\{x_t : t \in T\}$ there are various algorithms to determine the orders of the underlying ARIMA(p, d, q) process and coefficients. We will use the ARIMA function in the `fable` package [91], which implements a variation of the Hyndman-Khandakar algorithm [71]. This algorithm initially chooses the required order of differencing d using repeated KPSS tests [76] for $0 \leq d \leq 2$ to test the null hypotheses of stationarity. It first tests $d = 0$ (no differencing) for significance, then $d = 1$ and then $d = 2$, stopping on the first test that is insignificant.

Next four ARIMA(p, d, q) models are fitted for this d and for values $(p, q) = (0, 0)$, $(2, 2), (1, 0), (0, 1)$. Each fit is implemented using the `arima` function from the R `stats` package [100]. This fitting process uses a Kalman filtering approach to determine the maximum likelihood estimators for the coefficients.

The best model is assessed using the Corrected Akaike's Information Criterion (AICc), defined below. The model with the smallest AICc is consider to have the best fit. The algorithm then fits eight adjacent models where either $0, 1$, or -1 is added to each of p, q. Again the model with the smallest AICc is chosen. The process is repeated until the current model selected has p, q equal to the previous model selected. See [70, 71] for further details.

The AICc is defined as follows

Definition 5.8. The *Corrected Akaike's Information Criterion* (AICc) for an ARIMA(p, d, q) model of a time series $\{x_t : t \in T\}$ is

$$\mathrm{AIC_c} = \mathrm{AIC} + \frac{2(k+2)(k+3)}{|T| - k - 3}. \tag{5.5}$$

Here k is the number of parameters in the model (usually $k = p+q$) and $|T|$ is the number of elements in T, *i.e.* the number of observations in the time series. *Akaike's Information Criterion* (AIC) is

$$\mathrm{AIC} = |T| \log\left(\frac{\mathrm{SSE}}{|T|}\right) + 2(k+2). \tag{5.6}$$

and the *Sum Squared Error* (SSE) is

$$\mathrm{SSE} = \sum_{t=1}^{|T|} e_t^2, \tag{5.7}$$

where e_t is the residual at time t. Both the AIC and AICc are used as a measure of goodness of fit, where the SSE component seeks to measures the residual distance from 0 while the k terms add a penalty for the number of parameters used. Hence minimising the AIC or AICc is equivalent to finding a model with small SSE while preventing overfitting. See [47, §9.2] and [70, §7.5] for further details.

Note that the AIC and the AICc are both asymptotically efficient estimators, as detailed in [69].

5.2 Electricity demand and wind generation data

Here we begin to use the time series techniques presented in Section 5.1 and Appendix C to model the random variables required in the S-OPF implementation on South Australia in Chapter 6. The two time series data sets that we consider are:

- Total electricity demand for South Australia over the time period of June 2017 to May 2018, at 30min intervals, measured in MW. In our analysis we have assumed a constant MW rate for the entire 30min interval. The data is sourced from [11] and is also used and documented in [115].
- Wind power for a collection of thirteen wind farms in South Australia over the time period of June 2017 to May 2018, at 5min intervals. The data is measured in MW and is also sourced from [11]. In our analysis we average every 6 time intervals to form data that represents 30min intervals aligning with the electricity demand data.

This process is adapted from [115]. In the South Australian electricity system model adapted from [115], generators are identified by their respective DUID (Dispatchable Unit Identifier). These DUIDS for the wind farms in the South Australian model are listed in Section 5.4.3.

The interval between time series observations is 30 minutes which corresponds to the availability of the electricity demand data, as well as the time frame over which AEMO previously required electricity bids to be placed, see [16] for further details. The fitted models are used to generate samples, forming scenarios for S-OPF on South Australia, where the 30 minute interval corresponds with the time interval between each S-OPF problem we will solve. We now describe the modelling of wind generation and electricity demand in Sections 5.3 and 5.4.1.

5.3 Demand

Methods for modelling electricity demand in Australia and South Australia are diverse, and two comprehensive studies are [15] and [63], which we are following. Another study focusing on South Australia is [80] where the author has many similarities to our implementation (also applying time series methods with periodicity and ARIMA terms). Such times series methods are often called *classical* methods. However, the data set used in these studies covered the years from 2002 to 2005 and there are significant differences in South Australian demand profile since then. For example, roof top solar installation has reduced the demand on large scale generation in South Australia [97, 105].

Other studies on electricity demand forecasting use artificial intelligence/machine learning models such as Bayesian methods and neural networks, which is not our approach. See [110] and [66] further details. Beyond Australia, energy demand has been well studied with both machine learning and classical methods, and a review of these methods is available in [1].

Our approach to modelling electricity demand follows [63] which considers electricity demand for the NEM, and is very similar to the approach taken in the example presented in [70, §12.1] that models electricity demand for Victoria. Following this modelling approach was preferable since it allows streamlined model fitting and generation of samples with the `fable` and `feasts` packages. Our focus here is not to perfectly predict electricity demand but rather generate samples that exhibit representative behaviour for use within the S-OPF. Importantly, we aim not just to create a model for forecasting, but to have a model that can generate samples at a given future time.

5.3.1 Overview of modelling electricity demand

At a high level we model the electricity demand as a stochastic process by:

- Applying the Box-Cox transformation as per Definition 5.5, which determines $\lambda = 0.0190$.

- The model is split into a deterministic component d_t and a stochastic component Y_t such that the electricity demand is the sum of the components. The deterministic component uses a regression model with the following predictor variables: work day, maximum daily temperature, and the square of maximum daily temperature; as well as regressing on Fourier components that describe the seasonality in the data.

  Workday is a binary variable where a workday is a weekday that is not a public holiday. The data for public holidays is obtained from [102] which did not require cleaning, while temperature data was obtained from [50] where three missing values on the days were 12/10/2017, 4/4/2018, and 21/5/2018 were sourced from [106].

  Our approach does differ slightly from [63] and [70, §12.1] — while we find that temperature is strongly correlated with electricity demand we do not have half hourly temperature data available. But we do have access to maximum daily temperatures and hence this is used as a predictor. Further, we opted for using maximum temperature squared as a predictor rather than having temperature as a piecewise linear predictor.

  The stochastic component is modelled using an ARIMA(p, d, q). Note that the `fable` package in R [91] can model both the regression and the ARMA components together with the automatic fitting process as in Section 5.1.4. The order for the ARIMA model is determined as $p = 2, d = 0, q = 4$.

- Using the `forecast` function from the `fable` package [91] we generate a 48hr sample path as well as confidence intervals around the sample path and visually review them in Figure 5.1. This follows the example in [70, §12.1] where the preceding 48hrs of deterministic predictors are used as the predictors. This modelling appears reasonable, and the predicted values have the same behaviour as the average of demand at half hourly intervals as plotted in Figure 5.4.

- We generate realisations using the `generate` function from the `fable` package [91]. Here we require values for the deterministic components to generate new data, and for our purposes it is sufficient to use the values that we have fitted the data to. We comment on this in Section 5.5. We are able to generate multiple samples at a given time t, allowing us to use these in our S-OPF.

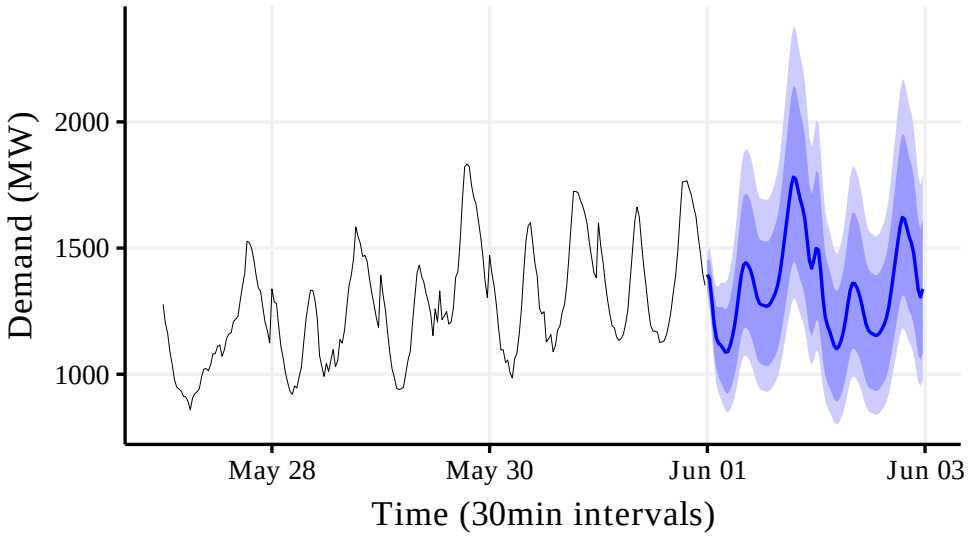


Figure 5.1: Electricity demand is forecast 48 hours into the future using the `forecast` function from `fable` [91]. As in the example presented in [70, §12.1] we use the previous 48 hours temperatures as predictors values. The forecast value is accompanied with 80% and 95% confidence intervals. These intervals gives an indication of the distribution of the demand model and where the samples will likely fall once generated.

5.3.2 Details on modelling electricity demand

Electricity demand is known for having daily, weekly and yearly seasonality as in [63] and [15]. This is evident in Figure 5.2 where we have plotted the South Australian electricity demand for each half hour from June 2017 to May 2018. Demand is on average higher in the cooler months of the year, however on hotter days during summer the electricity demand dramatically increases and there is increased variance.

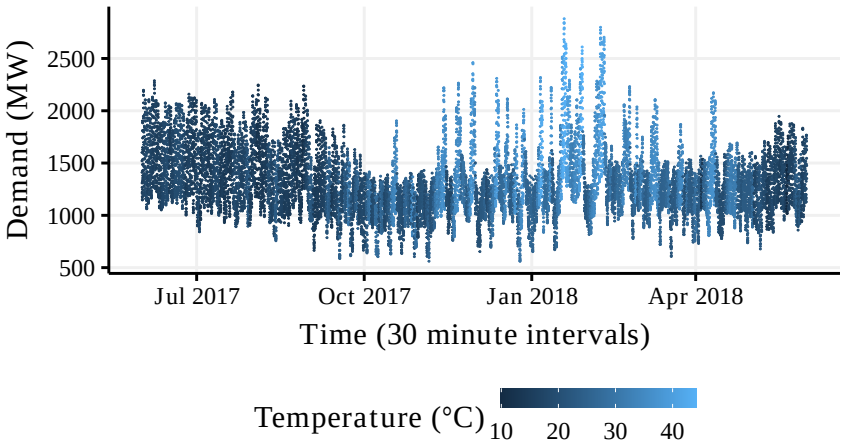


Figure 5.2: Historical South Australian half-hourly electricity demand (MW) for June 2017 to May 2018.

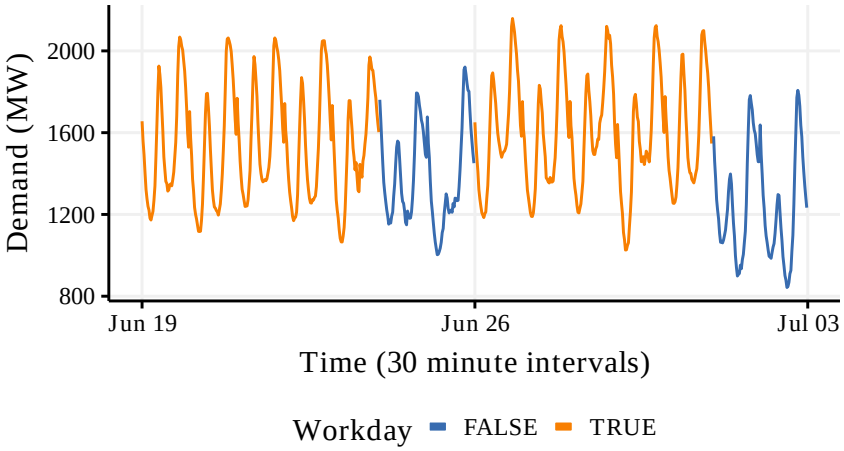


Figure 5.3: Historical electricity demand for South Australia plotted over a two week period in June 2017. This illustrates the weekly seasonal behaviour. On the non-workdays (which are the weekends for this period) have a similar daily behaviour, however the demand is lower than the workdays.

We consider the daily behaviour by plotting the average half-hourly demand over the year as in Figure 5.4. There are strong peaks in demand in the morning when people are waking up and in the evening when people are coming home. There is also a strong jump in demand at midnight which is typical for South Australia, where hot water systems are known to turn on between 11:30pm-12:00am to take advantage of off-peak electricity price

[21]. To model the distribution of electricity demand appropriately over time we need to account for this seasonal behaviour within our model.

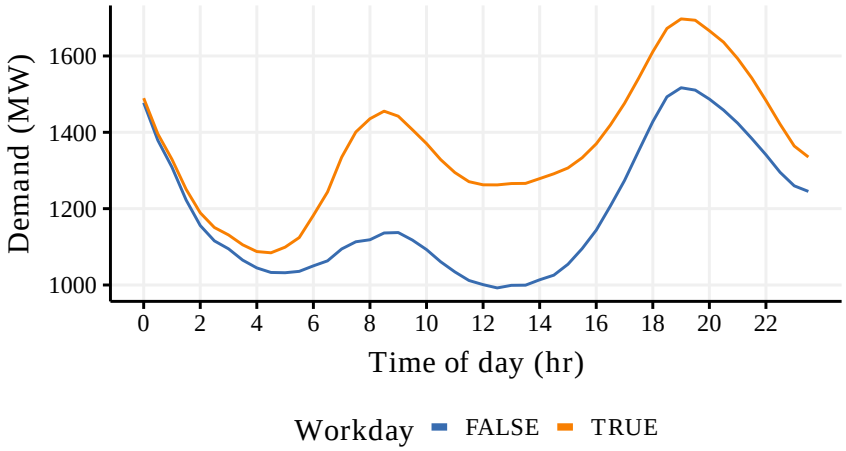


Figure 5.4: Average historical electricity demand for each half-hour in 24hr period for June 2017 to May 2018.

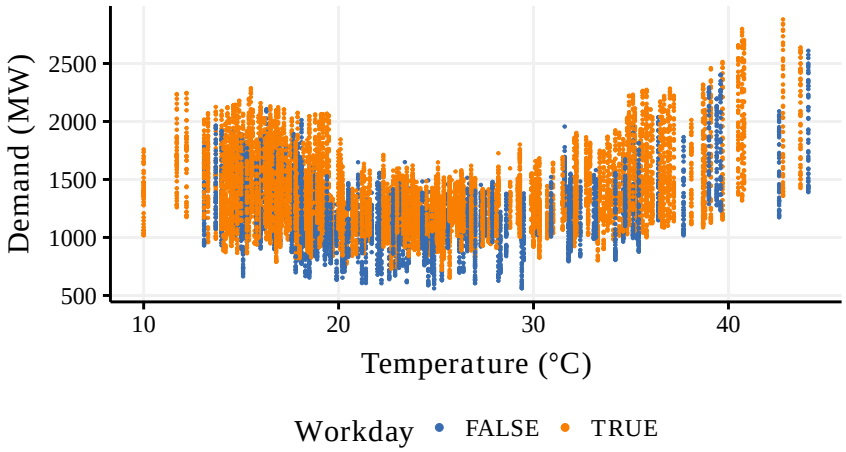


Figure 5.5: Daily maximum temperature in degrees Celsius against historical electricity demand for June 2017 to May 2018.

In Figure 5.5 we can see the non-linear relationship between temperature and demand. This is also noted in [63] and [15]. These papers model temperature by using certain piecewise functions that are activated when the temperature is above or below a critical point. Unlike these papers, we include temperature as a quadratic predictor.

From this we develop the demand model in Section 5.3.3. Note that we considered including an interaction term between Workday and Temperature, however this was not significant. We also considered using "time of day" as a variable, however this created issues with overfitting and we will be able to capture this behaviour with the Fourier daily terms.

The daily and weekly seasonal behaviour for electricity demand is evident in Figure 5.3. The colour identifies the *Workday* (Weekdays that are not public holidays). It appears that workdays follow a similar daily behaviour to non-workdays, however non-workdays appear to have a lower demand on average. The daily seasonality is show in Figure 5.4 where the average electricity demand for each half-hour in 24hr period for June 2017 to May 2018. There are clear peaks in demand at 00:00, 08:30 and 19:00. Within our model of electricity demand we want to capture the strong daily seasonal behaviour.

5.3.3 Demand model

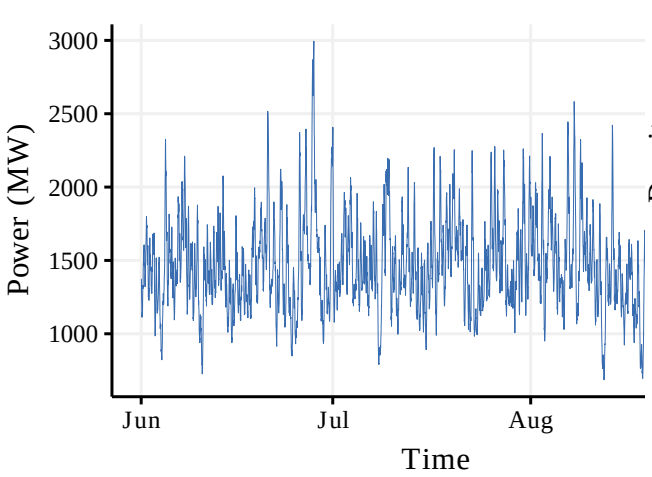
We let D_t be the demand at time t for South Australia. Then we have the model

$$\begin{aligned} D_t = c &+ a_1(\text{Workday True}) + a_2\text{Temperature} + a_3\text{Temperature}^2 + \\ &\sum_{i=4}^{14} a_i^s \sin\left(\frac{2\pi t}{48}\right) + a_i^c \cos\left(\frac{2\pi t}{48}\right) + && \text{(Daily)} \\ &\sum_{i=15}^{19} a_i^s \sin\left(\frac{2\pi t}{336}\right) + a_i^c \cos\left(\frac{2\pi t}{336}\right) + && \text{(Weekly)} \\ &\sum_{i=20}^{22} a_i^s \sin\left(\frac{2\pi t}{17520}\right) + a_i^c \cos\left(\frac{2\pi t}{17520}\right) + && \text{(Yearly)} \\ &\text{ARMA}(p, q). \end{aligned}$$

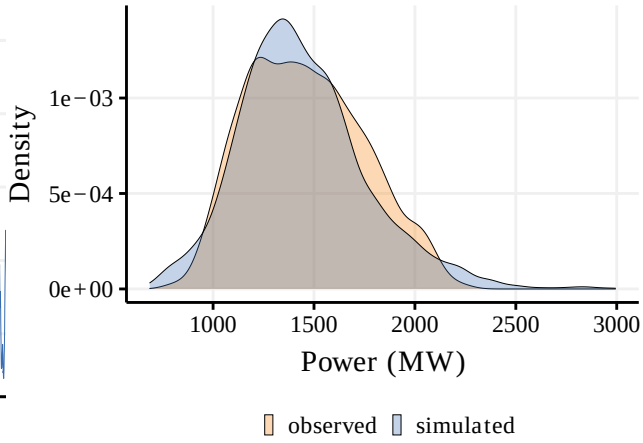
Using the `fable` package for fitting, we see that $p = 2$ and $q = 4$ and no differencing was required. We list the full model with coefficients in Appendix C.1.

5.3.4 Demand simulation

We now generate new data using this demand model. In Figure 5.6a we generate a single sample path over winter. In Figure 5.6b we see that the distributions of this sample path compared to the original observed data are very similar.



(a) Simulated sample path of electricity demand for South Australia. The path is simulated over a three month period.



(b) Comparison of densities for the sample path and real observed data. The simulated data has a similar density with slightly bigger tails.

Figure 5.6: Simulated electricity demand for South Australia.

5.4 Wind

Here we detail our approach for modelling wind power in South Australia using the time series methods from Section 5.1. As highlighted in [54] there are two main approaches for modelling wind power. The first approach relies on using wind speed data and understanding the relationship between wind speed and wind power to transform wind speeds into wind power as in [48]. Alternatively, and in some ways more simply, wind power can be modelled directly using wind power data.

Unlike the distribution of wind speed (which can be modelled by a Weibull distribution, see for example [42], [40]), wind power from a wind farm (or collection of wind farms) does not follow a standard distribution. This is most notable due to the boundedness of wind power within the physical capacity of the wind farm. For each wind farm in South Australia we have historic wind power data from [11] hence the second method is the one we take.

Our approach for modelling the distribution of wind power is guided by the the need for the modelling procedure to be applied to each wind farm in Section 6.1. For each season (summer/winter) and each wind farm we need to generate samples over multiple time steps. Integrating our sampling method whilst accounting for the boundedness of wind power is a key factor for our approach and motivates following the process in [54].

5.4.1 Overview of modelling wind power

At a high level, we model the wind generation for each wind farm separately as a stochastic process by:

- Setting the minimum generation to zero. Note that wind farms can draw small amounts of power giving a negative output, however our optimisation implementation requires that the power output is non-negative. To handle this we set any negative values to zero when cleaning data. We expect this to have very little impact on our modelling as these negative values were at most 0.2% of the positive values in aggregate over the period of interest. We note that some wind farms exhibited times of consistently zero output which may mean they are having repairs undertaken during that period. We have not explicitly adjusted for this, as it is a legitimate part of the process we are modelling.

- A Box-Cox transformation is applied for each wind farm as per Definition 5.5. The parameter λ is chosen separately for each wind farm and is detailed in Section 5.4.3.

- As in Chen et al. [54], we initially assessed differencing through a visual check of the differenced data. However, we ultimately decide on the order of differencing required using the KPSS tests through the automatic ARIMA function, with results in Section 5.4.3. One order of differencing is used in all models.

- Since wind output is highly seasonal we fit separate ARIMA models for summer (1st December to 28th February) and winter (1st June to 31st August). Summer and winter have very different weather patterns in South Australia, and these two seasons are the most interesting from the point of view of optimisation. We expect the most variability within these months, as visible in the Figure 5.7b and Figure 5.7d. We exclude autumn and spring for this reason.

- For each wind farm and for each season (summer/winter) we use the `generate` function from `fable` [91] to draw samples from the distributions of each wind farm. For each time t we implement the Limited-ARIMA (LARIMA) method from Chen et al. [54] where each time sample is limited by upper and lower limits of the wind farm, and the result is used to generate samples for the next time step. The lower limits are set to zero. The upper limit is the registered capacity for generation from that wind farm and is made available in [115] and originally obtained from [11].

- The LARIMA model shifts the sample mean from the theoretical mean of the ARIMA model. To combat this a sample path is drawn from which the mean is compared with that of the real data. The mean of the LARIMA model is then adjusted through trial and error until they are equal to within 0.1MW. This again follows Chen et al. [54].

