8-2018

Electrical Infrastructure Adaptation for a Changing Climate

Site Wang
Clemson University, sitew@g.clemson.edu

Electrical Infrastructure Adaptation for a Changing Climate

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Industrial Engineering

---

by
Site Wang
August 2018

Accepted by:
Dr. Scott J. Mason, Committee Chair
Dr. Russell Bent
Dr. Sandra D. Ekşioğlu
Dr. Harsha Gangammanavar

Abstract

In recent years, global climate change has become a major factor in long-term electrical infrastructure planning in coastal areas. Over time, accelerated sea level rise and fiercer, more frequent storm surges caused by the changing climate have imposed increasing risks to the security and reliability of coastal electrical infrastructure systems. It is important to ensure that infrastructure system planning adapts to such risks to produce systems with strong resilience. This dissertation proposes a decision framework for long-term, resilient electrical infrastructure adaptation planning for a future with the uncertain sea level rise and storm surges in a changing climate. As uncertainty is unavoidable in real-world decision making, stochastic optimization plays an essential role in making robust decisions with respect to global climate change. The core of the proposed decision framework is a stochastic optimization model with the primary goal being to ensure operational feasibility once uncertain futures are revealed. The proposed stochastic model produces long-term climate adaptations that are subject to both the exogenous uncertainty of climate change as well as the endogenous physical restrictions of electrical infrastructure. Complex, state-of-the-art simulation models under climate change are utilized to represent exogenous uncertainty in the decision-making process. In practice, deterministic methods such as scenario-based analyses and/or geometric-information-system-based heuristics are widely used for real-world adaptation planning. Numerical experiments and sensitivity analyses are conducted to compare the proposed framework with various deterministic methods. Our experimental results demonstrate that resilient, long-term adaptations can be obtained using the proposed stochastic optimization model.

In further developing the decision framework, we address a class of stochastic optimization models where operational feasibility is ensured for only a percentage of all possible uncertainty realizations through joint chance-constraints. It is important to identify the significant scalability limitations often associated with commercial optimization tools for solving this class of challeng-

ing stochastic optimization problems. We propose a novel configuration generation algorithm which leverages metaheuristics to find high-quality solutions quickly and generic relaxations to provide solution quality guarantees. A key advantage of the proposed method over previous work is that the joint chance-constrained stochastic optimization problem can contain multivariate distributions, discrete variables, and nonconvex constraints. The effectiveness of the proposed algorithm is demonstrated on two applications, including the climate adaptation problem, where it significantly outperforms commercial optimization tools.

Furthermore, the need to address the feasibility of a realistic electrical infrastructure system under impacts is recognized for the proposed decision framework. This requires dedicated attention to addressing nonlinear, nonconvex optimization problem feasibility, which can be a challenging problem that requires an expansive exploitation of the solution space. We propose a global algorithm for the feasibility problem's counterpart: proving problem infeasibility. The proposed algorithm adaptively discretizes variable domains to tighten the relaxed problem for proving infeasibility. The convergence of the algorithm is demonstrated as the algorithm either finds a feasible solution or terminates with the problem being proven infeasible. The efficiency of this algorithm is demonstrated through experiments comparing two state-of-the-art global solvers, as well as a recently proposed global algorithm, to our proposed method.

Dedication

This dissertation is dedicated to my loving wife, Lu Sun, who has always been unconditionally supportive of our family. Also to my parents, Haoying Xiao and Hong Wang, whom have always loved me selflessly and whose good examples have taught me to become who I am today.

Acknowledgments

First and foremost, I would like to express my most sincere gratitude to my advisor, Dr. Scott J. Mason, for his support, patience, motivation, and immense knowledge. His guidance has led me forward at all times during my research and my graduate studies. His consideration of my academic success and career development has pushed my growth far beyond my own expectations. I enjoyed my time working with Dr. Mason, both in person or remotely. I believe the lessons I learned from him will continue to motivate me to reach forward for something even bigger in my future.

Secondly, I would like to thank the brilliant researchers, Dr. Russell Bent, Dr. Carleton Coffrin, Dr. Donatella Pasqualini, Dr. Nathan M. Urban, Dr. Kaarthik Sundar, Dr. Harsha Nagarajan, and Dr. Hassan Hijazi, whom I worked with during my internship in Los Alamos National Laboratory (LANL). Thanks to the introduction by Dr. Scott J. Mason and Dr. Sandra Ekşioğlu, I was able to spend time in LANL as a student researcher during my Ph.D. It was a period of intense learning and exiting research on a set of cool problems. I truly appreciate and enjoyed the welcoming atmosphere created by these co-workers and their willingness to guide and enlighten me on how to do better research. In particular, I would like to express gratitude to my mentor, Dr. Russell Bent, who also serves as one of my committee members, for his patient, knowledgeable, and kind effort to help me navigate through my dissertation studies.

Thirdly, I would like thank another committee member, Dr. Harsha Gangammanavar, who served as a mentor to my Ph.D. He was a post-doc at Clemson University while we worked together on my first research projects. Dr. Harsha taught me how to conduct research by kindly, patiently demonstrating strong examples. As I was a young student who was eager to learn, code, and achieve, Dr. Harsha showed me the necessity to think and elaborate.

Fourthly, I would like to thank my committee member Dr. Sandra D. Ekşioğlu for her warming, encouraging, and patient guidance throughout the process of my dissertation.

Most importantly, I would like to send my deepest love to my dear wife, Lu Sun, who has been supportive this whole time. She is the most important reason for me to achieve higher standards through relentless learning, trying, and practicing. In my dark hours, she brings out the very best in me and encourages me to continue to do what I enjoy. Also, I want to express my unconditional love to my parents, Haoying Xiao and Hong Wang, who influence me the most by consistently motivating me to make my own decisions for my own life. I wouldn't become who I am today if it were not for them.

Last but not least, I would also like to thank my friends, Bryan Moore, Dr. Hadi Karimi, Mowen Lu, Rob M. Curry, Dr. Sreenath Chalil Madathil, Sufeng Niu, Shasha Wang, Tom Duignan, and Zahra Azadi for their important role in motivating and encouraging me.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The electrical grid, one of the most critical infrastructure systems, fundamentally supports modern society. The loss of electricity can introduce serious security threats along with massive financial losses. Although the modern infrastructure systems are designed to survive disturbances, severe natural events can still cause significant damages. For example, during a hurricane, the electricity transmission system can suffer localized inundation damage due to hurricane-induced water surge. As defined by National Oceanic and Atmospheric Administration (National Hurricane Center-Storm Surge Unit [2013]), "storm surge (SS) is an abnormal rise of water generated by a storm, over and above the predicted astronomical tide." Unexpected SS can jeopardize electricity substations and force them to shut down, which causes drastic generation and transmission capacity shedding and eventually leads to massive blackouts.

In 2009, Hurricane Sandy, which had the greatest SS in the northeastern US along coastal areas, struck New York City (NYC) as a Category II hurricane (Yates et al. [2014]). Although its associated winds were not impressively strong, a record-breaking five-foot surge happened at the Battery in lower Manhattan. At the same time in Long Island, this unprecedented SS destroyed 51 substations, nearly $2,500$ transformers, and $4,400$ distribution poles while causing more than 90% of the 1.1 million Long Island power authority customers to lose electricity. The excessive damages stressed bulk transmission systems to cause even more outages due to the interdependence of transmission systems, which led NYC to become crucially dependent on its local reserve generators. According to a report by Bloomberg [2013], local utility companies deployed preemptive actions, such as facility perimeter protection using temporary barriers and sandbags. Meanwhile, grid operator

enforced emergency dispatch policy to minimize potential downtime. However, when key substations were knocked out due to insufficient protection against the unexpected SS, massive power failures still occurred across a wide area. The severity of Hurricane Sandy was far worse than predicted (Abi-Samra and Malcolm [2011]).

Over the last 40 years, climate change is evident: the intensity, frequency, and duration of North Atlantic hurricanes have increased ever since 1980 (Webster et al. [2005]). Data from the U.S. Energy Information Administration suggests that weather-induced power outages have been increasing during the past two decades (see Executive Office of the President [2013]). An important immediate effect of the changing climate is sea level rise (SLR). A recent study by DeConto and Pollard [2016] uses an improved ice sheet mechanism to predict the SLR will reach up to one meter in the next century. According to Woodruff et al. [2013], SLR impacts electrical grids in a fundamentally different way than SS. First, the rising sea level increases the baseline of SS to superimpose the damages. If we assume Hurricane Sandy happens again, 50 years in the future, the damages associated with SS are exacerbated with SLR (Little et al. [2015]). Furthermore, SLR also imposes issues such as population displacement and accelerated coastal basin erosion, both of which can result in more uncertain, future risks for the electrical grid. To effectively manage risks in the long-term, it is necessary to incorporate the uncertainties of both SS and SLR when making electrical infrastructure planning decisions.

