NTNU
Norwegian University of
Science and Technology

Tackling Variability of Renewable Energy with Stochastic Optimization of Energy System Storage

Solving a Stochastic, Multistage AC Optimal
Power Flow problem with the Stochstic
Quasi-Gradient Method

Sondre Harbo

Industrial Economics and Technology Management
Submission date: August 2017
Supervisor: Alexei A. Gaivoronski, IØT

Norwegian University of Science and Technology
Department of Industrial Economics and Technology Management

Preface

This master thesis was written as conclusion of the student's time with the Industrial Economics and Technology Management program with a specialization in Managerial Economics and Operations Research at the Norwegian University of Science and Technology, NTNU Trondheim.

The thesis is evidently a product of the student's academic journey, with strong elements of both mathematical optimization and power system analysis, but also finance. This because the student has not only taken the subjects and courses required for the specialization profile in Managerial Economics and Operation Research, but also taken most of the courses offered by the department for the students with specialization in Finance and even been helping with teaching a few of these to younger students. Moreover, the technical background of the student, as part of the Industrial Engineering program, is within Energy and Environmental Engineering with specialization in Power Systems. Furthermore, the student has also been influenced by his exchange year at ETH, Zrich, where an interest in complexity science and risk was sparked and the student was exposed to how tools and techniques from one field might be able to solve unaddressed problems in other disciplines.

Hence, the idea behind this thesis is to use optimization methods and ideas developed in finance, a field in which dynamic and stochastic considerations have been a researched for many years, and apply them to the problem of rising volatility and energy storage in electrical power systems. Not only are the many methods of finance dealing with volatility of conceptual interest when addressing renewable energy variability, but the combination of the subjects lets the student apply and put to the test the breath of his background knowledge as well as dealing with some of the critical issues related to the solution of the climate disruption problems.

In the student's mind, the field of Operation Research is the bridge for applying concepts from finance on volatility onto power system analysis of optimal operation and stability. Specifically, methods of Stochastic Programming, Stochastic Optimization and Dynamic Programming is of great interest as they address the stochastic and dynamic elements related to renewable energy variability and energy storage. However, there also exists a methodology, know as the Stochastic Quasi-Gradient (SQG) method - partly developed and implemented by the supervisor of this thesis, professor Alexei Gaivoronski - that addresses both the dynamic and stochastic aspect, and additionally is in theory a faster solution approach than comparable heuristics and exact methods. This thesis is consequently focused on implementing the SQG method for a simulation of a energy system with energy storage and stochastic generation.

It might be of interest to know that the student has applied for admission to the Energy and Environmental Engineering program at NTNU, and plans to do another master thesis on this same topic. Hence, further work will soon be done on the topic, to provide more examples and analysis of how the SQG method is relevant and useful for analyzing energy storage in stochastic power systems.

Trondheim, August 30, 2017

Sondre Flinstad Harbo

Acknowledgment

Before delving further into the thesis, the author wishes to thank a number of people.

Firstly, I need to extend my most sincere gratitude to professor Alexei Gaivoronski. Not only would this thesis never been what it is without being able to utilize his excellent work on this topic and culminated in the implementation of the SQG solver. Moreover, I also feel honored to be able to be able to work with some of the issues he as such a great knowledge about. It is always an awe-making and humbling experience when Alexei answers one of my emails or questions with insights and information I need to do further research to fully understand. As a result, I feel very privileged whenever he has the opportunity to enlighten my research and work efforts.

Moreover, I am also very thankful towards academic faculty at Managerial Economics, Finance and Operations Research for excellent courses, exciting teaching and great learning of interesting and applicable concepts. I would also like to give thanks for the opportunity to pursue my own research questions and interests as part of my thesis. In addition, I am very happy to have been part of a class at Industrial Economics and Technology Management with may competent, clever and pleasant people. Their company has many a time sparked conversations of learning and exploration of possibilities. Some individuals to mention here are Magnus Johannsen who's expertise in programming and Evolutionary Algorithms has been inspiring, Edda Engmark and her inspiring insights on market coupling and scenario generation, and Kristian Sandaker and his unrivaled experience with computationally burdensome programming and application areas of empirical and statistical models. Moreover, I am also thankful towards PhD student Carl F. Rehn who helped me read though some of the paper in the latter stages.

Another group of people to whom I need to express my gratefulness is a few individuals from the department of Electric Power Engineering at NTNU. Professor Magnus Korpås, has for one thing provided many insight into how the SQG approach might be relevant and valuable to conduct analysis on the benefit of installing energy storage in power systems that are subject to uncertainty. PhD student Salman Zaferanlouei has been helpful to talk to about the implementation of the basic Alternating-Current Optimal Power Flow (AC-OPF) model. Master student Jørgen Erdal has provided a great conversational and conceptual sparring partner on several of the topics in this thesis, and provided valuable inspiration on the SPD modelling. Additionally, it is also appropriate to thank all of their department as well, for their many stimulating and thorough courses on Electric Power

Engineering.

I am also very appreciative for being able to use a picture from my previous employer Statkraft AS as cover illustration on this thesis.

Lastly, I need to thank my family for their continued and non-relenting support and belief in me and all my projects and endeavours, and putting up with all my exclamations on how exciting a little maths can be.

.

Than you all for great guidance, support and direction.

S.F.H.

Summary

This thesis shows how the Stochastic Quasi-Gradient (SQG) method may be utilized to analyze and optimize the use of energy storage in a power system to facilitate the inclusion of more volatility in the power system and put to use increased variable production by simulating an solving a Stochastic, Multistage Alternating-Current Optimal Power Flow (S-MS-AC-OPF) model.

The transformation away from fossil based fuels to more sustainable energy sources present many a challenge. One of these is that the renewable energy to be incorporated into the power system brings with it a high volatility that has not been present in the system before and that the grid and its physical components are not dimensioned to tackle. The technical solution to this is to include temporary energy storage in the energy system. Hence, for the success of the coming transition to a renewable energy system, there is a great need to develop methods that may analysis of how to facilitate the rising variability in the power grid using using energy storage.

From a perspective of optimization methods and power system analysis, this type of analysis is not trivial. This is because the analysis of energy storage requires the sought method to be of a dynamic character. It also has to be able to take heed of the uncertainty that will be present, and thus should be a method that deals well with problems of stochastic nature as well. Moreover, for the analysis to be of relevance from a electrical engineering point of view, the problem to be solved has to be able to tackle non-linearity and non-convexity, and hence being able to deal with local minima. This is to fully employ the constraints imposed by security, stability and physical limits of the power system. Therefore, the problem we want to optimize becomes quite complex. Luckily, there exists a method that may deal with all of these issues, and find a solution efficiently, namely the SQG method.

By implementing the SQG method with a S-MS-AF-OPF, and testing it with several approaches on several cases, we show how the SQG may serve the desired purpose. Especially the use of a gradient estimate directly from the AC-OPF solution provides a good solution within reasonable time. To conclude, the SQG method coupled with AC-OPF might be an effective tool for analyzing and optimizing energy systems with stochastic and dynamic aspects.

Sammendrag

Denne masteroppgaven viser hvordan Stokastisk Quasi-Gradient (SQG) metoden kan bli benyttet til å analysere og optimere bruken av energilagring i et kraft system og fasilitere en inkludering av mer volatilitet i kraftsystemet og benytte mer variable produksjon, gjennom simulering av en Stokastisk, Multisteg Vekselstrøm Optimal Kraftflyt (S-MS-AC-OPF) modell.

Transformasjonen vekk fra fossile brennstoffer til mer bærekraftige energikilder presenterer flere problemer. En av disse er at den fornybare energien som skal blir tatt inn i kraftsystemet tar med seg en stor volatilitet som ikke har vært tilstede i systemet før og som verken kraftnettet eller dets fysiske komponenter er dimensjonert til å takle. Den tekniske løsningen til dette er å inkludere midlertidig energilagring i energisystemet. Altså, for at overgangen til et fornybar energi system skal bli en suksess, er det et stort behov for å utvikle metoder som kan analysere hvordan man kan fasilitere den økte variabiliteten i kraftnettet bed å bruke energi lagring.

Med et perspektiv fra optimerings-metodikk og kraft-system-analyse, er denne typen analyse ikke triviell. Det er fordi en analyse av energilagring pålegger den ettersøkte metoden å ha en dynamisk karakter. Den trenger også å være i stand til å ta hensyn til usikkerheten som vil være tilstede, og bør derfor være en metode som håndterer problem av stokastisk natur bra. I tillegg, for at analysens skal være relevant fra et el-kraft-inteniørsk synspunkt, vil problemet som skal løses også måtte være ikke-linjært og ikke-konvekst, og derav må metoden være i stand til å håndtere lokale minimum. Dette er for virkelig å ta hensyn til begrensingene gitt av sikkerhet, stabilitet og fysiske begrensinger i kraftsystemet. Altså er problemet vi ønsker å løse ganske komplekst. Heldigvis finnes det en metode som kan ta hånd om alle disse problemene, nemlig SQG metoden.

Ved å implementere SQG metoden med en S-MS-AC-OPF, og teste den for flere tilnærminger og flere eksempler, vi viser hvordan SQG metoden kan tilfredsstille formålet. Spesielt bruken av en direkte utregnet gradient fra AC-OPF løsningen gir en god løsning på problemet innen rimelig tid. For å konkludere, kan SQG metoden, sammen med en AC-OPF være et effektivt verktøy for å analysere og optimere energi systemer med stokastiske og dynamiske aspekter.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The following introduces the thesis, by explaining how the topics are relevant, the motivation for the project work and the approach that was taken in this thesis. It also explains methods used and tries to gauge the contribution of the work. Lastly an outline for the rest of the thesis is presented.

1.1 Centrality of the topic

The climate challenge is by many recognized to be one of the greatest hurdles humanity has to tackle the coming generations. As the world tries to cope with the climate disruption it as imposed on itself, it faces numerous challenges. One of the major challenges is the raising variability with increased incorporation of renewable energy sources into the electric power system.

As countries, companies and people during the transition to a more sustainable society and economy construct wind farms, install photo-voltaic systems and buy electric vehicles, the inherent volatility of these renewable resources and green technologies will increase the variability in the power system dramatically. Whereas the generation and consumption of power has previously been rather stable on a short time horizon or at least predictable, the increased amounts of Renewable Energy Sources, as well as new trends on the demand side such as induction devices and charging of Plug-in Electric Vehicles, all makes the required energy production of controllable, dispatchable generation more unpredictable.

Moreover, the problem of rising variability in the power system is not only that of generating the right amount of power to meet the demand at any given time instance. It is also one of the power system's electrical component's capabilities, as many of these are developed for conditions where the power transferred, the voltage and current magnitudes, don't change too much from minute to minute. For instance, from any high-school science class, one might be aware that the coils present in a transformer transfers the power from one side to another through electromagnetic induction. Fundamental to this physical phenomena is the time-delay in this process, meaning that momentary changes cannot occur but produces extremely high current and voltage values in the transformer. Many other physical, technical components are also sensitive to rapid change power values, and on the system level essential issues such as voltage quality, reactive power and transmission frequency are all critically impacted by increasing variability.

The general solution to these issues is the introduction of energy system storage to the power system. Yet in doing so, there is still the question on how to do this in an effective way. There is a great need for analyzing how energy storage might be utilized to incorporate more renewable energy production into the current energy system, for instance how much wind energy could be built out in a specific area before one has to upgrade the grid, with and without energy storage. Equally important is to find the best way to utilize the energy storage in a most efficient way, so that as much as possible energy from the renewable sources may be utilized, or in other words find a method for optimal operation of the energy storage combined with other generating sources in the power system. Yet another issue is to facilitate the adoption of Electric Vehicles and the spontaneous charging they demand from grid.

In sum, for the success of the transition to a renewable future, there is a great need for methods that may facilitate an analysis of how to tackle the rising variability using in the system using energy storage.

1.2 Motivation

As argued above, there is a great need to better understand and address the rising variability of power generation and consumption that are a consequence of increased renewable energy resources in the grid, and how to use energy storage to facilitate this incorporation.

One key aspect of this problem is that of optimal operation, hence optimization techniques are in general of great relevance to this issue. Moreover, as we need to consider energy storage in the power system, the analysis needs to take on a dynamic character. Further, to tackle the variability aspect of the problem at hand, it is essential that ours methodology also is able to consider problems of a stochastic nature. Hence, we seek some method that may be able to optimize the dynamic charging and discharging of some energy storage in the electrical grid with stochastic generation and consumption.

To do this, we firstly need a way to analyze the complex power flow in the electrical grid in it self. For these types of problems the object function may be complex and non-linear, and the constraints are non-convex. There exists several approaches here and several simplifications. However, since we are concerned about analyzing situations near the constraints of the system, taking into account losses in the system and physical limits of the transformers and transmission line, an alternating-current optimal power flow (AC-OPF) framework seems to be the best. There also exists methods for solving the AC-OPF with dynamics across multiple time periods, yet these are of deterministic character.

Conversely, we also need the solution method to be be applicable to stochastic problems. Since the consideration of the variable renewable energy sources requires a stochastic or probabilistic description, the solution of the problem should be able to hedge against some of the more extreme cases and optimize the most likely ones. Moreover, in order to represent the variability in the most precise manner as possible, it is also beneficial for the model to be able to handle more complex probability distribution functions.

Consequently, this thesis presents the Stochastic Quasi-Gradient (SQG) method as stochastic optimization technique that may be used to solve problem at hand. It tackles the aforementioned issues of stochastic variables, non-linear, non-convex and dynamic behaviour through simulation of an underlying AC-OPF model. In addition, it also is able to incorporate any type of distribution function for the stochastic variables, making it much easier to couple with real observed data.

1.3 Researchers approach

This author has focused on the implementation of the SQG method to solve a multistage AC-OPF with dynamic and stochastic variables, through stochastic optimization of model simulation.

In developing the underlying AC-OPF simulation model, both standard AC-OPF models and dynamic models were considered. For the implementation, a standard solver in Matlab® was chosen, due to both familiarity for the author and reasonably good performance. From this, the main focus was to use the implemented AC-OPF simulation as the fundamental building block for the SQG-method, after a coupling with the SQG solver developed by Professor A.A.Gaivoronski. The implementation of the full SQG approach includes both consideration of the dynamic aspects of the problem, as well as the the stochastic properties.

The SQG is neither a heuristic nor an exact mathematical methods, yet it has components from both. Some parts of the SQG algorithm, has similarities to Global Optimization methods such as SA. On the other hand it uses the concept of analytically relations, such as gradient, and also Hessian if desired, to get to solution. Thus, it should be faster than brute-force heuristics, yet does't fall short due to the complexity of the problem as exact methods does.

To test the performance of the SQG approach for the stochastic, multiplied AC-OPF problem, two different cases was developed. One of these was a simple as possible case, to see whether the method worked for this type of problem. This basic case was also compared with an exact, yet discretized, solution. Further, a slightly bigger case model was developed, with a bigger grid and more time steps, to see how the model performed in this somewhat more realistic setting.

1.4 Contribution and methodology

The main contribution of this thesis is the implementation of a multistage, dynamic AC-OPF model with stochastic parameter solved by the SQG method. In doing so, a few developments has been made that contribute to the literature and modelling of these type of problems.

The contribution of this thesis is:

- Implements the AC-OPF for the SQG software
- Implement simulation based SDP of AC-OPF for comparison
- Provides examples on how the SQG implementation is competitive with SDP on small bus case

- Tests the S-MP-AC-OPF with the SQG for a bigger case with more stochastic parameters.
  - Explores different methods for estimating the stochastic quasi-gradient for the problem
  - Shows the SQG approach is capable of providing good solutions in reasonable amount of time for larger problems with several sources of uncertainty.
- Develops a strategic decision policy version of the S-MP-AC-OPF
  - Shows how to project a specific policy decision onto a hyper-plane given by the constraints of the policy decision.
  - Proposes policy decision rules that encode the aggregated value of the realized stochastic variables adjusted for the effect of their spacial distribution has on the grid.

1.5 Outline of the thesis

This report is divided into 8 chapters, in addition to Preface, Acknowledgements, abstract and appendices.

Firstly, an introduction is given in this very chapter, conveying how the topic is relevant, the motivation for the work, the approach take and contributions. Thereafter, a problem description and statement is provided in the second chapter. The third presents related literature and theory, before the forth presents the basic mathematical aspects of the model implemented. Chapter five presents the methodology and cases used in the thesis. In chapter six the implementation of the model is treated further, before chapter seven present results obtained from its simulation. The last chapter concludes the report and provides thoughts on further research.

Chapter 2

Problem description

This chapter gives a further introduction and description of the issues this thesis addresses. First, a precisely formulated problem statement, or research question, is presented. Thereafter, a further description of the problem and scope will be given.

It should be noted that some parts of both this last section, and the beginning of chapter 3 are perhaps a little too basic in the content presented. This however, has been deemed relevant to include non-the-less, as it might help provide background information and foster comprehension for a reader with no particular background in neither non-linear optimization nor stochastic optimization. Indeed, the reader in mind for this thesis is not only researchers in its respective fields, but also students from the same program as the author, for which many of the methods and methodologies of this thesis is unfamiliar.

2.1 Problem statement

In this thesis, the key concern is to asses how use energy storage optimally, given knowledge on the stochastic behaviour and variability of certain energy sources and consumers, and subject to the constrains imposed by the physical limits of the energy grid, and how the possibility to store energy might enable more variable energy to be introduced to the energy system. Hence, the research question for this thesis is

> How might the Stochastic Quasi-Gradient method be used to analyze and optimize multistage power system operation with energy storage under uncertainty?

2.2 Description of problem and scope

In the following, a more detailed problem description and background information is given, as well as sizing up the breath of the task and what is considered outside the scope. First, the most common and important issues of power system analysis and optimization are presented. Thereafter, the most relevant concerns for the optimization methods will be discussed, before a section on the combination of the to disciplines.

2.3 Power System Optimization

When dealing with power system optimization, one uses relations and phenomena from power system engineering and power system analysis to find the optimal way to produce energy for a system given its load and characteristics of energy energy distribution.

The energy produced and transported is subject to physical laws of energy conservation and constrained by the physical limits of the electrical components of the power system as well as rules set to enforce power security and stability.

To analyze this, several methods are available, such as the Economic Dispatch approach, regular Load Flow studies or Fast Decoupled Load Flow, and the standard Direct Current Optimal Power Flow (DC-OPF). Both the Economic Dispatch, but especially OPF methods, are readily incorporated with power market considerations as well, with power bids for both producers and consumers.

However, these approaches does not fully consider the impact of losses and reactive phase shift of the power in the system which is critical for a realistic analysis with respect to the physical constraints and security of the system. All these desired properties are characteristics unique to Alternating Current (AC) power systems, and only fully consider in another approach called AC-OPF method. This is because elements such as coils, capacitors, and long transmission lines experience electromagnetic temporal energy storage of power causing phase shift between the voltage and current profiles.

For DC this is not an issue, as the voltage and current are both a product of unidirectional flow of electric charge. For AC systems, which is most common for power grids, the current and voltage alternates periodically between negative and positive values as the charge flow changes direction. Hence, the reactive electrical components cause the phase between current and voltage to shift phase relative to the current. Since power is a product of the current and voltage at any given moment, a phase shift between the two means among other things that the active power delivered is less than the apparent power from the generators. Hence, this is critical information to consider for a optimal power flow analysis with varying loads and generation, as well as for the physical constraints and stability constraints.

For an example on how increasing volatility affects the grid, one might consider some assessments of how EV adoption is likely to affect the Norwegian grid by NVE with the publications Spilde and Skotland (2015) and Skotland et al. (2016). They show that the electrification of the transportation system will pose challenges to the grids transformer stations and voltage quality. Lines are generally not the main problem, yet some transformers, especially in rural areas, are overloaded in some cases. In terms of voltage quality and stability, there is a somewhat greater concern. For instance, measurements done by SINTEF has showed that even the connection of one 32 A one phase charger may cause significant reduction in voltage quality.

On a gird level, a large variation in power demand is problematic since the power produced needs to be the same as the consumed. Thus, it does not only make it necessary for the power producer to constantly adjust its production, which with big changes can be costly and inefficient. It also poses a threat to a consistent frequency but also stability of the grid. Hence the possibility to store and discharge energy lets one move load around, to do what is called *peak-shaving* and *valley filling*. In this way the difference between peak and trough is less which is beneficial for the grid. See figure 2.1 for an illustration.

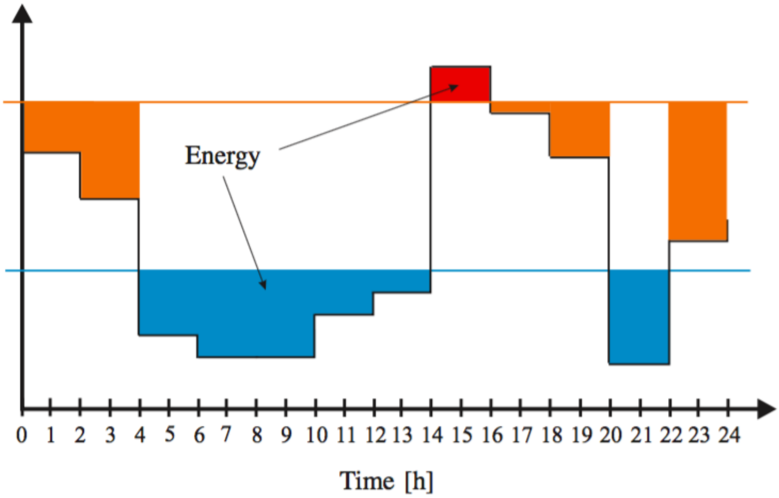

Figure 2.1: Peak-shaving and Valley-filling of High and Low Load Periods Galus et al. (2012)

Hence, we are also interested in analyzing the effect of providing Energy System Storage (ESS) for the grid, both in terms of optimal operation, but also on how it may help incorporate higher amounts of variable power without upgrading the physical components. This is because Renewable Energy Sources (RES) needs a place to store the energy they produce when their production is high, for then to be discharge into the grid when the production from the RES is low. Therefore we need to consider a dynamic approach, where certain decisions are made at different stages and where the system values might change from stage to stage also based on a previous decision. To apply the dynamic aspect, approaches such as a Dynamic OPF or Dynamic Programming might be used.

However, we also desire to include uncertainty in our models. The dynamic approaches mentioned above does not necessarily incorporate uncertainty easily and in a manner that is solvable within reasonable time. There is the possibility to use Stochastic Dynamic Programming, yet the solution time quickly becomes too great with this method for reasonably complex cases. Therefore, this thesis suggests the use of the Stochastic Quasi-Gradient (SQG) method to solve this stochastic, dynamic AC-OPF problem, through simulation of a regular AC-OPF.

When working on this thesis, some limitations had to be imposed on what not to do, and what was to be the main focus. Hence, when implementing the AC-OPF model we only consider relatively simple cases, and cases with no market coupling. This is because the focus of the thesis is mainly to implement and show that the SQG method is well suited for the problem we want to analyze. Moreover, the main effort has also not been to develop a most efficient method for running the AC-OPF model, but a method with reasonable speed to facilitate a swift solution time for the SQG solver.

2.4 Optimization Methods

There are many ways to characterize different types of optimization problems. One aspect is whether or not the problem is linear. A linear optimization problem, or a linear program, has a linear objective function dependent on its decision variables and constrained by some linear inequalities or qualities of both, as in model 3.2. There are many well developed methods for solving linear programs, for instance the SIMPLEX method or duality methods. A non-linear problem on the other hand is characterized by having non-linearity present in either its objective function or constraints or both. Examples of such non-linearity might be a quadratic cost function as objective, and a piece-wise linear function as constraint. For some types of non-linear problems, such as the ones mentioned above, there exists efficient and exact solution methods.

The biggest trouble with non-linear optimization problems is when the non-linearity give rise to a non-convex problem, meaning that there is the possibility of having both local and global optima, illustrated in figure 2.2 below.

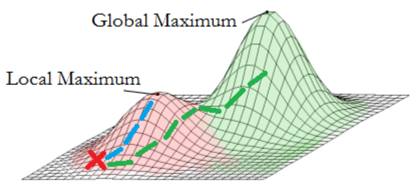

Figure 2.2: A non-convex (or really non-concave since it is about maximization) surface with global and local maxima.

When problems are characterized by non-convex features, it thus presents a challenge on how to find the global optimum. Intuitively, using the allusion of scaling a mountain clouded in mist; how does one know that one has reached the top, and not just some smaller peak on the hillside. One knows one has reached a top, since the current gradient is zero. Moreover, one might be able to check the area directly around the current position, and see that the curvature of the slope of the current point is negative in all directions and confirming that this is a top, not some edge one may walk along. Indeed, one might also have the ability to see some area around through the air, and note that one cannot see any other surfaces in the near proximity. However, it is still not conclusive that one has reached the top; it might hide somewhere behind the mist. In terms of optimization methods, the question is thus what kind of approach, strategy or algorithm should we use in order to get to the global optimum - the peak of the mountain. One thing this is likely to depend on is the point at where you start. Another thins, is how you decide the path you choose going forward to find the optimum. This is illustrated in figure 2.3, where a number of different starting positions and paths lead to different minima, but only one finds the global one.

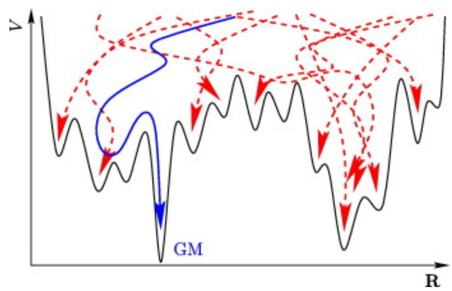

Figure 2.3: Different starting points and paths finding local and global minima on a non-convex line.

In tackling these type of global optimization problems, there are many different approaches, that may for instance be characterized as deterministic methods, stochastic optimization methods, or heuristics and meta-heuristics. The first tries to use some exact strategies to find the optimum, such as using inner and outer representations, cutting planes, branch and bound and so on. The second, stochastic optimization methods, are more or less exact methods that utilizes some random (hence termed stochastic) variables in their iterative solution process in order not to be "caught" in a local optimum. Hence, they are often labeled as Monte-Carlo methods, which generally make use of random sampling to generate some desired result. Examples here are stochastic approximation methods. The third, heuristics, may generally be described as an trial-and-error approach, that tries to find the best value using some rules-of-thumb, rules or algorithms using brute force intelligently. This category often overlaps with the previous one and share common features, especially since heuristics often also incorporate a random element in their search process, which may thus be also be labeled as a stochastic optimization method, or a randomized search method. Examples are Simulated Annealing, Evolutionary Algorithms and Swarm-based optimization.

However, the terms ”Stochastic” is not only used to describe a randomized, iterative solution process. In the field of operations research and optimization, the term stochastic programming is used to refer to mathematical decision making under uncertainty. That is, some part of the problem is not fully know, for instance the realization of some variables that will affect the objective function or constraints of the problem. One approach to solving such problems is to discretize the probability space and solve the corresponding deterministic simplification of the problem. Examples here are scenario trees, decomposition methods and Stochastic Dynamic Programming. Another would be to tackle the probabilistic nature straight on, for instance by chance constraints and probabilistic programming.

The method of Stochastic Quasi-Gradient proposed in this thesis, might be characterized as a Stochastic Optimization method for solving Stochastic Programs. Hence, it is not only able to deal with non-linearity, non-convexity and complex functions and to find a solution within reasonable range of the global solution. It also is aimed at solving optimization programs that are subject to uncertainty. Being a stochastic optimization method, and neither a fully exact approach nor a full-blown heuristic, it should be able to get close to the global optimum in a way that exact methods might have some difficulties with in the same time. Moreover, it will be able to use some analytic information from the solution process - the Stochastic Quasi-Gradient - which should aid its search so it may be faster than a heuristic based one.