- Then the mean-adjusted LARIMA model is ready to be used for generating samples for within the S-OPF on South Australia. This is done so using the `generate` function from `fable` [91]. We are able to generate multiple samples at a given time t, allowing us to use these in our S-OPF.

5.4.2 Details on modelling wind power

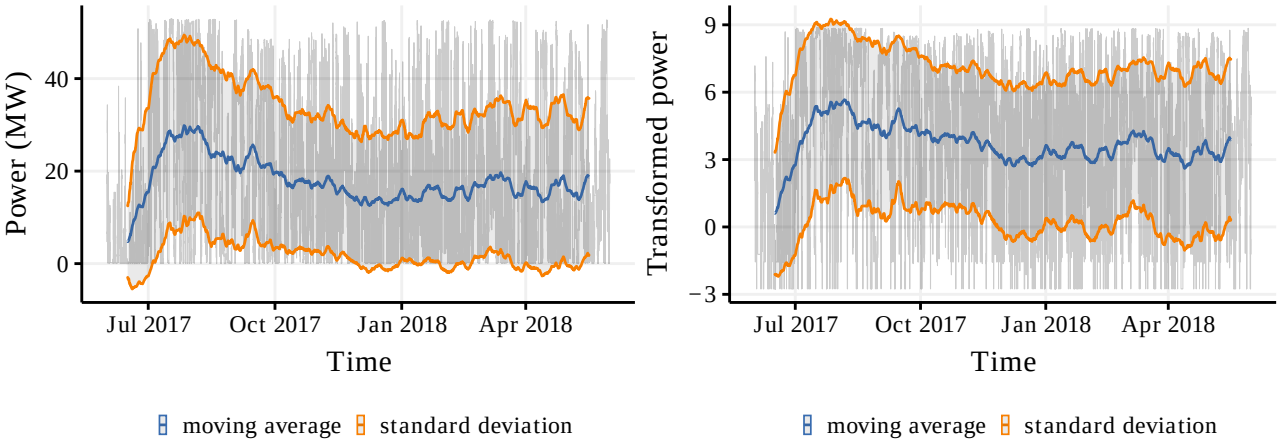
Expanding on the the modelling process described in Section 5.4.1 we describe some details by considering the wind farm with DUID BLUFF1.

Firstly we note that the maximum power generation is 53 MW and as described above we have constrained the minimum output to be zero. To gain a better understanding of the wind power output over time we plot the *moving average* (that is the sample mean) over 30 day rolling periods as in Section 5.1. In this case, we plot the *rectangular* moving average where for each half-hourly time we average over the previous and future fifteen days. Similarly we calculate the rectangular standard deviation and plot the moving average plus and minus the standard deviation (labelling this as the standard deviation).

These quantities give an indication about the behaviour of the expectation for the stochastic processes $\mathbb{E}[X_t]$ and the variance $\text{var}[X_t]$ as time t varies. In Figure 5.7a we see the moving average is not constant, indicating that $\mathbb{E}[X_t]$ is also not constant over time t. We note also that the standard deviation varies by small amounts over time although it is relatively consistent throughout the year. This indicates that the wind data is not strictly stationary nor second order stationary.

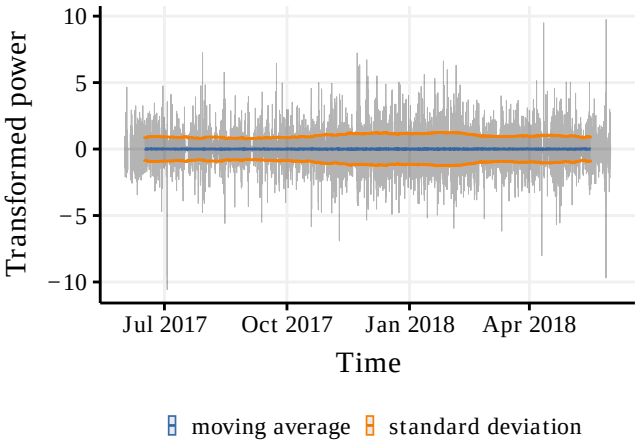
One approach for fitting time series models to bounded data is to first transform the data and then fit the required time series model. A transformation applicable for data that is bounded above and below is the scaled logit transformation [70, §13.3]. However, we found that in doing this the transformed data did not meet the assumptions of stationarity and further a lower bound of zero with a large amount of data from the wind farm also registering a zero value required a complicated logit transformation to handle this.

We find the approach of a Box-Cox transformation and then differencing better meets the assumption of stationarity and easily integrates within the `fable` framework for model fitting and sample generation. Following the procedure of Chen et al. [54] we will apply transforms to the data in order to obtain stationary data. We do this as there are known methods for fitting time series models when the data is strictly stationary or second order stationary. In [54] a number of Box-Cox transformations as in Definition 5.5 with the aim to obtain a transformed data set with stabilised variance. Each transformation is made by choosing a different value for λ. The transformation which minimises the residual sum of squares is chosen. We instead select a a value for λ in the Box-Cox transformation using the guerrero feature as in Definition 5.5 from [92].

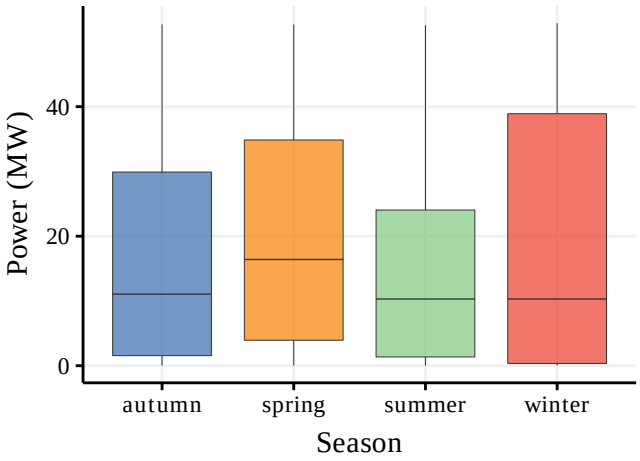


(a) The wind power generated from BLUFF1 from June 2017 to May 2018.

(b) Box-Cox transformed wind generation for BLUFF1 with ($\lambda = 0.36$).



(c) First order differencing applied to the Box-Cox transformed data.



(d) A box-plot demonstrates the seasonal variation in wind generation for BLUFF1.

Figure 5.7: Plots of BLUFF1 wind generation from June 2017 to May 2018.

The moving average mean and standard deviation are shown not to be constant in Figure 5.7b. This motivates composing this transformation with an another transformation in order to obtain stationarity.

Differencing is applied to obtain an approximately stationary time series as in Figure 5.7c. Here, the moving average of the transformed data is approximately constant which gives an indication that $\mathbb{E}[W_t] = \mu$ where μ is a constant. We also see that the standard deviation is relatively constant although varies slightly through out the year.

There also appears to be less variance during the cooler months and more variance over

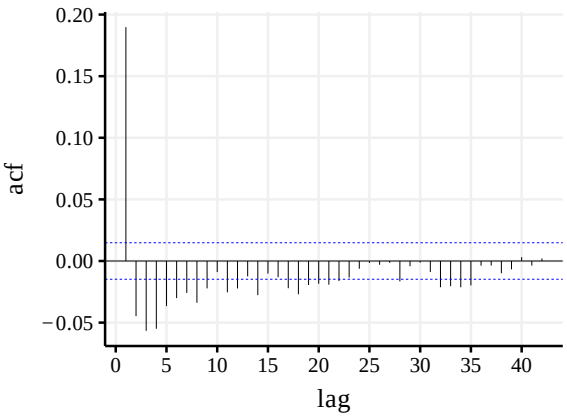
the hotter months, as in Figure 5.7c. The wind conditions exhibit different behaviours over summer and winter as in Figure 5.7d. However, during these separate periods the transformed data appear to be approximately stationary.

To capture this different behaviour in cooler and hotter months we will fit separate ARIMA models for the winter and summer periods only. We do not model the spring and autumn seasons here. The winter period will be between 1st of June 2017 and 31st of August 2017. The summer period will be between 1st of December 2017 and 28th of February 2018. We demonstrate the fitting process for BLUFF1 in winter here.

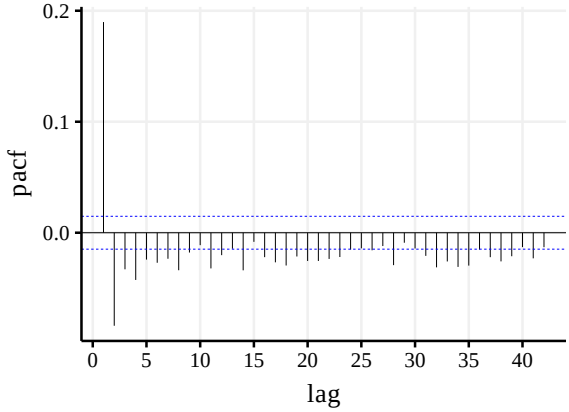
We apply the ARIMA automatic fitting function using the stepwise implementation and the greedy option. This is applied to the Box-Cox transformed BLUFF1 observations selected during the winter period. Note that we want to apply the ARIMA function once the data has also been differenced however the ARIMA function also selects the number of differences required and as expected is automatically selected to be differenced once. The fitted model is an ARIMA$(0, 1, 1)$ given by

$$x_t = 0.2161 W_{t-1} + W_t.$$

Here σ^2 is estimated as 0.716 and the $\text{AIC}_\text{c} = 11055.04$.



(a) ACF for transformed BLUFF1 data.



(b) The PACF for transformed BLUFF1 data.

Figure 5.8: There is a very significant correlation at lag 1 and then correlation falls for larger lags. Although lags above one are still significant there is some similarities with the ACF of an MA(1) process. The PACF for transformed BLUFF1 winter wind data appears to gradually taper off. This is in agreement with a PACF of an MA(1) process.

Using this fitted ARIMA model we produce a sample path for approximately three months to give an indication of the models ability to simulate wind power.

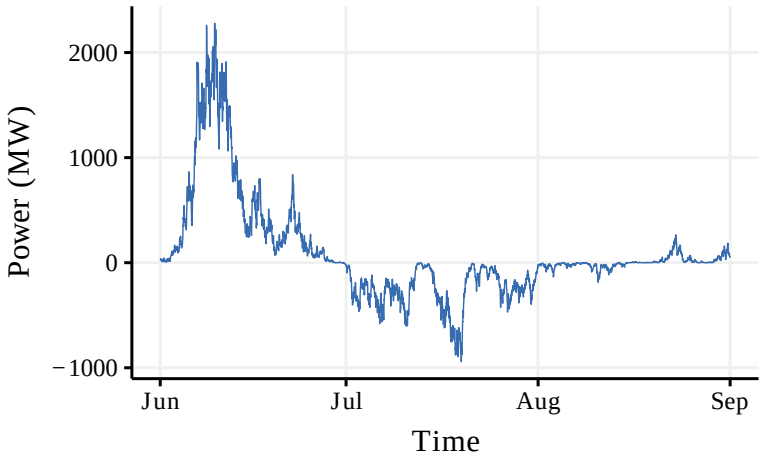


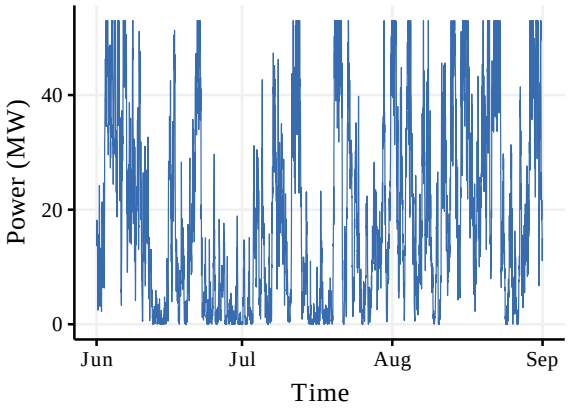
Figure 5.9: Simulated wind generation for BLUFF1 using an ARIMA(0,1,1).

This model alone does not cater for the boundedness of the wind power generation. We see in Figure 5.9 how the simulated power goes well beyond of the bounds 0 MW and 53 MW for this generator. This is where we need to adjust our ARIMA model. However unlike in [54] we do not consider the ARIMA model with limiter, which adjusts sample paths after generation, as this was not shown to be as accurate and as we do not need to simulate a long time into the future. For our purposes, we require forecasting the distribution one time-step ahead and generating several scenarios at each time point, hence we choose to use the Limited-ARIMA (LARIMA), also considered in [54].

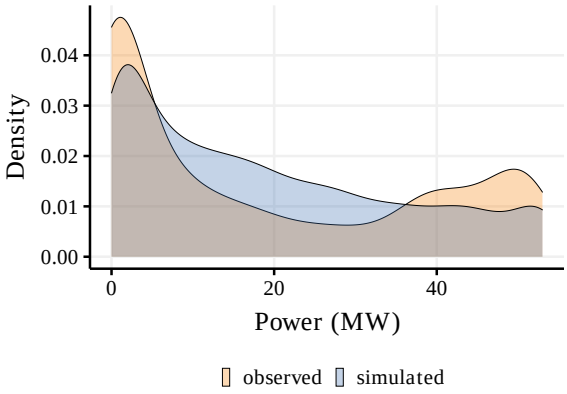
The LARIMA is a process that uses an ARIMA model when the output has physical bounds, as is the case for wind generation. When simulating the wind data for a time-step ahead samples are drawn using the fitted ARIMA(p, d, q) model, and then the samples are *limited* to be within the physical bounds.

To use the LARIMA model to simulate data over time, whenever a sample is taken and it is outside of the wind farms capacity the sample is replaced with the nearest limit. Using this model we simulate a sample path for the BLUFF1 wind farm. After simulating the sample path, we check the mean to determine how close the model is to the true mean. We note that we expect there to be some different due to the limits. We then adjust the model by including a small constant term to each estimate. This constant has been chosen for each model so that the mean for the sample path is within one decimal place of the mean from the data. We plot the result in Figure 5.10a.

In Figure 5.10b we plot the density of the LARIMA model against the observed data and see a reasonable fit, although the observed data has slightly heavier tails.



(a) A sample path for BLUFF1 using the LARIMA model with adjusted mean. The simulated wind power is now within the physical constraints of the wind farm.



(b) Comparison of densities for the sample path and real observed data. The simulated data has a similar to the observed data, although the observed data has heavier tails.

We would not expect the densities for the observed data and the simulated data to be identical, although we expect them to have similar characteristics. The simulated data is just one sample path of possible paths. Further the observed data has been taken from one year. Ideally, this modelling approach could be expanded and applied to multiple years of training data and compared with multiple sample paths. However for our purposes the models will be adequate to draw samples and form representative scenarios of wind generation.

5.4.3 Wind model

Table 5.1: Winter ARIMA(p, d, q)

DUID	Capacity (MW)	λ	p	d	q	c	AIC_c
BLUFF1	53.0	0.36	0	1	1	0.015	11 055.0
HDWF1	102.0	0.44	1	1	1	0.079	12 145.2
HDWF2	102.0	0.40	2	1	0	0.041	12 344.7
HALLWF1	95.0	0.55	4	1	2	0.020	14 585.0
WATERLWF	130.0	0.38	1	1	3	0.066	14 115.5
CLEMGPWF	57.0	0.49	0	1	1	−0.036	13 184.5
NBHWF1	132.0	0.52	4	1	2	0.061	16 033.8
HALLWF2	71.0	0.55	3	1	2	0.053	13 815.0
SNOWSTH1	126.0	0.33	0	1	1	0.064	11 053.9
SNOWNTH1	144.0	0.32	0	1	1	0.007	10 939.2
LKBONNY3	39.0	0.40	3	1	3	−0.008	10 500.7
LKBONNY2	159.0	0.39	2	1	2	−0.058	14 012.1
SNOWTWN1	99.0	0.52	2	1	3	0.031	13 559.4

Table 5.2: Summer ARIMA(p, d, q)

DUID	Capacity (MW)	λ	p	d	q	c	AIC_c
BLUFF1	53.0	0.36	0	1	1	0.360	13 193.2
HDWF1	102.0	0.44	1	1	1	1.060	14 556.2
HDWF2	102.0	0.40	2	1	0	0.65	13 905.1
HALLWF1	95.0	0.55	4	1	2	−0.043	17 103.1
WATERLWF	130.0	0.38	1	1	3	−0.097	14 018.1
CLEMGPWF	57.0	0.49	0	1	1	0.726	16 092.4
NBHWF1	132.0	0.52	4	1	2	0.907	18 420.1
HALLWF2	71.0	0.55	3	1	2	0.382	15 571.2
SNOWSTH1	126.0	0.33	0	1	1	0.108	12 172.0
SNOWNTH1	144.0	0.32	0	1	1	0.724	12 049.1
LKBONNY3	39.0	0.40	3	1	3	0.164	10 655.1
LKBONNY2	159.0	0.39	2	1	2	0.525	15 076.3
SNOWTWN1	99.0	0.52	2	1	3	0.066	16 157.2

5.5 Conclusion

In this chapter we have used real data to model the distributions for random variables overtime. These random variables are included as constraints within the S-OPF. Our motivation here was ultimately to model the distributions of wind and demand such that samples can be drawn and used within the S-OPF.

Our aim was not to "perfectly" model these variables but rather to produce models that will be representative and facilitate the S-OPF implementation on the South Australian Network. We used a combinations of transformations, decompositions and classic time series methods. The methods used here could be further improved by including cross validation and prediction to guide model selection, which we leave for future work.

These models will now be used to draw samples to form scenarios when implementing S-OPF on the South Australian network in Chapter 6 and Section 6.4.

Chapter 6

Stochastic-optimal power flow for South Australia

This chapter builds on the work of all previous chapters and contains two main parts. In the first part, covered in Sections 6.1 to 6.3, we extend the test cases of Chapter 4 by constructing a model for the South Australian electricity system based on the 2017 open NEM model produced by Xenophon and Hill [115]. We then use this model to run S-OPF on the South Australian network. Initially we run a deterministic OPF where only one scenario is drawn for each random variable. Then a full S-OPF is run using scenarios generated for electricity demand and wind power capacity by the time series models in Chapter 5.

In the second part, Section 6.4, we use S-OPF to understand where storage is optimally located within the network as South Australia transitions to a 100% renewable power system. We begin with a copperplate analysis where we quantify wind and storage requirements in the absence of network constraints. The copperplate analysis is used as a guide for how to proceed using our S-OPF. Using this analysis, we add generic storage units to all buses in the network and scale wind capacity by whole factors until the S-OPF power output is approximately 100% renewable. We proceed to use S-OPF to identify the most used storage unit locations. We successfully apply S-OPF to the South Australian network and identify where storage is most well utilised at buses with high power demand.

6.1 South Australian network model

To solve S-OPF on the South Australian power system we require a model of the network. There have been many studies on storage and power flow in the NEM, including studies on South Australia such as [49, 104, 111, 60, 43]. However, these studies either do not explicitly model the network topology or do not provide a network model in an open source nature.

Much of the data required to create a model of the NEM is publicly available. Xenophon and Hill [115] use this to construct the open NEM model. The open NEM model is publicly available at [114] and purpose built for power flow studies, so is well suited for our study.

We use the open NEM model to obtain a model of the South Australian transmission network. The open NEM model is formed to model the NEM as it was in 2017. Although the NEM has evolved since this time, this will form a strong basis for the characteristic NEM in South Australia and will be suitable for the purpose of this study, as the core assumptions, such as geographic constraints and demographics, change slowly.

Such a model compiles the information required to define and solve the S-OPF as presented in Chapter 4. This includes gathering relevant parameters for the network's buses, lines, generators and demand. Note that, as in Chapter 4, we do not model power flow through transformers in the South Australian network.

In the following sections we provide an overview for how this information is obtained in the open NEM model and, further, how we will extract the required components for the S-OPF on the South Australian network. Further details of the South Australian network components are provided in Appendix D.

6.1.1 Relevant components of the open NEM model

In this section we describe how components of the open NEM relevant to the South Australian model were obtained. The assumptions made in forming the open NEM model are important to understand as they will apply to S-OPF simulation. We recall the main relevant details here and we refer the reader to [115] for a full details of the open NEM model.

Lines and Buses

The topology of the network is constructed using Geoscience Australia data for power stations, substations and transmission lines obtained from [95, 94, 93]. A path of coordinates is provided for each transmission line. The coordinates for the endpoints of each transmission line are used to define network nodes/buses. The first and last coordinates are used to define *from* and *to* buses for each line.

Power stations and substations (whose coordinates do not necessarily coincide with buses) are assigned to the bus with the smallest haversine distance (the shortest distance between two points when measured on the surface of a sphere). The network can now be defined as a graph with nodes being the buses and edges the transmission lines. The path of coordinates to define each transmission line is used to calculate an approximate distance for each transmission line and this is later used to assign parameters for each transmission line such as the reactance.

Limited data on electrical parameters is available. The open NEM model calculates electrical parameters for each line from first principles by assuming certain physical characteristics. The main parameter of interest to run S-OPF on South Australia and one we explicitly use is the reactance for each transmission line as detailed in [115, p. 6]. Recall the reactance is used to calculate the PTDF matrix as in Section 3.2.2 and used to form constraints for S-OPF and defined in Definition 4.1.

Generators

As we discussed in the construction of buses, power stations (and substations) are assigned to their closest bus. However AEMO provides historic power output for generators identified with Dispatchable Unit Identifiers (DUIDs) [11] and not power stations. This is the data we used in Section 5.1 for fitting time series models to South Australian wind farms. Technical parameters for each DUID (such as registered capacities) are also available from [11]. Hence DUIDs are used to define generators in the open NEM model.

Further information such as fuel costs for generators is provided in the National Transmission Network Development Plan (NTNDP) [12]. The open NEM model links these data sets to provide generators by their DUID with a range of economic and technical operating parameters relevant for power flow studies. This includes the heat rate, fuel cost and VOM cost (variable operations and maintenance cost), from which the marginal cost is derived. Finally, the open NEM model creates a cross-reference table to assign DUID's to power stations. Hence all DUIDs with appropriate generation information are now assigned to buses.

We use the *short run marginal cost* provided for each generator within the South Australian network for the costs in our model as defined in [115]. This marginal cost has been calculated for each DUID using the following formula

$$\text{Marginal Cost}\left(\frac{\$}{\text{MWh}}\right) = \text{Heat Rate}\left(\frac{\text{GJ}}{\text{MWh}}\right) \times \text{Fuel Cost}\left(\frac{\$}{\text{GJ}}\right) + \text{VOM Cost}\left(\frac{\$}{\text{MWh}}\right).$$

Demand

Historic demand data is made available by AEMO for each NEM region at half-hourly intervals [11]. This demand data is known as the *operational demand* and includes both industry and consumer demand, however does not account for small scale generation and other small local effects. The full definition is available at [14]. We used this data in Chapter 5 to model electricity demand for South Australia.