In climatology research, scientific studies have assessed the evolving, uncertain risks to the electrical grid under a changing climate. By coupling a circulation-based hurricane model with a hydrodynamic model, Lin et al. [2012] projected that SS levels are to evolve by a magnitude comparable to the projected SLR in NYC. Meanwhile, the rising sea level can also increase the frequency of catastrophic hurricanes, which collectively expose significant vulnerability of NYC to SS. Another study by Tebaldi et al. [2012] focusing on global temperature change used a semi-empirical model to predict the substantial SS frequency increase on tail events (i.e., today's "century-level" SS events can become "decade-level" events by mid-century). Furthermore, another study by Lin et al. [2016] indicates a ~ 3 to ~ 17 return rate of Hurricane Sandy in next nine decades, which can significantly increase flood risks in NYC. The SLR is dependent on representative concentration pathways (RCP) simulations which are used to describe four representative climate futures of different greenhouse gas concentration in the atmosphere. In another study by Little et al. [2015] about joint projection with SLR and SS, it is suggested that higher RCP trajectory will further increase the coastal flood

risks of SS together with a higher SLR. Finally, Nateghi et al. [2014] suggests that increasing flood risks can lead to massive electric system outages and blackouts.

Even though the topic of climate change can be highly controversial, it is important to realize that the future of the electrical grid is under substantial, uncertain risks. Recognition of the emerging impacts can provide valuable insights in building a resilient system for extreme events (see Cook et al. [2016], Nicholls and Cazenave [2010], and Van Vliet et al. [2012]). During a catastrophic event such as a hurricane, utility companies rely on state-of-the-art outage prediction models combined with scenario analyses to coordinate preemptive operations and logistics to promote electricity transmission system resilience (see Nateghi et al. [2011] and Nateghi et al. [2014]). These practices are often conducted days ahead of the hurricane's landfall, which is a relatively short time frame. Hence, security efforts against extreme events is limited due to this short time frame for deploying effective protection.

Utilizing the above short-term, prediction-based framework can hinder the developments of an electrical grid with reduced the overall risk and a strengthened long-term resilience, as suggested by Francis et al. [2011] and Ranger [2011]. Decisions can be made ahead of time to better prepare the system for uncertain extreme events. The adaptation provided should not be limited to short time frame options. Although hurricanes can happen within a week, the climate adaptation should consider long-term time frame to avoid locally optimal conclusions. Last but not least, the adaptation needs to consider endogenous system development, such as power systems physics, electricity demand growth, and population distribution adjustments. Therefore, an advanced, long-term electrical grid adaptation plan is needed for a more resilient future. However, drafting a climate adaptation for electrical infrastructure is non-trivial due to the dual complication of power systems physics as well as the uncertainty of nature. Base on climatology study projections by Bierkandt et al. [2015], SLR and the evolving hurricane-induced SS are two major stressors to coastal electrical grid security. This motivates decision-makers to seek adaptation plans that incorporates a changing climate, i.e., a climate adaptation. A forward-thinking climate adaptation can economically develop, strengthen, and protect electrical a infrastructure system to address a system's vulnerability to extreme events. The need for climate adaptation has been called for in multiple governments reports, such as Bloomberg [2013], Department of Defense [2014], and of Louisiana [2017]. However, there has been little research works to develop valid methodologies for making valid climate adaptations.

There exist many state-of-the-art physical models that quantify how nature evolves, such

as Pasqualini [2016], Paolo et al. [2015], Schmidtko et al. [2014], and Kopp et al. [2016]. These studies also incorporate the assessment of which impact to electrical infrastructure. What is missing in practice is validated decision-making models that couple engineered decisions with an evolving nature. In practice, decision makers draft climate adaptations intuitively using either a robust decision-making (RDM) framework or geographical information system (GIS) heuristic methods based on their experiences or simulated scenarios (see Groves and Sharon [2013], Kasprzyk et al. [2013], Russo et al. [2013], and Simm et al. [2015]). The RDM framework is widely considered under an assumed deterministic setup without any freedom in specifying reliability levels for decision makers. This framework can often bias decision making process by focusing solely on low probability events while omitting the more common decisions. Other methods utilizing rule-based heuristics either simplify by aggregating uncertainty approximations or by neglecting the underlying system's physical restrictions. These result in a local solution with biased or rigid suggestions that have poor system resilience. Additionally, overlooking the physical restrictions of electrical grids can overestimate reliability and underestimate adaptation costs, which lead to investments for failures.

The notion of electrical grid climate adaptation can essentially be interpreted as a decision-making process in a physically restricted system with some quantifiable objectives. The goal is to achieve climate adaptation plan with the best objective possible while obeying power flow physics during operations under exogenous uncertainty. Therefore, the determination of a climate adaptation can be formally be expressed as a stochastic optimization model. To the best of our knowledge, an optimization-based climate adaptation framework has not been addressed in climatology research. However, optimization models have been extensively studied in power systems research for operations scheduling, economic analysis, and expansion planning applications (Wood and Wollenberg [2012]).

In modern society, many industries, governments, and commercial applications rely on optimization models to support decision-making processes. These models are used to improve operational efficiency, ensure engineering constraints are enforced, and maintain safety and reliability standards. Unfortunately, many aspects of the underlying conditions are either uncertain or unknown in practice as suggested by Birge and Louveaux [2011] and Wallace and Fleten [2003]. To address these issues, stochastic optimization has been studied to model and analyze complex systems under uncertain setups. Stochastic optimization determine decisions that are robust or are of low cost across random relaxations realizations. Within the stochastic optimization literature domain, researchers have characterized uncertainties using a variety of approaches; Birge and Louveaux [2011] suggested

that there is no single superior approach for all problems.

In recent works, stochastic optimization has been applied to cope with electrical grid applications, helping achieve higher quality solutions that are more actionable for real-world implementation (see Gangammanavar et al. [2016], Morales et al. [2009], and Constantinescu et al. [2011]). Such developments provide viable and flexible ways to link state-of-the-art physical simulation models with optimization models to help obtain better decisions. Electrical grid expansion problems under uncertainty provide valuable insights for methodology development on long-term climate adaptation. Gorenstin et al. [1993] formulate an early stochastic electric infrastructure expansion planning problem by considering uncertain demand growth, fuel costs, and financial constraints. Moreno et al. [2013] demonstrates a basic two-stage stochastic optimization model that is also used by a transmission system planning model. Other stochastic optimization modeling schemes, such as probabilistic constraints (see López et al. [2007] and Yu et al. [2009]), multi-stage formulations (see Ahmed et al. [2003] and Collado et al. [2012]), and sample average approximation (see Jirutitijaroen and Singh [2008]), have also been investigated. For applications with extreme natural events, short-term preemptive power system outage prediction is investigated extensively by Nateghi et al. [2011]. However, a study of the long-term climate adaptation of evolving SS under climate change impacts is lacking in the literature.

My proposed research focuses on developing an integrated Simulation-Optimization Climate Adaptation (SOCA) framework for resilient electric grid adaptation under time-evolving, uncertain climate changes. First, I propose and implement a mixed-integer stochastic optimization model that incorporates Joint Chance Constraints (JCC) and operational power flow physics. The JCC contain a risk parameter $\boldsymbol{\epsilon}$ that controls how often the feasible solution is obtained. As suggested by Kleywegt et al. [2002] and Luedtke and Ahmed [2008], stochastic optimization models with JCC generalize the robustness criteria to reflect the willingness of decision-makers to assess the trade-off between non-trivial risks and effective cost savings. The research goal is to show the value of long-term climate adaptation via advanced decision-making methodologies under uncertainty in mind. SOCA bridges the gap between the topics of climate change and optimization-based decision-making to provide valuable insights for decision makers. My proposed optimization model will be evaluated and compared to scenario-based or GIS heuristics commonly found in practice.

Given the complexity of the proposed optimization model, we then focus on algorithmic design to solve the general form of SOCA, which can be referred to as Joint Chance-Constrained

programs with Finite-Support and Feasible, Integer Recourse (JCC-FSFIR). A key feature of the JCC-FSFIR is its representation of uncertainty in the optimization problem: the uncertainty is modeled with a finitely supported scenario set that approximates all possible realizations of uncertainty. To develop a solution approach for JCC-FSFIR, we develop an algorithm which decomposes JCC-FSFIR into scenarios. This algorithm, which will be referred to as scenario-based heuristic configuration generation (SHCG), concludes objective bounds and find high-quality solution by creating solution configurations by solving multiple, smaller-sized, tractable FSFIR problems.

As a next step, we focus on incorporating realistic modeling of the electrical grid operations in the SOCA framework to provide more accurate estimations of climate adaptation. To do this, we investigate the feasibility of the alternative current power flow (ACPF) problem, which is a nonlinear nonconvex problem that is applicable to both the SOCA framework and the SHCG algorithm. Detecting a feasible solution for the ACPF problem requires an exploitation of the solution space, which can be computationally expensive. Alternatively, the same goal can be achieved by proving an ACPF's relaxation is infeasible. With this motivation, we combine the idea of tight piece-wise convex relaxation and adaptive domain discretization techniques to design an ACPF infeasibility proof (ACPF-IP) algorithm guaranteed to find either a feasible solution or prove the problem infeasible. ACPF-IP outperforms state-of-the-art global solvers which are based on spatial branch-and-bound techniques.