Again, it is not possible to do everything one might desire when working on a thesis, and some restraints has been made. As mentioned, the focus of this thesis is primarily to implement the SQG method for the multistage AC-OPF model, and show that the SQG method is a very relevant approach to use when analyzing energy systems with storage under uncertainty. As a quick comparison with an exact method, the author has implemented a similar case with the Stochastic Dynamic Programming approach. Likewise, it the student also considered to develop a comparative model based on a stochastic optimization heuristic, yet after trying out some easily available solvers requiring long run times, spending time of this was considered less of an priority. Further more, some attempts were made to develop methods to calculate the stochastic gradient directly from the AC-OPF solution and not using the finite difference method requiring much more time. However, very thorough mathematical proofs of the gradient calculations proposed were considered outside the scope of this thesis. Similarly is the case for the projection operator algorithm developed for one of cases to be solved by the SQG solver.

2.4.1 The use of SQG on AC-OPF problem with ESS

From these two paragraphs, it may be natural to see how the SQG approach might be a good solution approach to deal with multistage ACOPF with energy storage under uncertainty. One is that the SQG it addresses problems where it is not trivial to find the value of the objective function given a computed solution, and hence deal well with the complexity and non-convexity present in the AC-OPF models. Moreover, as the AC-OPF model for several time stages quickly becomes quite complex with the non-convexity of the underlying problem, the global optimization trait of the SQG approach is beneficial. Since the SQG method also is able to use any type of distribution function for the stochastic parameters, which simplifies and enables the usage for real life cases such as non-smooth consumer patterns and rather non-descriptive wind data. Moreover the SQG method may well be used to address multistage models of several time periods, which takes a long time to compute for exact methods and is a problem if one is to model for instance a whole day a head with for example a 60 minutes interval.

Chapter 3

Review of related literature

This chapter starts with a presents and some of the most important literature sources for this thesis, as well as discussing studies related to their topic and methodology.

In addition, a review will be conducted of some of the fundamental theoretical concepts required to understand the thesis' models and facilitate further discussion of methodology, implementation and results. As noted in chapter 2, some of the concept introduced here, might be considered a little too basic, but is included to facilitate the understanding for readers unfamiliar with these particular methods. As such, we present both some of the seminal as well as "state of the art" literature on these topics. This thesis utilizes concepts from two different, yet related, fields, which will be presented in their respective sections after an introduction to optimization is given.

3.1 Review of Literature

This work draws from several different streams of literature, as it may be placed in the conjunction of two different fields; Power System Analysis and Operations Research.

The most important sources of information for this thesis in terms of optimization methods in general has come from Nocedal and Wright (2006) and Birge and Louveaux (2011). The first of these as an comprehensive guide to numerical optimization, where also the works of Hillier and Lieberman (2010) and Lundgren et al. (2012) has been insightful. The second book has been a valuable guide in understanding the fundamentals of Stochastic Programming, but also the books by Kall and Wallace (1994), Pflug (1988) have been important. Especially the first two books, with Nocedal, Birge and the last two with Kall and Pflug, are considered seminal works in the field.

On the SQG method, the works of Ermoliev (1983), Ermoliev and Wets (1988), Gaivoronski (1988), Gaivoronski (2005) and Becker and Gaivoronski (2014) has all been of great value. Where the paper, and following book of Ermoliev introduces the SQG method theoretically, the works of Gaivoronski specifies how this method is implemented and the many advantages it has. Additionally, the work of Peeta and Zhou (2006) together with the Becker and Gaivoronski paper, has provided ideas and examples for what to consider when implementing, approximating values and tuning the SQG model.

For power systems, reference may be found with Grigsby (2012) and Crow (2012). The first provides an overview of electric power engineering and power system analysis in general, and the second gives a more in-depth guide to computational techniques for power systems. The works of Sperstad and Marthinsen (2016) and Castillo and O'Neill (2013) provides comprehensive guides to optimal power flow methods and their computational performance. For the consideration of dynamic programming, the book of Bellman (1957) is important. However, also the works of Zaferanlouei et al. (2016) and Erdal (2017)has been of influence for this thesis.

Many other papers have also been visited that are for the concern of length not brought up here. However the paper of importance and relevance for this thesis will be referenced in the sections below. Lastly, the work of course draws upon concepts and work started in Harbo (2016) with developed as a forebearer to both the ACOPF and SQG models proposed here.

3.2 Introduction to Optimization

In this section introduces and reviews the theoretical background of the mathematical optimization methods that are featured in this thesis, starting from basic concepts, discussing different classes - or basic characteristics - of optimization problems.

3.2.1 Basics of optimization

Starting from the very fundamentals, the field of optimization is one a subject with seeks to find the best solutions to a problem, often a decision of some sort, as described for instance in Lundgren et al. (2012) introductory book on Optimization. More specific, it is a field of applied maths, in which one formulates the problem in question mathematically, and uses its models and analytic methods to find the best decisions to make from the feasible alternatives. In the mathematical model like 3.1, one thus defines an objective function which is sought either minimized or maximized based on some decision variables the decision maker is in control of. Further, there are often several constrains imposed on the variables, giving the boundaries of possible solutions. In general an optimization problem may thus be expressed as

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathbf{X} \end{aligned} \tag{3.1}$$

where $f(\mathbf{x})$ is the objective function and $\mathbf{X}$ is the set of possible solutions for the vector of control variables $\mathbf{x}$. When subject to linear constraints, the problem may be formulated as

$$\begin{aligned} \min \quad z = \; & \mathbf{c}^T\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{3.2}$$

where $\mathbf{x}$ is the $1xM$ dimensional vector of control variables, $\mathbf{c}$ are the $1xM$ dimensional coefficients of the control variables in the cost function and $\mathbf{A}$ is the NxM dimensional coefficient matrix for the inequality constraints to be less or equal to their $Nx1$ counterpart of constraint values $\mathbf{b}$.

Then, introducing slack-variables σ to 3.2, we may express the dual problem as

$$\begin{aligned} \max \quad & \mathbf{b}^T\boldsymbol{\lambda} \\ \text{s.t.} \quad & \mathbf{A}^T\boldsymbol{\lambda} + \boldsymbol{\sigma} = c \\ & \boldsymbol{\sigma} \geq \mathbf{0} \end{aligned} \tag{3.3}$$

where λ is the Lagrangian multiplier treated in more depth in section 3.2.2.

3.2.2 Non-linear Optimization Using Lagrangian Multipliers

In many real-world cases, it is not desirable to reduce the model of the optimization problem at hand by simplified linear relationships. This is the case for instance in finance, for which many applications are discussed in Zenios (2007). One example is when attempting to minimize the risk of a portfolio of some assets (calculation the deviation from the mean, termed as volatility and measured by the standard deviation), given a lower bound on the expected return of the portfolio. Another example of a non-linear problem of particular interest to this thesis is the power flow of the AC-OPF model.

An optimization problem is non-linear if either the objective function or any of the constraints contain non-linear terms. The main issue with these type of problems is that they might give rise to non-convexity, meaning that thee might exist several local minima.

In general, non-linear optimization problems may be expressed similar to linear ones, as in Lundgren et al. (2012) we can formulate the general problem as

$$\begin{aligned} \text{min} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g_i(\mathbf{x}) \leq b_i \quad \forall i \in 1, 2, \cdots, N \end{aligned} \tag{3.4}$$

where some terms in the constraints $g_i(\mathbf{x})$ or $f(\mathbf{x})$ are non-linear.

As discussed in Bertsekas (1999) in detail, depending on the specifics of the problem formulation, the solution of such non-linear problems might be sought using methods like quadratic programming fractional programming and - perhaps most applicable to most nonconvex problems - branch and bound methods.

More importantly for our case in this report where we have derivable functions, the Karush-Kuhn-Tucker (KKT) conditions provide general conditions for optimality of non-linear problems apply, and gives rise to the technique of Lagrangian multipliers and nodal price analysis. The conditions for the point **x** to be a local minimia is

$$\Delta f(\mathbf{x'}) = v_i \Delta g_i(\mathbf{x'}), \quad v_i \geq 0, \quad \forall i \in 1, 2, \cdots, N \tag{3.5}$$

$$g_i(\mathbf{x'}) \leq b_i \quad \forall i \in 1, 2, \cdots, N \tag{3.6}$$

$$v_i(b_i - g_i(\mathbf{x'})) = 0 \quad \forall i \in 1, 2, \cdots, N \tag{3.7}$$

where the constraints in the first equation 3.5 gives the dual feasibility, the second constraints 3.6 defines primal feasibility and the last constraints 3.7 ensures complementary. From this we formulate the Lagrangian function as follows

$$\mathcal{L}(\mathbf{x,v}) = f(\mathbf{x}) + \sum_{i=1}^{N} v_i(b_i - g_i(\mathbf{x})) \tag{3.8}$$

where $v_i \geq 0$ is now the Lagrangian multiplier for the ith constraint. From the KKT equations above we see that is necessary for the Lagrangian function to be stationary for a point to be optimal, that is $\Delta\mathcal{L} = 0$ in mathematical terms.

Or restated in notation similar to section 3.2.2 the Lagrangian function may be expressed as

$$\mathcal{L}(x, \lambda, \sigma) = c^T x - \lambda^T(Ax - b) - \sigma^T x \tag{3.9}$$

where the necessary first order optimally conditions

$$\begin{aligned} A^T\lambda + \sigma &= c \\ Ax &= b \\ x &\geq 0 \\ \sigma &\geq 0 \\ x_i\sigma_i &= 0 \quad \forall i = 1, 2, \cdots, n \end{aligned} \tag{3.10}$$

Note the bold font previously used to explicitly specify the vector characteristic of a variable is now left out.

3.2.3 Line Search Methods

When trying to find the best solution of an optimization problem, there are different ways or strategies one may utilize. The simplex method mentioned earlier seeks to find the optimum of a linear, convex, problem by checking its vertices, that is points of intersection between different constrains, based on the knowledge that the optimum should lie in one of these points. It also moves from one point to the next along the line given by two constraints' intersection, in a way that yields most improvement per move.

Another approach is that of line search methods. Generally, they compute a search direction d^s and a step length ρ^s for each iteration. That is finding the next approximation for the solution $xs + 1$ by

$$x^{s+1} = x^s - \rho^s d^s \tag{3.11}$$

One intuitive method is here is the gradient approach. It tries to find the the optimal solution by figuring out the direction of the rate of improvement at its starting point, and then move a step of a certain length in that direction. More formally, as for instance described in Jameson (1995), if function $f(x)$ is a smooth function where the vector x has a of dimension n, its gradient $\xi = \nabla f$ given as

$$\xi_i(x) = \frac{\partial f}{\partial x_i} \tag{3.12}$$

If the objective is to minimize the fiction value, then the minimum $x*$ may be found through taking a number of step of length ρ in the direction of the negative gradient. Hence, for step s, the next approximation of the optimal value is given by

$$x^{s+1} = x^s - \rho^s \xi^s \tag{3.13}$$

where ρ needs to be chosen small enough that it leads x^{s+1} to become less.

More generally then, if one requires that the direction is one of decent - meaning that the movement in that direction yields a reduction of the fuction value, or $(d^s)^T \nabla f^s$ - the direction d^s is often has the form

$$d^s = -(\mathbf{B}^s)^{-1} \nabla f^s \tag{3.14}$$

where the matrix $\mathbf{B}^s$ is both symmetric and non-singular. For the gradient method, a method of steepest decent, $\mathbf{B}$ is simply the identity matrix.

Another line search method is Newton's method or Newton-Raphson method known from numerical analysis as a way to iterative find the zero's -or roots - of some function. In this case, the $\mathbf{B}^s$ matrix is the Hessian $\mathbb{H}^s = \nabla f(x^s)$ given as

$$\mathbb{H}_{ij}(x) = \frac{\partial^2 f}{\partial x_i \partial x_j} \tag{3.15}$$

producing the next step as

$$x^{s+1} = x^s - (\mathbb{H}^s)^{-1} \xi^s \tag{3.16}$$

3.2.4 Interior-point methods

Interior point - or barrier - methods are, together with active-set sequential quadratic programming methods, considered the most powerful way to solve large-scale non-linear problems. For understanding we may comparing it conceptually to the to the Simplex method, a long time standard approach to solving linear programs and the dominating solution method for more small scale linear problems. The Simplex method checks vertices along the feasible region in searching for the optimal solution, which generally requires a large number of inexpensive iterations. Yet, as the problem becomes bigger, e.g. increasing the number of constraints and variables, the solution time scales exponentially with size of problem.

The Interior point methods on the other hand use another strategy. It approaches the boundary in the limit, from interior (or exterior) of feasible region. The term "interior point" comes from the early development of the method that started at some initial point x^0 inside the feasible region - an interior point - and used barrier methods to make sure the iterations stayed within the feasible region. Such barrier methods deploy a barrier function

$$f(x) - \mu \sum_{i \in \mathcal{I}} \ln \big(g_i(x)\big) \tag{3.17}$$

where $g_i(\mathbf{x})$ represents the inequality constraints, and i specifies the a specific constraint of all the inequality constraints $\mathcal{I}$. The variable μ is the barrier parameter which is a positive number that should be specified to converge to 0 during the iterating, and the operator $\ln$ denotes the natural logarithm. As can be seen, the barrier function which prevents the iterations from leaving the feasible region as $-log() \rightarrow \infty$ as $\rightarrow 0$.

Hence, interior points never actually lie on the boundary, and thus is able to tackle non-linear cases as well. Indeed, the method has proven similarly efficient for linear as non-linear cases. In contrast to the Simplex method it uses a small amount of iterations, yet each iteration is more computationally expensive. Yet, for the nonlinear cases problems such as the needs to deal with non-convexity, how to update the barrier parameter under non-linearity and convergence of the solution are an additional challenge.

Formally, we may express the method as follows. Given as slight reformulation from the basic models expressed earlier in 3.2 and 3.4, we consider the optimization problem

$$\begin{aligned} \min_{x,\sigma} \quad & f(x) & \\ \text{s.t.} \quad & h(x) & = 0 \\ & g(x) - \sigma & = 0 \\ & \sigma & \geq 0 \end{aligned} \tag{3.18}$$

where $h(\mathbf{x})$ represents the equality constraints and $g(\mathbf{x})$ are the inequality constraints of the problem, which may or may not be of linear character, we may express the approximate barrier problem as

$$\begin{aligned} \min_{x,\sigma} \quad & f(x) - \mu \sum_{i \in \mathcal{I}} log\big(g_i(x)\big) \\ \text{s.t.} \quad & h(x) \qquad = 0 \\ & g(x) - \sigma = 0 \end{aligned} \tag{3.19}$$

where the slack variables σ no longer needs to be specified greater than 0 as the minimization of the barrier function prevents the elements of σ to move too close to 0.

The approximate problem given in equation 3.19 is then simpler to solve than the exact problem, and the Matlab® `fmincon` solver utilized in this thesis solves the problem primarily by using the line search Newton method of equation 3.16 discussed in section 3.2.3.

A further, full treatment of the interior point is not given here, as it would instill the needed to introduce a lot more theoretical concepts, and out focus here is the SQG method. For more information on this technique, see for instance Nocedal and Wright (2006) for a comprehensive presentation.

3.3 Stochastic Programming

A demanding issue when dealing with optimization problems is how to tackle uncertainty in the mathematical formulation and solution of optimization models. This is the focus for a group of optimization methods on stochastic optimization.

When speaking of Stochastic programming, there are two main approaches. One of these is what is called robust optimization, for which one is interested in finding some solution that satisfies all the specified constraints for all probabilities. The second is the approach of chance constraints, where the constraints are to be valid from some specified probability. There are a number of different approaches to solving stochastic optimization problems, for instance decomposition methods such as the L-shaped Decomposition Method, Benders Decomposition and Stochastic Decomposition, as well as Stochastic-Dynamic programming and the Scenario Tree approach that are briefly introduced in this thesis.

Yet most of these have in common that they require the problem have a certain degree of convexity. Neither do they deal well with non-linearity, and methods such as the Lagrangian multiplier approach that does may not easily tackle stochastic features. Equally critically, these other approaches often become computational demanding when introducing several time steps or stages. For instance does an expanded scenario tree becomes quite huge, and quickly very computational demanding. A method better suited to tackle these issues - issues that are key characteristics of the Optimal Power Flow models of interest that are non-convex and multistage - is the Stochastic Quasi-Gradient method that is the focus of this thesis.

3.3.1 Basis of Stochastic Programming

These models have historically emerged from recourse problems and probability constrained models, as Higle (2005) mentions in a brief and informal introduction to stochastic programming. These models arise when one of more variables of the model is best described by random variables. A more thorough presentation of Stochastic Programming is given in the books of Kall and Wallace (1994) and Birge and Louveaux (2011). As stated in their works, a stochastic problem can generally be formulated as

$$\begin{aligned} \min_{x \in X} \quad & \mathbb{E}_\omega f_0(x, \omega) \\ \text{s.t.} \quad & \mathbb{E}_\omega f_i(x, \omega) \leq 0, \quad \forall i \in 1, 2, \cdots, M. \end{aligned} \tag{3.20}$$

where one seeks to minimize the expected value (the $\mathbb{E}$ operator) of the objective function f_0 by choosing the right values for the control variables x given some constrains f_i and the realization of the random variables ω.

3.3.2 Recourse problem

In the case of an aforementioned recourse problem, we try to make some optimal decision on x with a corresponding cost c, called fist-stage decisions. These must be made before we have information about some uncertain events in the future, represented by $\boldsymbol{\xi}$, often defined by specific events or scenarios s or represented through some underlying stochastic variable ω, specified as $\boldsymbol{\xi}(s)$ and $\boldsymbol{\xi}(\omega)$ respectively. After these random events are realized, we are able to make some corrective measures $\mathbf{y}$ with corresponding costs $\mathbf{q}$. A stochastic linear formulation may be expressed as in Birge and Louveaux (2011) by

$$
\begin{aligned}
\min_{x} \quad & c^T x + \mathbb{E}_{\xi} Q(x, \boldsymbol{\xi}) \\
\text{s.t.} \quad & Ax = b \\
& x \geq 0
\end{aligned}
\tag{3.21}
$$

where

$$
Q(x, \boldsymbol{\xi}) = \min_{y} \left\{ \mathbf{q}^T \mathbf{y} | W\mathbf{y} = \mathbf{h} - \mathbf{T}x,\ y \geq 0 \right\} \tag{3.22}
$$

and the boldface characters symbolizes vectors that are subject to randomness, and $\boldsymbol{\xi}$ is a vector formed by the variables in q^T, **h** and **T**. W, **T** and **h** specifies the constraints of the second-stage corrective measures of **y**. The $\mathbb{E}_{\xi}$ operator is the mathematical expectation subject to $\boldsymbol{\xi}$.

3.3.3 Multistage Stochastic Programming

The book by Pflug and Pichler (2010) give a comprehensive treatment of multistage stochastic optimization. As before, Stochastic problems are characterized with uncertainty in the parameters describing them. Multistage problems, are problems that reach over several periods of time, in which one tried to find a optimal policy or strategy for solving the problem for all the desired stages. Hence they may also be called multi-period problems, yet for a multistage stochastic optimization problem the execution of the strategy or policy should allow for reactions to what previously has happened and react to the new situation, in a manner that optimizes the overall objective.

Multistage Stochastic problems arise in many real world cases, where one in to make several decisions over a period and where unforeseen events happen during the course of time. A common solution approach for discrete probability events is the use of Scenario Trees, where one weights the different outcome and decisions according to their probability. Another approach, arisen from dynamic programming where one solves a discrete problem for several time steps, is that of Stochastic Dynamic Programming.

3.3.3.1 Scenario Tree Approach

Scenario Trees may be applied after discretizing the random events into specified scenarios, s, and estimate (or assume) some probability for each of them. This is then used to calculate the expected value of the recourse function. As such, we may let $p_{t,s}$ be the probability associated with scenario s occurring at time step t, and x_t be the decision variable at the same time step. See figure 3.1 for an illustration.

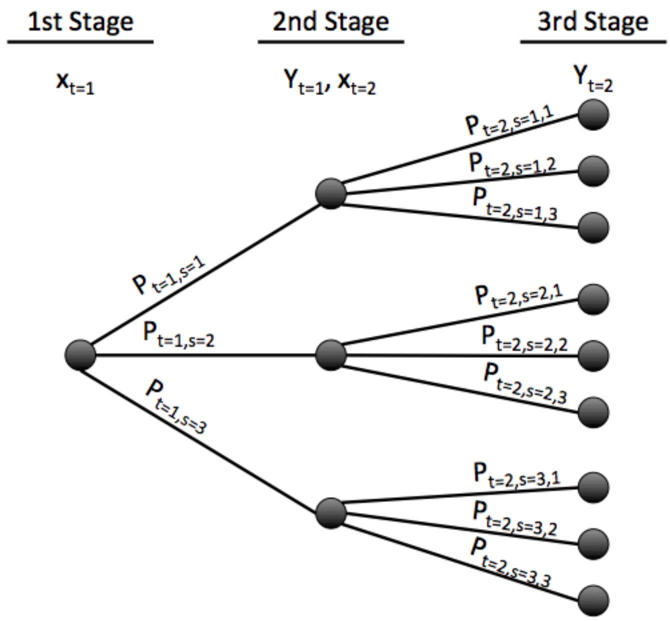

Figure 3.1: Scenario tree

One may choose to formulate the optimization problem of this in a so-called extensive form, as for instance

$$\begin{aligned} \min_{x} \quad & c_{t=1}^T x_{t=1} + \sum_{s=1}^{3} p_{s,t=1}\Big(\mathbf{q}_{t=1}^T \mathbf{y} + c_{t=2}^T x_{t=2} + \sum_{s=1}^{3} p_{s,t=2}\big(\mathbf{q}_{t=2}^T \mathbf{y}\big)\Big) \\ \text{s.t.} \quad & W\mathbf{y} = \mathbf{h} - \mathbf{T}x \\ & y \geq 0 \end{aligned} \tag{3.23}$$

where W defines the constraints parameters for the $\mathbf{y}$ corrective actions, $\mathbf{T}$ defines how the x's are related to $\mathbf{y}$ and each other, and $\mathbf{h}$ is some constants that may or may not, as all the boldface vectors, be subject to stochastic changes.

3.3.3.2 Stochastic Dynamic Programming

Stochastic Dynamic Programming, often abbreviated as SDP, is a method of stochastic optimization with roots in dynamic programming. It uses discretized probability distributions through Markow Chains and specifically Markow Decision Process to calculate the optimal policy for a specific problem, which all might be found in described Hillier and Lieberman (2010).

From dynamic programming, we may utilize its method for systematic evaluation of the optimal combination of some interrelated decisions. This approach discretizes the states and the time steps. Central to dynamic programming is the *Bellman Optimality Principle*, see for instance Bellman (1957) which says that for any given current state, an optimal policy for the following states is independent of policy decisions made in the previous states. Then, if we may somehow find the optimal policy and its value for the last stage, we may recursively calculate the optimal policy for each stage back to the beginning.

As stated in Hillier and Lieberman (2010), unlike for linear programs, there is no general formulation for dynamic programming, as the specific formulation used must be customized specifically to fit the problem in question. However, we may state this recursive relationship in mathematical notation as

$$f_n^*(S_n) = \underset{x_n}{min}\{c_{sx_n} + f_{n+1}^*(x_n)\} \tag{3.24}$$

where $f_n^*(S)$, the objective function of the optimal decision at stage n with state s_n, is a result of the cost for going between the states and the optimal objective function at the next stage. When starting from the last step N, the optimal policy at the current step n will always be of the form

$$f_n^*(S_n) = \underset{x_n}{min}\{f_n(s_n, x_n)\} \tag{3.25}$$

and can be used to calculate the optimal policy for all stages back to the initial stage.

Another feature of Bellman's Optimality Principle, is that the current state contains all relevant information of the previous states. When considering a stochastic version of the dynamic problem, one thus couple this with a definition of the stochastic process as an Markov Chain. These processes have the feature of what is called the Markovian Property, which means that the future events are independent of the past events. Thus, when considering the joint distribution of some events $X_0, X_1, ...$, we may formally express this as

$$
\begin{aligned}
&P\{X_{t+1} = j | X_0 = k_0, X_1 = k_1, ..., X_{t-1} = k_{t-1}, X_t = i\} \\
&= P\{X_{t+1} = j | X_t = i\} \\
&\text{for } t = 1, 2, ... \text{ and } i, j, k_0, k_1, ..., k_{t-1}
\end{aligned}
\tag{3.26}
$$

hence, to be able to calculate backward recursively the optimal expected value of the transition from the step $n+1$ to n, we only need to know the probability of transitioning between those two states. Thus, if one would be calculating forward, the probability distribution in this step connecting it to the next, is only dependent on its current state, not how it got there. Therefore we take in use transition probabilities of going from one stage to another, which is ordered in a transition matrix. It contains the probability from going the state in row n to state in column n

$$
\mathbf{P}^t = \begin{bmatrix} p^t_{00} & p^t_{01} \cdots p^t_{0N} \\ p^t_{10} & p^t_{11} \cdots p^t_{1N} \\ \vdots & \vdots \cdots \vdots \\ p^t_{N0} & p^t_{N1} \cdots p^t_{NN}. \end{bmatrix}
\tag{3.27}
$$

This then let us formulate the expected value of time step t as

$$
\mathbb{E}f^*(S_n, t) = \underset{x_n}{min}\{c_{sx_n}(t) + \sum_{k=1}^{N} \left(p^t_{n,k} f_{n+1}(x_k, t+1)\right)\}
\tag{3.28}
$$

3.3.4 Stochastic Quasi-Gradient Methods

The Stochastic Quasi-Gradient (SQG) methods were introduced in Ermoliev (1976), developed in Ermoliev (1983) and Ermoliev and Wets (1988), and early implemented by for instance Gaivoronski (1988). They are characterized as both a Monte-Carlo approach and an iterative sampling algorithm, and are designed to tackle nonlinear stochastic programming problems with continuous random distributions. They are numerical techniques with background in stochastic approximation, gradient projection and mathematical, algorithmic programming. In essence, instead of using exact values, they employ asymptotically consistent estimates to evaluate the functions in question and their derivatives. They are aimed at solving stochastic problems where both objective function and constraints may be of complex nature, for instance non-linear and non-convex. When applied to deterministic cases, they may be considered random search techniques (Ermoliev (1983)).

Within the field of Stochastic programming, SQG methods are considered a complement to the large scale linear stochastic programming methods(Gaivoronski (1988)). Such linear stochastic programs has been put quite an effort into and are often represented by discrete Scenario Trees and solved using Benders Decomposition. The linear approach solves the stochastic programming problem of 3.21 and 3.22 by its deterministic equivalent after discretizing the probability function of ω into a finite number of points, $\omega_i \ \forall \ i \in 1, ..., K$ where k is the number of discrete points, and replacing the spacial property of the original problem with sums, see equation 3.23. This approach is well suited for linear problems with large dimension for which the random variables are well described by discretizations of only a few points and where high precision is required and possible from these. Yet, these deterministic methods often encounter computational difficulties when the number of stages or required points of dicretization becomes moderate. This since the solution time scales with K^T (Gaivoronski (2005)), where K is the number of discrete points of the approximated probability distribution and T is the number of stages.