Demand is proportionally allocated to each bus based on population density in the region of the bus. The population data is obtained from the Australian Bureau of Statistics

(ABS) where the population is aggregated by Statistical Level 2 areas [4] and proportionally assigned to buses. This is not perfect as it does not distribute the demand according to other factors that contribute to it, such as industrial load, as the data for this was not available. However this will be suitable for applying our methods. For further details on the open NEM model see [115] and [114].

6.2 Constructing the South Australian network model

We now describe how we obtain the South Australian portion of the open NEM model that is applicable for forming S-OPF as in the formulation described in Chapter 4. The open NEM model provides almost all of the information required to form S-OPF. However, we build on the open NEM formulation to add power flow capacity limits for all lines in South Australia and adjust the minimum generation limits to better capture true system behaviour in the S-OPF.

We begin by downloading the Github repository from [114]. This repository contains data and code scripts written in Python language required to build the open NEM model. We run the separate scripts for constructing the topology of the network and for generators in the network. The script for generating the topology (buses and lines) of the NEM is contained within the `1_network` directory. Each script outputs a `pickle` file for nodes and the edges. (Alternatively the script also outputs csv files.) These files contain variables associated with each node and transmission line in the NEM. Some of the variables that we use are documented below in Table 6.1 and Table 6.2.

Table 6.1: Bus Data

Variable	Description	Units	Format
NODE_ID	Node identification number.	–	int
STATE_NAME	The Australian state the bus is located within.	–	str
NEM_REGION	The NEM region the bus is located within.	–	str
VOLTAGE_KV	The bus voltage.	kV	float
PROP_REG_D	The proportion of regional demand allocated to the bus.	–	float

Table 6.2: Line Data

Variable	Description	Units	Format
LINE_ID	Transmission line Identifier.	–	str
NAME	Name of the transmission line.	–	str
FROM_NODE	From node identification number.	–	int
TO_NODE	To node identification number.	–	int
X_PU	The per-unit line reactance.	p.u.	float
LENGTH_KM	Length of line.	km	float
VOLTAGE_KV	The line voltage.	kV	float

Using these files for the buses and lines we filter for the South Australian network components. We begin with the buses and filter the buses such that STATE_NAME is equal to South Australia. Next we filter for the transmission lines in South Australia by only considering lines where both the FROM_NODE and TO_NODE and to both have STATE equal to South Australia.

Generators

The generators are defined in generators.csv where they are identified with a DUID. The variables included in this data that we use are described in Table 6.3. We filter for the South Australian generators by requiring NEM_REGION being equal to SA1, which is the NEM region associated with South Australia. Note that each generator is assigned the nearest nodes using the haversine metric, the associated node is indicated with the variable NODE. We now have all generators that are located in South Australia and the node they are located at.

Table 6.3: Generators Data

Variable	Description	Units	Format
DUID	Dispatchable Unit Identifier.	–	str
STATIONID	Station identifier	–	str
NEM_REGION	The NEM region the buses is located within.	–	str
NODE	Node that generator is located at.	–	int
FUEL_TYPE	Primary fuel type.	–	str
FUEL_CAT	Primary fuel category.	–	str
MIN_GEN	Minimum dispatchable output.	MW	float
REG_CAP	Registered capacity.	MW	float
SRMC_2016-17	Short-run marginal cost for the year 2016–17.	$/MWh	float

Within the S-OPF implemented in Chapter 4 we assumed generators could be dispatched within limits, where there lower limit for all generators was assumed to be zero power. With some generators such as batteries this works well since they can easily be turned on or off. However some traditional fossil fuel generators, cannot be powered up and down as quickly as others. They may not be able to power up to full their capacity within the 30 minute dispatch, this is a limitation of the S-OPF in its current form.

There are four different types of fossil fuel generators in the model for South Australia and they have varying rates at which they can be turned on and off. The type of fossil fuel generators include:

- compression reciprocating engines (CRE)
- open cycle gas turbines (OCGT),
- combined cycle gas turbines (CCGT), and
- steam sub-criticals (SSC) generators.

We obtain the technology type for each DUID in South Australia from [18]. As indicated in [103], both OCGT and CRE generators can start up and shut down relatively quickly, while CCGT and SSC require much more time to start up and shut down. Further to this, there are increased costs associated with shutting down and starting up a power station. We do not seek to perfectly model this behaviour, as this would likely include optimising over a larger time horizon. However, we do want to avoid strange behaviour such as a power station turning off and on regularly when it would normally take a long time to do this. For CCGT and SSC generators we will have positive minimum generation limits to prevent such irregularity, meaning that these plants will always be running at a base load level. We use the stable minimum generation from [59, 103] as an indication for the minimum generation limit. We set the minimum output for all CCGT generators to 0.45 of the registered capacity and the minimum output for all SSC generators to 0.20 of the registered capacity.

Interconnectors

The interconnectors (lines that join NEM regions) are handled separately to the high voltage transmission lines in the network. South Australia has two interconnectors: the Murray Link, which is a high voltage direct current (HVDC) line and the Heywood interconnector an AC line. Both connect South Australia to Victoria [16, 61, 6]. The `FROM_NODE` and `TO_NODE` are contained within the output file `network_ac_interconnector_links.csv` and `network_hvdc_links.csv` for the Heywood interconnector and Murray Link respectively. Forward and reverse line power flow limits are contained within `network_ac_interconnector_flow_limits.csv` for the Heywood interconnector and in `network_hvdc`

_links.csv for the Murray link. We define the generators that correspond to the interconnectors at the appropriate South Australian buses for importing and exporting following the method in Chapter 4. For each generator the interconnector power flow limits are used as the generation limits.

We have now filtered for the South Australian network elements to determine the buses, and lines and their respective connections. We also have latitude and longitude location coordinates for the buses and a path of coordinates for the lines. In Figure 6.1 we have plotted the network using this information.

Storage in South Australia

The model for the South Australia is based on the system in 2017. However we are interested in storage requirements within the transitioning network. We add the storage units in South Australia that are currently operating as at May 2022.

Table 6.4: Storage units in the South Australian transmission network

name	max (MW)	capacity ($/MWh)	reference
Hornsdale Power Reserve	150.0	193.5	[68]
Dalrymple BESS	30.0	8.0	[58]
Lake Bonney BESS1	25.0	52.0	[72]

Line power flow limits

Although line limits for interconnectors are provided within the open NEM model, power flow limits for the transmission lines are not provided. As we saw in Section 4.6.4 line limits can have a significant impact on the solution of the S-OPF. Accounting for line capacity may make a difference to the optimal storage location. We want to apply S-OPF with the same formulation as in Section 4.1 to the South Australian network. Following this formulation we require the active power flow limits for each line in the network.

The energy market operator provides transmission equipment ratings (TER) including line power flow limits at [13]. Recall our S-OPF formulation only considers active power flow limits directly. However the equipment rating provided at [13] are for apparent power flow with unit (MVA). As an approximation we take the apparent power line limits to be the active power flow line limits. The line ratings are classified as either *dynamic* or *static*, where the line limits for lines with dynamic ratings vary with season (or temperature). If a dynamic rating is available for a transmission line we use this and take the rating that corresponds with the seasons (winter or summer) as defined in Chapter 5.

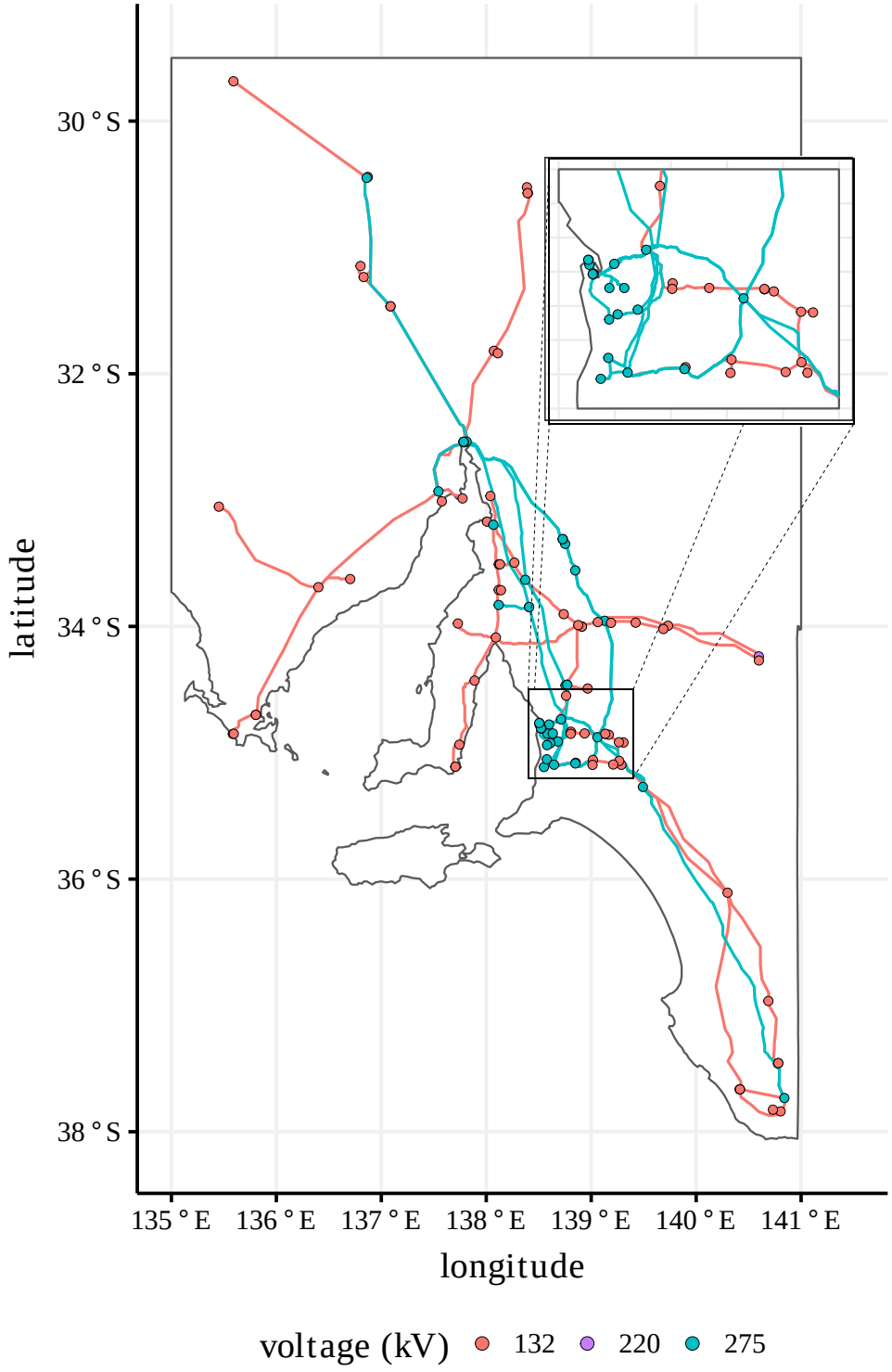


Figure 6.1: Graph of the South Australian transmission network. The network coordinates and the South Australian boundaries are obtain from the open NEM model [114].

To use these line limits we need to assign a rating for each line in the open NEM model. Unfortunately the `EQUIPMENT_ID` provided are not the same identification given in the open NEM model under `LINE_ID` or `NAME`. We manually assign equipment ID's to line names by comparing names and locations with the aid of AEMO diagram provided in [9]. For any line where an equipment ID was not easily determined, we assigned power flow limit of the nearest line that has same voltage.

We now have all of the parameters required to form S-OPF as in Chapter 4. This includes reactance for lines, network topology, line limits and generation limits. Hence we can now calculate the PTDF matrix and form constraints for S-OPF. In Appendix D we document all parameters for generators, lines and buses in the South Australian network.

6.3 Implementation

Here we implement S-OPF on the South Australian network and demonstrate expected behaviour within the results. This implementation is a direct extension of Section 4.6.6. We model all fossil fuel generators as first stage variables, and all other generators as second stage variables (including storage, wind generation and interconnectors) similar to Section 4.1.1. Note that unlike Section 4.6.2 we have not used the diesel generators as second stage generators since as it is not assumed that they are backup generators. In this implementation all fossil fuel powered generation is a first stage variable as it is assumed that dispatch needs to be allocated prior to knowledge of wind capacity and demand.

The marginal cost for generators is obtained from the open NEM model and documented in Table D.4. We follow the same methodology for the marginal cost applied to storage and interconnection as in Sections 4.2 and 4.3 and are displayed in Table 6.5.

Table 6.5: Marginal cost for interconnectors and storage units.

name	marginal cost ($/MWh)
Storage units	5.0
Interconnector export	4.0
Interconnector import	6000.0

Recall that both storage charging and exporting via an interconnector give negative power outputs that reduces cost. Charging to storage units is incentivised over exporting power via an interconnector since it is more cost effective. In our model, importing power via an interconnector is the most expensive form of power supply and hence modelled as a last resort. To be conservative, it is given a marginal cost of $6,000/MWh that generally exceeds the Victorian wholesale electricity price [24]. Note for quantities reported in p.u. and p.u. hours multiplying by 100 gives their respective MW and MWh values.

6.3.1 S-OPF deterministic case

As an initial benchmark, we run the simplest case of S-OPF where there is only one scenario for random variables at each time step. However, instead of sampling from the fitted models constructed in Chapter 5 in this case we use the historic data as the scenarios. That is, we use the historic demand and wind output for the random variables total demand and maximum wind capacity respectively. We run this OPF for one month from the 19th of June 2017 to the 19th of July 2017. Note in this case we do not include the additional batteries in the network indicated in Table 6.4. The output for each technology type is plotted in Figure 6.2.

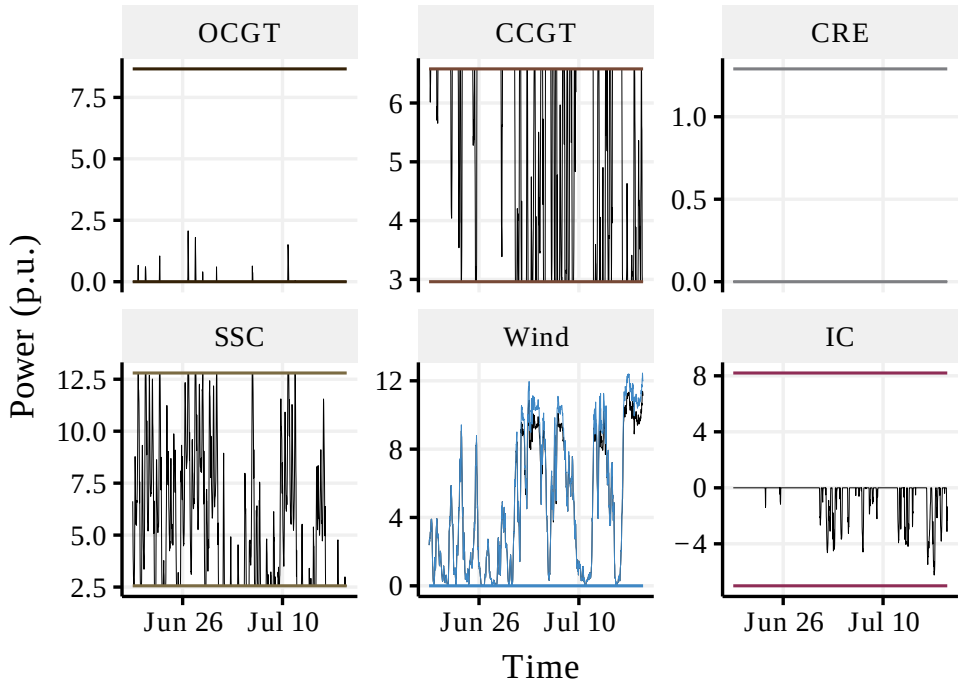


Figure 6.2: Power Dispatch for deterministic-OPF on the SA network. The power dispatch is grouped by the technology type. The upper and lower bound for each technology type is coloured by technology type. Almost always wind power capacity is fully utilised. When the wind capacity is not fully utilised this likely is due to the limited line capacity and the affect of choosing first stage and second stage generators. We see that the majority of power output from fossil fuel generators is provided by the two cheaper technology types CCGT and SSC. There is no power generated from the most expensive sources CRE and IC.

The S-OPF is performing as expected on the South Australian network. Although

the S-OPF output is not the same as historical power dispatch, it is producing results we would expect given the model parameters. Take for example the interconnectors represented by IC in Figure 6.2 and Figure 6.3. Typically the South Australia regularly relies on interconnection to maintain suffcient power supply, however the marginal cost we have applied interconnection over this time period does not allow for this behaviour.

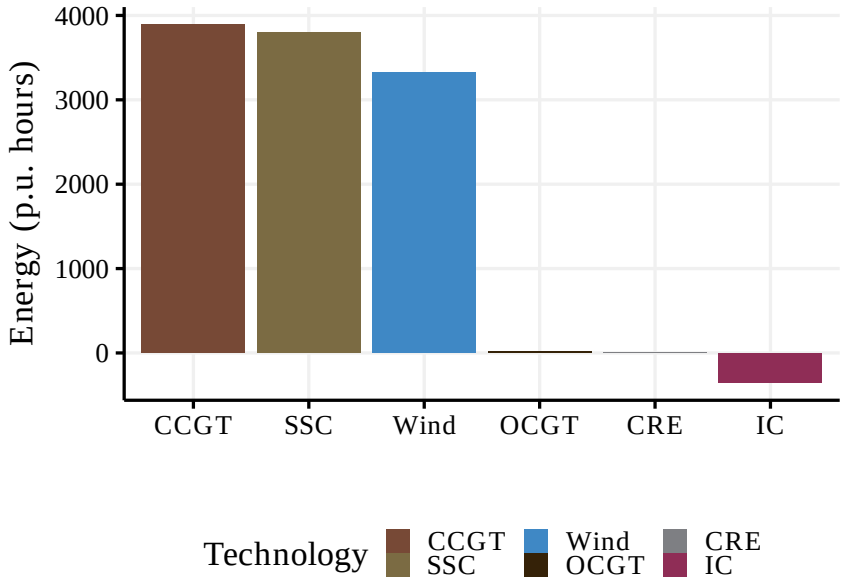


Figure 6.3: This plot describes the power generation mix. The highest and second highest portion of generation is provided by SSC and CCGT respectively. Out of of the fossil fuel power generators these have the lowest and second lowest marginal cost respectively. In comparison with the other two fossil fuel powered generators OCGT and CRE these have second highest and highest marginal cost respectively. Given the high marginal cost of these generators it makes sense that their power dispatch is much smaller or zero in the case of CRE. However these generators are important as they have the ability to power up and down more quickly. Finally we see when extra wind capacity is available but not immediately required, this is used to charge up the batteries. Further when batteries cannot be further charged power is exported through inter-connectors.

6.3.2 S-OPF with simulated samples

We now use the simulated scenarios generated using the methods described in Chapter 5 to run a full S-OPF on the South Australian network. The S-OPF is run for one cases that includes new battery storage in Table 6.4 and another case that does not include this storage for comparison. In Figure 6.5 we see when power demand is exceedingly high,

there is an increased reliance on fossil fuels. The ability for battery storage to satisfy this demand greatly reduces the reliance on fossil fuels and will be important to consider in the transitions to renewables for the South Australian network.

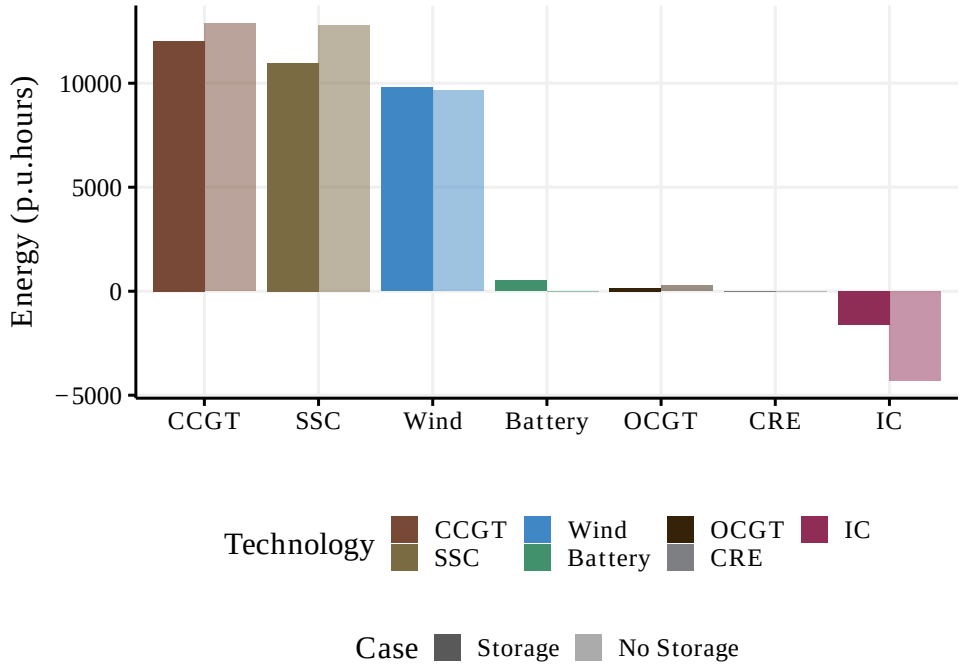


Figure 6.4: The total power output mix is significantly different for the cases with and without storage. As we expect, no storage results in significantly more fossil fuel generation from CCGT and SSC. However, there is also much more power exported on interconnectors. This is driven by the first stage decisions needing to be made prior to knowledge of demand and wind capacity. When there is no storage, no back up power supply other than the interconnectors can be used to meet demand other than wind, and these interconnectors are expensive compared to fossil fuels. This means the S-OPF prefers to over commit to fossil fuel generation in the first stage, rather than relying on importing from the expensive interconnectors. Any surplus power is then exported on the interconnectors to further reduce cost, rather than being used to charge batteries.