The research contributions of this dissertation are summarized as follows:

- *A simulation-optimization framework* for long-term electrical infrastructure climate adaptation design by linking state-of-the-art climate simulations with an optimization model. This framework is meant to address the short-comings of deterministic methods in practice. State-of-the-art climate simulations, which models evolving climate change and SS, are incorporated in the decision-making process by requiring the decisions to be aware of exogenous uncertainties.

- *An optimization model* that incorporates network-based power flow physics for multi-period electrical infrastructure climate adaptation designs. At each period, the optimization model allows decision makers to develop and harden the existing system in the face of growing and uncertain impacts. Unlike short-term outage prediction models, the proposed model investigates long-term climate adaptation under uncertainty, which allows for the consideration of

more viable options to reduce the overall risk and cost.

- *A stochastic setup* containing probabilistic constraint is applied in a stochastic optimization model to ensure system resilience with a probabilistic measurement. This stochastic setup requires the resulting adaptation to withstand the impacts of a hurricane with probability $1 - \epsilon$, where parameter complicated uncertain.

- *A comprehensive computational experiment design*, using 118-bus IEEE test system to represent the Norfolk, VA area. The experiment investigates a 50-year climate adaptation. Exogenous uncertainties are modeled by SLR projection model (Kopp et al. [2016]) and hurricane and SS simulators (Pasqualini [2016]). Based on the experimental result, the advantages of SOCA is demonstrated over several deterministic methods.

- *A solution approach* that efficiently solves a class of mixed-integer stochastic optimization models. The combinatorial nature of the proposed optimization model in the SOCA framework makes the stochastic optimization model NP-hard; it is nearly impossible to solve realistic-size problems using the state-of-the-art commercial software. The proposed algorithm targets a more general class of optimization programs other than just the climate adaptation problem and uses decomposition methods to find a high-quality solution with objective bounds.

- *A global algorithm* that performs infeasibility proof on a class of nonlinear, nonconvex programs. For these programs, proving problem infeasibility can require an exhaustive search of the decision space. Alternatively, our proposed algorithm applies tight piece-wise relaxation with dedicated algorithmic designs to achieve the same goal more efficiently. Moreover, global convergence of the proposed algorithm is discussed and demonstrated.

The rest of this dissertation is structured as follows. We first introduce the SOCA framework for coastal electrical infrastructure climate adaptations in Chapter 2. The model applied in the SOCA framework is a stochastic optimization model with JCC and a finite scenario support. To validate the SOCA, we conduct a series of numerical experiments using SOCA to compare it with a number of methods used in practice and observe that SOCA is able to achieve high-quality solutions that are resilient to uncertainty. SOCA is then used to conduct sensitivity analyses to further understand the relaxation between the costs of climate adaptation and the potential risks. Through our experiments, it was clear that the proposed optimization model is computationally challenging

due to JCC-FSFIR's combinatorial nature. By exploiting the formulation's structural properties, we develop objective bounds and propose a converging global algorithm to improve model tractability in Chapter 3. The proposed algorithm for solving JCC-FSFIR shows promising results for both large-scale applications and the classical stochastic knapsack problem. Next, we discuss the necessity and challenges of analyzing a more realistic model that reflects the complex physics of the electrical grid. In Chapter 4, a general piece-wise convex relaxation formulation for ACPF is described, followed by our ACPF-IP algorithm that focuses on it. Numerical experiments are conducted with ACPF-IP to demonstrate it efficiency compared to state-of-the-art global solvers and a recent global algorithm. Finally, in Chapter 5, we provide a summary of this dissertation discuss future research directions.

Chapter 2

A Simulation-Optimization Framework for Critical Electrical Infrastructure Adaptation to Sea Level Rise and Storm Surge

2.1 Introduction

A key challenge of planning a climate adaptation is how to incorporate future uncertainty in the decision-making process (Walker et al. [2013]). The uncertain sea level rise (SLR) and storm surge (SS) are often characterized by state-of-the-art probabilistic models (Lin et al. [2012], Staid et al. [2014]). Decision frameworks that treat uncertainty deterministically using statistical or probabilistic measures can misrepresent the benefits of the climate science due to their ignorance of desirable alternative outcomes (Price [2015]). While down-scaling uncertainty to a "most likely" future can lead to an optimal local decision, it falls short of the goal of resilience requirements and can result in unnecessary investments, as the future's realization is unknown. For example, as shown by McInerney et al. [2012], when focusing on a low probability but high impact event, resulting adaptations can be surrounded by a deep probability valley "trap" that only considers few extreme

events.

In general, climate adaptation frameworks are categorized by Walker et al. [2013] as: 1) assumption-based, 2) robust decision-making, or 3) adaptive policy-making frameworks. In the scope of electrical grids, common decision frameworks are assumption-based frameworks using 1) down-scaled scenarios based on Geographic Information System (GIS) analyses, which are a group of heuristics that utilize past experience and GIS metric, and 2) simulated, worst-case scenarios based on statistical metrics and deterministic heuristics to target adaptation robustness.

Many current decision frameworks overlook the physical restrictions of the electricity grid. Although it may be viable in trivial situations, the inter-connected grids operate under complex power flow physics. Developing a climate adaptation by neglecting such system dynamics can cause unnecessary power imbalances and excessive transmission burden when responding to uncertain impacts. This, in turn, can leads to infeasible operational conditions that trigger massive outages.

The challenges of developing a climate adaptation are many. First, it is non-trivial to formulate the decision model while encompassing the inherent uncertainties appropriately. Next, the size of realistic electrical grids presents fundamental computational challenges in identifying a quality solution. Finally, incorporating power system physics during decision-making can magnify the challenge by enforcing a non-convex solution space. Therefore, developing a climate adaptation framework is an interesting research question that requires both stochastic decision modeling and guided solution search approaches.

The rest of this chapter is organized as follows. In Section 2.2, a brief overview of our Simulation-Optimization Climate Adaptation (SOCA) framework is presented. Next, the optimization model is introduced in Section 2.3. A case study is presented in Section 2.4, followed by a series of numerical experiments in Section 2.4.1 to show how and why SOCA can be used as a more advanced decision framework. Finally, Section 2.5 summarizes the conclusions of the SOCA and offers future research directions.

2.2 Overview of SOCA

SOCA bridges climate simulation models with a generic stochastic optimization model (STO) to provide climate adaptations for electrical grids. In SOCA, SLR and SS are incorporated as the two uncertain sources of flooding damage to the system. The work flow of SOCA is

presented in Figure 2.1 (under the blue shaded area). In contrast to state-of-the-art outage prediction models by Nateghi et al. [2011] and Nateghi et al. [2014] (Figure 2.1's gray shaded area), SOCA focuses on long-term adaptation to reduce overall risks and costs. SOCA uses stochastic optimization modeling techniques, which consider the presence of randomness during optimization. The link between uncertainty and the decision-making process are simulated scenario, which characterize future uncertainty empirically. The STO takes randomly sampled simulated scenarios, transmission system data, and user parameters as inputs. The output of SOCA is a long-term climate adaptation that is resilient to uncertainty and obeys power flow physics constraints. In contrast to other long-term adaptation frameworks (Figure 2.1's magenta shaded area), the novelty of SOCA is its more advanced decision-making model that enlarges problem scope to a more accurate representation of uncertainty. Moreover, SOCA enables decision makers to perform flexible analyses by tuning user parameters, a feature which is not trivially embedded in other long-term adaptation frameworks.

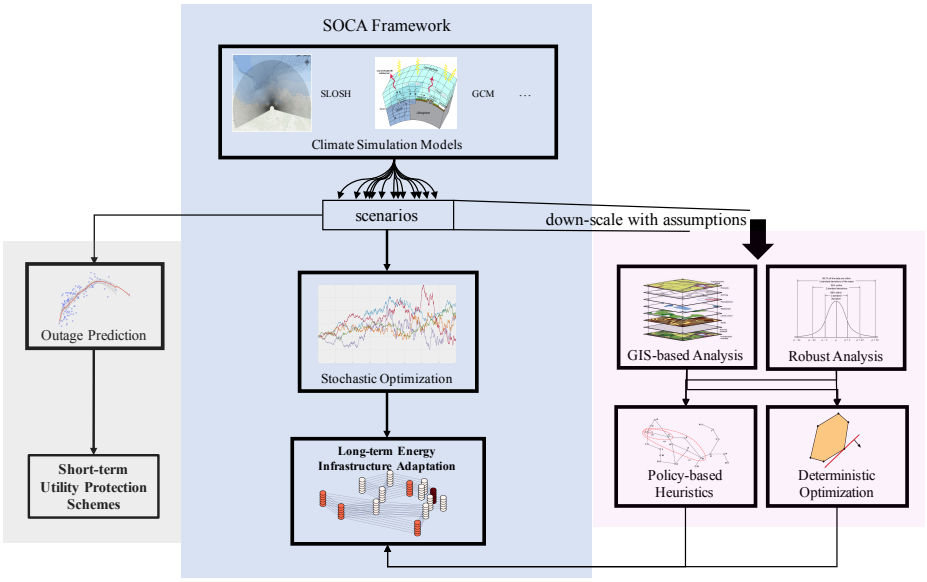


Figure 2.1: SOCA work flow comparing to other frameworks

2.3 Stochastic Optimization Model for Climate Adaptation

Power systems optimization encompasses a wide range of problem domains associated with various modeling techniques and solutions approaches. In recent years, stochastic optimization, which is tailored for decision-making under the presence of uncertainty, has attracted the attention

of many researchers (Birge and Louveaux [2011], Higle [2005]). Among power systems research studies, this modeling technique has been validated to address system intermittency more effectively and provide high quality solutions as shown in Sen et al. [2006] and Gangammanavar et al. [2016].