The SQG approach on the other hand, is well suited for problems with a smaller dimension, but where the number of stages are greater. Hence problems of dynamic character, especially stochastic dynamic optimization problems, are suitable cases for the SQG approach which solution time only scales linearly with T. The SQG approach also is well suited when the probability distributions are either complex or best described by continuous distributions, problems with many stochastic variables or problems of non-linear nature. However, due to its iterative, approximate nature, the convergence to a final, super precise solution often takes many more iterations that just coming well within the vicinity of the solution. Non-the-less this approach is of particular interest when optimizing simulation models depending on some finite number of parameters which may be challenging to pose as a linear or even non-linear programming model, and where the estimation of the underlying stochastic functions does not have high precision them self.

In the Stochastic Quasi-Gradient methodology, one solves the problem of minimizing

$$\min_{x \in X} F^0(x) = \mathbb{E}_\omega[f(x, \omega)]$$

from the objective equation in 3.20. where $\omega \in \mathbb{R}^k$ is a vector of random parameters and the decision variables are $x \in X \subset \mathbb{R}^n$. Here, X defines the set of feasible solutions. This is usually convex and simple to allow more easy computation of projection onto the set, yet more complex inequalities may in principle be used. They may for instance given by upper and lower bounds or linear constrains defined by $Ax \leq b$ like in 3.21 and 3.2. More generally the constraints may be expressed as

$$\text{s.t.} \quad F^i(x) = \mathbb{E}_\omega f_i(x,\omega) \leq 0, \quad \forall i \in 1, 2, \cdots, M.$$

as in 3.20.

As a general illustrative description, the iterative nature of the SQG approach start from an initial point x_0 and makes a step of size ρ_s in the direction opposite (in the case of a minimization problem) to the current estimate ξ^s of the gradient of $F^0(x)$ at point x_s. A more detailed description follows.

Choosing step direction

The most important part of implementing this problem is to calculate the step direction given by the estimate of the gradient, ξ^s, or the Stochastic (Quasi-)Gradient.

In notation inspired by Peeta and Zhou (2006) and Ermoliev (1983), then for a problem like 3.20 with $x \in X \leq \mathbb{R}^n$ we may assume that functions $f^v(x), v = 0, ..., m$ are convex. Then

$$F^v(z) - F^v(x) \geq (\hat{F}^v_x(x))^T(z - x) \; \forall z \in X \tag{3.29}$$

where ^{T} denotes the transpose of the vector resulting in an euclidean inner product and F^v_x is a generalized gradient, or subgradient when the function has has several group of variables it depends on. For such problems, i.e. min-max problems or two stage stochastic problems, deterministic iteration methods would approximate the solution with a sequence, $x^0, x^1, ..., x^s, ...$ using precise evaluations of the function $F^v(x)$ and its subgradient $\hat{F}^v_x(x)$ at the points of $x = x^s$.

Conversely in the Stochastic Quasi-Gradient approach, the sequence of approximations at iteration s, that is $x^0, x^1, ..., x^s$ is constructed from statistical estimates of $F^v(x)$ and $\hat{F}^v_x(x)$. These estimates are random numbers represented as $\phi^v(s)$ and random vectors represented as $\xi^v(s)$, instead of the precise values of the function $F^v(x)$ and the subgradient $\hat{F}^v_x(x)$. These estimates may well use information from the iterations process path and history and is defined by

$$\mathbb{E}(\xi^s | \mathbb{A}_s) = F^v(x^s) + a^v(s) \tag{3.30}$$

$$\mathbb{E}(\xi^s | \mathbb{B}_s) = F^v_x(x^s) + b^v(s) \tag{3.31}$$

where $\mathbb{A}_s$ and $\mathbb{B}_s$ are the σ-fields of the process history of the function and subgradient estimates and a_s and b_s are a bias terms that is to vanish asymptotically. That is

$$a^v(s) \to 0 \quad \text{and} \quad ||b^v(s)|| \to 0 \tag{3.32}$$

In the cases where this gradient is difficult to compute or does not exist in the classical manner, that is, if $b^v(s) \not\equiv 0$, the random vector ξ^s will be an gradient estimate of the appropriate generalization and is called the Stochastic Quasi-Gradient.

The resulting function value for a given iteration is found by exponential smoothing approximation, limiting the impact of previously found solutions on the current estimate whilst still letting their trace guide the value. The estimated function value is then set as

$$\hat{F}_s^0 = (1 - \gamma_s) \cdot \hat{F}_{s-1}^0 + \gamma_s \cdot F_s^0. \tag{3.33}$$

Such a smoothed estimate for the current value can also be effective for the stochastic grad as well.

When computing the Stochastic Quasi-Gradient in practice, there are two common methods. One is the use of finite differences approximation as for instance explained in Peeta and Zhou (2006) and Becker and Gaivoronski (2014). The second is to return a estimate of the gradient during the simulation process of the sub-problem. For a further discussion on this, see section 5.2.3.1 and 5.2.3.2 respectively, in chapter 5 on Methodology.

Choosing step length

Another critical task when performing the algorithm is the selection of step size ρ_s. How this is to be set is an issue that has been discussed by for instance both Pflug (1988) and Gaivoronski (1988) in the book of Ermoliev and Wets (1988). The latter proposes interactive and adaptive approaches that may utilize information from the solution process. Another approach that often is successfully in practice as noted by Becker and Gaivoronski (2014) is the use of piece-wise updated step size which changes the step size by multiplying with a predetermined factor according to

$$\rho_{s+1} = \begin{cases} r \cdot \rho_s & \text{if } s \in \Upsilon \\ \rho_s & \text{otherwise} \end{cases} \tag{3.34}$$

where Υ is the schedule for when to update the step size. For further description of such a schedule, see chapter 5 on Methodology. To facilitate the convergence of the solution though the SQG iterations, one must let the step size tend to null.

Projection on feasible set

When the estimate x_s is out of bounds, it is projected from the resulting point on to the set of X by the projection operator π so that

$$x^{s+1} = \pi_X(x_s + \rho_s \cdot \xi^s). \tag{3.35}$$

Thus, the projection operator π generates a series of points which may transform an arbitrary x' to the point $\pi_X(x') \in X$ by projecting it the shortest distance onto the feasible set X:

$$||x' - \pi_X(x')|| = \min_{x \in X} ||x' - x||. \tag{3.36}$$

As noted, when implementing and applying the the SQG method in practice, for instance as Becker and Gaivoronski (2014) has done in combination with simulation of complex dynamic network, the set of X is often a convex set of simple structure. This because the projection will possibly have to be done a great number of times during the iteration of the solution process.

3.4 Optimization of Power Systems

In this section, an basic description of Power System Analysis and Optimal Power Flow is given, as an background for the models and cases considered later in this report.

3.4.1 Power System Analysis

From the fundamentals of electrical engineering, we are well familiar with Ohm's and Kirchhoff's laws on voltage $U = I \cdot R$ and current $\sum I = 0$. Further more, we assume known that in alternating current (AC) systems for instance described in Grigsby (2012), complex power may be expressed as the sum of active and reactive power $\mathbf{S} = P + \jmath Q =$ **VI*** and also as the the multiplication of the phasors of voltage and the conjugate of current.

For ease of analysis we often transform all values to per unit [p.u.] numbers and represent the system through one-line diagrams thus simplifying it to one phase instead of three, as seen in figure 3.2 from Grigsby (2012).

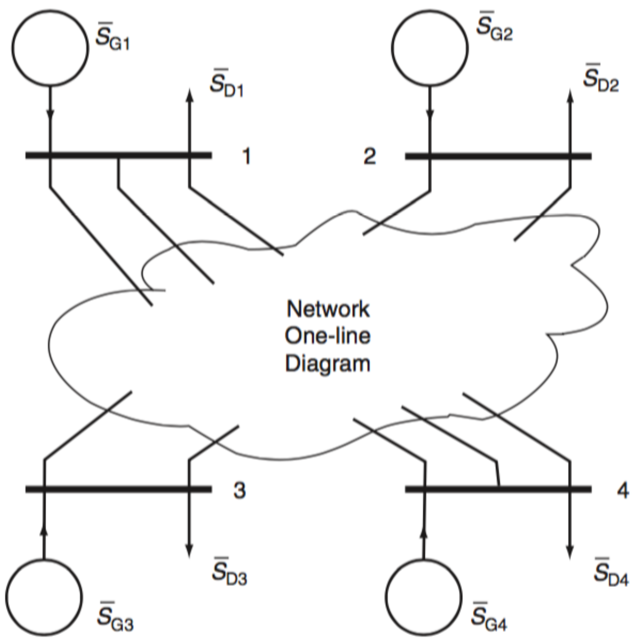

Figure 3.2: One-line system of four-bus network.

When developing the basic power flow equations for such a system, we need to consider the network topology and the characteristics (impedance, resistance) of the power lines and components. This is most often expressed in matrix notation, as with an n-bus system

$$[I_{Bus}] = [Y_{Bus}][V_{Bus}]$$

$$\begin{bmatrix} I_1 \\ \vdots \\ I_i \\ \vdots \\ I_n \end{bmatrix} = \begin{bmatrix} Y_{11} & \cdots & Y_{1i} & \cdots & Y_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{i1} & \cdots & Y_{ii} & \cdots & Y_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_{n1} & \cdots & Y_{ni} & \cdots & Y_{nn} \end{bmatrix} \begin{bmatrix} V_1 \\ \vdots \\ V_i \\ \vdots \\ V_n \end{bmatrix}$$

where $Y_{ij} = G + \jmath B$ is the admittance matrix of the network. Thus, the expression for power flow becomes

$$S_i = V_i \Big(\sum_{i=1}^{n} Y_{ij}^* \cdot V_{ij}^* \Big) \tag{3.37}$$

and

$$P_i = \sum_{j=1}^{n} |V_i||V_j| \big(G_{ij} \cos \delta_{ij} + B_{ij} \sin \delta_{ij} \big) = \sum_{j=1}^{n} |Y_{ij}||V_i||V_j| \cos(\delta_i - \delta_j - \theta_{ij}) \tag{3.38}$$

$$Q_i = \sum_{j=1}^{n} |V_i||V_j| \big(G_{ij} \sin \delta_{ij} + B_{ij} \cos \delta_{ij} \big) = \sum_{j=1}^{n} |Y_{ij}||V_i||V_j| \cos(\delta_i - \delta_j - \theta_{ij}) \tag{3.39}$$

where δ is the voltage angle and θ is the lag. As presented in Crow (2012) , one of the most common way to solve these problems are through iterations useing the NewtonRalphson Method and calculation the Jacobian

$$\begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} = \begin{bmatrix} \frac{\partial \Delta P}{\partial \delta} & \frac{\partial \Delta P}{\partial V} \\ \frac{\partial \Delta Q}{\partial \delta} & \frac{\partial \Delta Q}{\partial V} \end{bmatrix} \tag{3.40}$$

for the power flow equations, solving for the know power injections P and Q by

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |V| \end{bmatrix} \tag{3.41}$$

and finding the converging values of δ and $|V|$ through iteration.

The power flows equations from load flow studies expressed here are used as base for calculating the Optimal Power Flow of a system. Moreover, these equations clearly expresses the non-linear, non-convex physical properties that are required of our optimization methods when solving them.

3.4.2 Optimal Power Flow

The technique of Optimal Power Flow (OPF) is in essence an optimization of how power shall flow through a grid with respect to some objective function and by choosing the best values for the control variables. In many cases the objective is to minimize some cost or loss function dependent on the state variables and control variables defined in the formulation of the problem.

As seen in the previous section 3.4.1, and in for instance Cain et al. (2012), one way of formulating the OPF problem in a general manner is the following:

$$\begin{aligned} \text{Maximize} \quad & f(P) \\ \text{subject to} \quad & g(x,P) = 0 \\ & h_{min} \leq h(x,P) \leq h_{max} \\ & P_{min} \leq P \leq P_{max}, \end{aligned} \tag{3.42}$$

where $f(P)$ is the objective function. The equality constraints, $g(x,P)$, of the system expresses the physical law of power balance, or the conservation of energy, as stated in the equations 3.38 and 3.39 for active and reactive power respectively. The inequality constraints, $h(x,P)$, represents the physical limitations of the system that often are operational rules set by the operator so that power system components are not over- (or under-) loaded. Examples for such constraints are voltage limits, current limits of the transmission lines. The input and output variables P, is typically the generator values for the slack generator, and possibly other generator or buses. The variables x are the state variables.

Additionally, one has to classify the buses in the network as seen in table 3.1, to decide which variables are given in the OPF and which are used as control variables. Many books in power system analysis will cover this topic extensively, for in the book *Power Systems* by Grigsby (2012).

In the simplest version of the Optimal Power Flow problem, one assumes direct current (DC) and sett all bus voltage to 1pu. Yet, the more realistic case is an alternating current (AC) power flow, in which we also have to treat the phases of the system voltages and currents. Further detail on how these parameters and variables may be expressed is found in section 4.1 on the AC-OPF.

One of the most common ways for solving OPF problems is through the use of La-

Table 3.1: Classification of network buses

Bus type	Fixed quantities	Variable quantities	Physical interpretation
PV	real power, voltage magnitude	reactive power, voltage angle	generator
PQ	real power, reactive power	voltage magnitude, voltage angle	load, or generator with fixed output
Slack	voltage magnitude, voltage angle	real power, reactive power	an arbitrarily chosen generator

grangian multiplier discussed in section 3.2.2, which readily give insight on the price at the different nodes in the system.

3.4.3 Multistage Optimization of Power Systems

Power systems are a natural application in which to consider several time steps, and develop a multistage model for. This is especially due to the rise of increasingly more Energy Storage Systems (ESS) to be able to store variably generated renewable energy. Hence, one is interested in using this stored energy in some optimal manner, which make the models take on a dynamic character.

Other applications is a deterministic, dynamic ACOPF, often labeled DOPF, in which one solves the power flow for all time steps simultaneously coupled together by the energy storage in the power system. A review of such models may be found in Sperstad and Marthinsen (2016), as well as a comparison of the different solution methods implemented for this and their performance.

One article develop the AC-OPF to a dynamic problem, the article of Zaferanlouei et al. (2016) who presents a formulation of this problem where AC-OPF power flow with constraints as given in 3.42, 3.38 and 3.39 over a T step time period coupling the periods thought the dynamic equations of the battery, that

$$E_{ST,i}(t) = E_{ST,i}(t-1) + \Delta t \cdot \eta_{chrg} \cdot P_{Ch}(t) - \Delta t \cdot \frac{P_{Dch}(t)}{\eta_{dischrg}} \tag{3.43}$$

where $E_{ST,i}(t)$ is the energy stored at bus i at the end of time step t, $P_{Ch,i}(t)$ and $P_{Dch,i}(t)$ are the active charging and discharging power of a certain energy storage at bus i during time step t, $\eta_{Ch,i}$ and $\eta_{Dch,i}$ is the charging and discharging efficiency of the energy storage at bus i, and Δt is the amount of time between the time step increments.

Additionally, the energy stored is also clearly subject to some capacity constraints, such as

$$0 \le E_{ST,i}(t) \le E_{ST,i}^{max}. \tag{3.44}$$

and so may the discharge and charging values be too

$$\begin{aligned} 0 \quad &\le P_{Ch}(t) \quad &\le P_{Ch}^{max} \\ 0 \quad &\le P_{Dch}(t) \quad &\le P_{Dch}^{max} \end{aligned} \tag{3.45}$$

setting bounds on the dynamics of the problem and transfer of energy between states.

3.4.4 Stochastic Programming of Power Systems

The inclusion of stochastic variables in the AC-OPF, lets one directly model uncertainty as part of the problem, to find some solution that hedges against unwanted realizations.

From the notation introduced in the expressions 3.21 and 3.22 we may formulate this as

$$\begin{aligned} \min_{P_{st}} \quad & \mathbb{E}_{\xi} Q(t, P_{st}, \boldsymbol{\xi}) \\ \text{s.t.} \quad & E_{st}(t-1) \ge P_{st}(t) \ge E_{st}(t-1) - E_{st}^{max} \end{aligned} \tag{3.46}$$

where $Q(t, P_{st}, \boldsymbol{\xi})$ is the ACOPF model given the specific data at stage t given by 4.1, the storage policy P_{st} and the realization of the stochastic variables $\boldsymbol{\xi}$ at stage t.

Moreover, in this model we want to have the possibility to include generators and loads that are described by stochastic functions. For instance with a wind generator is well estimated to provide power over time according to a Weibull distribution. In such a case, we may express

$$P_{Wind}^{G}(V, t) = K \cdot V(t)^3 \tag{3.47}$$

where K is a constant and the wind velocity V at t is given thought the Weibull distribution

$$f(V) = \frac{k}{c}\Big(\frac{V}{c}\Big)^{k-1} e^{-(V/c)^k}. \tag{3.48}$$

Additionally we may want to add stochastic PV generation and house hold energy consumption, eg as fluctuations around a base production curve or load curve respectively. For instance as

$$P_L(t) = P_{avg} \pm P_{rnd}(\omega) \tag{3.49}$$

where P_{avg} is the average power for given time period (eg at noon) over the last week and $P_{rnd}(\omega)$ is some random realization for instance based on a normal distribution with $= P_{avg}$ and $\sigma(t)$ is calculated or assumed somehow. To make things relatively simple in this thesis, uniform distributions around some mean have been used for all the stochastic variables.

In this thesis, we seek to solve a multistage, stochastic program of a power system with ESS. In many regards this is a difficult problem to solve as it in principle quickly becomes rather heavy computationally. One common approach is the use of Scenario Trees and especially for the linear version of the problem - an economic dispatch problem. Another approach, is that of calculating water values using dynamic programming, or Stochastic Dynamic Programming.

However, many of these approaches scale exponentially in running time according to their level of discretization and the number of time steps, and soon become intractable. In this thesis we propose the use of the SQG method to solve the stochastic, multistage program.

3.4.5 Stochastic Multistage AC-Optimal Power Flow

In this section seeks to provide a somewhat through mathematical framework of the problem at the heart of the thesis; solving an AC-OPF for several time periods under the influence of stochastic variables. As this is a quite complex problem, containing non-linearity, non-convexity, dynamic and stochastic variables, it is not an abundance of literature that has implemented and tested this type of formulation.

We may borrow some of the notation and principles from the recourse problem in 3.21 and the scenario representation in 3.22, as we both have some first-stage decisions that we make before knowing the realization of the stochastic variables and cannot change, and corrective actions made after knowing the realization. However, we also have multiple steps as in 3.23 that we need to take into account.

Here a mathematical specification is given in a compact and general form, in the framework of the recourse problem. This is not a necessary framework for the problem formulation, but is included as it might aid the reader to conceptually understand how this formulation is constructed based on the previous discussion in 3.21. From this formulation will be expanded to more specific ones in 4.4, which one may base a SQG implementation of the problem on or expand the problem into suitable extensions. The details of the AC-OPF are simplified and left out for now, but will be stated in more depth in for instance Chapter 4. Generally we may say that our objective is to minimize the total cost of the energy system, by charging (negative $P_{Battery}(t)$) and discharging (positive $P_{Battery}(t)$) the battery optimally. It may be expressed as

$$\begin{aligned} \min_{P_{Battery}} \quad & \sum_{t=1}^{T} c_B P_{Battery}(t) + \mathbb{E}_{\xi} C(t, P_{Battery}, \boldsymbol{\xi}) \\ \text{s.t.} \quad & E_{st}(t-1) \geq P_{Battery}(t) \geq E_{st}(t-1) - E_{st}^{max} \end{aligned} \tag{3.50}$$

where the constant c_B represents the cost of utilizing the battery and C represents the cost of the energy production and loss in the grid. The bounds on $P_{Battery}(t)$ is for the charging restricted by how much energy, $E_{st}(t-1)$, is stored at the end of the last time step $t-1$, and for discharging how much energy is available left to disperse during time step t, $E_{st}(t-1) - E_{st}^{max}$.To find the energy stored in the battery at a given time step, we calculate

$$E_{st}(t) = E_{st}(t-1) - P_{Battery}(t) \tag{3.51}$$

and have $E_{st}(0)$ as a given value. Note, that we have left out the distinction between active and reactive power here, and only represented power production by P.

The decision is then to solve the deterministic AC-OPF minimizing the cost of a gas generator given the decided discharging of the battery $P_{Battery}(t)$ and realization of the stochastic variables in $\boldsymbol{\xi}$.

The problem for this case may in a very simplified manner be formulated as

$$\begin{aligned} \min_{P_{Gas}} \quad & \mathbb{E}_{\xi} c_G \big(P_{Gas}(t, P_{Battery}, \boldsymbol{\xi}) \big) \\ \text{s.t.} \quad & P_{Load} + P_{loss}(t, P_{Battery}, \boldsymbol{\xi}) - \sum_{g \subset G} P_g(t) = ... \\ & P_{Gas}(t, P_{Battery}, \boldsymbol{\xi}) \mid P_{Battery}(t), \boldsymbol{\xi}(t) \\ & P_{Gas}(t, P_{Battery}, \boldsymbol{\xi}) \geq 0 \end{aligned} \tag{3.52}$$

where g is the subset of power producing units G except the gas generator. The energy balance constraint here is thus a simple representation of the whole AC-OPF problem, with all its equities and inequality constraints.

Chapter 4

Mathematical Models

In this chapter, the problems sought to solve in this report are presented mathematically, with assumptions, notation and the optimization model. Where relevant, alternative or extended formulation is also considered.

The chapter starts by presenting the fundamental AC-OPF formulation used in this thesis, which lays the foundation for the simulations of the energy system with both energy storage and stochastic variables to be analyzed later. Thereafter a model in which the energy storage is included follows; a discrete dynamic AC-OPF. This is then contrasted with a formulation of a AC-OPF with uncertainty, but without the dynamic aspect, before another model is presented combining the two. Under the consideration of the last model with both a stochastic and dynamic aspect, a couple of alternative formulations are also considered, and a algorithm for a special case for the projection operator is developed.

Here, only the general mathematical models are presented. For a further discussion on the specific models for the different cases and their implementation, see chapter 5 and 6 respectively.

4.1 Basic AC-Optimal Power Flow Models

The technique called an Alternating Current Optimal Power Flow (ACOPF) is used to calculate how generators best produce their power given cost and losses due to resistance in the transmission grid, accounting for both active and reactive power.

First we introduce a general deterministic model for AC-Optimal Power Flow.

4.1.1 Assumptions

As standard for most Power Flow studies we assume as earlier that:

1. A balanced loading of the three phases in the power system.
2. The power system operates in steady-state.
3. Constant system frequency.
4. Power demand and production except in the slack bus is assumed known.
5. No energy storage is allowed in the system.
6. Prices is assumed to be constant for the basic model, but may be dynamic in extended formulations, for instance a reaction to the shadow prices of the power bidding of the different producers.

The first assumption here, allows us to simplify the parts using π-diagrams and constructing a single-line model of the system. This makes it simpler when building the mathematical model of transmission lines with corresponding impedance, buses, generators and loads. The second means that we disregard transient and dynamic behavior of the system, for instance in the case of lost loads, generator tripping, short circuits or other contingencies. This is also a pre-requisite of the third assumption, which also lets us disregard generator drooping and frequency response here.

4.1.2 Notation

The following general notation is applied in the mathematical model:

Indicies:

i, j Bus, or node, indicies

Sets:

G Distributed generators in the network
L Distributed loads in the network
N Number of buses in the network

Parameters:

$I_{line,rated}$	Rated capacity for current in the lines.
$Q_{G,i}^{min}, Q_{G,i}^{max}$	Minimum and maximum reactive power capacity of the generators.
V_{min}, V_{max}	Minimum and maximum voltage amplitude.
$\delta_{min}, \delta_{max}$	Minimum and maximum voltage angle.
P_i^L, Q_i^L	Demand, or load, for active and reactive power at a given bus i.
$Y_{i,j}, \theta_{i,j}$	The admittance and angle for the line between bus i and bus j in the system.
C_i^G	Price of production with generator at bus i.
$V_{slack} = 1p.u., \delta_{slack} = 0$	Voltage amplitude and angle of a bus.

Variables:

P_{slack}^G, Q_{slack}^G	Active and reactive power production from the slack bus generator.
V_i, δ_i	Voltage amplitude and angle of a bus.

4.1.3 Model

A general AC-Optimal Power Flow model is presented below, minimize cost of all generator, whilst upholding energy balance and physical constraints for all buses.

$$\begin{aligned}
\text{min} \quad & \sum_{i \in G} C^G \cdot P_i^G \\
\text{s.t.} \quad & P_i^G - P_i^L = \sum_{j=1}^{N} |\mathbf{V_i}||\mathbf{V_j}||\mathbf{Y_{i,j}}| \cos(\delta_j - \delta_i + \theta_{i,j}) \\
& Q_i^G - Q_i^L = \sum_{j=1}^{N} |\mathbf{V_i}||\mathbf{V_j}||\mathbf{Y_{i,j}}| \sin(\delta_j - \delta_i + \theta_{i,j} \\
& Q_i^{min} \leq Q_i \leq Q_i^{max} \\
& V_{min} \leq V_i \leq V_{max} \\
& |\mathbf{I}_{line,(i,j)}| \leq I_{line,rated} \\
& \delta_{min} \leq \delta_i \leq \delta_{max} \\
& i, j \in N.
\end{aligned} \tag{4.1}$$

In this model, the objective function is to minimize the cost of running the generators for a given production cost. The first two equality constraints are the power flow equations in this Alternating Current Network. It is derived from Kirchhoff's current law, which says that current in and out of every bus has to be equal The inequality constraints that follow represents physical limits in the power network. The firs inequality, states the upper and lower limits of reactive power that is possible to produce or consume at the different distributed generators. The second inequality, are the limits on how high or low voltage can be. This is usually within 0.95 and 1.05 p.u. - that is, within 95-105%. The line constraints are the line current between bus i and j being derived from Ohm's law $U = RI$ and here expressed by

$$\mathbf{I}_{line,(i,j)}(t) = \frac{\mathbf{V}_i - \mathbf{V}_j}{\mathbf{Y}_{i,j}}. \tag{4.2}$$

This model may be solved by many approaches, such as Lagrangian Multipliers or an iterative Newton-Raphson method, or the Interior Point Method that we use here.

4.2 Dynamic AC-Optimal Power Flow with Energy Storage

When one is to consider energy storage in power system optimization, such as letting the models incorporate batteries or pump-hydro power plants, the mathematical formulation takes on a dynamic character. This is an effect of the energy storage equations needed to be introduced, as seen in 3.43. For this thesis, the time increments will almost always being 1 hour, and thus left out. Moreover, one needs to constraint the minimum and maximum power of the battery as seen in 3.44, and possibly also it upper and lower charging and discharging values.

Such equations will be introduced in the following dynamic models, to connect the AC-OPF solution of each time step though the storage dynamics.

In this section a model is formulated which includes the possibility to store some energy in for instance a battery or a hydro power dam, and hence the model undertakes a dynamic character.

4.2.1 Assumptions

First we make some new assumptions.

1. The power demand and production except in the slack bus is assumed known by its stochastic distribution functions.

2. Energy storage is allowed in the system, but only supplies active power.

3. Prices is assumed to be constant for a certain time period for this model, and are not dependent on the dynamics of the model.

4.2.2 Notation

We now update the general notation from the previous model to having the possibility to be different for different time periods, and include the following new notation:

New indicies:

t Time instance

New sets:

T Time intervals

New parameters:

$P_{Ch,i}^{max}, P_{Dch,i}^{max}$	Maximum active charging and discharging power of a certain energy storage at bus i.
$E_{ST,i}^{max}$	Rated energy capacity of the energy storage at bus i.
$\eta_{Ch,i}, \eta_{Dch,i}$	Charging and discharging efficiency of the energy storage at bus i.
Δt	Time step increment.

New variables:

$P_{Ch,i}(t), P_{Dch,i}(t)$	Active power charging and discharging from the energy storage at bus i for time instance t.
$E_{ST,i}(t)$	Energy stored at time step t, which is given by the charge and discharge during t.