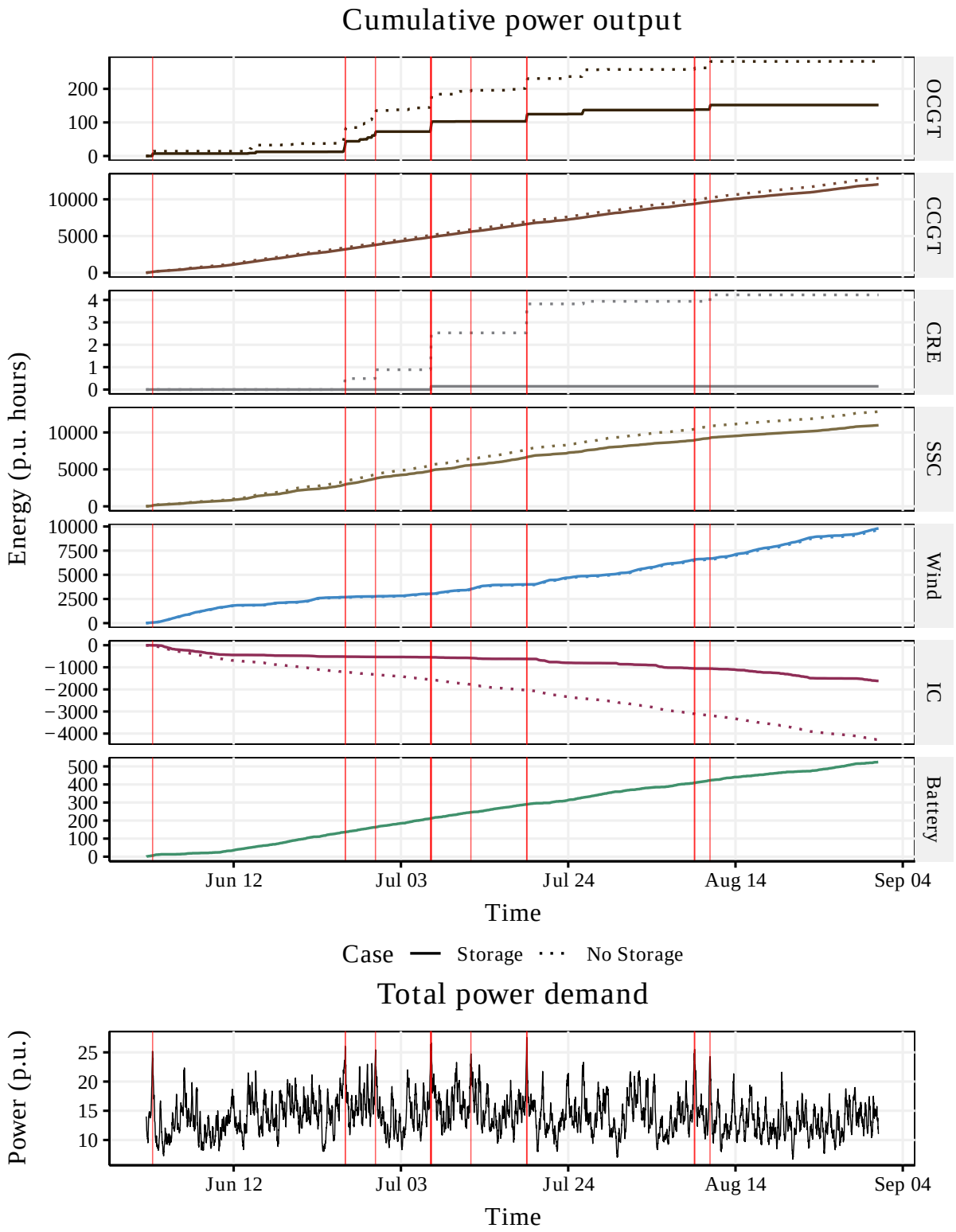


Figure 6.5: Cumulative generation by technology type; with and without storage. We use red vertical lines to indicate when demand exceeds 24 p.u. This corresponds with times that OCGT and CRE generators are relied on to satisfy demand. This indicates how excessive demand in the absence of storage increases reliance on expensive fossil fuel generation.

6.3.3 Comparison of price

While we do not expect the power dispatch from our S-OPF to be the same as the actual dispatch we do expect it to be similar and for the system to be operating as expected. In reality the power dispatch in not just optimised for the lowest cost, rather there is a bidding system where generators provide bids to provide power at a specified price.

- The price for electricity is determined by the marginal generator. Before each time step, generators bid their price for generation for that time step. The bids are ranked lowest to highest, and are selected for dispatch in that order until the demand is met. The price paid for the electricity is the highest bid of the generators selected for dispatch. This may exceed the bids of all other generators selected. Therefore this price generally overstates the actual costs of generation. So we expect our model to determine costs that are lower than the historic price.

- While there is competition in the market, there is not necessarily *perfect competition* [2] and it is known that the NEM experiences varying levels of generation competition [25]. Given that generators must return a profit on their marginal cost, we expect the true marginal cost of production to be less than the price bids, and hence less than the historic price. As our model considers price based on the marginal cost, we again expect this to be less than the historic price.

- The market price is expressed in dollars per MWh. To make a comparison with our cost we calculate the average cost in the same units. The deterministic OPF average price from June 1 - August 31 is \$50.0 per MWh. In the same quarter in 2017 the most common price bands for South Australian electricity were between \$50 and \$200 per MWh [23]. So our OPF price is slightly lower than but at the same order of magnitude as the historic price, which adds validity to our results and matches with the reasoning above. In Figure 6.6 we see that the average price of the S-OPF is comparable, with the No Storage case having an average price of \$57.2 per MWh, while the Storage case has a slightly lower average price of \$50.0 per MWh. Importantly we see less price spikes from the Storage case, as storage is able to be used instead of using expensive fossil fuel generation or importing using the very expensive interconnectors.

This concludes our initial validation and implementation of S-OPF on the South Australian network model. The results, including the average price, will be used as a base line in the next section where we analyse storage in the context of South Australia transitioning to a 100% renewable power system.

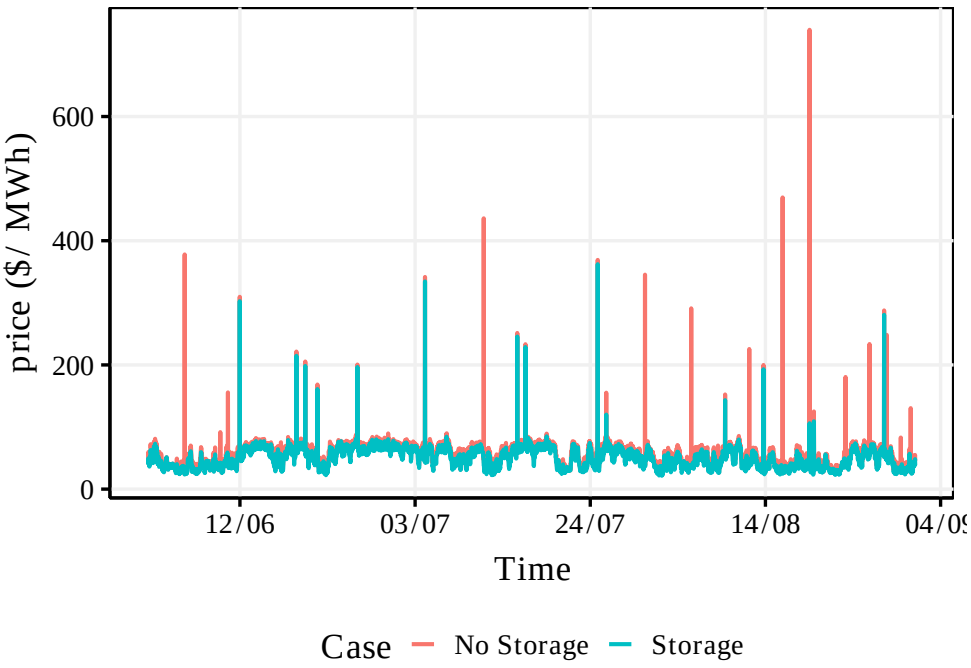


Figure 6.6: Simulated price from S-OPF output for winter. Again we compare the case of with and without storage. The No Storage case has an average price of $57.2 per MWh, while the Storage case has a slightly lower average price of $50.0 per MWh and less price spikes.

6.4 Optimising storage in South Australia

In Section 6.3 we considered S-OPF on South Australia including storage units that are currently in the South Australian Network as of May 2022. We compared how this storage changes the solution of the S-OPF in both winter and summer and saw that storage reduced the cost of power and reliance on fossil fuel generators.

As South Australia transitions to a net 100% renewable power system it is vital to ensure there is sufficient generation to meet demand at all times. As previously discussed, a cost efficient method for transitioning to 100% renewable grid is to store surplus electricity as energy (when wind and solar plentiful) and then use this stored energy to produce electricity when wind and solar cannot meet demand.

Not all generation needs to be provided from wind and solar sources. South Australia can import and export power to Victoria via interconnectors. Recall from Section 4.4 we define a 100% renewable network as any network that uses only renewable generations with storage and exports more electricity than it imports via interconnectors. In this section

we are interested in understanding which locations in the South Australian transmission system are optimal for storage to be installed. We use S-OPF as a tool for understanding where storage is optimally placed within the network and how much energy storage is required.

Recall the model for South Australian electricity is based 2017 open NEM model. At this time, large scale solar-PV was not connected to the South Australian grid and we do not model solar generation in this thesis. The random variables we have considered in the S-OPF are maximum wind capacity and total power demand as modelled in Chapter 5. We do not seek to model solar generation as a random variable in this thesis. However we acknowledge that similar time series models to those used in Chapter 5 could be incorporated in an equivalent way.

6.4.1 Copperplate analysis

In this section we run a copperplate analysis to give us an indication of how to consider transitioning to a 100% renewable electricity network with the South Australia using S-OPF. We seek to determine how much additional wind capacity and how many additional storage units may be required.

For this analysis, we assume that all wind capacity available is used to meet demand regardless of line limits. We assume all storage can be charged or discharged up to its available capacity, however we do include the power flow limits on the storage models. We assume all additional theoretical storage units have the same power flow limits and capacity limits as the Hornsdale Power Reserve, as in Table 6.4. We assume the charge on all storage units is zero for the first time step. We proceed as follows.

1. Take the simulated wind and demand data for the winter season as generated in Chapter 5

2. Ignore all line limits. Aggregate the wind capacity for each wind farm to a total at each time step. Aggregate the demand each bus to a total at each time step. Aggregate storage capacities and power flow limits for storage at each time step.

3. For a given number of additional theoretical storage units x, add their capacities and power flow limits to the existing storage totals. For a given wind multiple, y, scale the wind capacity by this amount. Run the following:

   (a) For each time step, determine the difference d between wind capacity and demand.

   (b) If this d is positive, then there is excess wind available. If there is storage capacity available, add this to the storage capacity (i.e. charge the storage) up to the power flow limits of the storage.

(c) If this d is negative, then there is not enough wind available to meet demand. In this case, use storage where available to meet this demand up to power flow limits, and correspondingly reduce the storage capacity available (i.e. reduce the charge).

(d) Record any amounts leftover, noting that negative amounts show there is a *shortfall* in the amount of wind and storage capacity available to meet demand.

4. Repeat this for each $x = 0, 20, 40, 60, \ldots, 2000$ and for each $y = 1, 1.11, 1.22, \ldots, 12$.

5. Analyse the results to determine which pairs of x and y result in no shortfall in meeting demand, either through using storage capacity or wind generation.

The results from this are plotted in Figure 6.7. At least 647 additional storage units and at least 5.3 times the wind capacity are required, however these numbers depend on each other as the graph shows.

If we add in the existing interconnectors to this copperplate model, and rerun the analysis, we find significantly less storage and scaling of wind is required for no shortfall to occur. Here we aggregate the interconnectors and storage power flow limits and capacities together, so that the capacity is now unlimited however the flow limits are still applicable. We record when either a shortfall occurs or when the interconnector and storage are net positive dispatchers. This latter event occurs when the interconnector is importing more than it is exporting.

We find that at least 20 additional storage units are required to ensure no shortfall occurs. We also find that at least 3 times the wind is required to ensure the interconnector and storage have net negative dispatch, i.e. the storage is charging and the interconnector is exporting more than importing. There is little dependence between the wind multiple and the number of storage units. We understand this is due to the now unlimited capacity from combining the interconnector flow and capacity limits with the storage flow and capacity limits. We achieve similar results if we separate the storage and the interconnector and force the storage to charge first.

This analysis is one of the reasons we have chosen the definition of 100% renewable to allow importing and exporting with an interconnector, as otherwise the amount of batteries required is at least 647 additional units, which also requires at least 12 times the wind generation. As these numbers are from this copperplate network, we expect even larger storage and wind requirements when including line limits in the S-OPF. However, including the interconnector greatly reduces the storage and wind requirements.

Overall this suggests we must increase the wind capacity of the current S-OPF model to at least 3 times its current capacity. We expect the line limits to then prevent the interconnector from exporting and importing as freely as this copperplate model, and therefore additional storage units will need to be placed, although we expect not quite as many as the circa 650 required in the first analysis.

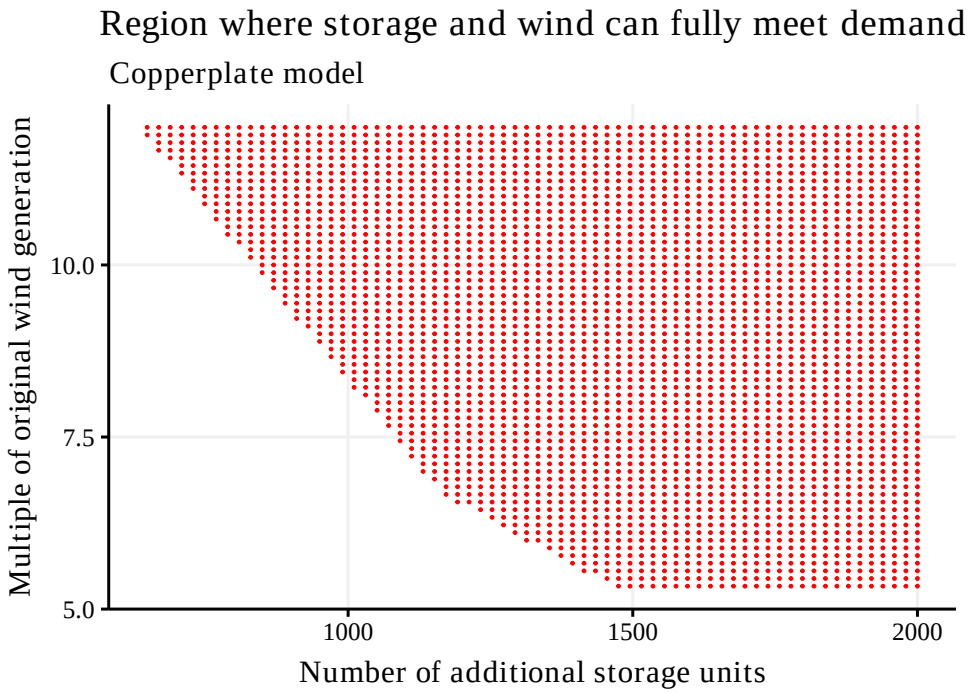


Figure 6.7: This graphs the parameter space pairs (x, y). Here x is number of additional theoretical storage units and y is the wind multiple. If the pair (x, y) allows all demand to be met by wind and storage at each time step, then the pair is plotted in red. The number of storage units required x depends on the amount of wind generation y, with higher wind generation requiring less storage. We see that at least 647 additional theoretical storage units are required, and this requires the full 12 times the amount of wind generation. We find that at least 5.3 times the amount of wind generation is required, and this requires nearly 1500 additional theoretical storage units.

6.4.2 Analysis using S-OPF

Our approach is to add additional storage units to all buses within the network and to analyse how this additional capacity is effects the cost of electricity and the dispatch profile for all generators. We use this to identify storage locations that are most well utilised for SA. There are a wide range of storage technologies with varying capacities and power limits. We use the battery storage that is currently installed in South Australia as a guide for the magnitude of storage that can be installed at each bus within the network. In South Australia we know that it is possible to install storage with order of magnitude of the Hornsdale Power Reserve, which at the time of installation was the largest battery of its kind. We consider additional storage units with the same 193.5 MWh capacity and 150MW power limit, similar to Section 6.4.1.

We place three storage units at every bus within the South Australian network and scale wind power capacity to see how power output approaches 100% renewable output. This results in 327 additional storage units added to the model, around half the circa 650 mentioned in Section 6.4.1.

In Section 6.3 importation via an interconnector was assumed to have the largest marginal cost. In the context of South Australia phasing out local fossil fuel generation this assumption is no longer applicable. South Australia does plan to rely on interconnectiora into the future however local fossil fuel generation is to be phased out. Hence we alter the marginal cost for importing electricity via interconnectors to be \$20/MWh. This is smaller than all other fossil fuel generation but larger than all other generation and hence power importation from interconnectors will be prioritised over fossil fuel generation. This approach is motivated by the desire at this point to not simply omit fossil fuel generation from the S-OPF, as fossil fuel generation guarantees feasibility of the S-OPF. Times when fossil fuel generation is relied on will be only when other generators cannot supply sufficient power given the network constraints. This will allow us to understand how renewables and interconnectors respond when they are preferred with respect to cost. Further, in this transitioning power system, since interconnectors are to be prioritised we expect the requirement for gas generation to be eliminated as the proportion of renewable generation is increased and additional storage utilised.

Previously in Section 6.3 we assume that CCGT and SSC technology type generators could not be turned off as they require much long start up and shut down times. However now we are interested in analysing the transitioning network where fossil fuel generation is eliminated. We now do not want to assume that fossil fuel generation will be relied on and hence reset the minimum output for CCGT and SSC generators to be zero.

A summary of the differences between Section 6.3.2 approach and this approach is below:

- Change in the cost of the interconnector, to prioritise its use compared to fossil fuels. It changed from a marginal cost of \$6,000 per MWh to only \$20 per MWh.
- Removal of the lower limits of fossil fuel generation with technology types CCGT and SSC. This allows fossil fuels to no longer be used in the transition to renewables.
- Addition of 327 additional storage units, see Section 6.4.3 for further details.
- Scaled wind generation, see Section 6.4.3 for details.

6.4.3 Simulations

With the aim to capture how storage is utilised when the system is under strain throughout winter and summer, we run the S-OPF over the full period defined for each season. Using

the copper plate analysis as a guide we add the 327 additional storage units and then increase the wind capacity until we are at 100% renewable. We then remove the less utilised storage. We analyse the results to determine which storage locations are the most and least utilised.

We are limited by the time that each simulation takes to run, which is on the order of 12 hours for each season with 20 samples at each time step. Part of the limitations in time come from the sequential dependencies of each step of this method, preventing distributed coding techniques to be used. This prevented us from applying a more complicated heuristic for this thesis.

We follow the steps listed below.

Step 1: Run S-OPF on the SA network with 3 additional storage units placed at each bus. While this increases the storage sufficiently, with the modelled power demand and wind capacity profile there is not yet enough renewable generation for the SA network to be 100% renewable. Results are discussed in Section 6.4.4.

Step 2: Scale initial wind capacity by integer factor (initially this factor is 2) and rerun Step 1. Consider if the network relies on 100% renewable energy. If it does not, increment the scale the wind (so that the factor is 3, then 4, etc) and repeat this process. Continue until the network relies on 100% renewable energy. We terminated this at scale factor 6. Results are discussed in Section 6.4.4.

Step 3: Analyse storage units that output the least and most power. We determine that the bottom 40% of used storage was contributing little to the overall use of storage: in summer it contributed to 11.0% of total outputted power from storage, while in winter it contributed to 17.3% as in Table 6.6. There was a large overlap with 83.2% of the bottom 40% of storage units being the same in each season. We take the intersection of the storage units that contributed to this bottom fortieth percentile for each season and remove these from the model. We then rerun the optimisation for each season again and considered how this changes the amount of renewable energy used. We discuss this in Section 6.4.5.

Step 4: As an extension, we add in the new interconnector between South Australia and New South Wales using the same model as the other interconnectors from Section 4.3 before rerunning the optimisation again. The costs for the interconnectors are as in Section 6.3, with the interconnector import changed to $20 per MWh as discussed in Section 6.4.2. This interconnector is located at bus 50, near Robertstown. Further details on this interconnector are in Section 6.4.6.

Step 5: We analyse all of the results. In particular, we looked at the locations of the top 20% of storage used. We discuss this further in Section 6.4.6

6.4.4 Results from Step 1 and 2

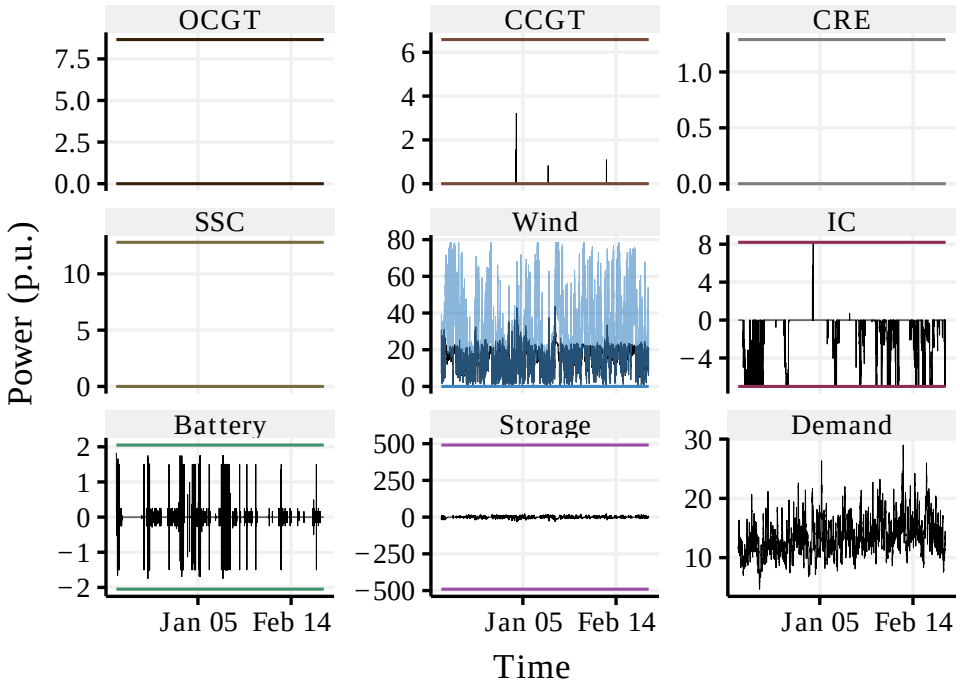
Following the process described in Step 1. and Step 2. with three additional storage units at each bus and wind capacity scaled by various factors we consider the dispatch profile for all generators, focusing on how the reliance on fossil fuel generation varies. There are significant differences in the dispatch of power from fossil fuels sources in summer and winter. After scaling wind by a factor of 6, in summer there is almost no reliance of fossil fuel generation while in winter there remains a small but significant reliance on fossil fuels. We refer to this as Case 1. Below we plot the power dispatch grouped by technology type.

In Figure 6.8 the power output is plotted for winter and summer with wind capacity scaled by a factor of 6 in each case. In both seasons there remains periods of time where there is reliance of fossil fuel generation in order to satisfy power demand. This occurs at times when simultaneously power demand is significantly high and wind capacity is low. In winter for example, during the period of time when fossil fuels are dispatched, the average demand is 16.8 p.u. while the wind capacity is 3.05 p.u. Comparing this with the average over the full winter period where the average demand is 14.2 p.u. and the wind capacity is 27.6 p.u. To have a one hundred percent renewable network there needs to be further renewable capacity or storage to cover these extreme situations.