The proposed climate adaptation problem shares some similarity with stochastic transmission expansion planning (STEP) problem, which was conceptually proposed to address exogenous uncertainty impacts on transmission expansion planning in power systems (Gorenstin et al. [1993]). In STEP problems, the integral expansion decisions increase problem complexity by introducing disjunctive decision space, which proves the problem to be NP-hard. The initial STEP research by Gorenstin et al. [1993] has less focus on detailed power systems modeling since the detailed modeling can superimpose computational intractability for realistic-sized problems (see Zhang et al. [2012a], López et al. [2007], and Zhang et al. [2013]). Main stream power systems research try to address this challenges through formulation improvements (such as Alguacil et al. [2003], Zhang et al. [2012b], and Teimourzadeh and Aminifar [2016]) and/or more dedicated algorithmic developments (such as Escobar et al. [2004], Moreno et al. [2013], Qiu et al. [2016a], Ugranli and Karatepe [2016], da Silva et al. [1999], and Rahmani et al. [2013]).

Among existing stochastic STEP models, probabilistic constraints (see Yu et al. [2009], Zhang et al. [2012a], and López et al. [2007]), scenario approximations Jirutitijaroen and Singh [2008], and $N-1$ security (see Zhang et al. [2013], Ugranli and Karatepe [2016], and Qiu et al. [2016a]) are popular methods used during the development of an expansion plan to satisfy assumed uncertain risk criteria. Traditional perspectives of applied uncertainty are demand growth, costs, and contingencies (see Gorenstin et al. [1993], Zhang et al. [2013], and Qiu et al. [2016b]) while more recent efforts focus on renewable generation resources with high intermittency (see Ugranli and Karatepe [2016], Qiu et al. [2016a], and Yu et al. [2009]).

STEP models develop the system by adding new units of resources to ensure demand increments are satisfied with the lowest cost. When power systems are exposed to natural impacts, additional hardening options are available to strengthen power system component resilience for a lower failure probability. In more recent work, models incorporating both expansion and hardening decisions have been introduced in power system resilience research by Nagarajan et al. [2016] and Yamangil et al. [2015]. These models concentrate on resilient power systems design under optimal operation conditions. In the face of extreme events such as hurricanes, a feasible operation serves as the last line of defense. Hence, assessing the benefits of power systems designs under feasible

conditions is a new and interesting research direction. In addition, to the best of our knowledge, there has been little research on stochastic optimization models incorporating an evolving exogenous uncertainty.

We now introduce a STO that aims to produce long-term adaptation decision x for transmission systems under evolving exogenous uncertainty $\tilde{\omega}$. Consider a transmission system with B buses and T multiple periods. Let sets $\mathcal{N} := \{1 \cdots B\}$ and $\mathcal{T} := \{1 \cdots T\}$ denote all buses and all periods, respectively. Further, let set $\mathcal{E} := \{(i,j), i,j \in \mathcal{N}\}$ denote all transmission lines. Adaptation decisions x are meant to fulfill the growing electricity demand and probabilistic resilience requirements under random variable $\tilde{\omega}$ (i.e., expand and protect system resources to keep the system feasible/operating under the impacts of a changing climate). The exogenous uncertainty $\tilde{\omega}$ considers two dependent random variables, $\tilde{\omega}^{sl}$ for SLR and $\tilde{\omega}^{ss}$ for SS, which impact the transmission system in different ways. To formulate the STO in a closed form, a scenario approximation technique is used to empirically represent $\tilde{\omega}$ through a finite scenario set Ω.

2.3.1 Adaptation Decisions

Additional generation capacities are required to fulfill the growing electricity demand during adverse weather events. *What*, *where*, and *how many* in terms of to expand new capacity are the primary questions of interest (López et al. [2007]). As generators can fail during extreme events of SLR or SS, expansion decisions g require more sophisticated considerations. A trivial solution may exist that only considers new capacity addition in flood-free buses. However, such triviality may not be feasible or economically viable since delivery of electricity must obeys the restrictions of transmission resources. During a hurricane, generator loss not only jeopardizes local supply but can also cause potential transmission congestion that creates the flow imbalances. For example, during Hurricane Sandy, the inner city of New York was more dependent on local generators since nearby floodplain power plants were shut down due to security issues or damages (Bloomberg [2013]). In this situation, demand loads are met with local resources while transmission lines hold power imbalances in the wider area. In the long run, demand profile change and geographical limitations also play important roles in making expansion decisions.

A majority of power plants require nearby water resources for cooling purposes (Bierkandt et al. [2015]), which inevitably makes the plants vulnerable to flooding damage. During an unexpected SS, operators are forced to shut down generation units, which can consequentially introduce

unnecessary transmission stresses that cause blackouts (see Bloomberg [2013] and Bienstock and Mattia [2007]). However, system component availability changes are subject to both the passive exogenous impacts and active protection. To overcome the potential loss of system components, hardening decisions (e.g., sandbags, plywood, temporary barriers, flood walls, parameter sealing, equipment raising, and other initiatives) can be applied to strengthen resource resilience and keep them operational (Bloomberg [2013]). These decisions are practical and cost-effective in preserving the stability of the power supply during extreme events for the long run.

In reality, as suggested by Nateghi et al. [2011] and Nateghi et al. [2014], utility companies rely on outage prediction models to schedule temporary hardening one week before the hurricane landfall. The decision scope is short which limits the possibility of more sustainable hardening decisions. In the long run, relying on short-term hardening decisions can be more expensive and less efficient than a longer-term fix. Moreover, hardening decisions need to co-optimized with expansion decisions in a complex transmission systems. There are more effective hardening options available to withhold more risk, such as equipment raising, flood walls, and parameter sealing. However, they take a longer time (e.g., $>$ one week) to build. Hence, a forward-thinking hardening plan incorporated along with the expansion would mitigate long-term risks and costs.

In summary, adaptation decisions x incorporate electricity generation expansion decisions $g_{it} \in \mathbb{Z}^+$ and facility hardening decisions $h_{it} \in \mathbb{Z}^+$, where $i \in \mathcal{N}, t \in \mathcal{T}$. For a given bus i in period t, g_{it} and h_{it} denote the total units of generators and hardening, respectively. These variable are bounded by the initial condition of the existing network as well as the maximum amount physically allowed. As it is assumed that STO does not consider relocating and removing existing facilities, variables g and h are monotonically increasing over time $t \in \mathcal{T}$. Adaptation variable restrictions are modeled in constraints (2.1), where $[g_i^{init}, g_i^{max}]$ and $[h_i^{init}, h_i^{max}]$ are initial and terminal restrictions for generation and hardening units at each bus $i \in \mathcal{N}$, respectively.

$$g_{i1} \geq g_i^{init}, \quad \forall i \in \mathcal{N} \tag{2.1a}$$

$$h_{i1} \geq h_i^{init}, \quad \forall i \in \mathcal{N} \tag{2.1b}$$

$$g_{i(t-1)} \leq g_{it}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T} \setminus \{1\} \tag{2.1c}$$

$$h_{i(t-1)} \leq h_{it}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T} \setminus \{1\} \tag{2.1d}$$

$$g_{iT} \leq g_i^{max}, \quad \forall i \in \mathcal{N} \tag{2.1e}$$

$$h_{iT} \leq g_i^{max}, \quad \forall i \in \mathcal{N} \tag{2.1f}$$

2.3.2 Modeling Probabilistic Climate Resilience through Scenario Approximation

A resilient power system should be able to maintain its stability under drastically changing conditions (Nagarajan et al. [2016]). In SOCA, the resilience criteria considered is the capability to maintain operational flexibility under $\tilde{\omega}$. At a minimum, the electrical grid should maintain "*feasible operation* (FO)," which is defined as maintaining electricity supply to the majority of demand with high probability $1 - \epsilon$. Since the modeling method can vary when representing FO, we use a literal form of Joint Chance Constraint (JCC) to address the resilience requirements here, but note that, detailed FO modeling will be described in later this section.

$$\Pr\{\text{system is FO given } \tilde{\omega}\} \geq 1 - \epsilon \tag{2.2}$$

To explicitly express constraint (2.2), $\tilde{\omega}$ requires a closed-form representation as a first step. This representation can be, but is not limited to, a distribution function, a static point prediction, or any other approximations. Realistically, uncertainty $\tilde{\omega}$ results from costly, complex physical simulation models, such as Pasqualini [2016], which have no closed-form formulation. In order to formulate the STO, a natural intuition is to simulate the outcomes of $\tilde{\omega}$ empirically, which is commonly referred to as Sample Average Approximation (SAA) (see Kleywegt et al. [2002] and Luedtke and Ahmed [2008]). We perform Monte Carlo sampling to approximate $\tilde{\omega}$ with a finite scenario set composed of S scenarios $\Omega =: \{\omega_1, \omega_2, \cdots, \omega_S\}$. For notational convenience, let set $\mathcal{S} := \{1, 2, \cdots, S\}$ be an index set for the scenarios.