4.2.3 Model

Incorporating these new assumptions and variables into the model, we have:

$$
\begin{aligned}
\min \quad & \sum_{t=1}^{T}\sum_{i\in G} C_i^G(t)\cdot P_i^G(t)\cdot \Delta t \\
\text{s.t.} \quad & P_{G,i}(t) - P_{L,i}(t) + P_{Dch,i}(t) - P_{Ch,i}(t) = ... \\
& \quad \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \cos(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t) \\
& Q_{G,i}(t) - Q_{L,i}(t) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \sin(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t) \\
& Q_{G,i}^{min} \leq Q_{G,i}(t) \leq Q_{G,i}^{max} \\
& V_{min} \leq V_i(t) \leq V_{max} \\
& |\mathbf{I}_{line,(i,j)}(t)| \leq I_{line,rated} \\
& 0 \leq P_{Ch}(t) \leq P_{Ch}^{max} \\
& 0 \leq P_{Dch}(t) \leq P_{Dch}^{max} \\
& E_{ST}(t) = E_{ST}(t-1) + \Delta t \cdot \eta_{chrg,i} \cdot P_{Ch,i}(t) - \Delta t \cdot \frac{P_{Dch,i}(t)}{\eta_{dischrg,i}} \\
& \forall t \in [0, t, 2t, ..., T] \quad \text{and} \quad i, j \in N.
\end{aligned}
\tag{4.3}
$$

In this model, the objective function is to minimize the cost of running the generators, for a given production cost and over all time periods.

4.3 Stochastic AC-Optimal Power Flow

When introducing uncertainty to the AC-OPF, some of the variables in the system will be represented by stochastic variables. Information on how these variables might be included, and thus a general formulation of the problem, might be found in section 3.4.4 and in model 3.46.

4.3.1 Model

With power production and loads given by some stochastic realization, the problem may be formulated in the same manner as in the deterministic case, only where the power values are random and the method tries to minimize the expected value of the objection function. In the model below, the random values are denoted as depending on the stochastic realization $\boldsymbol{\xi}$.

$$
\begin{aligned}
\min_{P_B, P_G} \quad & \mathbb{E}C\Big(P_G(t,\boldsymbol{\xi}) \mid P_B(t)\Big) \\
\text{s.t.} \quad & P_i(t,\boldsymbol{\xi}) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \cos\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& Q_i(t,\boldsymbol{\xi}) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \sin\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& P_i^{min} \le P_i \le P_i^{max} \\
& Q_i^{min} \le Q_i \le Q_i^{max} \\
& V_{min} \le V_i \le V_{max} \\
& \delta_{min} \le \delta_i \le \delta_{max} \\
& |\mathbf{I}_{line,(i,j)}| \le I_{line,rated} \\
& \forall i \in [1, .., N].
\end{aligned}
\tag{4.4}
$$

4.4 Stochastic Multistage AC-Optimal Power Flow

In this section seeks to provide a somewhat through mathematical framework of the problem at the heart of the thesis; solving an AC-OPF for several time periods under the influence of stochastic variables. An introduction and general formulation to this type of problem might be found in 3.4.5.

Note that a very important assumption for this model is that after deciding how much power to charge or discharge at the start of the period before the realized wind energy is known, this decision cannot be altered. Thus, $P_{Battery}(t)$ is fixed during the time step, in which the wind generation is realized and gas generator supplies the remanding energy demanded. Hence, it is not possible to suddenly decide to charge more if the wind power generated is realized to be in the upper range of its possible values. The power balance for each time step is assured by the gas generator, and not to be influenced by sudden change in charging policy during the time step.

4.4.1 Model

The decision variable for each singe time step is to determine how much power the dispatchable gas generator has to supply in order to make up for the remaining demand are are not met by the other power sources as well as covering all physical losses P_{loss} of power in the system. We assume that using the generator c_G is much more costly than the other power sources. We therefore seek how to optimally charge the battery so that it charges when energy is in abundance, and discharges when it is in shortage.

A model for this may be formulated as follows:

$$\min_{P_B, P_G} \quad \mathbb{E} \sum_{t=1}^{T} C\Big(P_G(t, \boldsymbol{\xi}) \mid P_B(t)\Big)$$

$$\begin{aligned}
\text{s.t.} \quad & P_i(t) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \cos\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& Q_i(t) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \sin\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& P_i^{min} \leq P_i \leq P_i^{max} \qquad (4.5) \\
& Q_i^{min} \leq Q_i \leq Q_i^{max} \\
& V_{min} \leq V_i \leq V_{max} \\
& \delta_{min} \leq \delta_i \leq \delta_{max} \\
& |\mathbf{I}_{line,(i,j)}| \leq I_{line,rated} \\
& E_{ST}(t-1) - E_{ST,2}^{max} \leq P_B(t) \leq E_{ST}(t-1) \\
& E_{ST}(t) = E_{ST}(t-1) - P_B(t) \\
& \forall t \in [1, 2, ..., T] \quad \text{and} \quad \forall i \in [1, .., N].
\end{aligned}$$

4.4.2 Reformulation of the decision variables to SOC

Where the implementation of the energy storage constrains in the S-MS-AC-OPF formulation from 4.5 is not as simply bounded as might be thought at first, another approach would be to reformulate the model in terms of SOC. That means, in stead of letting the decision parameter be how much to charge at a given time step, one lets it be the decision of what state of charge, or stored energy, one is to have at each time step.

In effect this becomes the same decision, as the charging or discharging of the battery for each time step is the difference of the energy levels of the battery at the current and the previous time steps. That is

$$P_{Battery}(t) = E_{ST}^{max} * \big(SOC(t-1) - SOC(t)\big) = E_{ST}(t-1) - E_{ST}(t) \qquad (4.6)$$

where

$$SOC(t) = \frac{E_{ST}(t)}{E_{ST}^{max}}. \qquad (4.7)$$

The benefit of this formulation is that the constrains on the decision variable becomes very simple, it reduces to only upper and lower bounds of no energy storage and maximum energy storage - the battery capacity E_{ST}^{max}. This it should be very easy for the SQG solver to project a step that is out of the feasible area, down onto the set X of feasible values for x as described in equations 3.35 and 3.36.

We may express the reformulated version of the stochastic, multistage AC-OPF optimization as

$$\begin{aligned} \min_{SOC} \quad & \sum_{t=1}^{T} \mathbb{E}_{\xi}\Big(\min_{P_G} c_G\big(P_G(t, P_B, \boldsymbol{\xi})\big)\Big) \\ \text{s.t.} \quad & P_L + P_{loss}(t, P_B, \boldsymbol{\xi}) - \big(SOC(t) - SOC(t-1)\big) - P_G(t, \xi) \\ & = P_G(t) \mid P_B(t), \boldsymbol{\xi}(t) \\ & 0 \leq SOC(t) \leq SOC^{max} \end{aligned} \qquad (4.8)$$

where the aim is to find the power to compensate with using the generators P_G, given the decided battery policy and all the other variables and realizations for that time step.

The equality in the formulation 4.8 above is a simple representation of the AC-OPF problem constraints. The reformulated, fully expanded version of the stochastic, multistage AC-OPF model becomes

$$
\begin{aligned}
\min_{E_{ST},P_G} \quad & \mathbb{E}\sum_{t=1}^{T} C\Big(P_G(t,\boldsymbol{\xi}) \mid E_{ST}(t), E_{ST}(t-1), \boldsymbol{\xi}(t)\Big) \\
\text{s.t.} \quad & P_i(t) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \cos\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& Q_i(t) = \sum_{j=1}^{N} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \sin\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& P_i^{min} \le P_i \le P_i^{max} \\
& Q_i^{min} \le Q_i \le Q_i^{max} \\
& V_{min} \le V_i \le V_{max} \\
& \delta_{min} \le \delta_i \le \delta_{max} \\
& |\mathbf{I}_{line,(i,j)}| \le I_{line,rated} \\
& P_B(t) = E_{ST}(t-1) - E_{ST}(t) \\
& 0 \le E_{ST}(t) \le E_{ST}^{max} \\
& \forall t \in [1, 2, ..., T] \quad \text{and} \quad \forall i \in [1, .., N].
\end{aligned}
\tag{4.9}
$$

The SQG solver here starts by deciding the states for the battery for all the time steps, before the resulting charging is calculated given a initial state of charge and passed to the AC-OPF simulation. Again the objective results are added up, and sent back to the SQG solver for further iterations.

4.4.3 Reformulation of the decision variables to charging policy rules

We now reformulate the decision variables to be some type of charging policy.

4.4.3.1 Charging rules based on inherited state of energy

One approach is again to use Energy level at the time steps as the focus of the decisions, whereas the second is to use the energy to charge.

4.4.3.2 Using energy stored as the parameter

We define the following variables:

Variables:

$x_{0,1}...x_{0,n}$	Decided state of charge to go to
$x_{1,1}...x_{1,n-1}$	Threshold value for deciding to go to a corresponding state, given the energy level at the start of the period.
z_{t-1}	Energy level at the end of last period
n	Level of discretization
N_{Buses}	The number of buses

which have the following relations

$$\begin{aligned} x_{0,1} \quad & \text{State to go to } \textbf{if} \quad 0 \leq z_{t-1} < x_{1,1} \\ & \vdots \\ x_{0,k} \quad & \text{State to go to } \textbf{if} \quad x_{1,k-1} \leq z_{t-1} < x_{1,k} \\ & \vdots \\ x_{0,n} \quad & \text{State to go to } \textbf{if} \quad x_{1,n-1} \leq z_{t-1} \leq 1 \end{aligned} \tag{4.10}$$

as well as

$$\begin{aligned} & 0 \leq x_{i,j} \leq 1 \quad \forall i = 1, j = 1, ..., n \quad \text{and} \quad i = 2, j = 1, ..., n-1 \\ & x_{0,j} \leq x_{0,j-1} \quad \forall i = 1, j = 2, ..., n \\ & x_{1,j} \leq x_{1,j+1} \quad \forall i = 1, j = 1, ..., n-2 \end{aligned} \tag{4.11}$$

Moreover, the charging will then be decided as the difference between the $x_{0,l}$ decided on for that time step given the z_{t-1}. That is

$$P_B = E_{st}^{max} \cdot (x_{0,l} - z_{t-1}) \tag{4.12}$$

and the new state of charge at the end of the time step will be given by the chosen policy. Thus $z_t = x_{0,l}$, where subscript l denotes the chosen policy for that times step.

The model then becomes:

$$
\begin{aligned}
\min_{\mathbf{x}, P_G} \quad & \mathbb{E} \sum_{t=1}^{T} C\Big(P_G(t, \boldsymbol{\xi}) \mid P_B(t, \mathbf{x})\Big) \\
\text{s.t.} \quad & P_i(t) = \sum_{j=1}^{N_{Buses}} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \cos\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& Q_i(t) = \sum_{j=1}^{N_{Buses}} |\mathbf{V_i}(t)||\mathbf{V_j}(t)||\mathbf{Y_{i,j}}(t)| \sin\big(\delta_j(t) - \delta_i(t) + \theta_{i,j}(t)\big) \\
& P_i^{min} \leq P_i \leq P_i^{max} \\
& Q_i^{min} \leq Q_i \leq Q_i^{max} \\
& V_{min} \leq V_i \leq V_{max} \\
& \delta_{min} \leq \delta_i \leq \delta_{max} \\
& |\mathbf{I}_{line,(i,j)}| \leq I_{line,rated} \\
& \forall t \in [1, 2, ..., T] \quad \text{and} \quad \forall i \in [1, .., N_{Buses}].
\end{aligned}
\tag{4.13}
$$

To implement this in the SQG solver, one needs to represent the inequalities in simple matrix form, such as

$$
\begin{bmatrix}
-1 & 1 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & -1 & 1 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 & 1 & -1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 1 & -1
\end{bmatrix}
\cdot
\begin{bmatrix}
x_{0,1} \\ x_{0,2} \\ \vdots \\ x_{0,n-1} \\ x_{0,n} \\ x_{1,1} \\ x_{1,2} \\ \vdots \\ x_{0,n-2} \\ x_{0,n-1}
\end{bmatrix}
\leq
\begin{bmatrix}
0 \\ \vdots \\ 0
\end{bmatrix}
\tag{4.14}
$$

4.4.3.3 Projection of policy decision variables onto its feasible set

For the SQG process, we often want the decision variables to be constrained in a very simple manner, as discussed in 3.3.4. The reason we are so concerned about the form of the constraints of the decision variables, x, that the SQG method is to optimize with respect to, has to do with the projection onto the feasible solution space. When one takes a step that is outside the feasible space of X, the algorithm should project the point onto the feasible region given by the constraints on x. In general terms, that means to find the point on the surface of the feasible region that is shortest distance from end point of step s, x^{s+1}. Thus, one straight forward way to find this would be to solve a quadratic program where on minimizes the distance squared:

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & (x^{s+1} - x)^2 \\
\text{s.t.} \quad & Ax \leq b \\
& 0 \leq x \leq 1
\end{aligned}
\tag{4.15}
$$

However, if the feasible region is complex, finding projection point that gives the shortest distance might take a little time and slow down the overall solution time. Thus it is of interest to see if one have a more explicit way to project an estimated point from the gradient step onto the feasible set.

In our case, it should be possible to do so very simply. Consider a point defined by a vector v consisting of n different numbers specifying its coordinates in the $\mathbb{R}^n$ solution space. For it to be a feasible solution it should be constructed in such a manner that

$v_1 \geq v_2 \geq v_3 \geq ... \geq v_n$, as it is for variables $x_{0,j}$ above. Then, to find the shortest way onto the hyper-plane surface in dimension $\mathbb{R}^n$ given by

$$\begin{aligned} v_1 &\geq v_2 \\ v_2 &\geq v_3 \\ &\vdots \\ v_{n-1} &\geq v_n \end{aligned} \tag{4.16}$$

to the point specified by v, we may start by considering a case in which only one of the constraints 4.16 are breached. In other terms, where some coordinate in the l^{th} dimension, v_l, is greater than v_{l-1}, yet all $v_{l+1}, ..., v_n$ are smaller or equal to v_{l-1}. Then one would project the point v onto the surface of $\mathbb{R}^{n-1}\mathbb{R}^n$ that is given by $v_l \leq v_{l-1}$. This is done by finding the first constraint it violates, that is the first number v_m that v_l is greater than. Here, being the "first" number refers to the first number of a coordinate of v when going though the coordinates v_i from $i = 1$ to $i = n$. Once the first number that v_l violates is found, the point on the surface of minimum distance to v is given by the point in the middle of $v_i = v_m \forall i \in m, ..., l$ and $v_i = v_l \forall i \in m, ..., l$ where the rest of v is unchanged.

Now, consider that the point of v violates the two of the constraints 4.16, as described above. That is, where all other coordinates are less than the previous ones except for two. Call these v_{l1} and v_{l2}, where v_{l1} is the first of the two. We then project this point onto the surface of $\mathbb{R}^{n-2}\mathbb{R}^n$ given by the constraints corresponding to the two numbers who are outside bounds. This time, it depends on which of the two numbers that violates the constraints are the greatest. If the second of those two, v_{l2}, is the greatest, the smallest distance will as before be found as the half-way point in the feasible region that lies between $v_i = v_{m2} \forall i \in m2, ..., l2$ and $v_i = v_{l2} \forall i \in m2, ..., l2$, where v_{m2} is the first number v_{l2} is greater than. However, if the first of the two violating points, v_{l1}, is greater, then point in the feasible region with shortest distance to v, is given as follows. The value of coordinates $m1, ..., l1$ are equal to $(v_{m1} + v_{l1})/2$ where v_{m1} is the first number that v_{l1} is greater than, and the values of coordinates $m2, ..., l2$ are equal to $(v_{m2} + v_{l2})/2$ where v_{m2} is the first number that v_{l2} is greater than after the values of coordinates $m1, ..., l1$ are changed.

Similarly, if a point v has k coordinates that all are bigger than the constraint imposed by any of the former coordinates, they are projected with shortest distance onto the surface in $\mathbb{R}^{n-k}\mathbb{R}^n$ defined by their corresponding inequalities, by the logic from the section above. Any violating coordinate of the point v that comes before the greatest number of $v_{l,greatest}$ will be affected when all points between $v_{m,greatest}$ and $v_{l,greatest}$ are changed to the half-way point. Any points following $v_{l,secondgreatest}$ will be affected when it finds half-way point between it and $v_{m,secondgreatest}$. Moreover, if any of the coordinates of v happens to be outside some upper or lower bound specified for all v_i, then a simple projection onto those bounds could be done either before the projection process onto the constraints 4.16 or after.

Thus, the projection of policy x of the model defined by 4.13, may use the projection method described above. However, to make things simpler, we let the state to go to for the different thresholds only be constrained by upper and lower bounds. Even though it would be relative easy making the values $x_{0,j}$ decreasing as j goes from i to n, a relaxed approach may make more sense, allowing the states of the thresholds to be more or less than each other. This because it might not always be so that you should charge the most when you are on lowest energy. The reason you might be low on energy might be because it is a time where energy from other sources than the dis-patchable generator might be low and it would be costly then to charge a lot. Instead, it might be wiser to charge a little, or stay in the same state until the shortage passes. Then the constraints of **??** instead becomes:

$$\begin{aligned} &0 \leq x_{i,j} \leq 1 \quad \forall i = 0, j = 1, ..., n \quad \text{and} \quad i = 1, j = 1, ..., n-1 \\ &x_{1,j} \leq x_{1,j+1} \quad \forall i = 1, j = 1, ..., n-2 \end{aligned} \tag{4.17}$$

Yet, it should also be noted that for the policy decision model treated here, any given policy will give a deterministic path of state of charges between the different time steps given an initial energy state. This because the inherited energy z is not affected by any of the stochastic realizations during the process, since the charging of the battery is set at the start of a time step and only generation from the generator is allowed to change during the time step in the current formulation. However, this could be changed, for instance by making the decision variable of the SQG iterations should be the power produced by the despicable generator.

4.4.3.4 Charging rules based on state of energy and stochastic realizations of the last step

Another approach would be to formulate the policy decisions in a manner so that the thresholds depend on the random variables realized in the previous step. Yet it should also reflect the state of energy of the battery. This could be done in a combinatory manner eg. by defining a two-dimensional matrix which holds the desired state to go to given a combination of current energy state and stochastic realizations. This would in many ways assimilate the concepts of transition matrix and cost-to-go matrix of the SDP approach.

Another, more straight forward manner to to this, would be to let the value of z represent the summation of some aggregate of the stochastic realization and the inherited energy from last step. More explicitly, we add the generated power of last step P_{Gen} of the dispatchable generator, and add the energy inherited from last step. Formally this is:

$$z = P_{Gen}(t-1) - E_{st}(t-1) \tag{4.18}$$

Since the realization of the stochastic variables, either it is energy production or consumption or both, is more or less directly reflected in the energy produced by the dispatchable generator that time step. The only difference by taking a sum of the realized variables and the resulting dispatchable energy generation is the losses of the gird and the energy discharged or charged by the battery. The first is only handy to have as part of the z value, since the losses indirectly encodes the distribution of the stochastic sources and consumers in the grid and adjust it according to the impact their geographical - or topological - placement in the grid has on the total power balance.

The energy $Est(t-1)$ of the end of time step $t-1$, says something about the ability to charge or discharge in this time step t, and is one reason for why it makes sense to include it in the summation. The summation between the two is there since a higher generated power $P_{Gen}(t-1)$ should indicate that it is desirable to discharge the battery, and a higher $Est(t-1)$ indicates that the state to go to should be lower. Moreover, by subtracting the energy at the end of the last time step from the energy produced by P_{Gen}, we effectively eliminate the effect the charging or discharging done in time step $t-1$ has on the energy produced by P_{Gen}. To see this, consider a battery had a energy level of 50 at the end of time step $t-2$, discharging either 10 or 20 resulting in a energy level of 40 or 30 respectively. Then, assuming losses are not present, say the energy produced by P_{Gen} with these discharges was 90 and 80, respectively, where the lower energy production of the second results form the higher discharge in this case. Then the value of z in both cases is $z = 90 - 40 = 80 - 30 = 50$. This then satisfies the second concern mentioned in the paragraph above, making z directly represent an aggregated value of the realization of the stochastic variables, favorably adjusted for the effect of their spacial distribution has on the grid.

The model from 4.13 remains the same, however the definition of the variables are somewhat altered.

New variable definitions:

$x_{1,1}...x_{1,n-1}$	Threshold value for deciding to go to a corresponding state, given the energy level at the start of the period and energy production of last time step.
z_{t-1}	An aggregate measure of the realizations of the stochastic variables in last time step.

The relation between the threshold values and policy decisions expressed in 4.10 stays the same, and we also here the relaxed constraints of 4.17. However, the bounds on the threshold values $x_{1,j}$ needs to be changed to

$$\begin{aligned} &0 \leq x_{1,j} \leq P_{Gen}^{max} \quad i = 1, j = 1, ..., n-1 \\ &x_{1,j} \leq x_{1,j+1} \quad \forall i = 1, j = 1, ..., n-2 \end{aligned} \tag{4.19}$$

and the the discharge of the battery must be updated to

$$P_B = E_{st}^{max} \cdot (x_{0,i}) - E_{st}(t-1) \tag{4.20}$$

where $x_{0,i}$ is the decided level of charge to go to for times step t dependent of the z of that time step.

Chapter 5

Methodology and Cases

Here, an overview of the methodology and test cases is presented, to aid the further discussion of the models in chapter 4, implementation of these in chapter 6 and their results in chapter 7.

5.1 Cases

To show how relevant the SQG method might be in solving the AC-OPF for several stages and under uncertainty, the implemented models were tested with two cases. The first of these was specifically constructed to be as simple as possible whilst still maintaining the key aspects of interest. This was then chosen to be a fully connected four bus system with only one source of uncertainty. The second case was intended to be a little more complex and with several sources of uncertainty, to fully investigate the capability of the SQG method. For this case a standard IEEE case was chosen and modified to obtain the desired properties.

Specific data for the cases is found in appendix B.2.

5.1.1 4 bus power system

For the implementation of the AC-Optimal Power Flow (ACOPF) model described in 4.1, we need a power grid network on which to model the energy system. This should be a system in which all physical properties, er resistance and impedance of lines, voltage levels and maximum current of lines, are probable and provides realistic results when solved.

In the 4 bus case - with four network nodes - which we consider first, we try to keep the problem as simple as possible. In that manner, we for instance let both the possible cost of wind generation, but more importantly for our formulation, the battery cost to be null. Note here, that neither the wind or the battery generates or consumes reactive power $Q_{L,4}(t)$. Thus the generator has only to supply reactive power $Q_{G,1}(t)$ according to the demanded reactive power and the phase changes across the network cables during the specific solution. The wind power generated is what is under stochastic influence in this case, and is given as by a uniform distribution with a specific range around a given mean wind energy for the time step.

The author constructed a trial case of a 4 bus system as presented in figure 5.1 below, to be used as an example case for initial implementation and testing.

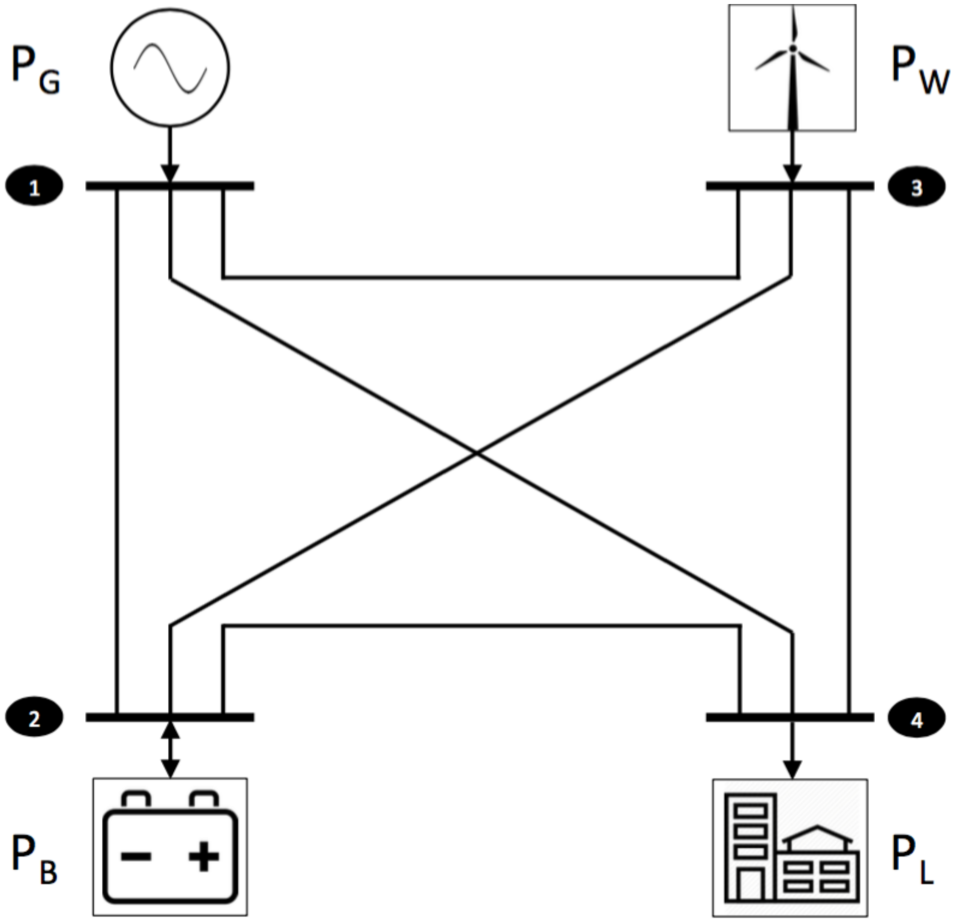

Figure 5.1: Illustration of 4 bus case.

Each power bus, or busbar, is represented by a node, and all buses are connected. Each bus is also connected to a power generator, consumer or storage.

- **Bus 1** is connected to a generator (e.g. coal, gas), and is the slack bus in the operation. That means it will produce the required amount of power to meet the demanded load, given the wind energy production and charging or discharging of the battery.

- **Bus 2** is connected to a battery park, that stores energy. It may charge or discharge its energy from or to the power grid. How much to charge or discharge is the policy decision to be made in the SQG-solved problem.

- **Bus 3** is connected to a wind park. It produces wind energy, but the exact amount is uncertain. It is as of now considered to be given as a random number in a uniform distribution.

- **Bus 4** is connected to households and industry consumers, which represents the power demand in the system. It is considered known and not stochastic in this specific case.

The data for the specified case can be found in the tables **??**, B.2 and B.3 in appendix B.1.1.

5.1.2 9 bus power system

The benefit of the SQG method was also to be tested more thoroughly by implementing it for a more complex case with an increased amount of stages. Setting up such a system, it was also thought to be desired in a manner that also is comparable to other researchers modelling results, we use data provided from the IEEE nine-bus system, as described in for instance Ariyo (2013) and presented in the figure 5.2 from the paper. However, we adjust the system so that its loads are stochastic, and it also has variable generation in PV and wind power.