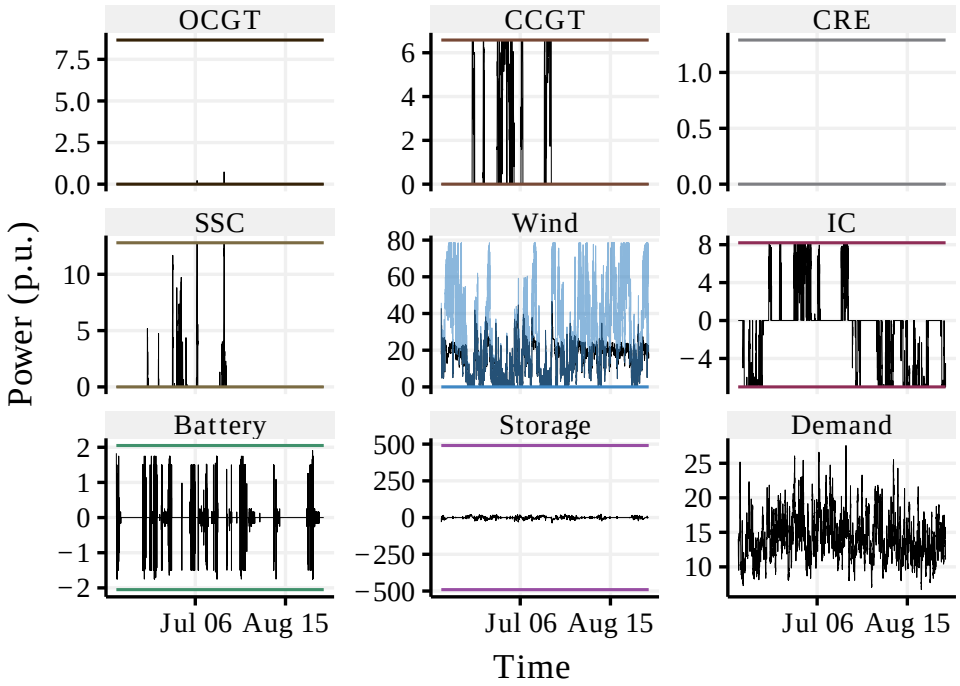
We mentioned in Section 4.4 that the transition to a 100% renewable electricity network should result in the costs approaching \$0. We see in Table 6.7 that the price has dropped from the circa \$50 per MWh in Section 6.3.3 to \$3.9 per MWh for winter and -\$0.3 per MWh for summer, which aligns with this.

Note that our approach to scale the wind capacity generated by the time series models has limitations, as in the real world additional wind generation would be located in different areas and hence subjected to different weather patterns. Therefore the correlation of this generation would be less than 1. This would make it less likely for very low periods of wind generation to occur at all wind farms at the same time. Another limitation in our approach is that new renewable generation also include solar farms which have a very different generation profile, which we do not model here.

In Figure 6.9 we compare the wind capacity with power outputs with box plots. While wind capacity has been scaled up not all of this additional capacity can be utilised. The highest power output from wind is only marginally higher than half the maximum capacity. Unlike in the copperplate analysis in Section 6.4.1 here network constraints limit how much wind power can be dispatched, and hence stored. Increasing storage capacity allows for wind capacity to be more utilised, however at times of low wind capacity the additional storage is not sufficient and fossil fuel generation is required. After scaling wind capacity by a factor of 6 in summer almost all generation is supplied from non fossil fuel sources. In winter fossil fuel sources still output approximately 4.5% of power, as in Table 6.7. The difference in distributions of wind capacity for summer and winter appear to show a significant difference in how fossil fuels are relied on in different seasons.



(a) Power dispatch for summer with wind capacity scaled by a factor of 6.



(b) Power dispatch for winter with wind capacity scaled by a factor of 6.

Figure 6.8: In Figure 6.8a and Figure 6.8b we plot the power dispatched over summer and winter respectively. The power output is plotted in black while the upper and lower power limits coloured for each technology type. The upper limit for wind is the scenario that occurs for the random variable wind capacity and hence varies over time. In summer there is almost no reliance on fossil fuels, while in winter 4.5% of power output is from fossil fuel generators.

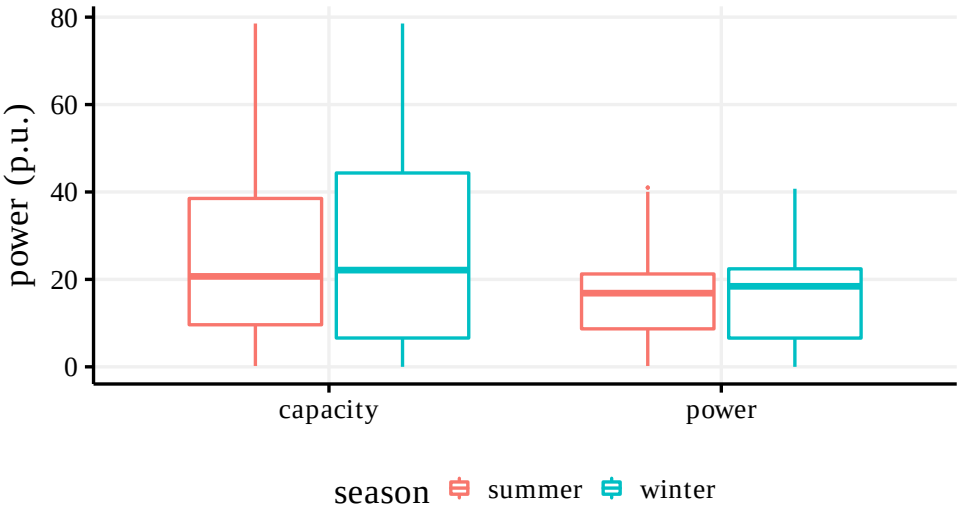


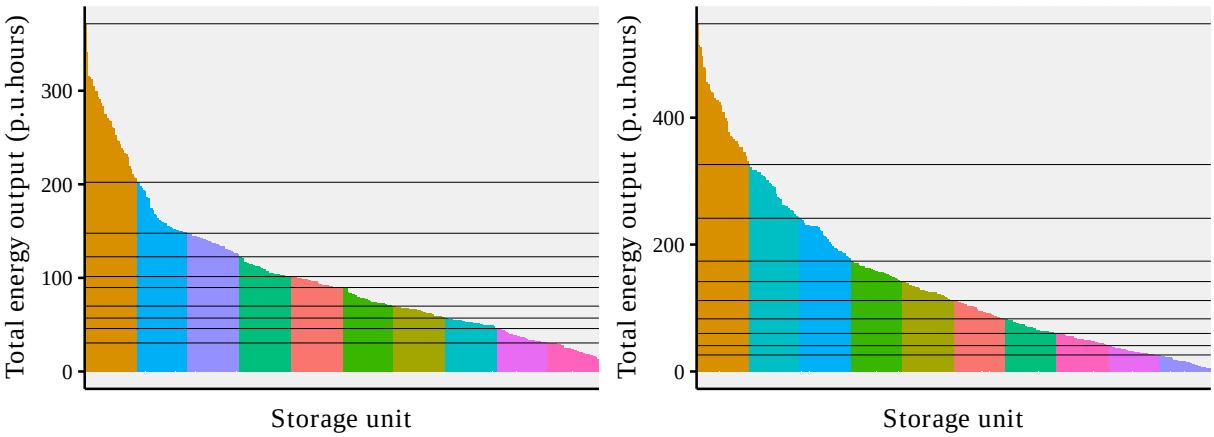
Figure 6.9: With the random variable wind capacity scaled by a factor of six not all of this capacity can be utilised. The maximum power output for wind over both seasons is approximately half of the maximum wind capacity. Network constraints limit how much wind power can be utilised. If the network constraint were no a limiting factor, any wind power that was available and not used by storage units would be exported via interconnections to further reduce cost.

As the wind generation for summer used less than 0.1% fossil fuels, we chose not to scale wind generation any further and now proceed to analysis how the additional storage is utilised. See Table 6.7 for more details.

6.4.5 Results of Step 3: Omitting less utilised storage

Following Step 3, we now omit some storage units. We first graph the total power output from each storage unit across each season in Figures 6.10a and 6.10b. We see that the change in proportion of power output is generally increasing for each decile. The proportion of total power output for each decile is given in Table 6.6.

We now omit the intersection of the bottom 40% of storage units used in each season. We chose to omit the bottom 40% as they contribute to less than 20% of the power output from storage in both seasons, as in Table 6.6. We expect that omitting these bottom 40% will result in other storage units being utilised instead to some degree. We find that the nodes that no longer have any storage units are the nodes 8, 9, 29, 31, 33, 39, 49, 63, 67, 74, 89, 95, 96, 99, 106, and 109, which, as in Figure D.1, are all in remote locations away from large demand regions such as Adelaide.



(a) Total power output for storage units in winter.

(b) Total power output for storage units in summer.

Figure 6.10: For each season, we plot in a descending order the total output for each additional storage unit. Each decile is coloured and the largest power output for each decile is indicated with a horizontal line. This highlights the disproportionate use of the additional storage units throughout the network. Note also how these plots differ, where storage output more power in summer than winter. However, the lower decile contribute more to total storage output in winter than in summer.

Note here we have omitted individual storage units and not all storage units from nodes. The motivation for this method was to discretely include/exclude storage allowing for simple integration with the current network code. Alternatively, we could aggregate power output for storage units by their node located and omit all storage from nodes with lower output, however we do not use this method here.

Table 6.6: Proportion of power output(%) for up to and including each decile.

decile	10	20	30	40	50	60	70	80	90	100
winter	2.3	5.8	11.0	17.3	24.6	34.0	44.7	58.1	73.9	100.0
summer	1.1	3.1	6.3	11.0	17.1	26.1	37.9	52.7	72.4	100.0

After omitting these storage units, we run the optimisation again for each season and refer to this as Case 2. This results in a larger reliance on fossil fuels for both seasons as in Table 6.7. Although the omitted storage had a small output over the full season in Case 1, its omission does increase the reliance on fossil fuel generation in Case 2. Removing this storage also results in increasing the price. We see in Table 6.7 the price for summer increases from -$0.3 per MWh to $0.3 per MWh on average, while the price for winter

increases from $3.9 per MWh to $4.9 per MWh. This is due to reducing the amount of relatively cheap storage that is able to supply power in Case 2.

We next explore how future interconnectors in conjunction with the storage from Case 2 will support South Australia's transition to renewables.

6.4.6 Step 4 and 5: Project energy connect interconnector and overall analysis

Here we introduce an additional interconnector to the S-OPF network model of South Australia, which is Step 4.

Currently South Australia has two interconnectors that connect South Australia to the NEM through Victoria and these are included in the South Australian Network model. However a new interconnector is under construction between South Australia and NSW [22]. This interconnector will run from Robertstown in South Australia's Mid North region through to Wagga Wagga in NSW with 800MW power capacity. This increased interconnector capacity will allow further trade of power between South Australia and the rest of the NEM and allow for further investment in renewable generation in the vicinity of the interconnector line.

Through comparing the route map for the interconnector [99] with the open NEM map we identify the bus for this interconnector in South Australia near Robertstown. Since we are modelling the interconnectors as generators rather than as lines, we can include the new interconnector in the network model without needing to recalculate the other network constraints. As we did in Section 4.3, we represent the interconnector as generator for importing and as a storage unit for exporting at this node with 800MW power capacity respectively.

We run the S-OPF again for each season with the storage in Case 2 and the new interconnector. This is referred to as Case 3. As expected the addition of the new interconnector makes a significant difference in reducing the reliance on fossil fuel generation as documented in Table 6.7. In summer, the fossil fuel use reduces from 0.5% for Case 2 to 0.1% for Case 3, while the winter use reduces from 5.8% to 2.1%.

As expected, the price was affected by including the interconnector. We see in Table 6.7 the price for summer drops from $0.3 per MWh to $0.1 per MWh, while the price for winter drops from $4.9 per MWh to $3.0 per MWh. This is due to the interconnector being able to supply cheaper power in Case 3 than the fossil fuels used in Case 2.

In Case 3, we found that the top 20 percentile account for approximately 40 percent of power output from all additional storage units, which is comparable to Case 1 as in Table 6.6. We found that many of the storage units in the top 20th percentile have more than one storage unit from the same bus. If we consider only buses where all three storage units are in the top 20 percentile for both seasons, then this gives bus 3,4,5,12,17,43,91

and 100. Power output from storage units at these buses contribute to 21% and 23% of the total power output from storage units for winter and summer respectively. We plot the location of the associated buses in Figure 6.11.

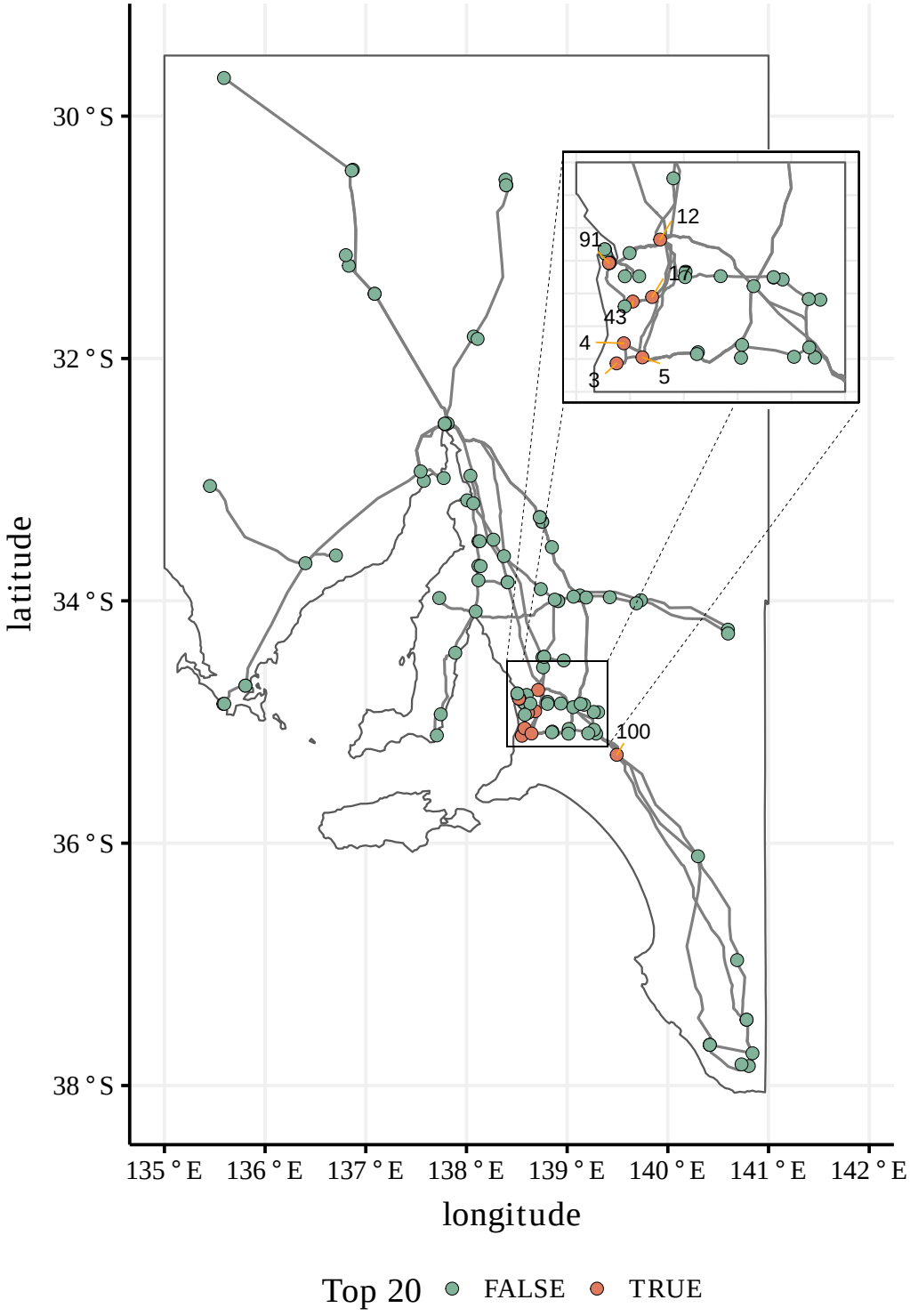


Figure 6.11: Taking only the buses where for both seasons (winter and summer) all three additional storage units have power output in the top 20 percentile. These buses are: 3,4,5,12,17,43,91 and 100. These buses are indicated as TRUE and the remaining buses are indicated as FALSE. We see that all buses are in close proximity to Adelaide, which is a region of high demand.

Recall the demand is allocated to buses on a proportional population basis. Out of eight buses in Figure 6.11 indicated as TRUE with the highest power utilisation from additional storage, six of these buses are in the top ten buses with respect to demand. The only buses not in the top 10 is bus 91 which has the thirteenth highest demand allocation, while bus 100 has far less demand. All of these buses except for bus 100 are located within Adelaide.

Some of these highly utilised storage units are located at or near buses with generators. At bus 91 is the Torrens Island A Power Station and at bus 3 is the Lonsdale Power Station and Pt Stanvac Power Station.

Another important consideration is the degree for a bus. That is the number of transition lines that are connected the the bus. If a bus has more transmission line connections then provided that lines have sufficient capacity there is greater ability for storage at that bus to charge and discharge power. From the buses indicated as TRUE in Figure 6.11 buses 12, 5 and 100 are in the top ten degree with degree 9, 6 and 6 respectively. Bus 12 is located slightly north of Adelaide while bus 100 is South of the city. All transmission lines from the south east of the state that run north towards Adelaide go through bus 100, which makes it a significant connection bus and may explain why storage is well utilised at this bus.

While there are similarities, including the new interconnector in the S-OPF makes a difference to how storage is utilised. For instance in summer in Case 3 all three storage units at bus 21 are in the top 20 percentile for power output, while in Case 2 only one storage unit at bus 21 is in top 20 percentile for power output. We suspect that this will be due to power from the interconnector flowing through this bus on the way to the high demand region of Adelaide. Hence when planning for storage it is important to consider how the transmission infrastructure will change and what effect this will have on system operation.

Table 6.7: Summary of usage power output and price for difference network cases

Case	Season	Fossil (%)	Other (%)	Mean Price ($/MWh)
Case 1.	Winter	4.5	95.5	3.86
Case 1.	Summer	0.0	100.0	-0.27
Case 2.	Winter	5.8	94.2	4.94
Case 2.	Summer	0.5	99.5	0.29
Case 3.	Winter	2.1	97.9	3.00
Case 3.	Summer	0.1	99.9	0.06

6.5 Conclusion

This chapter shows how we can use S-OPF on a model of the South Australian NEM with storage and interconnectors, and how we can understand the optimal locations for storage as the network transitions to a 100% renewable electricity network. This builds on the work of all previous chapters. We run our model over both summer and winter to account for a full range of seasonal conditions, as in Section 5.4.2.

In Sections 6.1 and 6.2 we first describe how we use the open NEM model [115] as a foundation to extract the South Australian portion of the model and add to it active power flow line limits for the transmission lines. Further, we add to the model the three battery storage units connected to the South Australian network since June 2017.

After initially running a determinist OPF in Section 6.3 on the South Australian network we use the full S-OPF formulation to compare the effect of the battery storage in the SA network. We conclude that our model gives a reasonable representation of the NEM model, with comparable costs, as well as behaving in line with our results from Chapter 4. That is, in Section 6.3.1 with the deterministic OPF we see that wind power is almost always used at its full capacity, and that the least expensive fossil fuel generators are dispatched before the more expensive fossil fuel generators, and before the interconnectors. We see with the S-OPF in Section 6.3.2 results that match this, and that adding in storage results in lowering the average costs from \$57.2 per MWh to \$50.0 per MWh for winter due to a lower reliance on fossil fuel generation and interconnector imports.

In Section 6.4 we use the SA model and our S-OPF formulation to investigate where storage is optimally placed within the SA network. We first analyse a copperplate network in Section 6.4.1 to gain an understanding of how much additional wind generation and storage capacity is required as we progress to a 100% renewable electricity network. We note that we need at least three times the amount of wind generation and potentially as many as 650 additional storage units, depending on our definition of 100% renewable electricity generation.

In Section 6.4.2 we discuss how we modify the S-OPF to aid in our analysis of the network as it transitions to 100% renewable electricity generation, before detailing our heuristic approach in Section 6.4.3 to analysing optimal storage locations. This results in adding around 300 additional storage units initially, as well as scaling the wind generation by 6 times as we show in Section 6.4.4. Finally, in Section 6.4.5 we remove the bottom 40% of storage units that add little to the generation. We note that these nodes are located mostly in remote locations.

A significant feature of the South Australian network is its interconnection to the rest of the NEM. Using our model for an interconnector in a transitioning network we model the additional effect that the new SA-NSW interconnector has on the network in conjunction with storage. The interconnector changes the location of the most used

storage units, as well as lowering the price per MWh.

Finally, we examine which buses contain the most well utilised storage units. We do this by identifying buses where all three storage units have power output in the top 20 percentile of all storage units for both seasons. We clearly see that these storage units are generally located in areas of high population and hence high demand.

Overall, we have demonstrated how our S-OPF method can be used to quantify the effect of storage and interconnectors in a transitioning electricity network, and how it can be used to determine the optimal locations of storage.

Chapter 7

Conclusion

This thesis presented a methodology to determine how generators and energy storage should be optimally used in a power network. It accounts for network constraints as well as the uncertainty of consumer demand and renewable energy generation over time. It applies the methodology to the South Australian electricity network, and determines optimal locations of energy storage as South Australia transitions to a 100% renewable energy electricity network. Although we apply our method to specific cases, the framework is general enough such that it can be adapted to other networks transitioning to renewables.

This thesis first covered important background material for our methodology. We defined a two-stage stochastic program with fixed recourse, the stochastic program of interest in this thesis. We prove several of our own results on the existence of solutions of these stochastic programs. In particular, we show that a finite optimal solution exists under certain non-restrictive conditions which apply for our methodology.

We focused on the sampling average approximation form where implementation uses Monte Carlo sampling methods. The sampling average approximation program has dual decomposition structure allowing the program to be decomposed into separate linear programs where efficient solution methods are applicable. We use the L-shaped method solution method, giving our own description of the algorithm and its convergence properties. We use these results to build our methodology.