When given a specific scenario $\omega_s \in \Omega$ comprised of SLR and SS data, a system of constraints is required to model FO. Without abusing the notation of formulating constraints (2.2), the condition of FO is lifted with a binary indicator $(f_s)_{s \in \tilde{\Omega}} \in \{0, 1\}$ denoting whether FO can be achieved with scenario ω_s (i.e., the system is FO when $(f_s = 1)$, otherwise $(f_s = 0)$). Then, JCC (2.2) is reformulated as linear constraints (2.3) which specifically requires FO must be obtained for at least

$\lceil S(1-\epsilon) \rceil$ scenarios.

$$\sum_{s \in \mathcal{S}} f_s \geq |\mathcal{S}|(1-\epsilon) \tag{2.3}$$

The notion of constraint (2.3) is a disjunctive approximation of JCC (2.2), which implies how system risks are evaluated. The convergence of this scenario approximation is proven by Pagnoncelli et al. [2009] and Luedtke and Ahmed [2008] if $S \to \infty$. A key to using this method is to determine how many scenarios $|\mathcal{S}|$ should be incorporated for convergence of the approximation. Theoretical estimation of $|\mathcal{S}|$ for convergence guarantee can be conservative, which is often criticized for its consequential computational intractability. However, it applies to conducted posterior evaluation for solution quality without convergence proof. We refer the readers to the work by Pagnoncelli et al. [2009], Luedtke and Ahmed [2008], and Mak et al. [1999] for more details.

2.3.2.1 Objective Function

The objective of STO is to minimize the total cost of climate adaptation $x := \{p, g\}$, which is modeled in (2.4) by tracking the difference of expansion and hardening at each period. Let c^g_{it} and c^h_{it}, where $i \in \mathcal{N}$ and $t \in \mathcal{T}$, denote the cost of adding one unit of generation or hardening resources, respectively, at bus i at period t.

$$Min \quad \sum_{t \in \mathcal{T}} (\sum_{i \in \mathcal{N}} c^g_{it}(g_{it} - g_{i(t-1)}) + \sum_{i \in \mathcal{N}} c^h_{it}(h_{it} - h_{i(t-1)})) \tag{2.4}$$

2.3.3 Modeling impacts of the exogenous uncertainty

A scenario ω_s is comprised of SLR scenario $\omega^{sl}_{ts} \in \mathbb{R}$, where $t \in \mathcal{T}$ and $s \in \mathcal{S}$, and SS scenario $\omega^{ss}_{its} \in \mathbb{R}$, where $i \in \mathcal{N}$, $t \in \mathcal{T}$, and $s \in \mathcal{S}$. It is assumed that SLR is universal for the entire system while SS can vary at different buses[1]. Let binary indicators $a_{its} \in \{0, 1\}$, where $i \in \mathcal{N}$, $t \in \mathcal{T}$, and $s \in \mathcal{S}$, denote the availability of bus i at period t. Since SLR and SS impact the system differently, two additional binary variables $a^{sl}_{its} \in \{0, 1\}$ and $a^{ss}_{its} \in \{0, 1\}$, where $i \in \mathcal{N}$, $t \in \mathcal{T}$, and $s \in \mathcal{S}$, are required to denote component availability under the impacts of SLR and SS, respectively. Naturally, if a bus i is available at period t ($a_{its} = 1$), this requires the bus to be available under both SLR

[1] In order to free buss natural/artificial protected from sea levels impacts in the optimization model, the elevation of these buss pre-processed by choosing a large value

($a^{sl}_{its} = 1$) and SS ($a^{ss}_{its} = 1$). The disjunctive formulation of this logic is presented in equation (2.5).

$$a_{its} = a^{sl}_{its} \cap a^{ss}_{its} \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.5}$$

Equation (2.5) is linearly modeled using constraints (2.6).

$$\forall i \in \mathcal{N}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.6a}$$

$$a_{its} \leq a^{sl}_{its} \tag{2.6b}$$

$$a_{its} \leq a^{ss}_{its} \tag{2.6c}$$

$$a_{its} \geq a^{sl}_{its} + a^{ss}_{its} - 1 \tag{2.6d}$$

We next model the impacts of SLR and SS given scenario ω_s using disjunctive functions (2.7) and (2.8) for a given adaptation decision x. In SOCA, it is assume that SLR shuts down a bus definitively (i.e., generator and associated transmission lines will be unavailable while the local load is dispensed to other places). In function (2.7), bus i will be turned off permanently if scenario s SLR exceeds its elevation e_i.

$$a^{sl}_{its} = \begin{cases} 1 & \text{if} \quad \omega^{sl}_{ts} < e_i \\ 0 & \text{if} \quad \omega^{sl}_{ts} \geq e_i \end{cases} \tag{2.7}$$

For a bus to operate normally under a hurricane, it is assumed that the bus either stays above the SS or is sufficiently hardened. Function (2.8) indicates that bus i will be turned off if insufficient hardening decisions are made. Parameter c_i implies the additional height protected from SS by one unit of hardening decision applied to bus i.

$$a^{ss}_{its} = \begin{cases} 1 & \text{if} \quad \omega^{ss}_{its} < e_i \\ 1 & \text{if} \quad \omega^{ss}_{its} < e_i + ch_{it} \\ 0 & \text{if} \quad \omega^{ss}_{its} \geq e_i + ch_{it} \end{cases} \tag{2.8}$$

With functions (2.7) and (2.8), the impacts of scenario ω_s is reflected using binary variables a, which will be later used to determine FO conditions through an operational dispatch model. With buses and transmission lines shut off, a partially disabled network can still be FO. Unfortunately, functions (2.7) and (2.8) cannot be directly embedded into an optimization model. The logic is

modeled using a set of linear constraints (2.9):

$$(2a^{sl}_{its} - 1)(\omega^{sl}_{its} - e_i) \leq 0, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.9a}$$

$$(2a^{ss}_{its} - 1)(\omega^{ss}_{its} - (e_i + c_i h_{it})) \leq 0, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.9b}$$

With given a scenario ω_s and adaptation decision variables g and h, constraints (2.9a) and (2.9b) formulate the function (2.7) and (2.8), respectively. Constraints (2.9a) implies variable a^{sl} by setting the variable = 0 if SLR is above the bus elevation, otherwise it equals 1. Constraint (2.9b) implies variable a^{ss} by setting the variable = 0 if the SS is beyond the reach of hardening, otherwise it equals 1. The non-linearity introduced with variable products in constraints (2.9b) can be equivalently linearlized using McCormick relaxation (Tsoukalas and Mitsos [2014]).

2.3.4 Modeling Dispatch Operation

As the impact of exogenous uncertainty on system topology are modeled through binary variables a, modeling the FO condition additionally requires a dispatch model that simulates system operation. It is ideal to consider the Alternative Current Power Flow model (ACPF) proposed by Carpentier [1962], which capture both Ohm's Law and Kirchhoff's circuit law with non-convex, nonlinear constraints. However, due to its computational intractability, it is often approximated by the Direct Current Power Flow model (DCPF), which is reviewed by Stott et al. [2009]. For further simplification, it is also plausible to omit the approximated power flow equations in DCPF and only consider a Capacitated Network Flow model (CNF). In practice, decision makers omit the interconnected network assumptions and simply consider a Capacity-Based model (CB), which evaluates system operation through aggregated supply and demand as shown in Gorenstin et al. [1993].

Although approximations can effectively alleviate the computational burden, they also introduce biased estimations of system operations and costs that could hurt solution validity. For example, over-estimating the system capability in delivering electricity can result in underestimated costs and resilience . On the other hand, a more realistic model will provide solutions that estimate costs accurately but the optimization model suffers from the increased model complexity.