In the 9 bus case, where the network has 9 nodes, we let three of the buses be transformers like in the original case, three of them be power generating buses, two be load buses ant the last be the energy storage E_{ST} that may discharge (or charge) power to the grid with $P_{Battery}$. Of the generating buses, one is a slack bus with a dis-patchable gas generator P_{Gas} that will compensate for what ever energy demanded that is not met by the production at any given instance. The two others will be a wind generator park, P_{Wind} as before, and a PV plant P_{PV}, which both will have stochastic production. As for the loads, we let one represent households $P_{Household}$ and the other businesses or industry $P_{Business}$, both with stochastic demand. In this case take note that the cost function for the generator here is a quadratic function $c_G = a * P_{Gas}^2 + b * P_{Gas} + c$ with no minimum point in the upper right quadrant, and hence the optimal solution should try to reduce the peak generation of the gas generator, by filling some of the throughs. The question is again how much the and let battery shall charge and discharge during the simulation horizon:

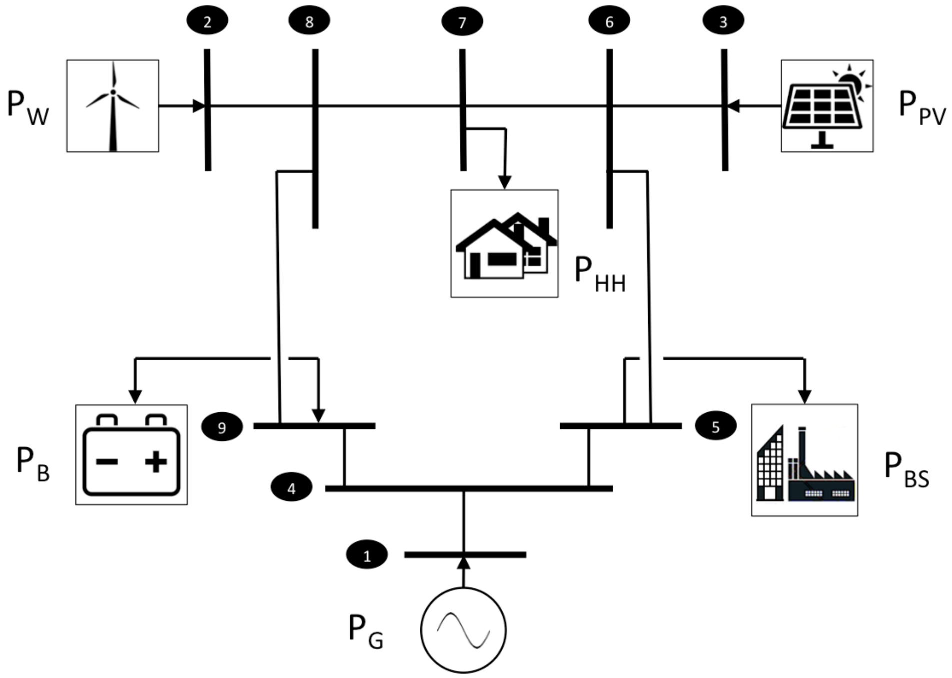

Figure 5.2: Modified IEEE nine-bus system.

The data for the 9 bus case can be found in the tables B.4 and B.5 in appendix section B.1.2.

For this case the mean wind and PV generation follows the curves presented in figure 5.4, and the mean power load from the Business and Household is presented in 5.4. For simplicity, the values they may realize to for each time step are all assumed to have a uniform distribution around their mean for that time step. The range of these distribution has for this case been chosen as ±25% for wind, ±10% for PV whenever the mean is more than null, ±5% Business and ±10% for Household.

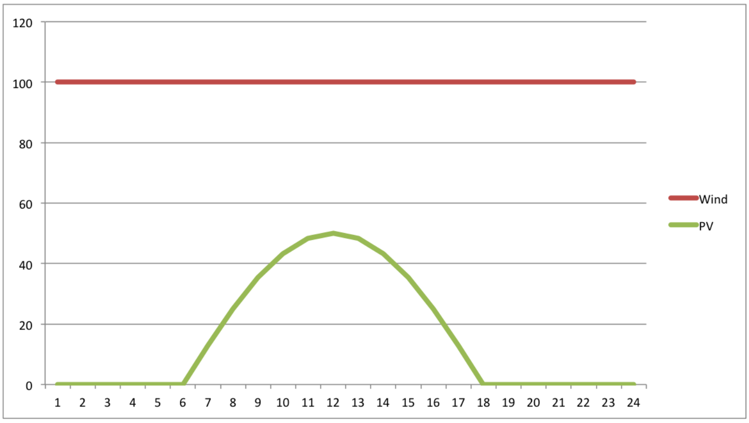

Figure 5.3: Mean Production Profiles for Stochastic Generation

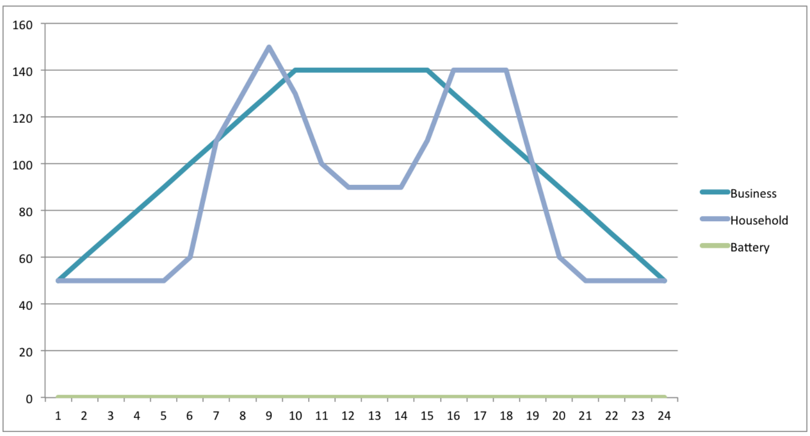

Figure 5.4: Mean Demand Profiles for Stochastic Loads

Note here, that neither the wind or the battery generates or consumes reactive power $Q_{L,4}(t)$. Thus the generator has only to supply reactive power $Q_{G,1}(t)$ according to the demanded reactive power and the phase changes across the network cables during the specific solution. The wind power generated is what is under stochastic influence in this case, and is given as by a uniform distribution with a specific range around a given mean wind energy for the time step.

5.2 Methodology

In the following section the solution methods will be given an overview, discussing some details of their methodology and presenting figures of their architecture or illustrative flowcharts.

5.2.1 Overview of solution methods

Matlab® was used to implement the theoretical concept and test the methods proposed in chapter 3 to tackle the problems described in chapter 2.

The basic AC-Optimal Power Flow model were solved using the *fmincon* solver of the Optimization Toolbox that is readily available in Matlab, which uses uses the interior point method. Both the SDP model and the SQG method utilizes this ACOPF model in their solution approaches.

For the stochastic approaches, the SDP model was developed from scratch in Matlab®, yet inspired by Erdal (2017) and Grillo et al. (2016). The SQG method was developed using the SQG solver of professor Alexei A. Gaivoronski (2016), with the specific case and framework built on top of this as required for the full model to ran.

5.2.2 AC-OPF solution methods

Where the non-linear AC-OPF might be hard to solve, there are some options in linearizing the AC-OPF problem, that is using the DC-power flow formulation and a linearity cost function. The Power Flow version on this model, where one focuses on the physical solution of the power flow in the system, is often solved with an iterative Newton-Raphson method, Lagrangian Multipliers using a gradient solution method, or other methods that may deal with a non-linear, non-convex problem of substantial size, see for instance Kirschenm (2011) for an informal presentation of the topic.

This thesis on the other hand, uses the full AC-OPF model with non-linearity and non-convexity and solvers it with the Interior-point method. In Castillo and O'Neill (2013) we find a study which presents a comparison of the computational performance of different non-linear optimization methods that are commonly applied to the ACOPF problem. Their study finds that the interior-point method is one of the methods that provide relatively fast and reliable solutions. Hence, we may chose to use Matlab's built-in optimization solver `fmincon` when solving the ACOPF problem, which uses the interior point method. The implementation of a faster interior point solver is possible, yet has not been prioritized in this thesis.

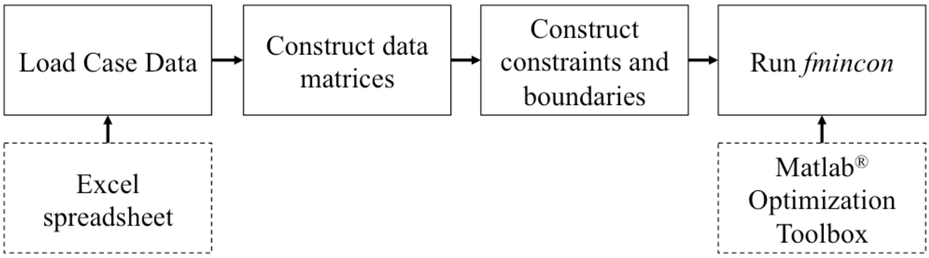

Figure 5.5: AC-OPF architectural flowchart

5.2.3 Stochastic Quasi Gradient Method

The SQG approach as discussed in 3.3.4 uses a privately implemented solver developed by Alexei Gaivoronski (2016) and for instance utilized in Gaivoronski (2005) and Becker and Gaivoronski (2014). An overview of the SQG approach, as presented in his 2005 paper, is displayed below in figure 5.6. As can be seen from the figure, both the optimized model, user interface and underlying AC-OPF simulation model is also developed to be run together with the SQG optimization engine.

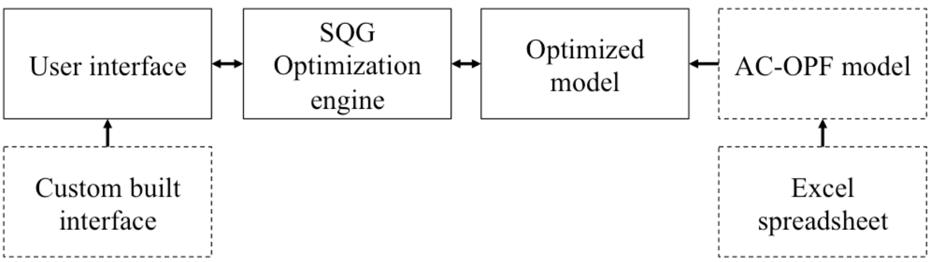

Figure 5.6: Overview of Stochastic Quasi-Gradient Architecture

Key question of the SQG method is how to retrieve the stochastic quasi-gradient, which is used to determine in which direction the iteration algorithm is to make its next step. A more detailed overview of how this process works is found in figure 6.2, divided into three main parts; initialization, the SQG iterations and the post-analysis. It may also be observed that for the SQG iteration, the AC-OPF is solved for all the time steps, before extracting the stochastic gradient.

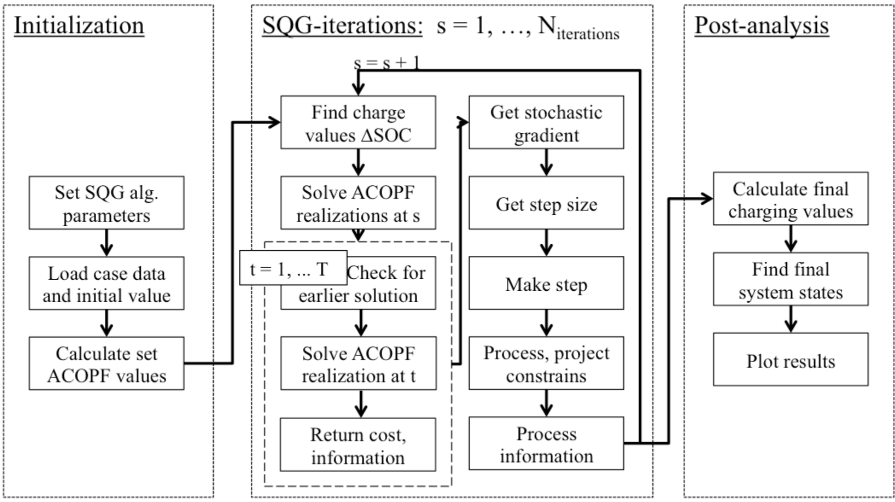

Figure 5.7: Iteration process of SQG algorithm for ACOPF

In order to find the gradient for the problem at each iteration, this thesis applies two different methods. One of these is the use of finite differences, explained in more detail in section 5.2.3.1. The other is to calculate the gradient in from some analytic expression. This secound approach is presented in 5.2.3.2.

The reason for the interest in different approaches in calculating the gradient is not only to compare different solutions and see if they come up with the same answer. It is also because a directly calculated gradient will speed up the solution process a great deal. From a conceptual perspective, the finite difference approach imposes a small perturbation of the system, and observes the gradient from the values this perturbation induces. To do this, it runs the system anew from the point of the latest gradient step, where only one decision variable is altered and the rest N_{SQG} decision variables are kept unchanged. Since the AC-OPF model takes a little time to calculate the resulting power flow even with only small deviations, the finite difference calculations take about $N_{SQG} + 1$ times as much time.

5.2.3.1 Finite Differences Approximation

One much used and relative general way of approximating the Stochastic Quasi-Gradient ξ_s is by the use of finite differences. This method seeks to find the direction of the gradient at a given point, by noting the difference of the value function when changing the decision variables sightly and checking this one by one.

Formally, the finite differences approximation of the subgradient F_x^v at a given point x^s during SQG iteration s for non-convex, differentiable functions may be expressed as

$$F_x^v(x^s) \sim \sum_{j=1}^{n} \frac{F^v(x^s + \Delta_s e^j) - F^v(x^s)}{\Delta_s} e^j \tag{5.1}$$

where e^j is the unit vector of the j-th axis, and $\Delta_s > 0$ is the step size of the perturbations away from x_s of the finite differences calculation. If the function is non-differentiable, the finite differences approximation may yield convergence, and a modified, somewhat randomized, version may be used where

$$F_x^v(x^s) \sim \xi^v(s) = \sum_{j=1}^{n} \frac{F^v(\bar{x}^s + \Delta_s e^j) - F^v(\bar{x}^s)}{\Delta_s} e^j \tag{5.2}$$

and the subgradient $F_x^v(x^s)$ is approximated using $\bar{x}^s = (x_1^s + h_1^s + ... + x_j^s + h_j^s + ... + x_n^s + h_n^s)$ and h_j^s are small, independent, random values uniformly distributed in the interval $\left[-\frac{\Delta_s}{2}, \frac{\Delta_s}{2}\right]$. The Stochastic Quasi-Gradient for iteration s is then given as

$$\xi_s = \{\xi^1(s), \xi^2(s), ..., \xi^j(s), ..., \xi^n(s)\} \tag{5.3}$$

5.2.3.2 Retrieving gradient during AC-OPF simulation

Another possibility when running an SQG-solution algorithm is to use some internal information from the simulation on the gradient of the sub-problem to estimate the step direction for the SQG algorithm.

The challenge when estimating the gradient for the battery, is that it would be beneficial to discharge at all the time steps considered in isolation since it reduces the generator cost due to the strictly increasing cost function in the feasible space for *P*for the 9 bus case. Yet, for some time steps to discharge, other need to charge so that energy is available. How to calculate the gradient then?

Firstly we need to have a notion for how the individual time steps can be improved by changing the discharged or charged energy from or to the battery. However it may be observed that one more unit of energy discharged by the battery at a time step, is equivalent to reducing the total demand of the loads in the power system by one unit and thus also the necessary power for compensation, given negligible power loss in the system, as expressed in equation 5.5 below. Hence, the gradient of the battery discharging is directly linked to the gradient of the gas generator. Using some of the general notation on the recourse problem from 3.52 may be also be expressed formally as

$$\begin{aligned}\frac{dQ(t,P_{Battery},\xi)}{dP_{Battery}} &= \frac{dC_G\big(P_{Gas}(t,P_{Battery},\boldsymbol{\xi})\big)}{dP_{Battery}} \\ &= \frac{dC_G\big(P_{Gas}(t,P_{Battery},\boldsymbol{\xi})\big)}{dP_{Gas}} \cdot \frac{dP_{Gas}}{dP_{Battery}}\end{aligned} \tag{5.4}$$

and since

$$\frac{dP_{Gas}}{dP_{Battery}} = -1 \mid P_{Loss} \approx 0 \tag{5.5}$$

we get

$$\begin{aligned}\frac{dQ(t,P_{Battery},\xi)}{dP_{Battery}} &= \frac{dC_G\big(P_{Gas}(t,P_{Battery},\boldsymbol{\xi})\big)}{dP_{Gas}} \\ &= -\,C'_G\big(P_{Gas}(t,P_{Battery},\boldsymbol{\xi})\big) \mid t, P_{Battery}, \boldsymbol{\xi}.\end{aligned} \tag{5.6}$$

Not only may we assume that the losses in these simple systems are not too great, but they would also generally scale approximately linearly with greater power in the system. We may therefor take the gradient of the objective function with respect to P_{Gas} and the constraints of the AC-OPF to be a good indication for how much there would be to gain from discharging the battery one more unit at an isolated time step.

It may also be noted that the derivation in equations 5.4-5.6 above, and indeed for all the implemented cases in this thesis, the efficiency of the battery charging and discharging is assumed to be the same, and equal to 1 for simplicity. If they are not, one has to adjust the equations for the deficiencies, which might also for instance be dependent of the power - that is rate of energy transfer - of the charge or discharge.

With regards to the implementation of these models, it is of interest to note that the Matlab®function `fmincon` used to solve the AC-OPF second-stage problems reaily returns the gradient of its decision parameters. Hence it is very easy to extract the information on $C'_G\big(P_{Gas}(t, P_{Battery}$ during the iterations without any loss in computational time. Moreover, the statement in 5.8 may also be assessed and confirmed through this functionality of the `fmincon` solver, by letting $P_{Battery}$ be able to change minusculely. When doing so, one may observe that the gradient returned is almost identical for $P_{Battery}$ and P_{Gas}.

Despite having a good indication on how the objective function of the separate time steps might be improved by alternating the battery charging policy, this does not necessarily translate directly into gradients for the first stage decision parameters. As noted in the introduction paragraph of this section, there is the problem that the gradient for all time steps indicate that it is beneficial to charge more, if the cost function is strictly increasing. Hence, all gradients will tell the solver to try increase their values. However - and this also holds true for whatever form the cost function has - to be able to discharge the battery at one point, it will have to be charged at an earlier point. Thus, when reaching the maximum energy level, the energy stored for a certain time step cannot be increased further to improve the function value. Instead, the energy levels of previous time steps has to be reduced.

In other word, since the battery discharging for time step t is given by 4.6, or in short as

$$P_{B,9}(t) = E_{ST}(t-1) - E_{ST}(t), \tag{5.7}$$

it is equivalent increase $E_{ST}(t-1)$ as to decrease $E_{ST}(t)$ in order to achieve a change in $P_{B,9}(t)$. More formally,

$$\begin{aligned} \frac{dQ(t, P_{Battery}, \xi)}{dP_{Battery}} &= \frac{dQ(t, P_{Battery}, \xi)}{d\big(E_{ST}(t-1) - E_{ST}(t)\big)} \\ &= -\, C'_G\big(P_{Gas}(t, P_{Battery}, \boldsymbol{\xi})\big) \mid t, E_{ST}(t), \boldsymbol{\xi} \\ &= \quad C'_G\big(P_{Gas}(t, P_{Battery}, \boldsymbol{\xi})\big) \mid t, E_{ST}(t-1), \boldsymbol{\xi}, \end{aligned} \tag{5.8}$$

or in short and with respect to the decision on how much energy to store in a certain time step,

$$\frac{dQ(t, P_{Battery}, \xi)}{dE_{ST}} = C'_G\big(P_{Gas}(t, P_{Battery}, \boldsymbol{\xi})\big) \mid t, E_{ST}(t-1), \boldsymbol{\xi}. \tag{5.9}$$

In those cases that the cost function is a second order polynomial centered around a minimum point at some preferred point of operation (eg. due to efficiency-loss in the generator when deviating from this point), the gradient calculated as in 5.9 will be able to produce negative values, indicating that the battery should charge up. Yet, when the cost function is strictly increasing, it will not be able to produce a negative gradient for $Q(t, P_{Battery}, \boldsymbol{\xi})$ by utilizing $C'_G\big(P_{Gas}(t, P_{Battery}, \boldsymbol{\xi})\big)$ directly. Indeed, wee need to manipulate the gradient of $C'_G\big(P_{Gas}\big)$ somehow. More specifically, it is needed to shift the gradient values in such a manner that the need for discharging for those time steps at which the gas generator has to produce a lot of energy, in one way manages to push down the need for other time steps to discharge at all and those with least production start to charge instead to facilitate discharging where the gradient is the greatest.

To achieve this, it would for instance be possible to shift the gradient values with respect to stored energy at each time step so that they are centered around the mean of the gradients of the related AC-OPF solutions. However, since the cost function is a polynomial of second degree, the use of the regular mean would produce artificially high gradient values with respect to energy storage. Thus, it is needed to shift the values with regards to a weighted mean of second degree, such as the Root-Mean-Squared value of the gradient number series.

For a set of n numbers, x_1, x_2, ... , x_n, we calculate the Root Mean Square as

$$RMS_{x,n} = \sqrt{\frac{1}{n}(x_1 + x_2 + ... + x_n)}, \tag{5.10}$$

and the gradient with respect to energy storage at a specific time step may be approximated by

$$\frac{dQ(t, P_{Battery}, \xi)}{dE_{ST}(t)}) = C'_G\big(P_{Gas}(t, P_{Battery}, \boldsymbol{\xi})\big) - RMS_{C'_G(P_{Gas}),n} \tag{5.11}$$

The use of this approach to calculate the gradient with respect to the energy stored at a time step has been implemented and provides sensible results for the 9 bus case for 24 time periods.

5.2.4 Stochastic Dynamic Programming

To be able to compare the SQG method with an exact method for the same case, the simplest case was also solved with a Stochastic Dynamic Programming approach. An overview of how this process works is displayed in figure 5.8, where a backward recursion method is applied. The model was developed from scratch in Matlab®, and will be discussed further in 6.4.1.

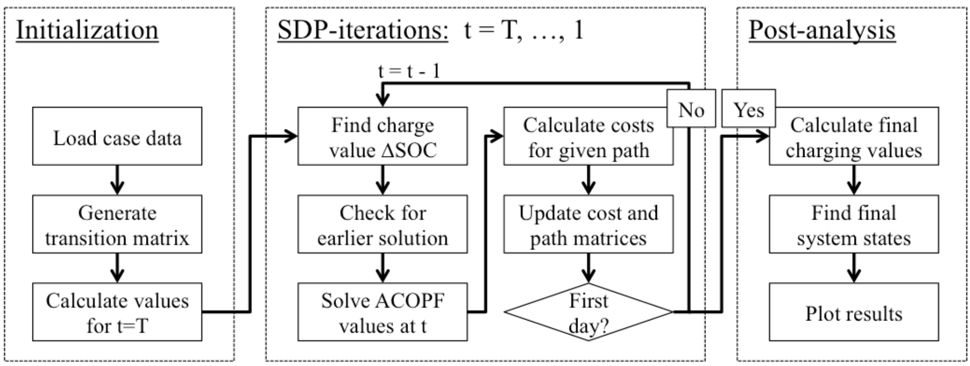

Figure 5.8: Illustration of SDP method for the AC-OPF.

Chapter 6

Implementation

This chapter lays forth how the method, models and cases from previous chapters were implemented. Everything this thesis presents is implemented in the programming language Matlab®.

The first hurdle for the thesis work was in developing the underlying AC-OPF simulation model. For the SQG method to be of interest, it is necessary that the underlying model does not take too much time to run, as it is to be simulated a great number of times during the optimization process. Both standard AC-OPF models and dynamic models were considered, starting with the former as it requires simpler implementation.

Thereafter, the focus was to develop and implement the solution of a stochastic, multistage, AC-OPF model using the SQG method. In doing so, it was necessary to develop both the simulation model further to be fed to the SQG solver, though another layer of code for the optimization of the simulation of all the time steps.

Moreover, the simplest case was also implemented using a simulation based SDP with discretization, to compare it with the SQG solution.

6.1 AC-Optimal Power Flow

A significant portion of the work in the initial phase were used to implement the basic AC-OPF model, find and tune a good solution method for it. Here, the `fmincon` solver of

Matlab® was used, but faster solvers should be commercially available. For the `fmincon` solver, there are several options available, not only for which optimization technique it is to use, but also on what parameter values the chosen algorithms should have. The Interior Point method was chosen, since research like Castillo and O'Neill (2013) and **??** has indicated that this method works particularly well for the AC-OPF problem.

Also some effort was put into tuning this the Interior Point algorithm, but often the default values of the solver generated the best results. Furthermore, a few other standard global optimization solvers were implemented and tested, but did not show immediate promise. Examples here are Global Search and Particle Swarm from the *Global Optimization Toolbox* if Matlab®. Indeed, the attempt of the other solvers to solve the same problem took longer time. As neither the closer manipulation of the Interior Point Method, nor the implementation of other global solvers, gave any benefits in terms of quickened solution time, they are not treated more in depth here.

6.1.1 Algorithm for solving the basic AC-OPF problem

Algorithm 1 shows how the AC-OPF solution is found.

Algorithm 1 Solving the basic AC-OPF problem

1: **Initialization:**
   a) Load the power system for the desired case from a specified excel file.
   b) Use the data to directly and the defined base values to construct the required matrices for all the buses, such as generation, $P_{G,i}$ and $Q_{G,i}$, load, P_L and $Q_{L,i}$, their minimum and maximum values, as well as that of voltage V_i^{max}, V_i^{min}.
   c) Construct matrices with the line data from the file, and use these to construct the Y-bus matrix for the specified case. Also load the flow limits.
   d) Construct matrices for buss voltages and angles from the from the file.
2: **Calculate bounds** on decision variables according to the limits for the generators, $P_{G,i}^{max}$, $P_{G,i}^{min}$ and $Q_{G,i}^{max}$, $Q_{G,i}^{min}$, as well as and bus voltages.
3: **Define constraints**, that is the equality matrix if desired, and load the non-linear equality and inequality constrains and the cost function
4: **Run optimmization** utilizing the interior point method using Matlab® built-in function `fmincon`.

6.2 The SQG method

When implementing the solution approach with the SQG method, the student has been fortunate to be able to utilized an optimization engine developed and implemented in

Matlab® by Alexei Gaivoronski (2016). On top of this, the student has implemented code that lets the SQG optimize the simulation model for the specific problem, as well as code that specifies how the simulation of the AC-OPF is to be treated by the SQG iterations.

Two different cases, discussed in 5, was developed and implemented to test the performance of the SQG approach for the stochastic, multistage AC-OPF problem. Further efforts was made in tuning the SQG model by changing its algorithm parameters, running simulation upon simulation of the different cases for different values of the parameters. Moreover, several methods to find the gradient of the problem was developed and implemented, as was the use of different type of decision variables.

6.3 Implementing the S-MP-AC-OPF for the SQG solver

Solving the stochastic, multistage AC-OPF problem with the SQG solver, is an iterative process. One first decides how much the battery is to charge or discharge for each of the time steps within the model horizon. Then the AC-OPF is solved for all the time steps, calculating the the cost of using compensating power from the costly gas generator given the decided battery charging policy and the realization of wind generation. It returns the sum of the objective values of the AC-OPFs for the time steps, being the total cost of running the generator during the time horizon. After that, the SQG calculates a step direction and length in which to change the charging policy - the decision variables - either by the use of a classical calculated gradient returned from the ACOPFs somehow, or by the stochastic quasi-gradient calculated through use of finite differences in the solver.

However, as noted in the last section, it is preferable to have the constraints of the decision variables of the SQG to be simple, we may choose to reformulate the aforementioned problem to be of simpler structure.