We detailed background in power systems analysis with the main focus on linear approximation of active power flow. We defined DC-OPF and in particular DC-OPF using the PTDF formulation, from which we use as the basis for our S-OPF. We develop our own stochastic-optimal power flow formulation that defines the DC-OPF as a two-stage stochastic program with fixed recourse. Having adapted a model for generic storage and created a model for interconnectors for the networks of interest, we devised a framework for using the S-OPF to simulate optimal dispatch of power over time. We generate scenarios using time series models and apply our method to test networks, and we quantify the cost benefit of our S-OPF with a comparable deterministic program.

Forming the sampling average approximation stochastic program with a range of sample sizes we calculate confidence intervals for the true solution using the single replication procedure and use this to understand the relationship between sample size and solution accuracy. We conclude that the S-OPF behaves as expected, provides significant insight into power dispatch in the presence of storage and interconnectors, and that the accuracy improves as the sample size increases and is sufficient for our purposes.

Adapting our methodology to the South Australian electricity network, we fit time series models to 2017-18 wind and demand data. These models are then used to generate a distribution of scenarios for these random variables, which show a strong fit with the historical data.

Finally, we implemented our S-OPF methodology on the South Australian network. To do this, we first constructed a model of the network that was used to form constraints within the S-OPF. Using this model together with the scenarios from our time series models, we implemented the S-OPF over time in a number of cases. This is applied in the context of South Australia transitioning to a 100% renewable power system. We use a greedy approach, scaling wind capacity and including theoretical storage at all buses before applying an elimination strategy to omit less utilised storage. We compare our results with a copperplate analysis.

We find more storage and wind generation is needed than in the copperplate network analysis, and that our method gives more detail on locations of storage. This implies that line limits and network constraints do make a difference to optimal storage use and locations, that both should be considered in planning, and that our methodology can assist with this. We conclude that locations of high demand and high connectivity are preferenced by our methodology, and indicate where storage may optimally placed.

There are many avenues for future work. This includes refining the models used for the generators, storage and interconnectors, to closer approximate their behaviour. For example, modelling the interconnector with varying price to reflect demand and supply in the adjacent network. While our analysis used a greedy approach for optimising storage locations, in future work other methods for introducing storage could be used. Similarly while we scale the capacity for wind power currently installed, we acknowledge that other approaches could be considered in the future. This may affect the optimal storage within the network. Our time series models can also be extended to include further predictors and machine learning techniques could be incorporated. We note that many aspects of our model use 2017-18 data as this was used in [115]. Updating to the current data and including new network features such as large scale solar generation we leave for future work.

Overall, this thesis has constructed and tested a S-OPF methodology, and analysed results after applying it to networks transitioning to renewables. We conclude that the S-OPF appropriately accounts for both the network constraints and the uncertainty for optimising power dispatch. We have shown how the S-OPF can be used to understand

the role and optimal locations of storage in a transitioning network and demonstrated this on a real network with insightful results.

Appendix A

Power systems derivations

In this section we detail the proof of Lemma 3.2 as in Chapter 3.

Before proving Lemma 3.2 we first consider that the node-arc incidence matrix N does not have full rank as in Lemma A.1.

Lemma A.1. *The matrix N has rank $n-1$ for a connected graph/network. If we remove any row from N to form $\tilde{N}$ this will be an equivalent system and $\widetilde{N}$ will have full rank of $n-1$.*

Proof. The row sum of N is zero, meaning that any one of the rows can be written as a linear combination of the other rows. This means that rank(N) is at most $n-1$.

Suppose that rank(N) is strictly less than $n-1$. This is true if and only if removing any one row of N would still leave linearly dependent rows. This would occur if and only if there exists a nontrivial linear combination of the rows that equals the zero vector. That is, if the rows of $\widetilde{N}$ are given by $b_1, b_2, \ldots, b_{n-1}$ then there exists $c_1, c_2, \ldots, c_{n-1} \in \mathbb{R}\backslash\{0\}$ such that,

$$c_1 b_1 + c_2 b_2 + \cdots + c_{n-1} b_{n-1} = 0.$$

Suppose without loss of generality $c_1 \neq 0$ then we can scale all coefficients so that $c_1 = 1$. Recall we use the convention that if $i < j$ then a line connecting these two nodes will be denoted (i, j). In N row 1 will correspond to the lines that are connected to node 1. So for any node k connected to node 1 we have $N_{(1,(1,k))} = 1$ and hence $N_{(k,(1,k))} = -1$ which implies $c_k = 1$ in order for the row sum to be zero.

Repeating this argument for all other rows we get that $c_k = 1$ for all k. So we have

$$b_1 + b_2 + \ldots + b_{n-1} = 0.$$

However, we also know that the rows sum to zero, so

$$b_1 + b_2 + \cdots + b_{n-1} + b_n = 0.$$

This must imply that $b_n = 0$, the zero vector. This would mean that there is no line that connects node n to the other nodes, which is a contradiction. So the rows of $\widetilde{N}$ are linearly independent and rank(N) is equal to $n-1$. Note that this does not depend on which row is removed. □

Now we can prove Lemma 3.2 as follows,

Proof. Assume the network/graph is connected. We have

$$\begin{bmatrix} \widetilde{P} \\ P_n \end{bmatrix} = P = Np = \begin{bmatrix} \widetilde{N} \\ b_n \end{bmatrix} p$$

Since the system is assumed to be balanced (so $P_n = -\sum_{i=1}^{n-1} P_i$ where $P = (P_1, \ldots, P_n)$), then we have that

$$P = Np \iff \widetilde{P} = \widetilde{N}p = \widetilde{N}YN^T\theta$$

From the definition of N we know that the column sums of N^T are zero. This implies that for any constant vector $a = c(1, 1, \ldots, 1)^T$ for $c \in \mathbb{R}$, then

$$N^T(\theta + a) = N^T\theta + N^Ta = N^T\theta + 0 = N^T\theta. \tag{A.1}$$

So without loss of generality we choose θ_n, the final element of θ, to be 0 (or equivalently choose c such that $\theta_n = -c$). If we denote $\widetilde{\theta}$ as the subvector of θ consisting of the first $n-1$ entries, then we have that

$$N^T\theta = \tilde{N}^T\widetilde{\theta}.$$

Hence we have the equivalent system $\widetilde{N}Y\widetilde{N}^T\widetilde{\theta} = \widetilde{P}$. Since Y is a positive definite matrix then we write

$$\widetilde{B}_{\text{bus}} = \widetilde{N}Y\widetilde{N}^\top,$$

which is also positive definite and hence invertible. So we can write

$$\widetilde{\theta} = (\widetilde{N}Y\widetilde{N}^T)^{-1}\widetilde{P} = \widetilde{B}^{-1}\widetilde{P}$$

and hence can write the vector of power flows as

$$p = YN^T\theta = Y\widetilde{N}^T\widetilde{\theta} = Y\widetilde{N}^T(\widetilde{N}Y\widetilde{N}^\top)^{-1}\widetilde{P},$$

as required. □

A.1 Power flow decomposition using Moore-Penrose sudo-inverse

Another method to solve for the voltage phase angles θ is used by Nesti, Zocca, and Zwart [90]. Recall we have that $P = B_{\text{bus}}\theta$ where the B_{bus} term is singular. We can use a *Moore-Penrose* pseudo-inverse $B_{\text{bus}}^{+} \in \mathbb{R}^{n\times n}$ for B_{bus} and hence we obtain θ as

$$\theta = B_{\text{bus}}^{+}P.$$

Note that such an inverse can be calculated using singular value decomposition, see Ben-Israel and Greville [34] for further details. It can be shown that $B_{\text{bus}}^{+} = (B_{\text{bus}} + \frac{1}{n}J)^{-1} - \frac{1}{n}J$ where $J \in \mathbb{R}^{n\times n}$ is the matrix with every entry being one. We do not use this method here and refer the reader to [90] for further details.

Appendix B

Algorithms for modelling storage in the S-OPF

Here we give further details on how the *Continuous Linear Model* constraints from [62] are equivalently formulated and applied in practice within our S-OPF over time. Within the simulation we are solving an optimal power flow problem at each time step for a number of time steps. At each time the OPF is solved, the capacity of the storage (charge) and the rate (power) that it can be dispatched at needs to be updated for the next time. We describe how this is done below such that charging and discharging constraints are satisfied.

Firstly the storage units need to be initialised such that at time 1 the storage units satisfy the relevant constraints.

Algorithm 1: Algorithm for initialising storage power output and capacity.

function Initialise_Storage($P_{base}, \delta t, P^{\max}_{ES,n}, E^{\max}_{ES,n}, p$);
Input : P_{base} power base of power system, δt time step between dispatch in hours, $P^{\max}_{ES,n}$ the maximum power input and output (MW), $E^{\max}_{ES,n}$ the maximum capacity of the storage unit (MWh), p proportion of max capacity for initial capacity $E_{EN,n,0}$.
Output: $E_{ES,n,t}$ storage capacity at time t, $P^{\max}_{ES,n,t}$ maximum power output at time t, $P^{\min}_{ES,n,t}$ maximum power input at time t.
Set $E_{EN,n,0} = pE^{\max}_{ES,n}$
Set $P^{\max}_{ES,n,t} = \min(E_{EN,n,0}/\delta t, P^{\max}_{ES,n})$
Set $P^{\min}_{ES,n,t} = -\min((E^{max}_{ES,n} - E_{EN,n,0})/\delta t, P^{\max}_{ES,n})$
return $(E_{ES,n,t}, P^{\max}_{ES,n,t}, P^{\min}_{ES,n,t})$;

Secondly, the power input, output and charge need to be updated for each storage

unit at each time step t.

Algorithm 2: Algorithm for updating available storage power output and charge for time t.

function Update_Storage($P^{net}_{ES,n,t-1}, E_{ES,n,t-1}, P^{\max}_{ES,n}, E^{\max}_{ES,n}, \delta t$);
Input : P_{base} power base of power system,δt time step between dispatch in hours, $P^{\max}_{ES,n}$ the maximum power input and output, $E_{ES,n,\max}$ the maximum capacity of the storage unit, p proportion of max capacity for initial capacity $E_{ES,n,0}$.
Output: $E_{ES,n,t}$ storage capacity at time t, $P^{\max}_{ES,n,t}$ maximum power output at time t, $P^{\min}_{ES,n,t}$ maximum power input at time t.
Set $E_{ES,n,t} = -P^{net}_{ES,n,t-1}\delta t + E_{ES,n,t-1}$
Set $P^{\max}_{ES,n,t} = \min(E_{ES,n,t}/\delta t, P^{\max}_{ES,n})$
Set $P^{\min}_{ES,n,t} = -\min((E_{\max} - E_{ES,n,t})/\delta t, P^{\max}_{ES,n})$
return $(E_{ES,n,t}, P^{\max}_{ES,n,t}, P^{\min}_{ES,n,t})$;

At each time step t the charging and discharging constraints for storage are equivalently given by

$$-P^{\max}_{ES,n,t} \leq P^{net}_{ES,n,t} \leq P^{max}_{ES,n,t},$$

Appendix C

Time series background

In this section we recall some additional definitions and fundamental results from time series analysis relevant for Chapter 5. We follow the references [45] and [70]. We use the notation of a lower case to denote observations, e.g. x, and an upper case to denote random variables, e.g. X.

Definition C.1. A *time series* is a set of observations $\{x_t, t \in T\}$, where each observation is made at a particular time t. We assume $T \subset \mathbb{R}$ is discrete and consists of equally spaced elements.

We usually consider the observations x_t to be observations of random variables X_t, which gives the notion of a stochastic process as follows. This will allow us to form a model to analyse time series observations.

Definition C.2. A *stochastic process* is a collection of random variables $\{X_t, t \in T\}$, indexed over a collection of time points T. We assume $T \subset \mathbb{R}$ is discrete and consists of equally spaced elements. We denote μ_t the mean of X_t and σ_t the standard deviation of X_t. For any subset $T_1 \subset T$, we denote the joint probability distribution of $\{X_t : t \in T_1\}$ by f_{T_1}.

Definition C.3. A stochastic process X_t is *stationary* (also known as *strictly stationary*) if its properties are unaffected by a change of time origin. That is, if $T_1 = \{t_1, t_2, \ldots, t_m\} \subset T$ and for any k such that $T_1 + k = \{t_1 + k, t_2 + k, \ldots, t_m + k\} \subset T$ then $f_{T_1} = f_{T_1+k}$.

For a stationary process the distribution of X_t is the same for all t, this implies the mean and variance are constant so that,

$$\mu_t = \mu, \tag{C.1}$$
$$\sigma_t^2 = \sigma^2. \tag{C.2}$$

Note however that while the observations $\{x_t : t \in T_1\}, T_1 \subset T$ can be considered a sample from the same random variable X_{t_1}, they are not likely to be independent.

However, for a stationary process, we can consider the observations $\{x_t : t \in T_1\}, T_1 \subset T$ and $\{x_t : t \in T_1 + k\}, T_1 + k \subset T$ to be a sample from the same collection of (possibly dependent) random variables $\{X_t : T_1 \subset T\}$. Often this is used to check whether a time series may satisfy the definition of stationary. For example, one can consider whether the observations from different *rolling periods* have the same (or similar) sample mean and sample variance, where by rolling periods we mean all observations over the time periods $[a + k, b + k] \cap T$, for all k and for a given $a, b \in \mathbb{R}$. If not, this would indicate the underlying process is not stationary.

To give a measure of the dependency between the random variables, we define the autocovariance and autocorrelation function.

Definition C.4. For a stochastic process $\{X_t, t \in T\}$ we define the *autocovariance* function $\gamma(r, s)$ at $r, s \in T$ as the covariance between X_s and X_r, so that

$$\gamma(r, s) = \text{cov}[X_r, X_s]. \tag{C.3}$$

If $\{X_t, t \in T\}$ is stationary, then $\gamma(r.s) = \mathbb{E}[(X_r - \mu)(X_s - \mu)]$, which only depends on the difference $k = |s - r|$, called the *lag*. In this case, we denote the autocovariance function as $\gamma(k)$.

Definition C.5. We define the *autocorrelation* function at $r, s \in T$ as

$$\rho(r, s) = \frac{\mathbb{E}[(X_r - \mu)(X_s - \mu)]}{\sqrt{E[((X_r - \mu)^2 E[((X_s - \mu)^2)]}}.$$

For a stationary process this can be simplified since the variance is constant over the random process so that the autocorrelation only depends on the lag k. This gives

$$\begin{aligned} \rho(k) &= \frac{\mathbb{E}[(X_t - \mu)(X_{t+k} - \mu)]}{\sigma^2}, \\ &= \frac{\gamma(k)}{\gamma(0)}. \end{aligned}$$

Note for a lag $k = 0$ we obtain $\rho_0 = 1$.

Both the autocovariance and the autocorrelation can be used to check if the stochastic process is stationary, as again one would expect the sample autocovariance and sample autocorrelation to be the same (or similar) over rolling periods for a stationary process. However, if the data contains a trend then the sample autocorrelation has a slow decay

in absolute value while if the data has strong deterministic periodicity then the sample autocorrelation will display the same periodicity.

A weaker definition of stationarity is the notion of *second order stationary* or (*weakly stationary*). This weakens the previous definition to consider only the first and second moments of the distributions, rather than the distributions themselves.

Definition C.6. We say a process is *second order stationary* if its mean is constant and its autocovariance function depends only on the lag, that is,

$$\begin{aligned} \mathbb{E}[X_t] &= \mu, \\ \mathrm{Cov}[X_t, X_{t+\tau}] &= \gamma(\tau). \end{aligned}$$

It is not possible to calculate autocovariance and autocorrelation given a sequence of time series observations $x_{t_1}, x_{t_2}, \ldots x_{t_N}$. However we can estimate the autocovariance and autocorrelation. Here we take the definitions as used in R and defined in [108].

Definition C.7. The *sample autocovariance* at lag k is given by

$$c_k = \frac{1}{N} \sum_{t=1}^{N-k} (x_t - \bar{x})(x_{t+k} - \bar{x}),$$

where $\bar{x} = \frac{1}{N} \sum_{t=1}^{N} x_t$.

Definition C.8. The *sample autocorrelation* at lag k is given by

$$r_k = \frac{c_k}{c_0}.$$

For a second-order stationary time series the sample autocovariance and sample autocorrelation are good approximations for the autocovariance and the autocorrelation respectively as N becomes large. More precisely, they are approach the true autocovariance and autocorrelation asymptotically as N increases. See Brockwell and Davis [47, §7.2] for further details.

The sample autocovariance and sample autocorrelation can be used to identify appropriate stochastic time series models. Another useful measure for identifying the appropriate probability model for a time series is the partial autocorrelation.

Definition C.9. The *partial autocorrelation* function ϕ_{kk} where k is the lag, arises from solving the Yule-Walker equations which are a set of recurrence relations that can be derived as in [45]. The Yule-Walker equations are given by

$$\rho_j = \phi_{k1}\rho_{j-1} + \cdots + \phi_{k(k-1)}\rho_{j-(k-1)} + \phi_{kk}\rho_{j-k}, \text{ for } j = 1, 2, 3, \ldots, k.$$

The partial autocorrelation ϕ_{kk} is the autocorrelation between x_t and x_{t+k} once the linear relationship between $x_{t+1}, x_{t+2}, \ldots, x_{t+k-1}$ is removed.

The set of Yule-Walker equations can be approximated by substituting the autocorrelation ρ_j with sample autocorrelation r_j from which we can solve for the coefficients ϕ_{kk} for all k. Then the partial autocorrelation gives an indication of dependence of a x_{t_1} on the previous observations $\{x_t : t < t_1, t \in T\}$. This is related to the property of autoregession, which is a common property for a stochastic process, as we discuss in the next section. In particular, an autoregressive process of order p will have ϕ_{kk} non-zero when $k \leq p$ and zero for $k > p$. We define these concepts precisely in the next section.

C.1 Demand model fit

Here we give the coefficients and standard error for the fitted demand model from Section 5.3.

Table C.1: Linear model for Box-Cox transformation of power demand with $\lambda = 0.01900043$. (Linear model with ARIMA(2,0,4) errors)

term	coefficient	s.e.	term	coefficient	s.e.
ar1	1.3637	0.1018	ar2	−0.3993	0.0973
ma1	−0.3219	0.1027	ma2	0.1923	0.0112
ma3	0.1336	0.0212	ma4	0.0351	0.0253
temperature	−0.0067	0.0020	temperature^2	0.0001	0.0000
isworkdayTRUE	0.0035	0.0035	fourier($C1-48$)	0.0603	0.0048
fourier($S1-48$)	−0.1216	0.0048	fourier($C2-48$)	−0.0331	0.0024
fourier($S2-48$)	−0.1162	0.0024	fourier($C3-48$)	0.0296	0.0014
fourier($S3-48$)	0.0412	0.0014	fourier($C4-48$)	0.0062	0.0009
fourier($S4-48$)	0.0368	0.0009	fourier($C5-48$)	0.0106	0.0006
fourier($S5-48$)	0.0000	0.0006	fourier($C6-48$)	0.0095	0.0004
fourier($S6-48$)	0.0078	0.0004	fourier($C7-48$)	0.0074	0.0003
fourier($S7-48$)	0.0104	0.0003	fourier($C8-48$)	0.0064	0.0003
fourier($S8-48$)	0.0098	0.0003	fourier($C9-48$)	0.0067	0.0003
fourier($S9-48$)	0.0058	0.0003	fourier($C10-48$)	0.0056	0.0003
fourier($S10-48$)	0.0051	0.0003	fourier($C1-336$)	0.0922	0.0109
fourier($S1-336$)	−0.0201	0.0108	fourier($C2-336$)	−0.0400	0.0097
fourier($S2-336$)	0.0233	0.0097	fourier($C3-336$)	0.0130	0.0084
fourier($S3-336$)	−0.0093	0.0084	fourier($C4-336$)	−0.0077	0.0072
fourier($S4-336$)	−0.0046	0.0072	fourier($C5-336$)	0.0158	0.0063
fourier($S5-336$)	0.0147	0.0063	fourier($C1-17520$)	−0.0700	0.0122
fourier($S1-17520$)	0.0333	0.0115	fourier($C2-17520$)	0.0733	0.0112
fourier($S2-17520$)	0.0932	0.0112	fourier($C3-17520$)	−0.0211	0.0112
fourier($S3-17520$)	0.0033	0.0112	-	-	-

- σ^2 estimated as 0.001296,
- log likelihood=33399.34,
- AIC=-66704.69, AICc=-66704.43, BIC=-66339.45.

Appendix D

South Australian transmission network

Here we detail additional inputs for the South Australian network model from Chapter 6 as well as a map of the network.