For finalizing the model formulation, the STO for climate adaptation is formally defined with notation $\mathcal{P}(\Omega, \epsilon) \rightarrow x^*$, where Ω denotes the input scenario Ω, ϵ denotes the risk parameter for JCC (2.3), and x^* denotes the output adaptation design. Let y be the consolidated dispatch decision

variables for describing system operations. Details of decision y vary when different dispatch models are utilized. For a given scenario ω_s, let $\xi(\omega_s) \to f_s$ denote a general dispatch model that concludes decisions f_s. FO is achieved when there exists a feasible dispatch decision y (i.e., $f_s = 1 \Leftrightarrow y \in \xi(\omega_s)$). This disjunctive logic is presented in (2.10).

$$\begin{cases} f_s = 1 \quad \text{if} \quad y \in \xi(\omega_s) \\ f_s = 0 \quad \text{if} \quad y \notin \xi(\omega_s) \end{cases} \tag{2.10}$$

Recall that $f_s = 1$ means the system is able to satisfy ξ with the "majority" demand request, where "majority" is defined through a ratio parameter $\lambda \in [0, 1]$. Let D_t, where $t \in \mathcal{T}$, denote the total demand at period t.

For CB with a given scenario ω_s, aggregated available generation capacity is required to be at least the majority demand λD. Dispatch decision y is defined with generation variables p^g_{its}, where $i \in \mathcal{N}$, $t \in \mathcal{T}$, and $s \in \mathcal{S}$. Each generation variable p^g is bounded by the product of unit generation capacity and total available generation units [2]. Let $\bar{g}_i$, where $i \in \mathcal{N}$, denotes the unit generation capacity parameter in MW. Then, the CB dispatch model is modeled in (2.11).

$$p^g_{its} \leq \bar{g}_i g_{it}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.11a}$$

$$\sum_{i \in \mathcal{N}} \bar{g}_i p^g_{its} a_{its} \geq f_s \lambda D_t, \quad \forall t \in \mathcal{T}, s \in \mathcal{S} \tag{2.11b}$$

These constraints control variables f to relax the dispatch decision's feasible region by each scenario. To summarize, STO (Ω, ϵ) with CB is characterized with (2.1), (2.3), (2.4), (2.9), and (2.11).

For CNF with a given scenario ω_s, $f_s = 1$ means the majority demand λD is satisfied under a capacitated network (i.e., power must be delivered through a interconnected network while obeying flow balances at each bus). In this case, dispatch decisions y are defined with power flow decisions p_{ijts} for each transmission line $(i, j) \in \mathcal{E}$, power generation variable p^g_{its}, and demand fulfillment decision p^d_{its}, where $i, j \in \mathcal{N}$ and $t \in \mathcal{T}$. Let $\bar{p}^D_{it}$ denote the electricity demand parameters in MW at bus i in period t. For each transmission line $(i, j) \in \mathcal{E}$, let p^{min} and p^{max} denotes the minimum and maximum flow allowed, respectively. Given this notation, the CNF dispatch model is modeled

[2] This variable is introduced for the consistency of model introduction. In model implementation, this variable can be omitted by combining constraints in (2.11)

in (2.12):

$$\forall t \in \mathcal{T}, s \in \mathcal{S}$$

$$p^d_{its} \leq \bar{p}^d_{it} a_{its}, \quad i \in \mathcal{N} \tag{2.12a}$$

$$\sum_{i \in \mathcal{N}} p^d_{its} \geq f_s \lambda D_t \tag{2.12b}$$

$$p^{min}_{ij} a_{its} \leq p_{ijts} \leq p^{max}_{ij} a_{its}, \quad \forall (i,j) \in \mathcal{E} \tag{2.12c}$$

$$p^{min}_{ij} a_{jts} \leq p_{ijts} \leq p^{max}_{ij} a_{jts}, \quad \forall (i,j) \in \mathcal{E} \tag{2.12d}$$

$$p^g_{its} \leq \bar{g}_i g_{it}, \quad \forall i \in \mathcal{N} \tag{2.12e}$$

$$M(f_s - 1) \leq \sum_{j|(j,i) \in \mathcal{E}} p_{jits} - \sum_{j|(i,j) \in \mathcal{E}} p_{ijts} + \bar{g}_i a_{its} g_{its} - p^d_{its} \leq M(1 - f_s), \quad \forall i \in \mathcal{N} \tag{2.12f}$$

Constraints (2.12a) fulfill demand if the bus is free from SLR and SS. Constraints (2.12b) requires the total fulfilled demand must satisfy the majority demand λD. Constraints (2.12c) and (2.12d) shut off a transmission line if either end bus is unavailable. Generation variable p^g is bounded with available generation resources in constraints (2.12e). Finally, constraints (2.12f) enforce flow balance at each bus (i.e., the power flow into a bus must equal the power flow out of the bus, plus the power consumed, minus the power generated). Note that constraints (2.12e) and (2.12b) are relaxed with $f_s = 0$ when no feasible dispatch decision y is identified. This relaxation is conducted through a large value M. To summarize, STO $\mathcal{P}(\Omega, \epsilon)$ with CNF is characterized with (2.1), (2.3), (2.4), (2.9), and (2.12).

The use of approximations for power flow physics is an useful and common practice justified by their design and operations (Coffrin and Van Hentenryck [2014]). For DCPF with a given scenario s, $f_s = 1$ means the majority demand λD is satisfied under DC approximated power flow. The key assumptions of DCPF require 1) transmission conductance to be relatively small compared to susceptance, 2) small voltage angle differences, and 3) voltage magnitudes of each bus considered equalling 1. Collectively, these assumptions linearize the voltage products and trigonometric functions in ACPF.

Here, DCPF is built upon CNF by enforcing additional constraints, which means constraints (2.12) must be satisfied previously. Dispatch decision y is defined with all CNF dispatch decisions together with voltage angle decisions θ_{its}, where $i \in \mathcal{N}$, $t \in \mathcal{T}$, and $s \in \mathcal{S}$. Let $[\theta^{min}_i, \theta^{max}_i]$ denote the allowed voltage angle interval. For each transmission line $(i,j) \in \mathcal{E}$, let b_{ij} denotes the line reactance.

To formulate DCPF, a reference bus with zero voltage angle is required. The index of the reference bus is denoted using R. DCPF requires the delivery of electricity to obey the network constraints presented in (2.12). Additional constraints are required to approximate power flow physics more accurately:

$$\theta_{Rts} = 0 \quad t \in \mathcal{T}, s \in \mathcal{S} \tag{2.13a}$$

$$\theta_i^{min} f_s a_{its} \leq \theta_{its} \leq \theta_i^{max} f_s a_{its}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.13b}$$

$$(f_s + a_{its} a_{jts} - 2)M \leq p_{ijts} - b_{ij}(\theta_{its} - \theta_{jts}) \leq (2 - a_{its} a_{jts} - f_s)M \quad \forall (i,j) \in \mathcal{E}, t \in \mathcal{T}, s \in \mathcal{S} \tag{2.13c}$$

Constraints (2.13a) regulate the reference bus voltage angle while constraints (2.13b) regulate bus voltage angle limits. Approximate power flow equations are enforced in (2.13c), which is relaxed with value M when no feasible dispatch decision y can be identified. To summarize, STO $\mathcal{P}$ with DCPF is characterized with (2.1), (2.3), (2.4), (2.9), (2.12), and (2.13).

2.4 Case Study

We now focus on conducting a series of numerical experiments using an IEEE standard test system from Coffrin et al. [2014] that has 118 buses and 177 transmission lines. The standard test case is adjusted to represent the Norfolk, VA area based on additional information from Pennsylvania-New Jersey-Maryland (PJM) electric regions (see PJM [2017]). Generation expansion is allowed on 104 buses while expanded generation types are determined based on information from the test case. Hardening is allowed for all buses, while unit hardening protection height c is chosen based on local generation capacity. Due to limited data sources, an *ad-hoc* cost system is used for determining the costs of expansion and hardening based on local terrain and generation type. The STO horizon is 100-years with a resolution of five decision periods: adaptation plan for year 2020, 2030, 2050, 2070, and 2090. The system layout is presented in Figure 2.2, where both the realistic network (left side) from PJM [2017] and test system (right side) are presented. Exogenous uncertainty is characterized by coupling a novel hurricane simulator by Pasqualini [2016] with the surge simulator Sea, Lake, and Overland Surge from Hurricanes (Jelesnianski et al. [1992]) . The scenario inputs are generated by climate scientists thanks to at the Los Alamos National Laboratory.