6.3.1 Expanded formulation of the upper and lower bounds on stored energy

Looking at the model formulation proposed in 3.50 and dynamics given from 3.51, one may take notice that the bounds given on the decision variable, changes over time depending over what the decision variable was in the last time step. Thus, they are for one thing don't lend them self to be directly implemented into a runable optimization program. Further more, they are are also not as well suited as regular bounds to perform the projection described in 3.35 and 3.36. Hence, we need to do some arithmetic manipulation to turn these bounds in to a matrix of suitable inequalities. To do so, we may start with equation

3.51,

$$E_{st}(t) = E_{st}(t-1) - P_{Battery}(t).$$

As we see here there is a backward dependency, where the energy stored at end of time step t is given by the energy at the end of previous time step $t-1$ and the discharging during the time step t. For the previous time step this would have been

$$E_{st}(t-1) = E_{st}(t-2) - P_{Battery}(t-1). \tag{6.1}$$

and if consider this dynamics across two time steps, we have

$$E_{st}(t) = E_{st}(t-2) - P_{Battery}(t-1) - P_{Battery}(t). \tag{6.2}$$

Indeed, we may follow this logics all the way back to the first time step and the energy specified for the battery at the start of the time horizon.

Hence, given a certain amount of energy at the start of the period, $E_{st}(0)$, and that we know the charging policy for all steps until a certain time step, we may calculate the energy stored in the battery at the end of that time step. More formally, for any time step t, we may find the energy stored as

$$E_{st}(t) = E_{st}(0) - \sum_{\tau=1}^{t} P_{Battery}(\tau). \tag{6.3}$$

Comparing this to the upper and lower bounds posed in 3.50,

$$E_{st}(t-1) \geq P_{Battery}(t) \geq E_{st}(t-1) - E_{st}^{max} \tag{6.4}$$

and first looking at the upper bound, we have that

$$P_{Battery}(t) \leq E_{st}(t-1) = E_{st}(0) - \sum_{\tau=1}^{t-1} P_{Battery}(\tau). \tag{6.5}$$

Rearranging for $P_{Battery}$ which are the decision variables, we have that

$$P_{Battery}(t) + \sum_{\tau=1}^{t-1} P_{Battery}(\tau) \leq E_{st}(0) \tag{6.6}$$

or that

$$\sum_{\tau=1}^{t} P_{Battery}(\tau) \leq E_{st}(0). \tag{6.7}$$

Similarly for the lower bound we have

$$P_{Battery}(t) \geq E_{st}(t-1) - E_{st}^{max} = E_{st}(0) - \sum_{\tau=1}^{t-1} P_{Battery}(\tau) - E_{st}^{max} \tag{6.8}$$

and get that

$$\sum_{\tau=1}^{t} P_{Battery}(\tau) \geq E_{st}(0) - E_{st}^{max}. \tag{6.9}$$

This may easily be adopted into matrices of inequalities. Such a matrix, for all time steps T, may be written as

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ -1 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & 0 & \cdots & 0 \\ -1 & -1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 & \cdots & 1 \\ -1 & -1 & -1 & -1 & \cdots & -1 \end{bmatrix} \cdot \begin{bmatrix} P_{Battery}(1) \\ P_{Battery}(1) \\ P_{Battery}(2) \\ P_{Battery}(2) \\ \vdots \\ P_{Battery}(T) \\ P_{Battery}(T) \end{bmatrix} \leq \begin{bmatrix} E_{st}(0) \\ E_{st}^{max} - E_{st}(0) \\ E_{st}(0) \\ E_{st}^{max} - E_{st}(0) \\ \vdots \\ E_{st}(0) \\ E_{st}^{max} - E_{st}(0) \end{bmatrix} \tag{6.10}$$

and may readily be implemented into code.

6.3.2 Tuning the SQG solver

To make the SQG method preform well, it is beneficial to tune the solution algorithm. As noted in Becker and Gaivoronski (2014) the parameters of most relevance is to set the initial step and its sequence ρ, Υ discussed in 3.3.4, the sequence and size of the finite differences steps Δ_s as discussed in 5.2.3.1, and the sequence of multipliers γ_s for estimating the objective function from equation 3.33.

For the step size sequence, it is really only necessary to specify the initial step size ρ, as the rest of the sequence can be given from this based on some default schedule. In addition, both values Δ_s and γ_s may well be chosen to be constant, or given by some initial value then to be changed by a predefined default schedule of the solver. Thus, the testing is left to only three parameters, which may be simply be done by trial and error.

However, some more insight may also be used when tuning these. For the initial step size ρ, it is beneficial if this does not put the iteration far out of bounds for every decision parameter and every step, not should it make the steps too small. For the step size of the finite differences Δ_s, there is no point in making it too small, since the error caused by the stochastic parameters often is considerable. Hence it may for instance be between $0.01 - 0.0001$ times the size of the initial step size. The value of the averaging parameter γ_s has to be between 0 and 1, where a larger value means that less of the history is incorporated in the current solution.

As a note on the nature of stochasticity, it might be mentioned that the theoretical precision of the solution increases roughly inversely to the square root of the observations of the objective function. This means that in order to get a relative precision of 0.01, it is needed about $C \cdot 1000$ iterations, where C is some constant specific to the problem.

6.3.3 Algorithms for solving AC-OPF using SQG

The aim of the master thesis is to develop and implement a program in which we solve an AC-OPF problem with stochastic generation and load for several time periods. The procedures for the implemented code up to now is presented in this section. More in detail on the mathematics of these problems are found in chapter 3, 5 and 4.

Figure 6.1 may serve as an illustration of this process, in which the SQG optimizes a simulation.

Algorithm 2 Simulation of AC-OPF with stochastic wind generation for one time step

1: **Initialization:**

a) From the `TrySimulateAC-OPFwSQGv02.m` file:
- Decide the maximum energy of the battery, $E_{ST,2}^{max}$, the mean wind production in the simulation, $\bar{P}_{W,3}$ and the range of the random realization around this mean value, $\omega_{range} \in [0, 1]$.
- Calculate the initial guess for how much battery charging is to be done for all time steps, $x^0 = P_{L,4} - \hat{P}_{G,1} - \bar{P}_{W,3}$

b) From `SimulateAC-OPF_Opt_v02.m`:
- Load case data with `getProblemDataACPOF.m`.

c) Let the initial battery level be half of the maximum energy stored.

2: **Generic step,** s **of the simulation:**

a) **Calculate bounds** for the AC-OPF decision variables ordinarily.

b) **Calculate the realized wind generation**, $P_{W,3}$,

$$\omega = P_{W,3} = \bar{P}_{W,3} * (1 - \omega_{range}) + 2 * \zeta * \bar{P}_{W,3} * \omega_{range} \tag{6.11}$$

where ζ is a random number between one and zero, that is $\zeta = \mathcal{U}\Big(0, 1\Big)$

c) **Update the generator data**, AC-OPF upper and lower bounds and AC-OPF initial guess values with the realized wind generation $P_{W,3} = \omega$.

d) **For the battery,** the value of load (or generation) is fixated with the charging policy of the current step, x^s. That is $P_{Battery} = x^s$

e) **Define constraints**, that is the equality matrix if desired, and load the non-linear equality and inequality constrains and the cost function

f) **Run optimization** utilizing the interior point method using Matlab® built-in function `fmincon`.

g) **Return** the resulting value of the object function, $\Phi(s)$, as well as the power flow solution, where one of the variables is the battery level decided by the AC-OPF calculation.

3: **Iteration with SQG software:**

a) **If** the optimization engine has run for the desired number of steps, it is to terminate and finalize. **If not**, the algorithm will continue with iteration $s + 1$

b) The observed value for $\Phi(s)$ is sent to the SQG optimization engine, and uses it to generate updated values of the policy parameters x_{s+1} which is to be $P_{Battery}$ for the new iteration

c) Return to 2.

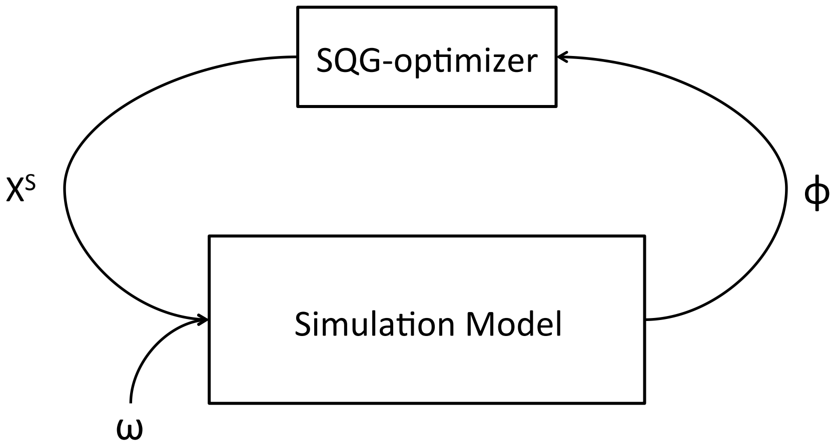

Figure 6.1: SQG overview

6.3.3.1 Algorithm for solving AC-OPF under uncertainty for several time steps

Algorithm 3 displays the optimization of the AC-OPF simulation model using the SQG method for several time steps, and SOC or stored energy, $E_{st}(t)$, at the different time steps, as the decision parameters of the SQG optimization.

The problems that are sought to solve here are defined in **??** for the simple four bus system with stochastic wind generation, and 4.9 for the nine bus system where both wind and PV generation, as well as household and business consumption, are stochastic. For the sake of briefness, an algorithm for each is not presented separately, but are generally outlined here. Also, the iteration of the SQG optimization engine may use either a Quasi-Gradient calculated using finite differences or a gradient returned directly from the simulation in order to decide the next x_s. Again, they will not be posed in separate algorithms here, but be presented though a general one.

Algorithm 3 Optimization of Simulation Model

1: **Initialization:**

a) From the `TrySimulateAC-OPFwSQGv0M.m` file:

- Decide the maximum energy of the battery, $E^{max}_{ST,2}$, number of time steps in the simulation, T, the mean generation and load of the buses for all time steps and their corresponding range of stochastic realized values.

- Set the initial guess for how much battery charging is to be done for all time steps $x^0(t)$

b) From `SimulateAC-OPF_Opt_v0M.m`:

- Load case data with `getProblemDataACPOF.m`.

2: **for generic AC-OPF simulation step, s: do**

- Set the battery level of the first period to be half of the maximum energy stored, and the objective value for all the time steps $\Phi^{tot}_T(s)$ to 0.

3: **for time step $t = 1, ..., T$ do**

a) **Calculate bounds** for the simulation decision variables ordinarily.

b) **Calculate the realized load and generation**, in the same manner as in equation 6.11.

c) **Update the generator data**, AC-OPF upper and lower bounds and AC-OPF initial guess values with the realized wind generation and loads.

d) **For the battery,** the value of load (or generation) is fixated with the charging policy of the current step, x^s. That is $P_{Battery}(t) = x^s(t-1) - x^s(t)$

e) **Define constraints**, that is the equality matrix if desired, and load the non-linear equality and inequality constrains and the cost function

f) **Run optimization** utilizing the interior point method using Matlab® built-in function `fmincon`.

g) **Return from simulation** the resulting value of the object function, $\Phi(s, t)$, as well as the power flow solution, where one of the variables is the battery level decided by the AC-OPF calculation. If a direct gradient is used to calculate the next step x^{s+1}, this is also returned.

h) **Add** the value of the solution, $\Phi(s, t)$, for time step t to the total value of the solution $\Phi^{tot}_T(s)$ of the T time steps. That is $\Phi^{tot}_T(s) = \Phi^{tot}_T(s) + \Phi^{tot}_T(s)$.

i) **If** $t = T$**, Return** the resulting value of the object function, $\Phi^{tot}_T(s)$, as well as the power flow solutions, where one of the variables is the battery level decided by the AC-OPF calculation. If relevant, return the gradient with respect to energy level as defined in 5.11.

4: **end for** t

5: Iteration with SQG software:

a) **If** the optimization engine has run for the desired number of steps, it is to terminate and finalize. **If not**, the algorithm will continue with iteration $s + 1$

b) The observed value for $\Phi^{tot}_T(s)$ is sent to the SQG optimization engine, and uses it to generate updated values of the policy parameters x_{s+1} which is to be $E_{st}(t)$ for the new iteration, either by using finite differences, or gradients passed from the iterations with respect to t.

6: **end for** s

Note that in the Matlab®files referred to above, `M` here denotes which case the file is running. For the 4 bus case, `M` is `2`, for the 9 bus case `M` is `3`.

This process is illustrated in figure 6.2.

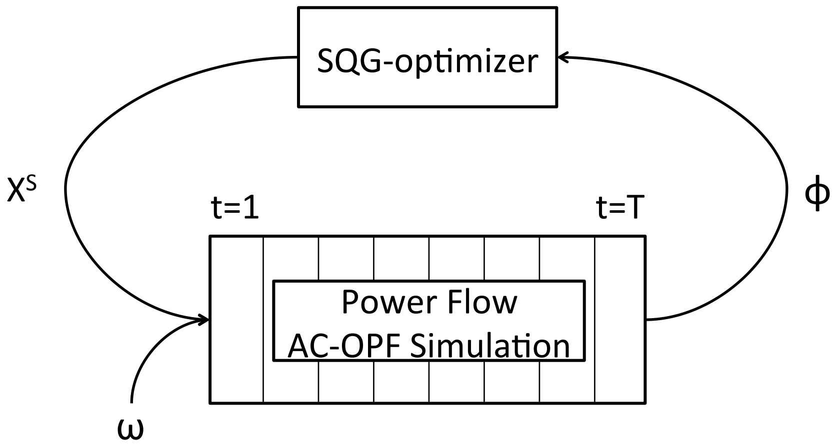

Figure 6.2: SQG overview

6.3.3.2 Algorithm for projection of policy point onto its feasible set

An algorithm for a projection of point v onto the feasible set defined through the constraints that $v_i \geq v_{i+1} \forall i = 1, ..., n-1$ could be as follows:

Algorithm 4 Projection of policy point onto feasible set

Initialization: find the number of elements in the vector v by n =length(v) and make a duplicate $v^{new} = v$.
for $i = 2, ..., n$ **do**
    if $v_{i-1} < v_i$ **then**
        for $k = 1, ..., i$ **do**
            if $v_i < v_k$ and $v_i^{new} < v_k^{new}$ **then**
                $v_i^{new} = (v_i + v_k)/2$
                $v_k^{new} = v_i$
                for $l = k, ..., i$ **do**
                    $v_l^{new} = v_i^{new}$
                end for
            end if
        end for
    end if
end forreturn v^{new}

Intuitively, this approach makes sense in the following way. Since the first value of v, v_1, puts a constraint on all following values to be less than it, it makes it more likely that all values following are within these constraints if v_1 is relatively close to the upper bound on v. Thus, for each value that violates the constraints posed by earlier values, it makes it easier for the rest to be within bounds if some value before the violation would be higher. Thus, for all violations, we find the minimum distance to project them back to the feasible space. The greatest violations will need to be prioritized, and moves back onto the feasible space by changing the first - and thus most effectually for the rest - value it violates. A further proof of this will not be given here, yet figures B.2 and B.3to illustrate this is found in appendix.

6.3.4 Implementation of the cases of Stochastic, Multistage AC-OPF using SQG

6.3.4.1 Implementation Case 1: AC-OPF of 4 bus system

For this model to make sense, we make a few assumptions.

1. The power generation from the wind park is assumed to be given. We here use an expected value based on the designed uniform distribution.
2. The optimal operation point of the generator is known, as is the load demanded.
3. The charging or discharging of the battery is set as the power demand minus the power produced from the wind and the power of the favored generator operation point. Note that this does not take into account the losses and phase shifts, which is what the AC-OPF method is to calculate.
4. Generator cost is given as a parabolic equation with lowest cost at the favored operation point.
5. The efficiency of the battery charging and discharging is assumed to be the same, and equal to 1 for simplicity
6. The model considers several time steps, in which the values as for instance minimum and maximum wind power may vary.
7. Energy storage is allowed in the system within the battery pack, but only supplies active power.

The model is much the same as in 4.5, except that the wind power production is given as a stochastic value distributed uniformly between $P_{W,3}^{min}$ and $P_{W,3}^{max}$. The objective is still to minimize the cost of running the generator. However, the decision variable for the SQG optimization is now how much power to charge or discharge the battery using the basic AC-OPF model as the simulation. That is, to find an optimal chairing policy for the battery given the stochastic wind production, simulating the OPF of the system with the decision variable for the simulation in the SQG is now how much power produce by the generator.

Assumption number four here means that we further assume that the only cost is related to utilizing the generator, and the cost is lowest close to a desired operation point. Hence we use a cost function of the following form

$$C(P_{Gas}) = c \cdot (P_{Gas})^2 + b \cdot P_{Gas} + a \tag{6.12}$$

where this implementation has used the specific values of $a = 323000$, $b = -8000$ and $c = 50$ resulting in a favorable operation point at $P_{Gas} = 80$ as illustrated in figure 6.3 below

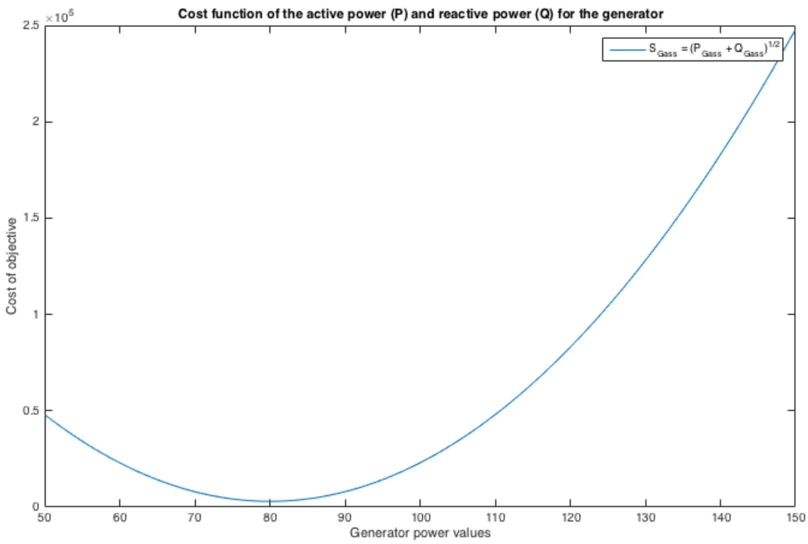

Figure 6.3: Cost function of the 4 bus case

In this model, the decision variable is how power to produce in the generator. The generator at bus 3 will supply the remaining power after the battery has decided how much to charge or discharge, and the wind energy has been realized.

In the development of the codes, we first implement this model as a simple case with only one time step, before developing it further with energy storage. In the simplest case, where the wind energy is assumed to be a specific value, this this system may be solved as basic AC-OPF problem. With the wind as a stochastic variable, we utilize the SQG approach to find a solution under uncertainty. After that we develop a model with energy storage over several time-steps.

6.3.4.2 Implementation Case 2: AC-OPF of 9 bus system

For the 9 bus case the implementation, the increased number of stochastic sources and loads in the system, as well as the increased number of time steps require some more consideration about the exact implementation of the system. One of these is whether to use a cost function as in 6.3 around an assumed operation point, or whether to use a strictly increasing cost function for all generator values greater than 0 as illustrated in 6.4. This is further discussed in chapter 7.

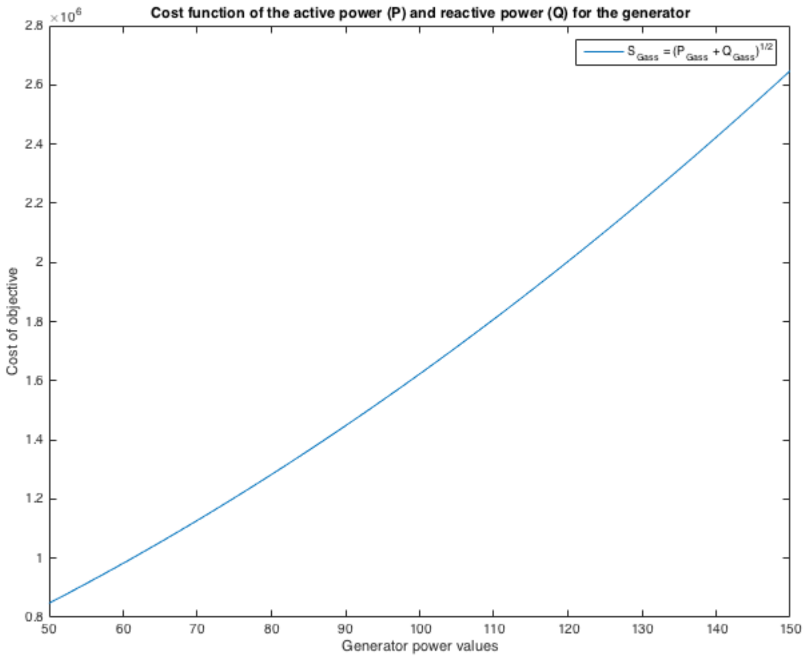

Figure 6.4: Cost function of the 4 bus case

6.4 Solving stochastic, multistage AC-OPF with SDP

To have something to compare the SQG method to, a SDP solution was implemented for the problem at hand.

In implementing this model, two main things was done to keep the solution time from becoming far too great even for low levels of discretization. The first of these, were to store previously calculated solutions during the run. Since the model operates with fully discrete values, and only a limited number of these, many of the specific value combinations reoccur during the simulation. Hence, by checking whether a previous solution exists, the algorithm is able to run through all the instances in much less time.

Another measure taken to reduce the running time, was to only use limit the upper and lower iteration value for wind energy by the maximum and minimum wind energy at that step.

6.4.1 Algorithm for solving the AC-OPF problem using SDP

Algorithm 5 shows how the Stochastic, Multistage AC-OPF solution is found thought the use of SDP.

Algorithm 5 SDP algorithm for AC-OPF

Initialization:
    a) Load the power system for the desired case from a specified excel file.
    b) Set algorithm parameters.
    c) Calculate transition matrix from all N_{wind} to all N_{wind} for time t

    d) Initialize transition cost, cost-to-go and path matrices
Calculate possible and optimal cost of last time step.
for $t = T - 1, ..., 1$ **do**
    for $soc_t = 1, ..., N_{SOC}$ **do**
        for $wind_t = 1, ..., N_{wind}$ **do**
            for $soc_{t+1} = 1, ..., N_{SOC}$ **do**
                Check for earlier solutions. If found, use and skip running AC-OPF.
                Run AC-OPF model with charging as ΔSOC and P_w as $wind_t$
                for $wind_{t+1} = 1, ..., N_{wind}$ **do**
                    Calculate transition cost from equation 3.28
                end for
            end for
            Find optimal SOC and corresponding cost.
        end for
    end for
end for

Chapter 7

Results and discussion

This chapter presents some of the results generated in thesis work.All along brief discussions are made on the observations in each case.

First presented is the solution of the basic AC-OPF model which lies the foundation of all simulations later in this thesis, for both the 4 bus and the 9 bus case. Thereafter the results from the SQG solution of the stochastic, multistage AC-OPF is put forward, illustrating how the object function is estimated, and how the model performs for both cases, one oer several time steps, and different formulations. A brief comparison with the SDP method is made, as is further testing illustrating the benefit of the SQG method for this stochastic problem.

The computer used for all simulations and computations is a MacBook Air from mid 2013 with a $1,3GHz$ Intel $i5$ core and $4GB1600MHz$ ram.

7.1 Basic AC-Optimal Power Flow

In this section, the results from he underlying AC-OPF model is presented. It is here only implemented for one time step, but for both the 4 bus and the 9 bus case. In modelling the AC-OPF for several time steps, this model has been run consecutively and updating the energy stored at each step.

7.1.1 Case 1: 1 time step

In the following, the results from the 4 bus AC-OPF solution is presented and discussed.

7.1.1.1 Results

Figure 7.1 shows the generated active and reactive power at the different buses in the 4 bus constructed power system of case 1. Here, the blue bars represent the active power delivering usable effect to the system, whereas the yellow bars are the reactive power being present in the system without contributing to any actual consumption. Figure 7.2 shows the corresponding loads.

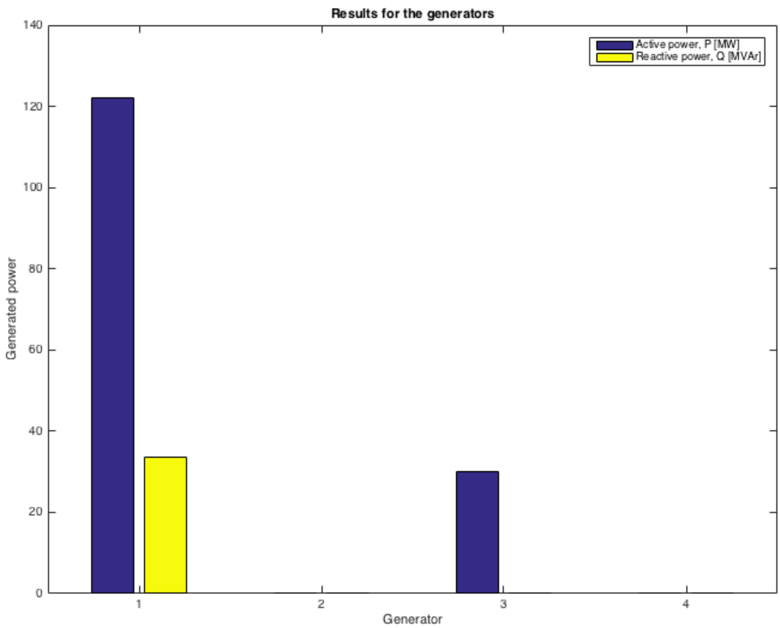

Figure 7.1: Generator values for the 4 bus basic AC-OPF

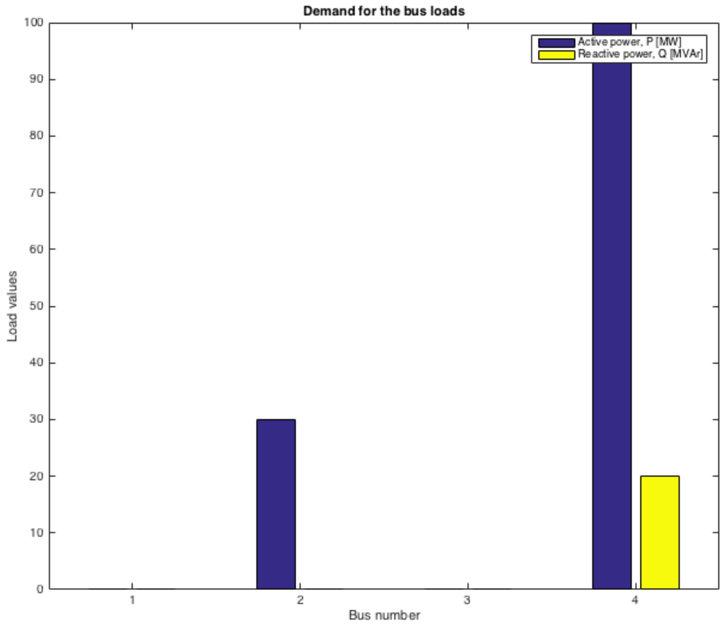

Figure 7.2: Load values for the 4 bus basic AC-OPF

7.1.2 Discussion

As can be seen from figures 7.1 and 7.2 together, the generator supplies a little more then $120MW$ in order to meet the demand of $120MW$ from the household loads. We also see that the battery has chosen to charge during this time step, and that it is the same as the wind energy of $30MW$. The generator is also supplying reactive power to meet the reactive power load from the load. It should also be noted that we here see the total supplied power by the generator is greater that the sum of the other generation and losses. Hence, the power balance is satisfied after losses in the system is accounted for.

7.1.3 Case 2: 1 time step

The 9 bus AC-OPF results is presented and discussed here.

7.1.3.1 Results

The figure 7.3 displays the values of generated energy of the three power sources in the 9 bus system. Again, the blue bars represent the active power, and the yellow the reactive power. Figure 7.4 presents the load values.