Table D.1: South Australian Buses

node Int64	demand Float64	node Int64	demand Float64
1	0.000 257 382	56	0.006 144 44
2	0.013 301 5	57	0.000 373 805
3	0.097 755 1	58	0.005 191 39
4	0.067 991 5	59	0.002 222 67
5	0.019 753 1	60	3.88809×10^{-5}
6	0.013 945 3	61	0.000 128 097
7	0.046 059	62	0.001 353 73
8	2.83770×10^{-5}	63	0.005 716 56
9	4.59142×10^{-5}	64	0.002 356 74
10	0.000 391 053	65	0.000 500 22
11	8.68743×10^{-6}	66	0.002 162 41
12	0.082 540 1	67	0.001 450 82
13	0.000 600 457	68	0.007 118 82
14	6.61319×10^{-5}	69	0.007 629 68
15	0.000 721 092	70	0.078 155 6
16	0.008 104 49	71	0.122 532
17	0.061 443 2	72	0.019 528 4
18	0.001 139 11	73	8.05071×10^{-5}
19	0.004 261 77	74	0.000 633 503
20	0.002 156 35	75	0.008 430 75
21	0.002 247 29	76	0.010 850 8
22	0.006 690 85	77	0.000 899 131
23	0.010 32	78	0.000 845 716
24	0.000 622 535	79	0.004 055 02
25	0.007 224 44	80	0.001 985 23
26	0.017 969 1	81	0.002 456 37
27	0.012 277 4	82	0.000 561 118
28	0.001 819 29	83	0.001 431 9
29	0.000 922 841	84	0.002 353 08
30	0.000 127 2	85	0.000 155 359
31	1.83845×10^{-5}	86	3.49556×10^{-5}
32	0.013 420 4	87	0.000 530 21
33	0.008 828 83	88	0.000 620 743
34	0.000 572 543	89	0.000 891 243
35	0.001 016 54	90	0.006 180 73
36	0.001 233 51	91	0.015 203 1
37	0.001 874 37	92	0.004 707 44
38	0.000 395 01	93	0.001 134 12
39	0.000 237 113	94	7.12938×10^{-5}
40	0.001 297 48	95	0.000 840 199
41	0.001 425 92	96	1.73887×10^{-5}
42	0.000 835 082	97	0.003 262 52
43	0.061 795 5	98	0.000 738 28
44	0.000 957 351	99	0.000 248 329
45	0.001 389 06	100	0.003 057 96
46	0.000 691 921	101	0.001 738 98
47	0.047 614 6	102	0.006 852 18
48	2.71189×10^{-6}	103	0.000 100 571
49	0.003 565 98	104	0.004 760 25
50	0.000 179 249	105	0.004 090 19
51	0.000 367 877	106	0.006 001 9
52	0.000 895 273	107	0.000 164 614
53	0.001 269 88	108	0.005 084 69
54	0.000 203 619	109	0.004 624 52
55	0.000 822 149		

Table D.2: South Australian Lines A

name String7	from node Int64	to node Int64	line lim Float64	reactance Float64	name String7	from node Int64	to node Int64	line lim Float64	reactance Float64
$\ell_{50,51}$	50	51	1.37	0.012 078	$\ell_{86,108}$	86	108	1.78	0.036 587 6
$\ell_{51,52}$	51	52	1.37	0.044 448 8	$\ell_{108,90}$	108	90	1.41	0.025 256 5
$\ell_{52,83}$	52	83	1.37	0.050 398 3	$\ell_{73,94}$	73	94	2.34	0.300 402
$\ell_{83,53}$	83	53	1.37	0.010 295	$\ell_{75,76}$	75	76	1.18	0.034 472 5
$\ell_{22,53}$	22	53	1.65	0.173 533	$\ell^1_{79,79}$	79	78	1.05	0.000 224 327
$\ell_{53,22}$	53	22	1.1	0.176 873	$\ell^2_{79,78}$	79	78	1.05	0.000 293 558
$\ell_{53,50}$	53	50	1.41	0.121 253	$\ell_{80,77}$	80	77	1.83	0.000 195 053
$\ell_{54,50}$	54	50	1.37	0.014 499 2	$\ell_{81,77}$	81	77	1.78	0.074 178 4
$\ell_{15,54}$	15	54	1.37	0.031 842 3	$\ell_{81,100}$	81	100	6.594	0.157 96
$\ell_{22,23}$	22	23	1.07	0.007 407 24	$\ell_{50,88}$	50	88	5.72	0.054 706
$\ell_{100,103}$	100	103	1.83	0.065 257 4	$\ell_{100,88}$	100	88	6.03	0.031 814 7
$\ell_{68,100}$	68	100	0.6	0.244 348	$\ell_{85,87}$	85	87	1.41	0.022 285 4
$\ell_{75,81}$	75	81	1.78	0.028 597 8	$\ell_{87,86}$	87	86	1.41	0.006 305 32
$\ell_{76,78}$	76	78	1.05	0.087 336 9	$\ell_{86,89}$	86	89	1.37	0.000 303 827
$\ell_{77,69}$	77	69	1.83	0.123 439	$\ell_{90,12}$	90	12	1.41	0.037 262 3
$\ell_{69,68}$	69	68	1.83	0.210 708	$\ell_{43,17}$	43	17	9.02	0.004 061 05
$\ell_{78,68}$	78	68	1.05	0.364 066	$\ell_{5,4}$	5	4	6.346	0.004 639 1
$\ell_{68,100}$	68	100	1.78	0.243 15	$\ell_{7,70}$	7	70	4.76	0.004 903 9
$\ell_{100,5}$	100	5	5.97	0.063 304 8	$\ell_{70,72}$	70	72	5.91	0.006 341 8
$\ell_{2,5}$	2	5	1.78	0.040 338	$\ell_{7,72}$	7	72	5.91	0.005 224 31
$\ell_{101,2}$	101	2	1.78	0.038 200 4	$\ell^1_{91,72}$	91	72	5.95	0.000 111 497
$\ell_{84,85}$	84	85	1.41	0.008 282 5	$\ell^2_{91,72}$	91	72	5.95	0.000 103 121
$\ell_{85,103}$	85	103	1.41	0.033 652 8	$\ell^3_{91,72}$	91	72	5.95	0.000 111 601
$\ell_{16,90}$	16	90	1.41	0.003 719 28	$\ell^1_{11,48}$	11	48	5.95	0.000 112 9
$\ell_{102,103}$	102	103	1.78	0.008 239 12	$\ell^2_{11,48}$	11	48	5.95	0.000 107 982
$\ell_{103,104}$	103	104	1.78	0.012 291 4	$\ell^3_{11,48}$	11	48	5.95	0.000 113 376
$\ell_{104,101}$	104	101	1.78	0.037 308 3	$\ell^4_{11,48}$	11	48	5.95	0.000 125 214
$\ell_{105,101}$	105	101	1.78	0.008 935 66	$\ell_{12,72}$	12	72	8.57	0.010 712 4
$\ell_{3,5}$	3	5	6.1825	0.005 279 7	$\ell^1_{92,18}$	92	18	9.02	8.65628×10^{-5}
$\ell_{3,4}$	3	4	4.51	0.004 967 22	$\ell^2_{92,18}$	92	18	9.02	8.27359×10^{-5}
$\ell_{4,17}$	4	17	9.02	0.014 100 1	$\ell_{47,12}$	47	12	7.14	0.006 557 03
$\ell_{17,12}$	17	12	5.95	0.011 774	$\ell_{18,47}$	18	47	9.02	0.006 764 96
$\ell_{18,19}$	18	19	11.43	0.000 886 174	$\ell_{5,48}$	5	48	8.57	0.034 390 9
$\ell_{19,48}$	19	48	11.43	0.001 952 96	$\ell_{17,72}$	17	72	5.95	0.023 352 2
$\ell_{64,25}$	64	25	1.41	0.012 852 9	$\ell_{12,26}$	12	26	1.41	0.048 703 1
$\ell_{64,35}$	64	35	1.41	0.053 010 5	$\ell_{22,23}$	22	23	1.07	0.007 351 23
$\ell_{20,33}$	20	33	1.05	0.079 634 5	$\ell_{97,98}$	97	98	1.83	0.033 636 2
$\ell_{66,20}$	66	20	1.05	0.087 364 5	$\ell_{93,98}$	93	98	1.37	0.112 023
$\ell_{20,98}$	20	98	1.05	0.167 376	$\ell^1_{93,27}$	93	27	1.41	0.037 295 4
$\ell_{20,65}$	20	65	1.05	0.085 265 5	$\ell^2_{93,27}$	93	27	1.41	0.037 236 7
$\ell_{109,67}$	109	67	1.05	0.049 963 3	$\ell_{50,10}$	50	10	4.29	0.040 391 3
$\ell_{32,57}$	32	57	2.34	0.020 389	$\ell_{10,58}$	10	58	5.91	0.068 501 9
$\ell_{57,58}$	57	58	4.57	0.031 453 1	$\ell_{1,10}$	1	10	5.91	0.000 114 64
$\ell_{57,60}$	57	60	2.34	0.049 15	$\ell_{50,44}$	50	44	4.29	0.027 401
$\ell_{55,58}$	55	58	2.89	0.076 117	$\ell_{94,95}$	94	95	2.34	0.010 736 6
$\ell_{58,37}$	58	37	2.4	0.133 382	$\ell_{94,96}$	94	96	2.34	0.000 604 838
$\ell^1_{32,58}$	32	58	2.34	0.148 584	$\ell_{73,74}$	73	74	2.34	0.008 441 74
$\ell^2_{32,58}$	32	58	2.34	0.144 57	$\ell_{28,57}$	28	57	0.96	0.274 413
$\ell_{58,30}$	58	30	2.34	0.279 918	$\ell_{28,29}$	28	29	0.73	0.065 891 4
$\ell_{28,56}$	28	56	0.73	0.254 476	$\ell_{28,63}$	28	63	0.73	0.242 089

Table D.3: South Australian Lines B

name String7	from node Int64	to node Int64	line lim Float64	reactance Float64
$\ell_{106,56}$	106	56	0.73	0.000 111 309
$\ell_{107,56}$	107	56	0.73	0.059 948 1
$\ell_{38,107}$	38	107	0.73	0.001 707 09
$\ell_{8,9}$	8	9	2.34	0.025 062 7
$\ell_{8,30}$	8	30	2.34	0.071 304 9
$\ell_{8,36}$	8	36	2.34	0.187 601
$\ell_{65,99}$	65	99	1.05	0.004 151 63
$\ell_{65,64}$	65	64	0.91	0.116 63
$\ell_{66,67}$	66	67	1.05	0.133 171
$\ell^1_{58,61}$	58	61	2.89	0.001 523 28
$\ell^2_{58,61}$	58	61	2.89	0.001 548 03
$\ell^1_{58,62}$	58	62	2.38	0.001 751 42
$\ell^2_{58,62}$	58	62	2.38	0.001 681 08
$\ell^1_{59,62}$	59	62	2.38	0.000 136 529
$\ell^2_{59,62}$	59	62	2.38	0.000 119 218
$\ell^3_{59,62}$	59	62	2.38	0.000 110 087
$\ell^4_{59,62}$	59	62	2.38	0.000 107 368
$\ell_{30,31}$	30	31	2.34	0.000 127 731
$\ell^1_{36,37}$	36	37	2.34	0.005 799 13
$\ell^2_{36,37}$	36	37	2.34	0.005 894 39
$\ell_{37,49}$	37	49	2.34	0.299 386
$\ell_{73,58}$	73	58	2.34	0.173 901
$\ell_{88,12}$	88	12	1.41	0.020 292 3
$\ell_{12,50}$	12	50	4.51	0.074 881 5
$\ell_{88,6}$	88	6	11.82	0.019 556
$\ell_{42,64}$	42	64	1.41	0.077 925 8
$\ell_{55,42}$	55	42	1.41	0.037 407 7
$\ell_{39,34}$	39	34	1.41	0.002 226 98
$\ell_{39,42}$	39	42	1.41	0.028 332 9
$\ell_{45,12}$	45	12	4.51	0.054 434 5
$\ell_{64,58}$	64	58	4.29	0.046 264 1
$\ell_{82,81}$	82	81	2.74	0.076 583 3
$\ell_{78,82}$	78	82	2.51	0.000 103 524
$\ell_{93,13}$	93	13	2.89	0.000 172 868
$\ell_{12,21}$	12	21	2.89	0.017 116 2
$\ell_{13,21}$	13	21	2.89	0.000 784 151
$\ell_{21,55}$	21	55	2.89	0.051 751 2
$\ell_{26,93}$	26	93	1.37	0.019 355
$\ell_{71,48}$	71	48	5.95	0.009 318 49
$\ell_{6,5}$	6	5	11.82	0.010 115 9
$\ell_{44,41}$	44	41	5.91	0.015 618 5
$\ell_{40,41}$	40	41	5.91	0.000 109 345
$\ell_{46,24}$	46	24	5.0	0.014 474 7
$\ell_{64,45}$	64	45	4.51	0.042 959 6
$\ell_{45,24}$	45	24	4.51	7.20285×10^{-5}
$\ell_{14,98}$	14	98	1.37	0.009 358 39
$\ell_{32,57}$	32	57	1.17	0.020 417 6
$\ell_{41,58}$	41	58	5.91	0.066 113 5
$\ell_{97,55}$	97	55	1.83	0.092 088 1

Table D.4: South Australian Generators

gen String15	node Int64	index Int64	stage Int64	cost Float64	low lim Float64	up lim Float64	stoch Int64	storage Int64	cap Int64
AGLHAL	1	1	1	105.2	0.0	180.0	0	0	0
BLUFF1	1	2	2	0.0	0.0	53.0	1	0	0
LONSDALE	3	3	1	129.7	0.0	21.0	0	0	0
PTSTAN1	3	4	1	127.3	0.0	58.0	0	0	0
DRYCGT1	7	5	1	110.4	0.0	52.0	0	0	0
DRYCGT2	7	6	1	110.4	0.0	52.0	0	0	0
DRYCGT3	7	7	1	110.4	0.0	52.0	0	0	0
HDWF1	10	8	2	0.0	0.0	102.0	1	0	0
HDWF2	10	9	2	0.0	0.0	102.0	1	0	0
HALLWF1	10	10	2	0.0	0.0	95.0	1	0	0
TORRB3	11	11	1	73.7	0.0	200.0	0	0	0
TORRB4	11	12	1	73.7	0.0	200.0	0	0	0
OSB-AG	11	13	1	63.8	0.0	180.0	0	0	0
TORRB2	11	14	1	73.7	0.0	200.0	0	0	0
TORRB1	11	15	1	73.7	0.0	200.0	0	0	0
WATERLWF	14	16	2	0.0	0.0	130.0	1	0	0
QPS3	19	17	1	85.0	0.0	24.0	0	0	0
QPS2	19	18	1	85.0	0.0	24.0	0	0	0
QPS5	19	19	1	85.0	0.0	128.0	0	0	0
QPS4	19	20	1	85.0	0.0	24.0	0	0	0
QPS1	19	21	1	85.0	0.0	24.0	0	0	0
ANGAST1	27	22	1	129.7	0.0	50.0	0	0	0
CLEMGPWF	34	23	2	0.0	0.0	57.0	1	0	0
NBHWF1	40	24	2	0.0	0.0	132.0	1	0	0
HALLWF2	44	25	2	0.0	0.0	71.0	1	0	0
SNOWSTH1	46	26	2	0.0	0.0	126.0	1	0	0
LKBONNY3	79	27	2	0.0	0.0	39.0	1	0	0
SNUG1	79	28	1	129.2	0.0	63.0	0	0	0
LADBROK2	80	29	1	89.9	0.0	40.0	0	0	0
LADBROK1	80	30	1	89.9	0.0	40.0	0	0	0
LKBONNY2	82	31	2	0.0	0.0	159.0	1	0	0
TORRA2	91	32	1	78.4	0.0	120.0	0	0	0
TORRA1	91	33	1	78.4	0.0	120.0	0	0	0
TORRA3	91	34	1	78.4	0.0	120.0	0	0	0
TORRA4	91	35	1	78.4	0.0	120.0	0	0	0
PPCCGT	92	36	1	61.2	0.0	478.0	0	0	0
MINTARO	97	37	1	101.5	0.0	90.0	0	0	0
SNOWNTH1	99	38	2	0.0	0.0	144.0	1	0	0
SNOWTWN1	99	39	2	0.0	0.0	99.0	1	0	0
POR01	106	40	1	126.8	0.0	50.0	0	0	0
POR03	106	41	1	126.8	0.0	23.0	0	0	0
IC_Ml_IM	22	42	2	6000.0	0.0	220.0	0	0	0
IC_Ml_EX	22	43	2	0.4	−200.0	0.0	0	1	∞
IC_HEYW_IM	81	44	2	6000.0	0.0	600.0	0	0	0
IC_HEYW_EX	81	45	2	0.4	−500.0	0.0	0	1	∞

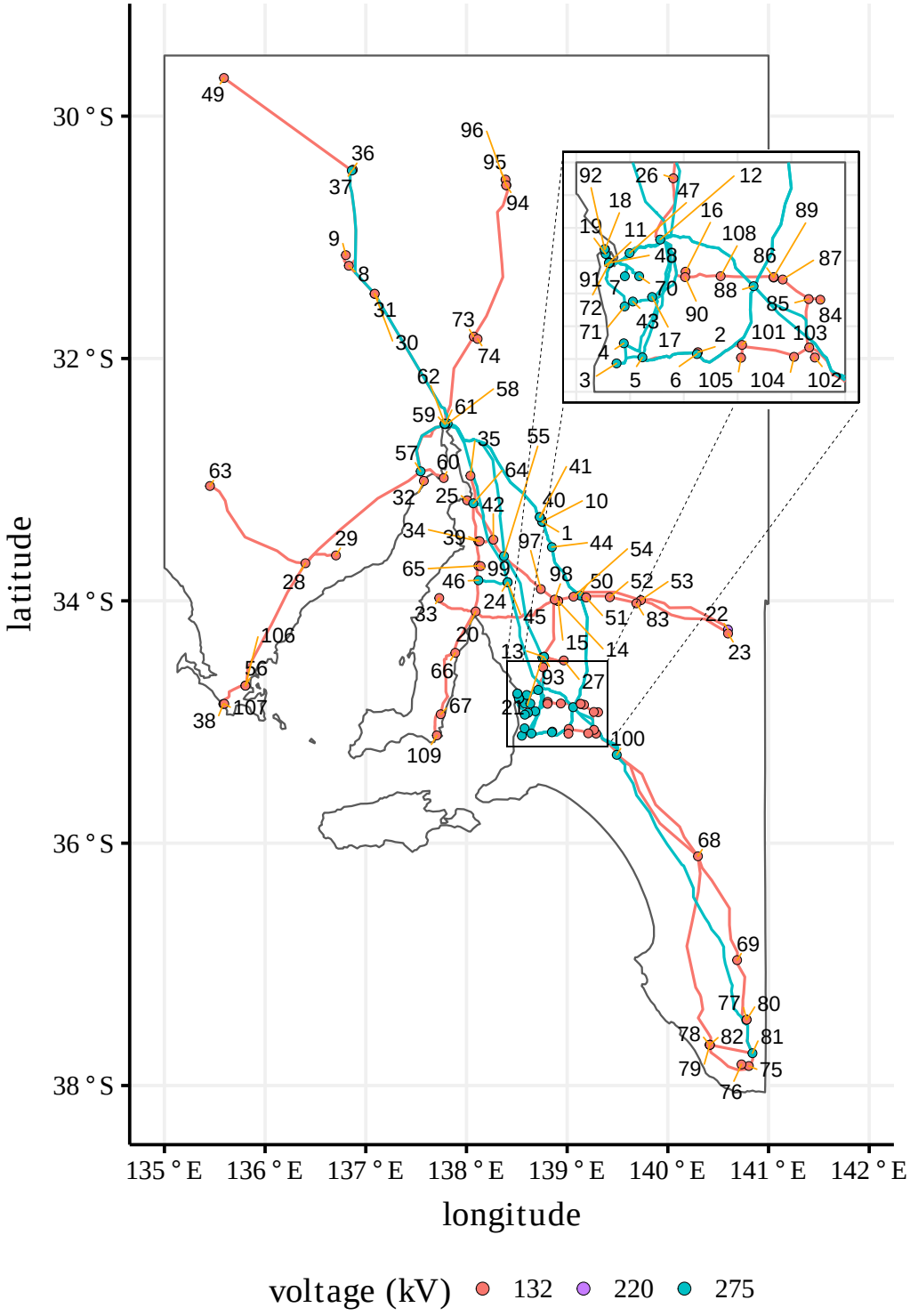


Figure D.1: Graph of the South Australian transmission network with node IDs. The network coordinates and the South Australian boundaries are obtained from the open NEM model [114].

Bibliography

[1] H. K. Alfares and M. Nazeeruddin. "Electric load forecasting: Literature survey and classification of methods". In: *International journal of systems science* 33.1 (2002), pp. 23–34.

[2] M. Ali Khan. "Perfect Competition". In: *The New Palgrave Dictionary of Economics*. London: Palgrave Macmillan UK, 2017, pp. 1–15. DOI: 10.1057/978-1-349-95121-5_1633-2. URL: https://doi.org/10.1057/978-1-349-95121-5_1633-2.

[3] Aurecon group. *Hornsdale Power Reserve Year 2 Technical and Market Impact Case Study*. Tech. rep. 2020. URL: https://www.aurecongroup.com/markets/energy/hornsdale-power-reserve-impact-study.

[4] Australian Bureau of Statistics. *1270.0.55.001 - Statistical Area Level 2 (SA2)*. 2016. URL: https://www.abs.gov.au/ausstats/abs@.nsf/lookup/by%20subject/1270.0.55.001~july%202016~main%20features~statistical%20area%20level%202%20(sa2)~10014 (visited on 06/25/2022).

[5] Australian Energy Market Commission. *Fact sheet: How the spot market works*. 2017. URL: https://www.aemc.gov.au/sites/default/files/content/d6cc8e9d-6a9f-4648-bef7-b25cad5df460/5-Fact-sheet-How-the-spot-market-works.pdf (visited on 07/27/2021).

[6] Australian Energy Market Commission. *Last resort planning power - 2018 review*. Tech. rep. 2019. URL: https://www.aemc.gov.au/sites/default/files/2019-02/FinalReport.pdf.

[7] Australian Energy Market Operator (AEMO). *100 Percent Renewables Study – Modelling Outcomes*. Tech. rep. 2013. URL: http://thebrightfutureproject.weebly.com/uploads/2/7/1/4/27148779/100-percent-renewables-study-modelling-outcomes-report.pdf.

[8] Australian Energy Market Operator (AEMO). *5MS Commencement*. URL: https://aemo.com.au/initiatives/major-programs/past-major-programs/five-minute-settlement/5ms-program-management/5ms-commencement (visited on 01/10/2021).

[9] Australian Energy Market Operator (AEMO). *AEMO — Diagrams and previous maps*. 2019. URL: https://aemo.com.au/en/energy-systems/electricity/national-electricity-market-nem/nem-forecasting-and-planning/forecasting-and-planning-data/diagrams-and-previous-maps (visited on 06/18/2021).

[10] Australian Energy Market Operator (AEMO). *AEMO — Energy Explained: Voltage*. 2020. URL: https://aemo.com.au/newsroom/energy-live/energy-explained-voltage (visited on 08/04/2020).

[11] Australian Energy Market Operator (AEMO). *AEMO — NEM web data*. 2018. URL: https://www.aemo.com.au/energy-systems/electricity/national-electricity-market-nem/data-nem/market-data-nemweb (visited on 06/17/2021).

[12] Australian Energy Market Operator (AEMO). *AEMO — NTNDP database*. 2016. URL: https://www.aemo.com.au/energy-systems/major-publications/integrated-system-plan-isp/national-transmission-network-development-plan-ntndp/ntndp-database (visited on 06/17/2021).

[13] Australian Energy Market Operator (AEMO). *AEMO — Transmission Equipment Ratings*. 2021. URL: https://aemo.com.au/en/energy-systems/electricity/national-electricity-market-nem/data-nem/network-data/transmission-equipment-ratings (visited on 06/18/2021).

[14] Australian Energy Market Operator (AEMO). *Demand Terms in EMMS Data Model*. 2021. URL: https://www.aemo.com.au/-/media/files/electricity/nem/security_and_reliability/dispatch/policy_and_process/demand-terms-in-emms-data-model.pdf?la=en&hash=4095438A02E6638F369C4D7CB31F41C3.

[15] Australian Energy Market Operator (AEMO). *Electricity Demand Forcasting Methodology Information Paper*. Tech. rep. 2020.

[16] Australian Energy Market Operator (AEMO). *Fact Sheet The National Electricity Market*. 2017. URL: https://www.aemo.com.au/-/media/Files/Electricity/NEM/National-Electricity-Market-Fact-Sheet.pdf.

[17] Australian Energy Market Operator (AEMO). *Managing frequency in the power system*. 2018. URL: https://aemo.com.au/en/learn/energy-explained/energy-101/managing-frequency-in-the-power-system.