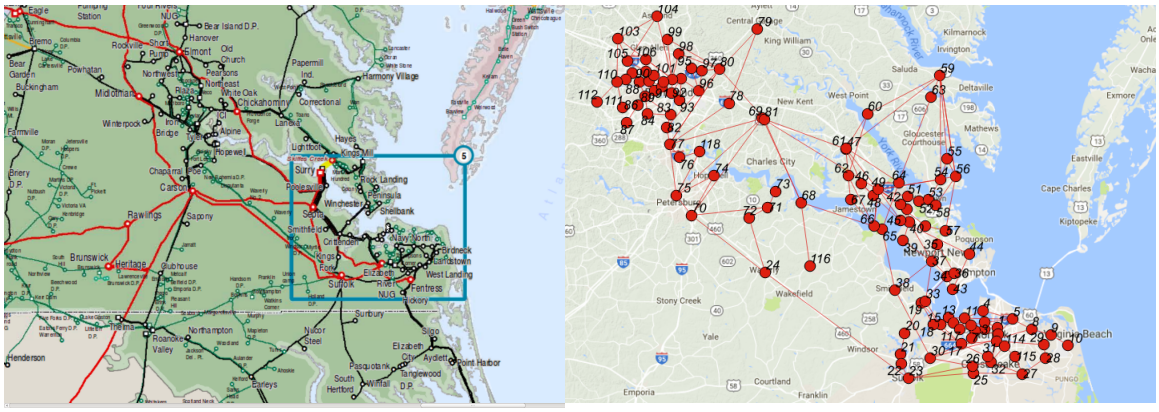


Figure 2.2: Testing Network representations: (Left) PJM Electric Regional Map for Norfolk and RichmondPJM [2017]; (Right) Adjusted IEEE 118-Bus Test SystemCoffrin et al. [2014]

2.4.1 Numerical Experiments

2.4.1.1 Comparison of Dispatch Models

When developing policies for power systems, a popular practice is to approximate the system by aggregating resources using CB. Although CB can be computationally easy, it can lead to less reliable decisions which result in little room for error during implementation. An experiment was created to demonstrate the differences among three models (CB, CNF, and DCPF introduced in 2.3.4) by simulating their adaptation decisions in other dispatch models.

STO is tested with all dispatch models using 100 simulated scenarios for one period, which result in optimal adaptation decisions x^* including x^*_{CB}, x^*_{CNF}, and x^*_{DCPF}. The parameter ϵ is set equal to 0.0 for a robust decision to eliminate the noise from risks allowed.

To evaluate solution quality, we define a special case of STO with $\mathcal{P}(\{\omega\}, 0, x = \bar{x}) \rightarrow \hat{f}$, where adaptation variables x are set to $\bar{x}$ for a single scenario input ω and binary output $\hat{f}$ captures whether FO can be achieved ($\hat{f} = 1$) or not ($\hat{f} = 0$). This special case is a single scenario simulation of adaptation design $\bar{x}$. A *resilience evaluation* of decision $\bar{x}$ denotes a series of simulations $\mathcal{P}(\{\omega \in \hat{\Omega}\}, 0, x = \bar{x})$ using scenarios from a test scenario set $\hat{\Omega}$. The outputs of these simulations $\hat{f}$ are summarized to form a *resilience ratio* between 0 and 1, which equals the total number of scenarios out of $|\hat{\Omega}|$ that achieve FO. Note that this *resilience evaluation* will also be applied for other experiments in this chapter.

The results for the three adaptation solutions are summarized in Table 2.1, where each row focuses on one adaptation solution. The associated costs of the solutions are recorded in column

"Estimated Costs." The resulting adaptations are evaluated under all dispatch models for 2000 externally sampled scenarios. Evaluation results are summarized in the three columns, one for each of the three different dispatch models.

		Estimated Cost	CB	CNF	DCPF
x^*	x^*_{CB}	13166	97.5%	0.00 %	0.00 %
	x^*_{CNF}	27228	98.1%	96.80%	0.00 %
	x^*_{DCPF}	31282	98.1%	97.65%	96.80%

Table 2.1: Solution of different dispatch modes and resilience evaluation results

As expected, CB has the lowest cost, but resilience evaluation indicates x^*_{CB} fails when simulated against more realistic dispatch models. The cost x^*_{CNF} is roughly twice the cost of x^*_{CB}, which indicates a significant underestimation of adaptation costs when using CB. Although the cost of x^*_{CNF} is only about 10% lower than the cost of x^*_{DCPF}, the simulation results of x^*_{CNF} are not able to satisfy a single scenario when approximated power flow equations are incorporated. This suggests that marginal objective value difference can also affect decisions. From this experiment, STO with DCPF is found to be the best approach with available computational tractability. Solution x^*_{DCPF}, which results from the most complicated dispatch model proposed, remains highly resilient across other dispatch models. In summary, we recognize the necessity and adequacy of utilizing DCPF dispatch model for the subsequent experiments.

2.4.2 Comparing Optimization with the Heuristic Methods

In this experiment, we demonstrate the benefits of utilizing an optimization-based framework for climate adaptation as compared various deterministic heuristic-based frameworks. In practice, heuristics are commonly considered for electrical infrastructure planning. Some assume a non-probabilistic uncertainty space by down-scaling simulated scenarios using statistical metrics. Others rely on simple rule-based or historical-data-based principles.

Heuristics decouple the climate adaptation problem into an *expansion problem* and a *hardening problem*. These methods neglect the anticipations of local changes which can propagate their effect through a system globally, and hence, yield a local solution. Recall that the goal of SOCA is to properly incorporate uncertainty in the decision-making process for higher solution quality in real world instances. If any heuristic works well, it implies that there exists certain trivial rules in the decision-making process. To begin with, we first describe four heuristics methods motivated from

previous research by Groves and Sharon [2013], Russo et al. [2013], government reports by Simm et al. [2015], and other unofficial communications.

- **Passive Hardening (PH):** First, determine expansion decisions by solving a *static expansion planning* problem. This problem can be reduced from the proposed STO by considering an artificial scenario without any SLR or SS. Then, based on the input scenario, harden a bus from its worst local SS after that bus is flooded at any period. For example, if bus i was flooded by a 2-meter SS in period t, harden bus i to ensure it is safe from this 2-meter surge in period $t+1$. If the same bus is subject to a 4-meter SS in period $t+1$, then harden the bus to 4 meters at period $t+2$. PH mimics what realistic decision makers do in reacting to previous events. This method has a natural time lag, which can expose significant vulnerability in an evolving climate.

- **Highland Expansion (HE):** First, determine expansion decisions similarly to PH but only add new generation capacity at buses with elevation higher than the 90-the percentile of SS based on the input scenario. Then, determine hardening decisions just like PH. HE tries to prioritize generation addition to more secured buses to avoid potential flooding risks. It can create inoperable plans since generation units in urban floodplain areas may heavily rely on local operating reserves Bloomberg [2013].

- **Bathtub Hardening (BH):** First, determine expansion decisions just like PH. Then, determine *hardening decisions* by preemptively hardening buses subject to the highest SS of the input scenario if that bus is not flooded by SLR. For example, if the scenario indicates bus i is subject to a 2-meter SS in period t, harden this bus to the worst SS in the same period. BH intends to stretch hardening decisions to its limitation by considering a conservative course of action preemptively. This method can be over-estimate of the costs when less representative scenario input is given. Intuitively, BH is supposed to provide high resilience. However, decoupling expansion and hardening decisions may trigger this heuristic to fail.

- **Extreme Hardening (EH):** First, determine expansion decisions similarly to PH but consider an additional 1% of demand growth. Then, preemptively harden buses to ensure the entire system is protected from the worst SS based on the input scenario. Finally, EH has the most conservative scheme with excessive demand growth expected to make up for the

decoupling of adaptation decisions. Meanwhile, additional hardening is provided to ensure the system is safe from any surprises. It is expected that EH will provide very resilient but expensive adaption plans.

The results of this experiment are summarized in Figure 2.3. The input scenarios are down-scaled from a large scenario set using percentile metrics: 50-th, 90-th, 95-th, and 100-th. For example, a 50-th percentile scenario is generated by summarizing the 50-th percentile SS and SLR at each period (and at each bus if necessary). Each heuristic is fed these four percentile scenarios while the result is represented by a line and four dots in Figure 2.3. In contrast to these heuristics, a deterministic optimization model (DET) is used with the same inputs. This DET can be reduced from STO by providing a single scenario and setting $\epsilon = 0$. The results of DET are shown as the orange line with four dots in Figure 2.3. Finally, STO is tested with 200 Monte Carlo-sampled scenarios changing ϵ from 0.0 to 0.5. The results are presented as a red line with multiple triangles for different ϵ values. The changing ϵ demonstrates how the concept of risk is interpreted differently between the deterministic and stochastic frameworks. For example, a 10% risk using STO yields has a fundamentally different result than considering 90-th percentile scenario to DET or heuristics.

All adaptation solutions are evaluated with an additional 2000 scenarios for 10 replications, and the average resilience ratio values are collected and presented in Figure 2.3. The x-axis indicates resilience ratio while the y-axis denotes the associated adaptation costs. The desired adaptation should be driving towards the right-bottom corner (a resilient adaptation with low costs).

Figure 2.3 suggest that STO is the more advanced framework since no other method can matches its level of resilience. The only resilient solution comes at a price when using EH with the maximum input scenario. SOCA balances costs with resilience by only focusing on a limited number of scenarios. Apart from SOCA, DET is the second best approach when fed with the maximum scenario. The results also suggest that resilience is not a monotonic function of costs. In this experiment, it is clear that 1) there exists no trivial rules in making climate adaptations and 2) implementing SOCA shows great benefits.