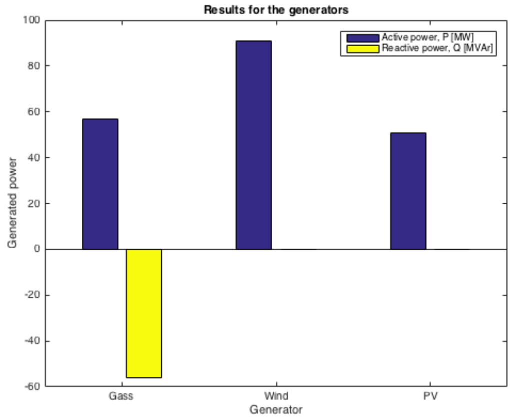

Figure 7.3: Generator values for the 9 bus basic AC-OPF

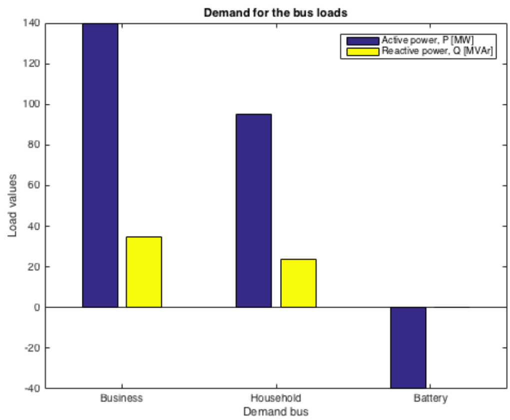

Figure 7.4: Load values for the 9 bus basic AC-OPF

7.1.3.2 Discussion

Again, we see that the generator has the produce more produce more power in total, active and reactive, that is used by the different system loads, minus the energy produced by other generators.

7.2 SQG solution of ACOPF

This section presents the most important results when solving the stochastic, multistage AC-OPF using the SQG method. It starts by presenting the solutions for one time step for the 4 bus case, then several time steps. Comparisons are also made between the gradient and finite difference approaches, and with the SDP solution. Thereafter, much the same is presented for the 9 bus case, except that the SDP is left out due to its computational burden.

7.2.1 Case 1: 1 time step

First, the 4 bus case is presented with the results from the one time step simulation on estimating the objective function.

7.2.1.1 Results

In figure 7.5 we see how the SQG algorithm estimates the objective function for the AC-OPF model. The different lines represent the number of simulations ran for each of the estimates, where the SQG algorithm solves for the optimal value for a set of discretizised decision variables along the x-axis. Figure 7.6 shows the same graph essentially, however, with a greater resolution at the proximity of the optimal point. The vertical stippled line shows the value of the decision variable at the optimal solution, whereas the horizontal stippled line shows the objective function value at the same solution.

Estimating the objective function

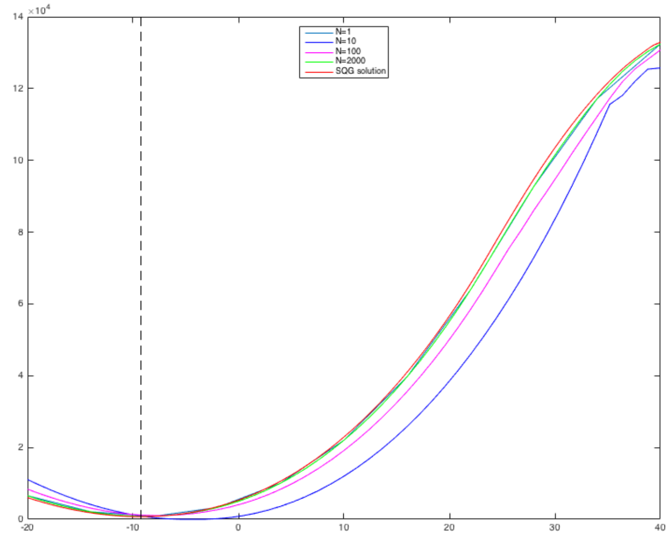

Figure 7.5: Estimation of objective function for 4 bus system

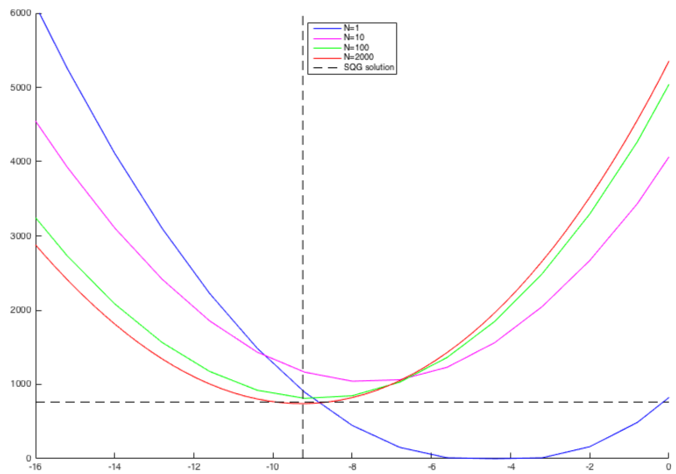

Figure 7.6: Estimation of objective function compared with SQG solution for 4 bus system

7.2.1.2 Discussion

From the graphs it is clear that the more simulations used, the better estimate of the solution is found. Another point to notice about these graphs is that it is quite similar to figure 6.3 with the actual cost function. Indeed, it is almost the same, just shifted leftwards as the decision parameter is how much to charge or discharge the battery, not the generator.

7.2.2 Case 1: 4 time steps

Here the four time step solution of the AC-OPF model is presented with battery and stochastic wind generation. Both the finite differences approach and the gradient approach will be discussed.

7.2.2.1 Results

A standard profile for the generator values is seen in figure 7.7. The only thing that is different in this figure from solution to solution is the values for the generator in blue, resulting from the charging of the battery. Yet, during the simulation, the value for wind, the yellow line, changes

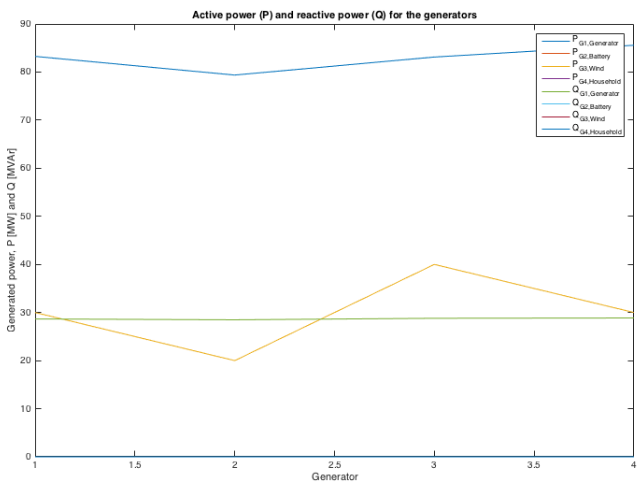

Figure 7.7: Typical generator profiles for 4 bus case

Figure 7.8 shows the battery charging policy profiles found by using the finite difference version of the SQG method for the four time steps with uncertainty. In figure 7.9 one sees the same graph, ony solved with the directly calculated gradient from the AC-OPF simulation.

Finite differences

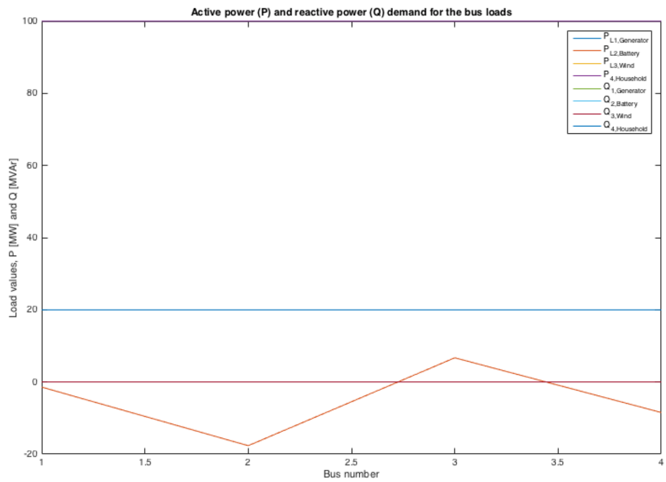

Figure 7.8: Battery charging with Finite Differences for 4 bus case

Direct gradient

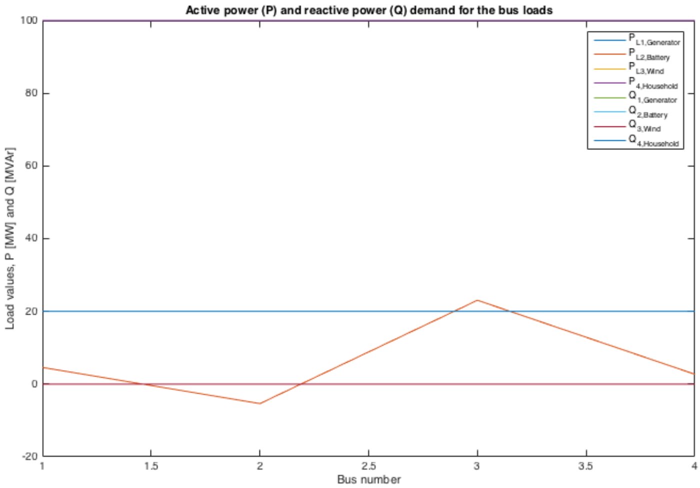

Figure 7.9: Battery charging with direct gradient for 4 bus case

7.2.2.2 Discussion

From the figures it is evident that the solution found is different for the two methods. I the finite difference case, the battery is not used in the first time step, only to be substantially discharged during the following step. In the gradient solution, the policy found requires charging in the first step, but does not discharge as much during the second. This seems to be a wrong approach, as it should be beneficial to discharge so that the generator is at about $80MW$ after wind production. Indeed the charging profile should in many ways mimic that of the wind production profile so that the generator is as stable as possible.

A close look on the solution function values is found in appendix C.1. Here figure C.1 shows the function value and observations for the finite difference calculations, ending at a value of 2871. The gradient solution end at a value of 4185 as indicated by C.2, showing that the finite difference solution has found a better value. On the other hand, the gradient solution is takes much shorter time to compute. It is able to compute 500 iterations of the SQG solution process in about 400 seconds on the MacBook Air of the student, whereas the finite difference approach needs about 1700 seconds, scaling to about T times as much time due to the calculation of the perturbations.

7.2.2.3 Comparison with SDP

Figure 7.10 shows the SDP solution of the same problem as above. For this graph, a discretization level of 101 has been used.

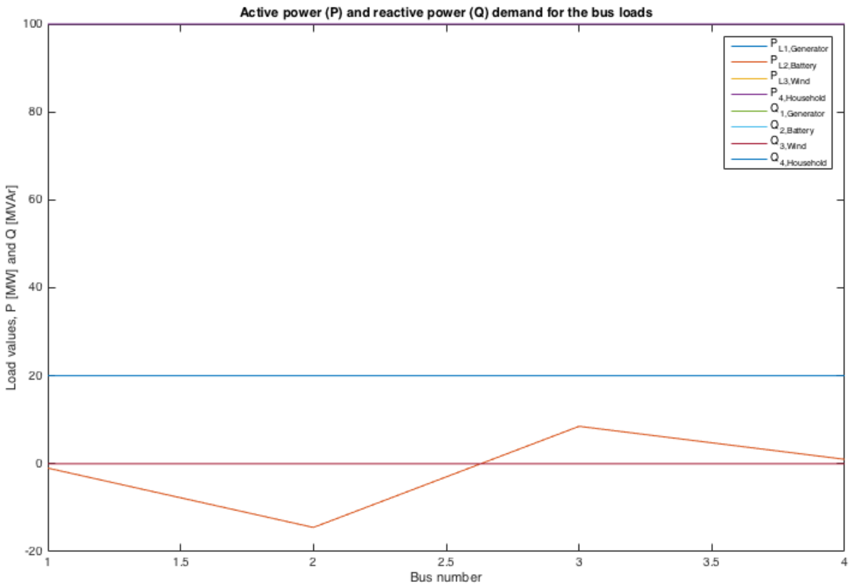

Figure 7.10: Battery charging with SDP for 4 bus case

It is to note that this solution is not identical to neither the finite difference solution in figure 7.8 nor the direct gradient in figure 7.9, yet bears most resemblance to the former.

In regards to the performance of the SDP model, the solution time rapidly becomes quite long, especially if one is to have a level of discretization that is able to return a seemingly logic solution. For an approach which only uses a total discretization level for both SOC and wind energy of 11, the solution time is about 40 seconds. Yet, since here most of the possible wind energy energy realizations is only represented by two or three outcomes here (see figure B.1 in the appendix), the resulting solution seems not to make that much sense. For a discretization level of 51, the solution is seems better, yet the solution takes much longer time, 1300 seconds.

scales further.

7.2.3 Case 2: 1 time step

In the following subsection, the results from the solutions of the second case is presented, showing how the SQG algorithm is able to estimate the objective function and optimal policy value.

7.2.3.1 Results

Figure 7.11 shows the estimated objective function value for a given number of SQG iterations, where 7.12 shows a close up near the solution point given by the stippled lines.

Estimating the objective function

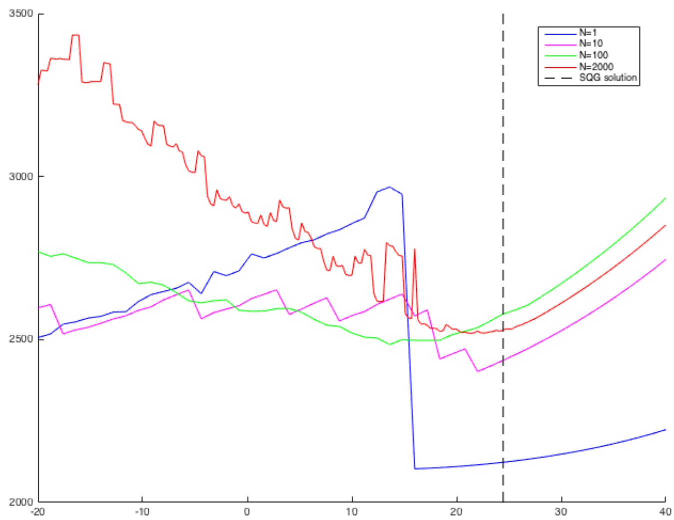

Figure 7.11: Estimation of objective function for 9 bus system

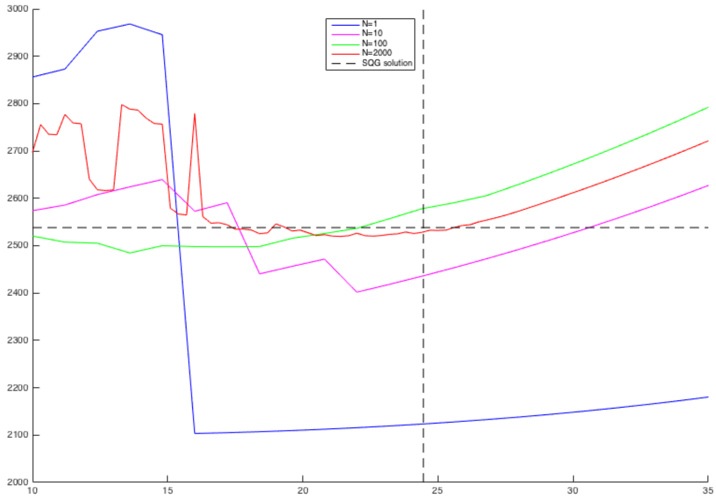

Figure 7.12: Estimation of objective function compared with SQG solution for 9 bus system

7.2.3.2 Discussion

In these figures, it is clearly shown that there is a lot of uncertainty present in the simulations, yet that the SQG method is able to approximate a good solution to these after enough iterations. It also shows that the deviations from the later found value can be substantial for the initial cases and for different parts of the solution space.

7.2.4 Case 2: 24 time step

Here the results from the 9 bus system for a 24 hour period is presented. First by showing a general profile for the generators, before the charging profiles for the finite differences and direct gradient calculations are dislpayed.

7.2.4.1 Results

In figure 7.13, a standard profile for the system generators is shown, where the randomness is kept out. The red line denotes the mean wind production, which during the simulations are random. The yellow line shows the mean PV production, also realized as a stochastic variable in the simulations. The blue and the purple lines represents the generated active and reactive power from the gas generator for the specific solution this graph is extracted from.

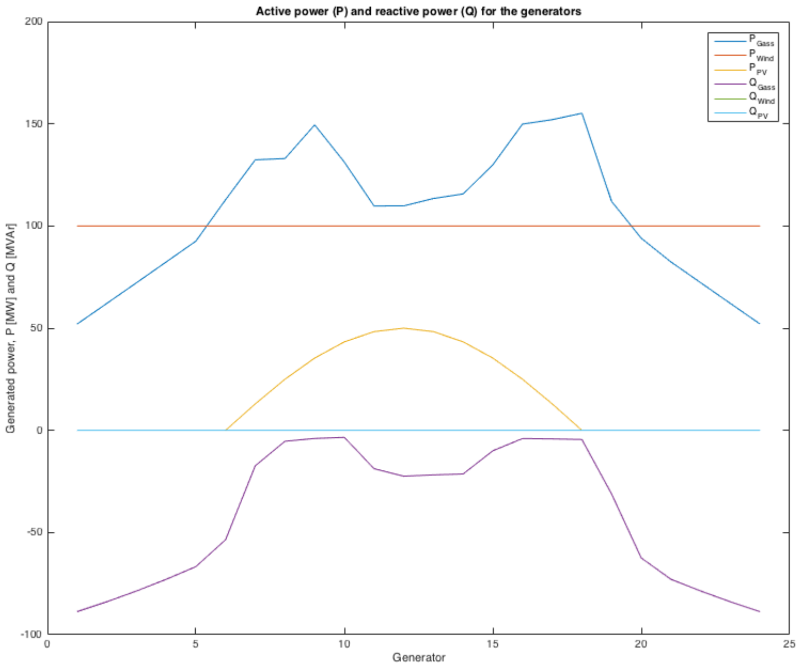

Figure 7.13: Generator values for SQG solution for 9 bus system

The figures 7.14 and 7.15 shows the energy consumption and charging profiles for the finite differences and the directly calculated gradient. The yellow line in these graphs is the charging policy.

Finite differences

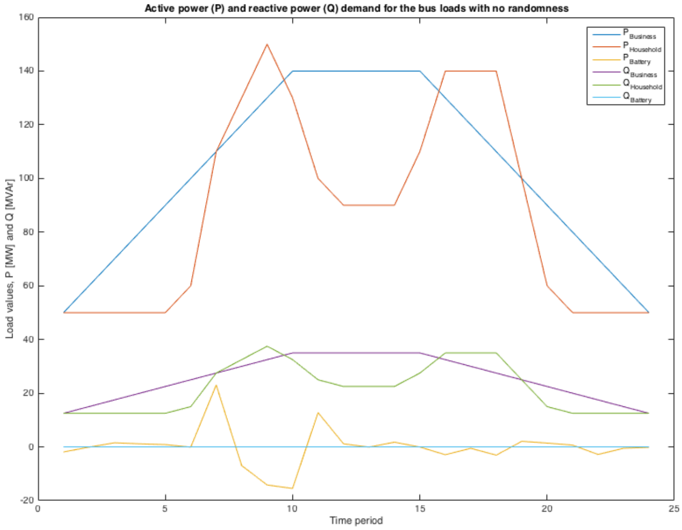

Figure 7.14: Generator values for Finite differences SQG solution for 9 bus system

Direct gradient

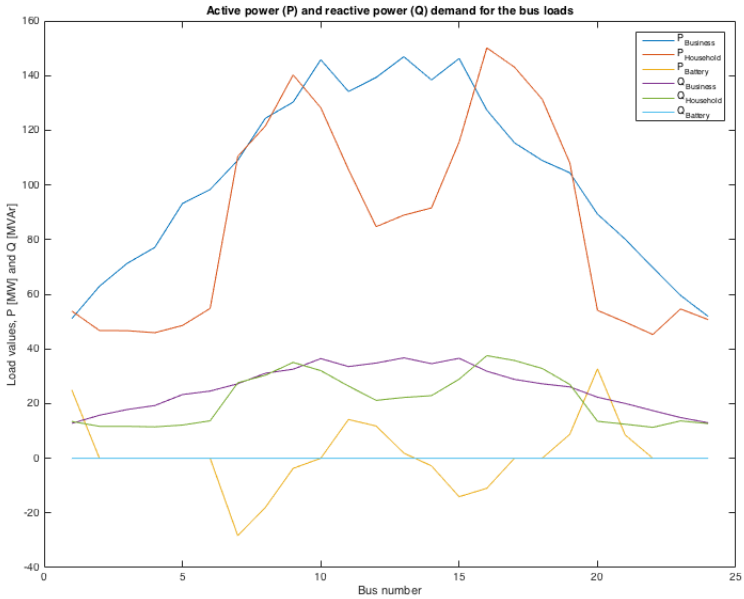

Figure 7.15: Generator values for Gradient SQG solution for 9 bus system

7.2.4.2 Discussion

This time around, it is the finite difference methodology that is not able to produce any sensible result. We see that it only suggests to utilize the battery in the first part of the time period, and only does marginal charging and discharging thereafter. The gradient solution of 7.15 on the other hand, produces what seems a more sensible. It discharges during the times of high demand, and charges in the times of lower demand. One thing that might not seem that logic is that it ends the simulating with a full battery; since the energy in the battery at the start of the period can be considered free power, it would make sense to use it all. Yet, this solution is actually more clever, since it charges in the last step by knowing that it now is in a time period with unusually low demand.

Looking at the objective value approximations from the figures C.3 and C.4 in the appendix, we see that the firs graph for the finite difference approach approaches a lower value than the second graph from the direct gradient calculation. With regards to solution time, the gradient approach is able to solve the 9 bus system with 250 iterations in about 450 seconds, whereas the finite differences approach requires about 5000 seconds for the same case and number of SQG iterations. A reason for the gradient method to perform better here, is that the author spent more time tuning the models of the 9 bus case compared to the 4 bus case.

7.2.5 Finding optimal policy decision rules

In addition to the charging policy for each time step, another approach discussed in 4.4.3 is to use the SQG method to find some decision rules for optimal charging policy.

7.2.5.1 Results

Running decision rule the model with three variables as the decision parameters, we get the following results.

xOut =

10.1640 23.5531 46.1557

Here, the first two values represents the energy level to go to, whilst the third represent the associated threshold value given by equation 4.18 as the last periods generation minus the battery energy of the start of the period.

In figure 7.16, we see the evaluation of the objective function values for corresponding to the solution above.

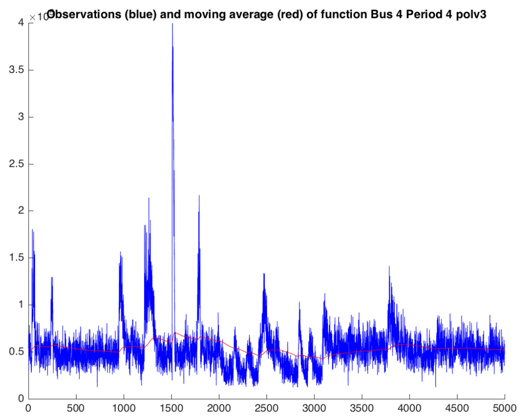

Figure 7.16: Objective function for policy rules calculations for the 4 bus system

7.2.5.2 Discussion

The figure 7.16 ends at about 6000 which is not a very good solution value. Yet, a more fine grained decision variable vector for the policy rules might allow a better solution. Non-the-less, this is even after 5000 SQG iterations, taking about 8100 seconds. From figure C.5 in the appendix, which shows that the value for the first decision variable has a difficult time converging, it is evident that the simulation needs event more time to come up with a good solution. This is an even bigger problem when having more variables. Yet, for big and complex cases, this type of decision rule analysis might allow the problem to be solved quick by having a limited amount of decision variables.

7.2.6 Further testing

In this subsection, results are presented from test using different values or different range for the random wind generation in the 9 bus system, gauge the benefit of the stochastic solution with the SQG method.

In figure 7.17 one finds the objective function values for the regular values of the 9 bus case. The approximated value for the solution ends up at around 11000 after 200 iterations.

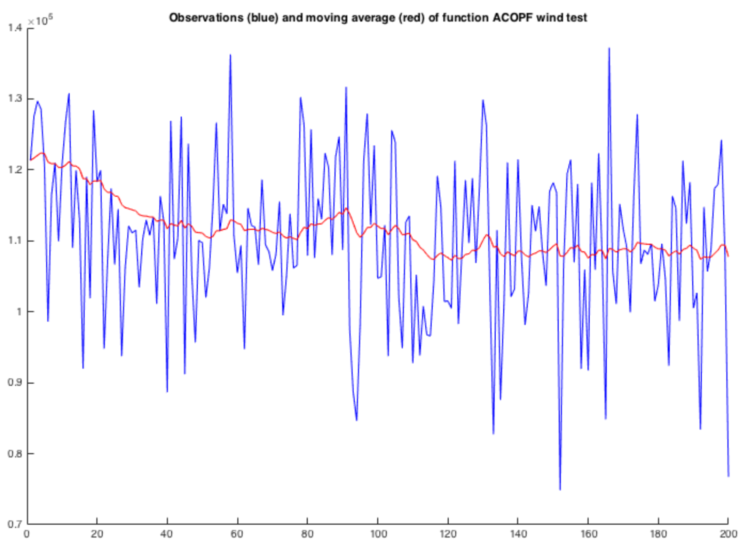

Figure 7.17: Normal objective function values for Gradient SQG solution for 9 bus system

7.2.6.1 Different range for the random variables

In figure 7.18 and 7.19 the same graph is presented, yet with increased or decreased range respectively for the stochastic value of wind generation. In the fist graph, we see the solution starting out at a much higher value and converging to about 13000 whereas the second starts and later converges to a value around 10250.

Increased range of stochastic variables

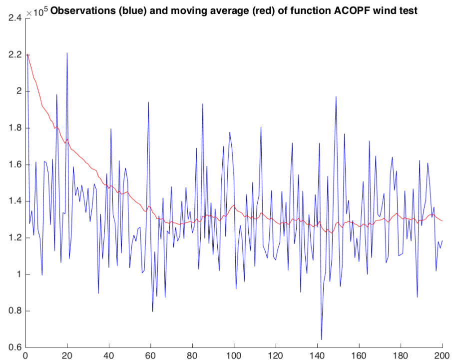

Figure 7.18: Objective function values for Gradient SQG solution for 9 bus system with increased stochasticity

Decreased range of stochastic variables

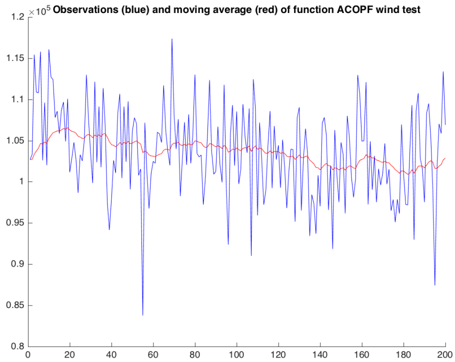

Figure 7.19: Objective function values for Gradient SQG solution for 9 bus system with decreased stochasticity

From these figures, it is clear that higher stochastic variations increases the cost for the system, and that the higher the volatility, the more improvement is made with the stochastic solution after some iterations.

7.2.6.2 Different mean value for the wind production

The figures 7.20 and 7.21 shows the same graphs again, but now it is the the mean value for the wind generation that is shifted so that we may see if the batteries might facilitate more wind in the grid.