[18] Australian Energy Market Operator (AEMO). *NEM Registration and Exemption List*. 2022. URL: https://aemo.com.au/en/energy-systems/electricity/national-electricity-market-nem/participate-in-the-market/registration (visited on 07/21/2022).

[19] Australian Energy Market Operator (AEMO). *South Australian Electricity Report*. Tech. rep. 2020. URL: https://www.aemo.com.au/-/media/Files/Electricity/NEM/Planning{_}and{_}Forecasting/SA{_}Advisory/2020/2020-South-Australian-Electricity-Report.pdf.

[20] Australian Energy Market Operator (AEMO). *South Australian electricity report 2021*. 2021. URL: https://www.sa.gov.au/topics/energy-and-environment/energy-supply/sas-electricity-supply-and-market (visited on 08/20/2020).

[21] Australian Energy Market Operator (AEMO). *South Australian Electricity Report South Australian Advisory Functions*. Tech. rep. 2019. URL: https://www.aemo.com.au/Electricity/National-Electricity-Market-NEM/Security-and-reliability/Future-.

[22] Australian Energy Regulator (AER). *AER approves South Australia – NSW interconnector regulatory investment test*. 2020. URL: https://www.aer.gov.au/news-release/aer-approves-south-australia--nsw-interconnector-regulatory-investment-test.

[23] Australian Energy Regulator (AER). *Quarterly volume weighted average price by contribution of price bands - South Australia*. 2022. URL: https://www.aer.gov.au/wholesale-markets/wholesale-statistics/quarterly-volume-weighted-average-price-by-contribution-of-price-bands-south-australia.

[24] Australian Energy Regulator (AER). *Quarterly volume weighted average price by contribution of price bands - Victoria*. 2022. URL: https://www.aer.gov.au/wholesale-markets/wholesale-statistics/quarterly-volume-weighted-average-price-by-contribution-of-price-bands-victoria.

[25] Australian Energy Regulator (AER). *State of the Energy Market, Chapter 2, Electricity Transmission*. Tech. rep. 2021.

[26] Australian Government. *Quarterly Update of Australia's National Greenhouse Gas Inventory: December 2019*. 2020. URL: https://www.industry.gov.au/sites/default/files/2020-05/nggi-quarterly-update-dec-2019.pdf.

[27] Australian Renewable Energy Agency (ARENA). *Demand response - Australian Renewable Energy Agency (ARENA)*. 2022. URL: https://arena.gov.au/renewable-energy/demand-response/ (visited on 06/19/2022).

[28] W. Bai, D. Lee, and K. Lee. "Stochastic Dynamic AC Optimal Power Flow Based on a Multivariate Short-Term Wind Power Scenario Forecasting Model". In: *Energies* 10 (Dec. 2017), p. 2138. DOI: 10.3390/en10122138.

[29] K. Baker. "Solutions of DC OPF Are Never AC Feasible". In: *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*. e-Energy '21. Virtual Event, Italy: Association for Computing Machinery, 2021, pp. 264–268. DOI: 10.1145/3447555.3464875. URL: https://doi.org/10.1145/3447555.3464875.

[30] K. A. Baker. *Coordination of Resources across Areas for the Integration of Renewable Generation: Operation, Sizing, and Siting of Storage Devices*. eng. 2014.

[31] B. Banerjee, D. Jayaweera, and S. Islam. "Smart Power Systems and Renewable Energy System Integration". In: ed. by D. Jayaweera. Vol. 57. Studies in Systems, Decision and Control. Cham: Springer International Publishing, 2016, pp. 15–28. DOI: 10.1007/978-3-319-30427-4_1. URL: http://link.springer.com/10.1007/978-3-319-30427-4http://dx.doi.org/10.1007/978-3-319-30427-4{_}1.

[32] G. Bayraksan and D. P. Morton. "Assessing Solution Quality in Stochastic Programs via Sampling". In: *Decision Technologies and Applications* (2009), pp. 102–122. DOI: 10.1287/educ.1090.0065. URL: http://pubsonline.informs.orghttp//www.informs.org.

[33] J. A. Bazerque. "Stochastic Optimization of Power Systems with Risk Constraints And Sparsely Distributed Storage". eng. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 3824–3828.

[34] A. Ben-Israel and T. N. E. Greville. *Generalized inverses: theory and applications*. 2nd ed. CMS books in mathematics. Springer, 2003.

[35] J. F. Benders. "Partitioning procedures for solving mixed-variables programming problems". In: *Numerische Mathematik* 4.1 (1962), pp. 238–252. DOI: 10.1007/BF01386316. URL: https://link.springer.com/article/10.1007/BF01386316.

[36] A. R. Bergen and V. Vittal. *Power Systems Analysis*. 2nd ed. Prentice-Hall, 2000.

[37] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. "Julia: A Fresh Approach to Numerical Computing". In: *SIAM Review* 59.1 (2017), pp. 65–98. DOI: 10.1137/141000671. URL: https://epubs.siam.org/doi/10.1137/141000671.

[38] M. Biel and M. Johansson. "Efficient Stochastic Programming in Julia". In: *INFORMS Journal on Computing* (2022). DOI: 10.1287/ijoc.2022.1158. arXiv: 1909.10451. URL: https://arxiv.org/abs/1909.10451v3.

[39] D. Bienstock. *Electrical Transmission System Cascades and Vulnerability: An Operations Research Viewpoint*. Society for Industrial and Applied Mathematics Publications, 2015.

[40] D. Bienstock, M. Chertkov, and S. Harnett. "Chance-constrained optimal power flow: Risk-Aware Network Control under Uncertainty". In: *SIAM Review* 56.3 (2014), pp. 461–495. DOI: 10.1137/130910312. URL: http://www.siam.org/journals/ojsa.php.

[41] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, New York, NY, 1997. DOI: https://doi.org/10.1007/978-1-4614-0237-4.

[42] P. P. Biswas, P. N. Suganthan, and G. A. Amaratunga. "Optimal power flow solutions incorporating stochastic wind and solar power". In: *Energy Conversion and Management* 148 (2017), pp. 1194–1207. DOI: 10.1016/j.enconman.2017.06.071.

[43] A. Blakers, B. Lu, and M. Stocks. "100% renewable electricity in Australia". In: *Energy* 133 (2017), pp. 471–482. DOI: 10.1016/j.energy.2017.05.168.

[44] H. Blanco and A. Faaij. *A review at the role of storage in energy systems with a focus on Power to Gas and long-term storage*. 2018. DOI: 10.1016/j.rser.2017.07.062.

[45] G. E. P. Box and G. M. Jenkins. *Time series analysis: forecasting and control*. 1970.

[46] S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[47] P. J. Brockwell and R. A. Davis. *Time Series: Theory and Methods*. Springer Series in Statistics. New York, NY: Springer New York, 1991. DOI: 10.1007/978-1-4419-0320-4. URL: http://link.springer.com/10.1007/978-1-4419-0320-4.

[48] B. G. Brown, R. W. Katz, and A. H. Murphy. "Time Series Models to Simulate and Forecast Wind Speed and Wind Power". In: *Journal of Climate and Applied Meteorology* 23.8 (1984), pp. 1184–1195. URL: http://www.jstor.org/stable/26181389.

[49] J. C. P. Bullas. *Energy Storage Requirements for the South Australian Grid*. Tech. rep. Honours Thesis, The University of Adelaide, 2018.

[50] Bureau of Meteorology. *Weather Station Directory*. URL: http://www.bom.gov.au/climate/data/stations/ (visited on 03/14/2022).

[51] F. Cebulla, J. Haas, J. Eichman, W. Nowak, and P. Mancarella. "How much electrical energy storage do we need? A synthesis for the U.S., Europe, and Germany". In: *Journal of Cleaner Production* 181 (2018), pp. 449–459. DOI: 10.1016/j.jclepro.2018.01.144.

[52] S. Chatzivasileiadis. "Lecture Notes on Optimal Power Flow (OPF)". In: *ArXiv pre-print* (2018). arXiv: 1811.00943. URL: http://arxiv.org/abs/1811.00943.

[53] S. Chatzivasileiadis. *Optimization in Modern Power Systems DTU Course 31765 Lecture Notes*. Tech. rep. 2018. arXiv: 1811.00943v1.

[54] P. Chen, T. Pedersen, B. Bak-Jensen, and Z. Chen. "ARIMA-based time series model of stochastic wind power generation". In: *IEEE Transactions on Power Systems* 25.2 (2010), pp. 667–676. DOI: 10.1109/TPWRS.2009.2033277.

[55] G. B. Dantzig and P. Wolfe. "Decomposition Principle for Linear Programs". In: *Operations Research* 8.1 (1960), pp. 101–111. DOI: 10.1287/opre.8.1.101.

[56] I. Dunning, J. Huchette, and M. Lubin. "JuMP: A modeling language for mathematical optimization". In: *SIAM Review* 59.2 (2017), pp. 295–320. DOI: 10.1137/15M1020575. arXiv: 1508.01982.

[57] R. Durret. *Probability: theory and examples*. Cambridge Univeristy Press, 1991.

[58] ElectraNet. *ElectraNet, About the battery*. URL: https://www.escri-sa.com.au/about/.

[59] ElectraNet. *Generator Technical and Cost Parameters*. 2020. URL: https://www.electranet.com.au/wp-content/uploads/projects/2016/11/508986-REP-ElectraNet-Generator-Technical-And-Cost-Parameters-23July2020.pdf.

[60] B. Elliston, I. MacGill, and M. Diesendorf. "Least cost 100% renewable electricity scenarios in the Australian National Electricity Market". In: *Energy Policy* 59 (2013), pp. 270–282. DOI: 10.1016/j.enpol.2013.03.038.

[61] Energy Regulator Australian. *Murraylink - electricity transmission interconnector*. URL: https://www.aer.gov.au/networks-pipelines/service-providers-assets/murraylink-electricity-transmission-interconnector.

[62] C. Eyisi, A. S. Al-Sumaiti, K. Turitsyn, and Q. Li. "Mathematical Models for Optimization of Grid-Integrated Energy Storage Systems". In: (2019). arXiv: 1901.06374. URL: http://arxiv.org/abs/1901.06374.

[63] S. Fan and R. J. Hyndman. "Short-term load forecasting based on a semi-parametric additive model". In: *IEEE Transactions on Power Systems* 27.1 (2012), pp. 134–141. DOI: 10.1109/TPWRS.2011.2162082.

[64] V. M. Guerrero. "Time-series analysis supported by power transformations". In: *Journal of Forecasting* 12.1 (1993), pp. 37–48. DOI: 10.1002/for.3980120104.

[65] Y. Guo, K. Baker, E. Dall'Anese, Z. Hu, and T. H. Summers. "Data-Based Distributionally Robust Stochastic Optimal Power Flow-Part II: Case Studies". eng. In: *IEEE transactions on power systems* 34.2 (2019), pp. 1493–1503.

[66] H. Hamedmoghadam, N. Joorabloo, and M. Jalili. "Australia's long-term electricity demand forecasting using deep neural networks". In: (2018). arXiv: 1801.02148. URL: http://arxiv.org/abs/1801.02148.

[67] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I*. Grundlehren der mathematischen Wissenschaften. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993. DOI: 10.1007/978-3-662-02796-7. URL: http://link.springer.com/10.1007/978-3-662-02796-7.

[68] Hornsdale Power Reserve, Neoen. *Hornsdale Power Reserve — South Australia's Big Battery*. URL: https://hornsdalepowerreserve.com.au/ (visited on 06/27/2022).

[69] C. M. Hurvich and C.-L. Tsai. "Regression and time series model selection in small samples". In: *Biometrika* 76.2 (1989), pp. 297–307. DOI: 10.1093/biomet/76.2.297. URL: https://academic.oup.com/biomet/article-lookup/doi/10.1093/biomet/76.2.297.

[70] R. Hyndman and G. Athanasopoulos. *Forecasting: Principles and Practice (3rd ed)*. 2021. URL: https://otexts.com/fpp3/ (visited on 12/30/2021).

[71] R. J. Hyndman and Y. Khandakar. "Automatic Time Series Forecasting: The forecast Package for R". In: *Journal of statistical software* 27.3 (2008).

[72] Iberdrola Energy. *Lake Bonney Wind Farms — Iberdrola Energy*. URL: https://www.infigenenergy.com/our-assets/owned-renewable-energy-assets/lake-bonney/ (visited on 06/27/2022).

[73] R. A. Jabr, S. Karaki, and J. A. Korbane. "Robust Multi-Period OPF with Storage and Renewables". In: *IEEE Transactions on Power Systems* 30.5 (2015), pp. 2790–2799. DOI: 10.1109/TPWRS.2014.2365835.

[74] R. Kannan, J. R. Luedtke, and L. A. Roald. "Stochastic DC Optimal Power Flow With Reserve Saturation". In: *Electric Power Systems Research* 189 (2019). arXiv: 1910.04667. URL: http://arxiv.org/abs/1910.04667.

[75] S. Kharel and B. Shabani. "Hydrogen as a Long-Term Large-Scale Energy Storage Solution to Support Renewables". In: *Energies* 11.10 (2018), p. 2825. DOI: 10.3390/en11102825. URL: http://www.mdpi.com/1996-1073/11/10/2825.

[76] D. Kwiatkowski, P. C. Phillips, P. Schmidt, and Y. Shin. "Testing the null hypothesis of stationarity against the alternative of a unit root. How sure are we that economic time series have a unit root?" In: *Journal of Econometrics* 54.1-3 (1992), pp. 159–178. DOI: 10.1016/0304-4076(92)90104-Y.

[77] M. Lenzen, B. McBain, T. Trainer, S. Jütte, O. Rey-Lescure, and J. Huang. "Simulating low-carbon electricity supply for Australia". In: *Applied Energy* 179 (2016), pp. 553–564. DOI: 10.1016/j.apenergy.2016.06.151.

[78] B. Lu, M. Stocks, A. Blakers, and K. Anderson. "Geographic information system algorithms to locate prospective sites for pumped hydro energy storage". In: *Applied Energy* 222 (2018), pp. 300–312. DOI: 10.1016/j.apenergy.2018.03.177.

[79] A. Madansky. "Inequalities for Stochastic Linear Programming Problems". In: *Management Science (pre-1986); Linthicum* 6.2 (1960), p. 197. URL: http://proxy.library.adelaide.edu.au/login?url=https://www.proquest.com/scholarly-journals/inequalities-stochastic-linear-programming/docview/205811448/se-2?accountid=8203.

[80] L. Magnano and J. W. Boland. *Generation of synthetic sequences of electricity demand: Application in South Australia*. 2007. DOI: 10.1016/j.energy.2007.04.001.

[81] W. K. Mak, D. P. Morton, and R. K. Wood. "Monte Carlo bounding techniques for determining solution quality in stochastic programs". In: *Operations Research Letters* 24.1 (1999), pp. 47–56. DOI: 10.1016/S0167-6377(98)00054-6.

[82] S. Martin. *Australia commits to 2050 net zero emissions plan but with no detail and no modelling*. 2021. URL: https://www.theguardian.com/australia-news/2021/oct/26/scott-morrison-says-australia-2050-net-zero-emissions-plan-based-on-choices-not-mandates (visited on 07/31/2022).

[83] J. Matousek and B. Gärtner. *Understanding and Using Linear Programming*. 1st ed. Springer-Verlag Berlin Heidelberg, 2007. DOI: 10.1007/978-3-540-30717-4.

[84] M. Mazengarb. *South Australia minister aiming for 100 per cent renewables before 2030*. 2020. URL: https://reneweconomy.com.au/south-australia-minister-aiming-for-100-per-cent-renewables-before-2030-2030/.

[85] O. Megel, J. L. Mathieu, and G. Andersson. "Hybrid Stochastic-Deterministic Multiperiod DC Optimal Power Flow". In: *IEEE Transactions on Power Systems* 32.5 (2017), pp. 3934–3945. DOI: 10.1109/TPWRS.2017.2651409.

[86] M. A. Mirzaei, A. S. Yazdankhah, B. Mohammadi-Ivatloo, M. Marzband, M. Shafie-khah, and J. P. Catalão. "Stochastic network-constrained co-optimization of energy and reserve products in renewable energy integrated power and gas networks with energy storage system". eng. In: *Journal of cleaner production* 223 (2019), pp. 747–758.

[87] B. Mountain. *How the South Australia blackout occurred: what the data tells us — RenewEconomy*. 2016. URL: https://reneweconomy.com.au/how-the-south-australia-blackout-occurred-what-the-data-tells-us-65806/ (visited on 07/27/2021).

[88] J. R. Munkres. *Topology*. 2nd. Upper Saddle River, NJ. Prentice Hall, Inc., 2000.

[89] N. Nazir and M. Almassalkhi. "Stochastic multi-period optimal dispatch of energy storage in unbalanced distribution feeders". eng. In: *Electric power systems research* 189.C (2020), pp. 106783–.

[90] T. Nesti, A. Zocca, and B. Zwart. "Emergent failures and cascades in power grids: a statistical physics perspective". In: *Physical Review Letters* 120.25 (2017). DOI: 10.1103/PhysRevLett.120.258301. arXiv: 1709.10166. URL: http://arxiv.org/abs/1709.10166http://dx.doi.org/10.1103/PhysRevLett.120.258301.

[91] M. O'Hara-Wild, R. Hyndman, E. Wang, G. Caceres, T.-G. Hensel, and T. Hyndman. *fable: Forecasting Models for Tidy Time Series*. 2021. URL: https://cran.r-project.org/package=fable.

[92] M. O'Hara-Wild, R. Hyndman, E. Wang, D. Cook, T. Talagala, and L. Chhay. *feasts: Feature Extraction and Statistics for Time Series*. 2021.

[93] K. Orr and B. Allan. *Electricity Transmission Lines*. 2015. DOI: http://dx.doi.org/10.4225/25/553DC89AD105C.

[94] K. Orr and B. Allan. *Electricity Transmission Substations*. 2015. DOI: http://dx.doi.org/10.4225/25/5549B03B2401E.

[95] K. Orr and N. Skeers. *Power Stations*. 2014. DOI: http://dx.doi.org/10.4225/25/544EE47D2C1DE.

[96] H. Pandzic, Y. Wang, T. Qiu, Y. Dvorkin, and D. S. Kirschen. "Near-Optimal Method for Siting and Sizing of Distributed Storage in a Transmission Network". eng. In: *IEEE transactions on power systems* 30.5 (2015), pp. 2288–2300.

[97] G. Parkinson. *South Australia sets stunning new record, solar meets 106 pct of demand — RenewEconomy*. 2021. URL: https://reneweconomy.com.au/solar-meets-all-of-south-australia-demand-and-more/ (visited on 07/06/2022).

[98] D. Pozo, J. Contreras, and E. E. Sauma. "Unit Commitment with Ideal and Generic Energy Storage Units". In: *IEEE Transactions on Power Systems* 29.6 (2014), pp. 2974–2984. DOI: 10.1109/TPWRS.2014.2313513.

[99] Project EnergyConnect. *Project Overview fact sheet*. 2022. URL: https://www.projectenergyconnect.com.au/moreInformation.php (visited on 07/17/2022).

[100] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2020. URL: https://www.r-project.org/.

[101] A. Ruszczyński. *Nonlinear Optimization*. Princeton University Press, 2006. DOI: 10.2307/j.ctvcm4hcj. URL: http://www.jstor.org/stable/10.2307/j.ctvcm4hcj.

[102] SafeWork SA, Government of South Australia. *Public holidays*. URL: https://www.safework.sa.gov.au/resources/public-holidays (visited on 03/14/2022).

[103] B. Skinner. *Barker Inlet: A new technology responding to the market*. 2020. URL: https://www.energycouncil.com.au/analysis/barker-inlet-a-new-technology-responding-to-the-market/ (visited on 07/21/2022).

[104] R. Standing. *Energy Storage Requirements for the South Australian Grid*. Tech. rep. Honours Thesis, The University of Adelaide.

[105] The Commonwealth Scientific and Industrial Research Organisation (CSIRO). *Australia installs record-breaking number of rooftop solar panels - CSIRO*. 2021. URL: https://www.csiro.au/en/news/news-releases/2021/australia-installs-record-breaking-number-of-rooftop-solar-panels (visited on 07/06/2022).

[106] Time and Date AS. *Weather in October 2017 in Adelaide, South Australia, Australia*. 2017. URL: https://www.timeanddate.com/weather/australia/adelaide/historic?month=10{\&}year=2017 (visited on 03/14/2022).

[107] R. Van Slyke and R. Wets. "L-Shaped Linear Programs With Applications to Optimal Control and Stochastic Programming". In: *SIAM Journal on Applied Mathematics* 17.4 (1969), pp. 638–663. DOI: 10.1137/0117061. URL: https://epubs.siam.org/page/terms.

[108] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Fourth. 2002.

[109] D. Washington and B. Seibert. *Blackout report leaves renewables debate dangling in the breeze —InDaily*. 2016. URL: https://indaily.com.au/news/local/2016/10/05/wind-farms-contributed-to-state-wide-blackout-interim-report/ (visited on 12/14/2022).

[110] D. Weeraddana, N. L. D. Khoa, L. O. Neil, W. Wang, and C. Cai. "Energy consumption forecasting using a stacked nonparametric Bayesian approach". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12461 LNAI (2020), pp. 19–35. arXiv: 2011.05519. URL: http://arxiv.org/abs/2011.05519.

[111] F. Willsmore. *Thesis: When the Wind Blows: Vulnerability of Power Networks with High Wind Penetration*. 2020.

[112] S. Wogrin and D. F. Gayme. "Optimizing Storage Siting, Sizing, and Technology Portfolios in Transmission-Constrained Networks". eng. In: *IEEE transactions on power systems* 30.6 (2015), pp. 3304–3313.

[113] A. J. Wood, B. F. Wollenberg, and G. B. Sheblé. *Power generation, operation, and control*. 3rd ed. Somerset: Wiley, 2014.

[114] A. K. Xenophon. *Geospatial Modelling of Australia's National Electricity Market*. 2018. URL: https://github.com/akxen/egrimod-nem.

[115] A. Xenophon and D. Hill. "Open grid model of Australia's national electricity market allowing backtesting against historic data". In: *Scientific Data* 5.1 (2018), pp. 1–21. DOI: 10.1038/sdata.2018.203. URL: www.nature.com/scientificdata.

[116] D. Yacar, D. A. Tejada-Arango, and S. Wogrin. "Storage allocation and investment optimisation for transmission-constrained networks considering losses and high renewable penetration". eng. In: *IET renewable power generation* 12.16 (2018), pp. 1949–1956.

[117] M. S. Ziegler, J. M. Mueller, G. D. Pereira, J. Song, M. Ferrara, Y. M. Chiang, and J. E. Trancik. "Storage Requirements and Costs of Shaping Renewable Energy Toward Grid Decarbonization". In: *Joule* 3.9 (2019), pp. 2134–2153. DOI: 10.1016/j.joule.2019.06.012.