Increased value of mean wind production

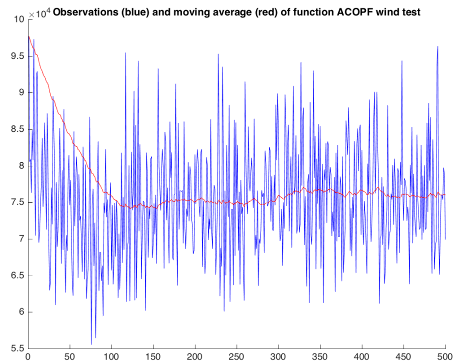

Figure 7.20: Objective function values for Gradient SQG solution for 9 bus system with increased wind generation

Decreased value of mean wind production

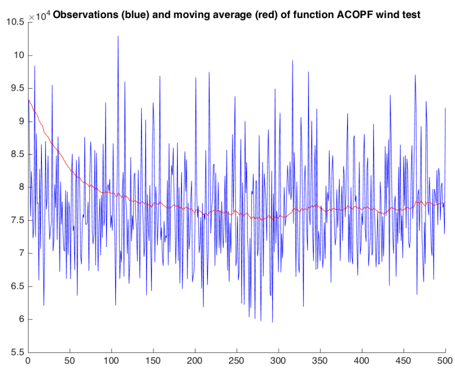

Figure 7.21: Objective function values for Gradient SQG solution for 9 bus system with decreased wind generation

In both figures, the approximated value converges to about the same value. However, the first figure with an increased amount of wind generation, starts of at a higher value and takes a little longer before it manages to converge. This indicates that the battery is able to adjust so that the increased wind energy that at first may create additional cost, is incorporated in the same fashion as less wind with a more optimal charging policy.

Chapter 8

Concluding remarks and future research

This thesis has centered around how to use the Stochastic Quasi-Gradient (SQG) method to analyze how best to operate the power generators together with a energy storage system in the advent of greater variability thought inclusion of renewable energy sources in the power system.

To address this, the thesis has developed mathematical models and methods, and implemented a stochastic and multistage version of the Alternating Current Optimal Power Flow (AC-OPF) problem to be solved by the SQG method. This chapter brings some concluding thoughts on the endeavor, and elaborates on future areas of research continuing on the topic and models treated here.

8.1 Conclusion

This thesis has showed that the SQG method might be an effective tool for analyzing and optimizing energy systems with stochastic and dynamic aspects.

For the 4 bus case, the finite differences method for estimating the stochastic quasi-gradient provided the best solution, which also was closest to the best solution from the stochastic dynamic programming approach.

In the 9 bus case, however, the estimate of the stochastic quasi-gradient directly calculated gradient from the AC-OPF solution gave a better and smarter solution than the finite difference approach. The reason the solution it finds may be considered good, is that it both manages to charge and discharge the battery when energy is in abundance and shortage, and charges the battery in the later stage towards midnight when the power is in abundance even though one might think it would gain from discharging more during that time.

In regards to solution time, the gradient performs clearly best of the models. The finite difference approach gives interesting results, but in a much longer time for the bigger model. Further tuning and tweaks of both solution approaches for the individual cases might improve them further, yet since the finite difference approach has to calcualte all the perturbations again and again, it will always be considerably slower. For the small case in this thesis, the SDP method worked ok, given a fine enough level of discretization. However, in larger cases, with more time steps, this method is bound to loose relevance as the solution time it needs grows exponentially.

What goes for the accuracy of the models, a greater number of SQG iterations is needed to produce more precise results. As mentioned, to have a relative precision of say 0.01 one needs somewhere around 1000 iterations. However, considering the error present in the stochastic functions in the first place, a precision of much more than 0.025 is not necessarily to be expected.

The decision rule analysis of the charging policy, does not come forth as such a good approach here. This is mainly because the solutions found for several variables are not very logic, and that it uses a lot of time to produce these. Even a small case took longer time than any other model.

From the further analysis with the models, we also made note of the benefit of the stochastic solution approach to optimal charging. The tests indicated that the inclusion of energy storage might enable the incorporation of more variability in the electrical grid. Our tests also indicated that the more volatility present, the more improvement can be made with by employing a charging policy based on the SQG solution.

In chapter 2 we asked:

> How might the Stochastic Quasi-Gradient method be used to analyze and optimize multistage power system operation with energy storage under uncertainty?

To conclude, we may note that the are several ways the SQG may be applied to this problem, yet that the approach with a directly calculated gradient performs well for the largest of the cases tested here, providing a good solution quite fast that lets energy storage facilitate the incorporation of variability in the electrical grid.

8.2 Further work

This report also has showed their is more work that may be done, potential not only on the further development, tuning and application of the models presented, but also in exploring new ways to utilize the SQG approach in the field of Power System Analysis. The work of this thesis is of course limited by how much it is possible to do within a certain time frame. However, there are may ways this work may be continued forward, both in terms of methodology and in terms of application.

As for improvement in methodology, some possibilities are:

- Implement or develop an even faster solver of the AC-OPF model.
- Develop a Stochastic Optimization heuristic to solve the same case for comparison, for instance an Evolutionary Algorithm or Swarm Optimization
- Expand the AC-OPF model include more direct measures of stability, such as the N-1 criterion or continuation power flow.
- Expand the AC-OPF model to include a market coupling, for instance through bidding of power from the different sources to be solved by a market clearing algorithm.
- Develop a theoretical proof for the suggested relation on calculating the SOC gradient
- Develop a theoretical proof for the suggested relation on projection of policy decision variables onto feasible set.
- Implement parts of the model using parallel programming, to speed up the solution process.
- Implement the use a discrete, dynamic AC-OPF for the whole period to calculate the gradient for the solution in the SQG process.
- Implement a version where the dispatchable generation is the decision variable not charging, with more complex constraints for the SQG solver.
- Find new ways to implement the decision rule version of the model, to gain more insights from this approach.

As mentioned previously in this thesis, the work with the fundamental AC-OPF model and its Interior Point solver was halted once an approach with sufficiently good performance was found. Therefore, it might be very relevant to improve the solution time of the underlying AC-OPF model, as it has the potential to greatly speed up the SQG iterations. Moreover, a comparison with a typical randomized search method such as an Evolutionary Algorithm might could be interesting to see whether the SQG performs better than it does for this problem. Further expansions of the AC-OPF model is also possible to analyze other power system phenomena in greater detail, such as the N-1 criterion or a fully market coupled model. Another issue left unresolved is the theoretical development of proofs for the calculation of the SOC gradient and the projection suggested in this thesis, and further advances in the implementation of the method.

In order to apply this method to implement and analyze problems that might provide valuable insight for real life cases, one may for instance look at

- Use the method in this thesis to valuate how much economic profit a energy grid operator can gain if it is able to exercise some control of when the Electric Vehicles in the grid are charging.
- Valuate control of household demand, through for instance price signals in a smart grid system.
- Include electricity price, and for instance water inflow to hydro power reservoirs, as a stochastic variables.
- Use real grid data to develop a full fledged model for an existing power system.
- Use real data to implement forecasts of the variable generation of variable generation and demand.
- Develop an online model that uses real time data to calculate the optimal level of a distributed battery in a micro grid for any time instance.
- Use the model to analyze how much wind can be included in a power system with and without batteries, given it has to satisfy the stability and safety limits of the current grid.

The suggestions mentioned above are in now way exhaustive or exclusive, yet might provide an illustration to how versatile the method might be and serve as inspiration for other researchers to test out the SQG method. As mentioned in the preface of the thesis, the student will continue working with the application of the SQG method for analysis of power system variability and energy storage. Hence, some of the mentioned points from the lists above will be addressed quite soon.

Bibliography

Ariyo, F. K. (2013). Electrical network reduction for load flow and short-circuit calculations using powerfactory software. *Electrical and Electronic Engineering*, 3:1–7.

Becker, D. M. and Gaivoronski, A. A. (2014). Stochastic optimization on social networks with application to service pricing. *Computational Management Science*, 11:531562.

Bellman, R. E. (1957). *Dynamic Programming*. Princeton University Press, Princeton.

Bertsekas, D. P. (1999). *Nonlinear Programming*. Athena Scientific, Massachusetts, US, 2nd edition.

Birge, J. R. and Louveaux, F. (2011). *Introduction to Stochastic Programming*. Springer, New York, US, 2nd edition.

Cain, M. B., O'Neill, R. P., and Castillo, A. (2012). History of optimal power flow and formulations. Technical Report Optimal Power Flow Paper 1, Federal Energy Regulatory Commission, Washington DC, US.

Castillo, A. and O'Neill, R. P. (2013). Computational performance of solution techniques applied to the acopf. Technical Report Optimal Power Flow Paper 5, Federal Energy Regulatory Commission, Washington DC, US.

Crow, M. L. (2012). Computational methods for electric power systems. In Grigsby, L. L., editor, *Power Systems*, chapter 5. CRC Press, Florida, US, 3rd edition.

Erdal, J. S. (2017). Stochastic optimization of battery system operation strategy under different utility tariff structures. Master's thesis, NTNU Trondheim, Norway.

Ermoliev, Y. (1976). *Methods of Stochastic Programming*. (in Russian), Nauka,Moscow.

Ermoliev, Y. (1983). Stochastic quasigradient methods and their application to systems optimization. *Stochastics*, 9:1–36.

Ermoliev, Y. and Wets, R., editors (1988). Oxford University Press, Springer-Verlag.

Gaivoronski, A. A. (1988). Stochastic quasigradient methods and their implementation. In Ermoliev, Y. and Wets, R. J.-B., editors, *Numerical Techniques for Stochastic Optimization*, pages 313–351. Springer-Verlag.

Gaivoronski, A. A. (2005). Stochastic quasigradient methods and their implementation. In Wallace, S. W. and Ziemba, W. T., editors, *Applications of Stochastic Programming*, pages 37–60. MOS-SIAM.

Gaivoronski, A. A. (2016). SQG v.5 reference manual. Unpublished, Norwegian University of Science and Technology.

Galus, M. D., Wietor, F., and Andersson, G. (2012). Incorporating valley filling and peak shaving in a utility function based management of an electric vehicle aggregator. *IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*.

Grigsby, L., editor (2012). *Power Systems*. CRC Press, Florida, US, 3rd edition.

Grillo, S., Pievatolo, A., and Tironi, E. (2016). Optimal storage scheduling using markov decision processes. *IEEE Transactions on Sustainable Energy*, 7:755–764.

Harbo, S. F. (2016). Agent based modelling, simulation and optimization of electric power flow with plug-in electric vehicles. Unpublished specialization project report in TIØ4500, Norwegian University of Science and Technology.

Higle, J. L. (2005). Stochastic programming: Optimization when uncertainty matters. *INFORMS - Tutorials in Operations Research*.

Hillier, F. S. and Lieberman, G. J. (2010). *Numerical Optimization*. McGraw-Hill International, New York, US, 9th edition.

Jameson, A. (1995). Gradient based optimization methods. *MAE Technical Report No. 2057*, Princeton University.

Kall, P. and Wallace, S. W. (1994). *Stochastic Programming*. John Wiley and Sons, Chichester, 2nd edition.

Kirschenm, D. (2011). Optimal power flow. University Lecture: `https://www.researchgate.net/file.PostFileLoader.html?id=58511588cbd5c2a0f5159ae7&assetKey=AS%3A439133249249281%401481708933486`. [Online; accessed 16-June-2017].

Lundgren, J., Rnnqvist, M., and Vrbrand, P. (2012). *Optimization*. Studentlitteratur, Malm, Sweden, 1st edition.

Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization*. Springer, New York, US, 2nd edition.

Peeta, S. and Zhou, C. (2006). Stochastic quasi-gradient algorithm for the off-line stochastic dynamic traffic assignment problem. *Transportation Research*, 40:179206.

Pflug, G. (1988). Stepsize rules, stopping times, and their implementation in stochastic quasigradient algorithms. In Ermoliev, Y. and Wets, R. J.-B., editors, *Numerical Techniques for Stochastic Optimization*, pages 353–372. Springer-Verlag.

Pflug, G. C. and Pichler, A. (2010). *Multistage Stochastic Optimization*. Springer, New York, US.

Skotland, C. H., Eggum, E., and Spilde, D. (2016). Hva betyr elbiler for strmnettet? Technical report, The Norwegian Water Resources and Energy Directorate (NVE), Oslo, Norway.

Sperstad, I. B. and Marthinsen, H. (2016). Optimal power flow methods and their application to distribution systems with energy storage. Technical report, SINTEF Energy Research, Trondheim, Norway.

Spilde, D. and Skotland, C. (2015). Hvordan vil en omfattende elektrifisering av transportsektoren pvirke kraftsystemet? Technical report, The Norwegian Water Resources and Energy Directorate (NVE), Oslo, Norway.

Zaferanlouei, S., Ranaweera, I., Korpås, M., and Farahmand, H. (2016). Optimal scheduling of plug-in electric vehicles in distribution systems including pv, wind and hydropower generation.

Zenios, S. A. (2007). *Practical Financial Optimization*. Blackwell, Oxford, UK, 1st edition.

Appendix A

Acronyms and Definitions

A.1 Acronyms

AC Alternating Current

ESS Energy System Storage

EV Electric Vehicles

NVE The Norwegian Water Resources and Energy Directorate

OPF Optimal power flow

RES Renewable Energy Sources

SDP Renewable Energy Sources

SOC State of Charge, eg. how much power there is available on a PEV's battery compared to its capacity.

SSB Statistics Norway, *Statistisk Sentralbyrå* in Norwegian.

A.2 Definitions

Dispatchable generation Power production that is easy and quick to turn on and off.

Simulation The word simulation is in this thesis is often used for referencing the low-lever part of the models running the deterministic AC-OPF optimization given the realization of stochastic variables and charging policy. The results returned are used by the SQG-optimization engine to optimize the stochastic dynamic AC-OPF.

Stochastic Optimization The use of stochastic variation in an iterative optimization method in order to help find the global optimum.

Stochastic Programming Refers to decision making under uncertainty, by modeling and formulation of exact mathematical programs to optimize the outcome by manipulation of some decision variables.

Multi-Period Stochastic AC-OPF or MP-S-AC-OPF is an AC-OPF model with ESS and stochastic variables, solved by the SQG method.

Stochastic Dynamic AC-OPF or SD-AC-OPF, is an AC-OPF model with ESS (thus dynamic) and stochastic variables, solved by Stochstic Dynamic Programming (SDP).

Appendix B

Additional Information on the Models

B.1 Case data

B.1.1 Case 1: the 4 bus power system

Table B.1 presents the given demand and generation (average in the stochastic case) of the different buses, that the AC-OPF takes as input to its solution enforcing the values by upper and lower boundaries. In table B.2 the corresponding constrains on the buses voltage maximum and minimum, as well as power bounds on active and reactive power. Table B.3 shows the line constraints for the network. Note that the values are the same for all three phases. The system is assumed to have base values of $100MVA$ for base power and $115kV$ as base voltage.

Table B.1: Given demand and generated power for the 4 bus case

Bus number	Pd	Qd	Pg	Qg
1	0	0	0	0
2	10	0	0	0
3	0	0	30	0
4	100	20	0	0

Table B.2: Constraints value for the buses for the 4 bus case

Bus	Vmax	Vmin	Qmax	Qmin	Pmax	Pmin
1	1,1	1,1	200	-200	200	0
2	1,1	0,9	0	0	0	0
3	1,1	0,9	0	0	30	30
4	1,1	0,9	0	0	0	0

Table B.3: Constraints value for the lines for the 4 bus case

$\textbf{Line}_{from,to}$	$\textbf{R}_{i,j}$	$\textbf{X}_{i,j}$	$\textbf{I}_{i,j}^{max}$
2,4	0,5	0,3	250
2,3	0,5	0,3	250
2,1	0,5	0,3	250
1,3	0,3	0,2	250
1,4	0,3	0,2	250
3,4	0,2	0,1	250

B.1.2 Case 2: the 9 bus power system

Table B.4 shows the constrains on the buses voltage maximum and minimum, as well as power bounds on active and reactive power. In table B.5 the line constraints for the network are presented. Note again that the values are the same for all three phases, and that the system is assumed to have base values of $100MVA$ for base power and $115kV$ as base voltage.

Table B.4: Constraints value for the buses for the 9 bus case

Bus number	Discription	Vmax	Vmin	Pmax	Pmin	Qmax	Qmin
1	Gas, slack	1,1	1,1	300	0	300	-300
2	Wind	1,1	1	300	0	0	0
3	PV	1,1	1	300	0	0	0
4	Transformator	1,1	0,9	-	-	-	-
5	Business	1,1	0,9	-	-	-	-
6	Transformator	1,1	0,9	-	-	-	-
7	House	1,1	0,9	-	-	-	-
8	Transformator	1,1	0,9	-	-	-	-
9	Battery	1,1	0,9	50	0	0	0

Table B.5: Constraints value for the lines for the 4 bus case

Line$_{from,to}$	$\mathbf{R}_{i,j}$	$\mathbf{X}_{i,j}$	$\mathbf{B}_{i,j}$	$\mathbf{I}_{i,j}^{max}$
1,4	0	0,0576	0	250
4,5	0,017	0,092	0,158	250
5,6	0,039	0,17	0,358	150
3,6	0	0,0586	0	300
6,7	0,0119	0,1008	0,209	150
7,8	0,0085	0,072	0,149	250
8,2	0	0,0625	0	250
8,9	0,032	0,161	0,306	250
9,4	0,01	0,085	0,176	250

B.2 SDP data

Transition probability matrix with discterization level of 11.

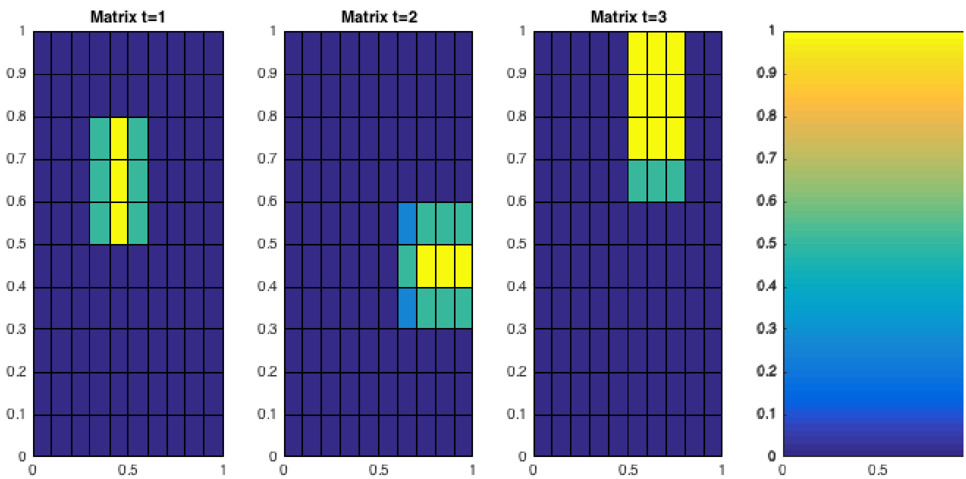

Figure B.1: Transition matrix.

B.3 Figures on projection of policy rules

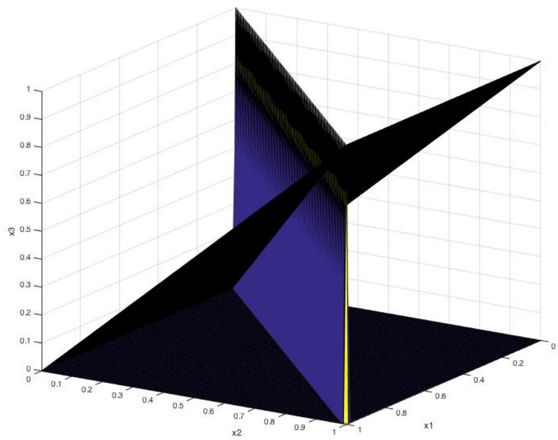

Figure B.2: Feasible space for policy projection in three dimensions.

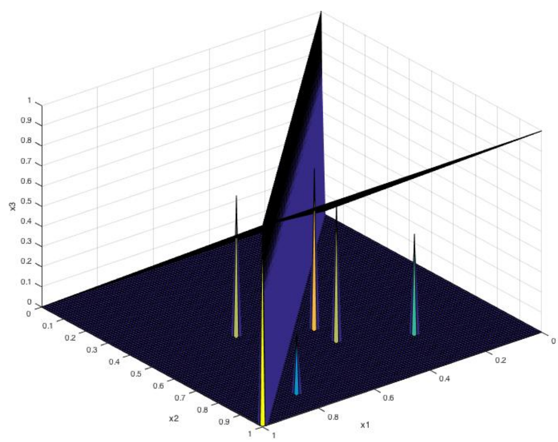

Figure B.3: Feasible space for policy projection in three dimensions with illustrative points.

Appendix C

Additional Results and Graphs from the Models

Due to concerns on length for this report, a number of figures may have been left out that might be of interest to the reader.

In this appendix, a few extra graphs and other relevant results will be presented.

C.1 Further figures form the simulations

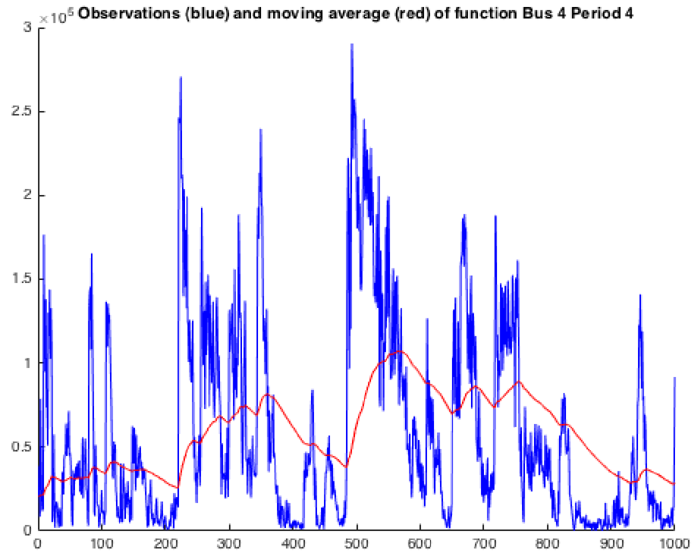

Figure C.1: Function approximation and observations for the 4 bus case with finite differences

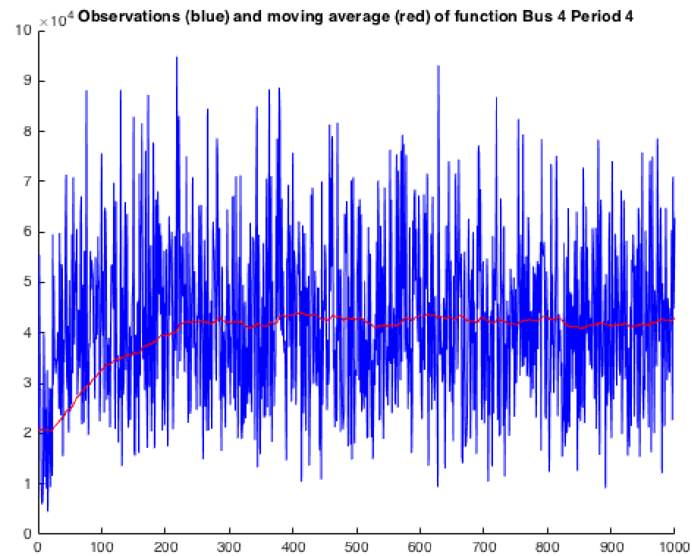

Figure C.2: Function approximation and observations for the 4 bus case with direct gradient

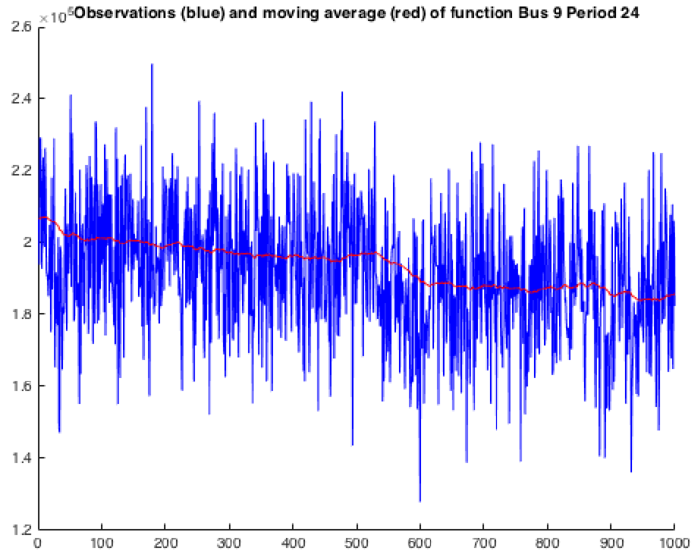

Figure C.3: Function approximation and observations for the 9 bus case with finite differences

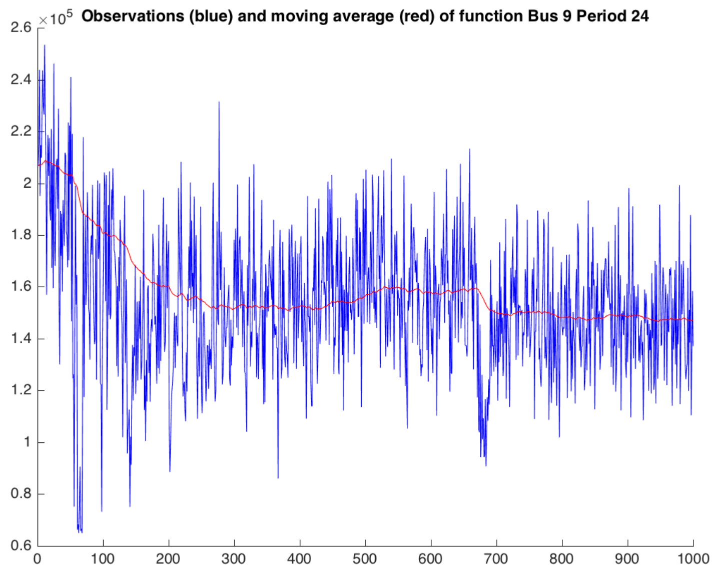

Figure C.4: Function approximation and observations for the 9 bus case with direct gradient

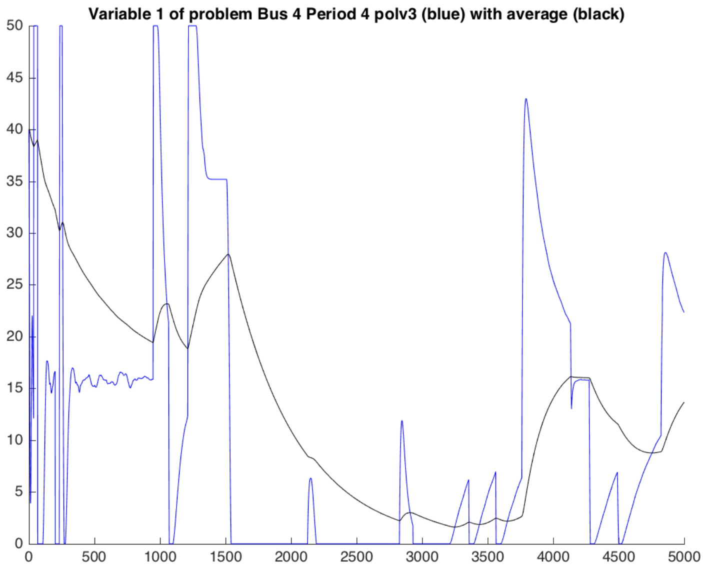

Figure C.5: Variable approximation of policy rule simulation