2.4.3 DET vs. STO

In the previous experiments, we observe that DET with a 100-th percentile scenario can yield a similar cost to STO with $\epsilon = 0$. Both frameworks try to address the goal of "robustness" while

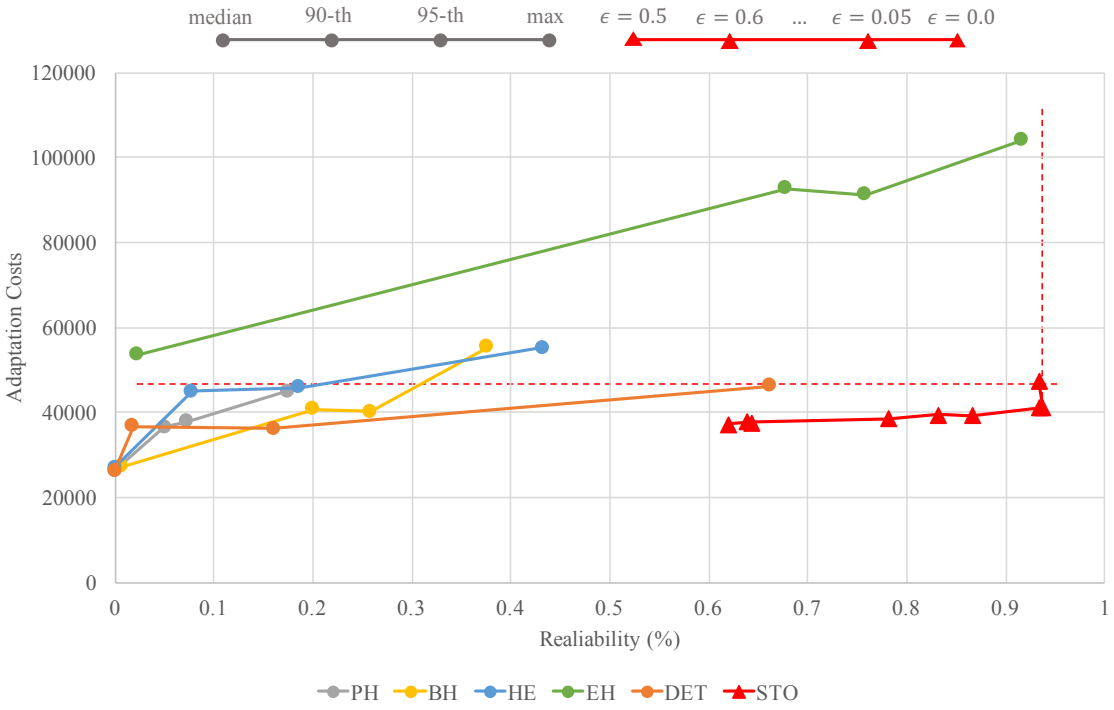


Figure 2.3: Comparison of optimization-based and heuristic-based methods

the resilience ratio suggests a difference of nearly 30%. A closer look at both adaptation solutions is presented in Figure 2.4. Each solution is separated into expansion (red maps) and hardening (blue maps). At each bus of the test system, we present the adaptation by stacking decisions over time from bottom to top. The color depth of each layer represents how many generation or hardening units are deployed, with a deeper color meaning more units. To better contrast the two solutions, the differences are marked with green boxes.

From the results, STO spends more on expansion by building more generation capability. DET intuitively trades-off expansion for hardening given its awareness of the worst scenario at each bus. Between the two solutions, the choice of bus for adaptations are obviously different. STO is not aware of the worst SS but an empirical distribution of uncertainty is formed at each bus, which causes STO to be more aware of what part of the system needs to be left out to satisfy the probabilistic risk. Furthermore, distinct differences can also be observed in the decision timing between the two solutions.

We next explore SOCA's flexibility to provide trade-offs between cost and resilience by tuning the parameter ϵ. In robust optimization-based frameworks, constraints (2.2) are usually

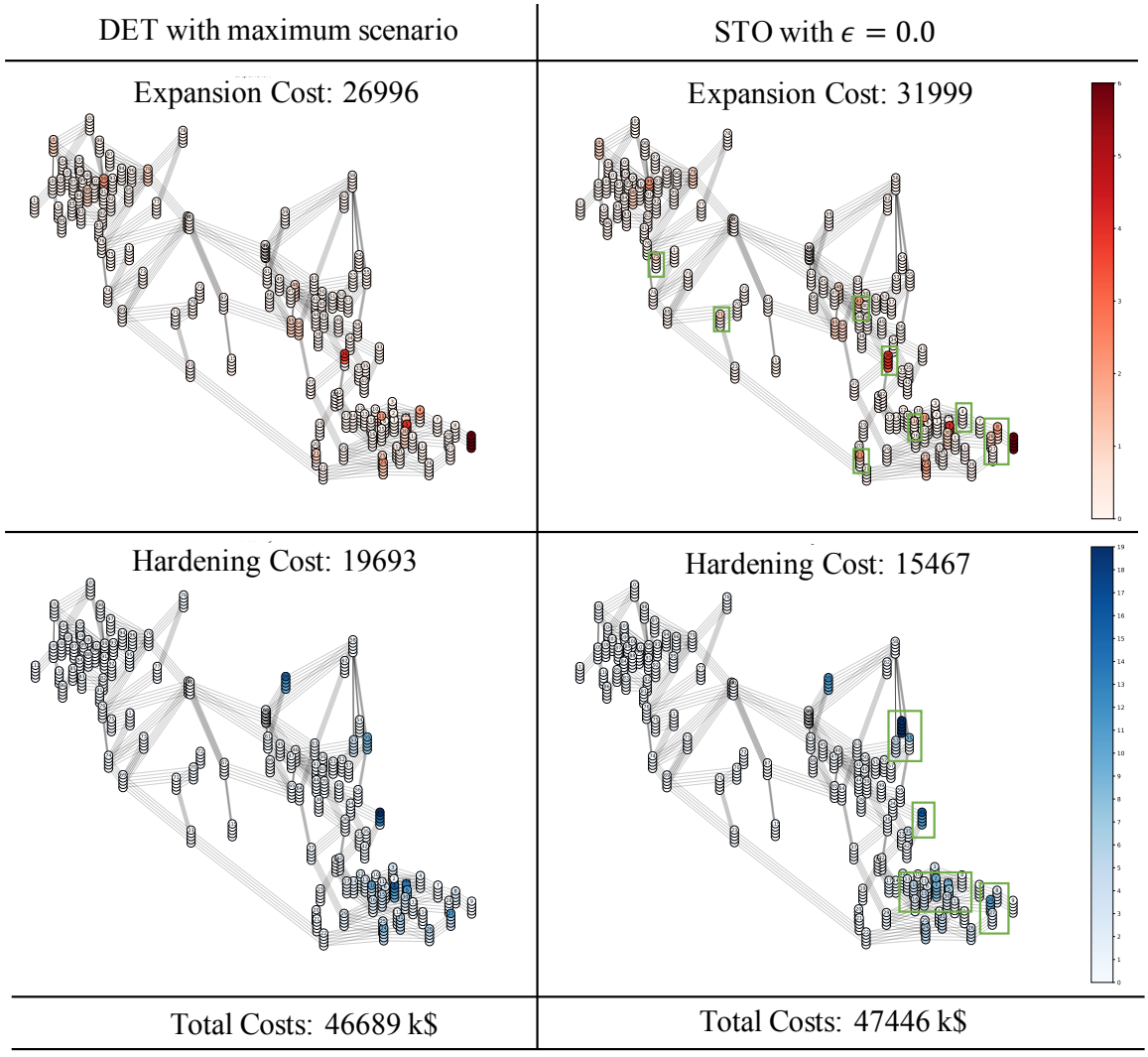


Figure 2.4: Detailed Solution Comparison between STO with $\epsilon = 0$ and DET 100-th percentile scenario

considered as a hard constraint, where ϵ is a sufficiently small value that requires all included scenarios to be feasible. Alternatively, STO enables ϵ to be a more general, real value. Recall this change of assumption is attractive due to the potentially high cost in fulfilling extreme events, which reflects a policy maker's willingness to agree to a non-trivial violation in exchange for a sufficient decrease in costs. However, this can create tremendous computational burden given the disjunction introduced during feasible scenarios selection.

We compare STO and DET in finding an adaptation plan that allows a maximum 5% or 10% uncertain risk. For STO, these purposes are achieved through parameter ϵ in constraints (2.3). For DET, 95-th or 90-th percentile scenario inputs are considered as counterparts, respectively. In Table 2.2, the details of each solution, basic details about expansion or hardening including costs, total buses chosen, total units added, average costs per MW/meter, and a median height of chosen buses [3] are presented. To better represent the details, the terminal expansion and hardening counts are plotted cumulatively by bus elevation in Figure 2.5.

	STO			DET		
	$\epsilon = 0\%$	$\epsilon = 5\%$	$\epsilon = 10\%$	100-th	95-th	90-th
Expansion Cost	31999	29506	29047	26696	26676	27468
Buses	13	11	10	9	9	9
Unit	27	23	25	21	21	22
$ per MW	18.50	15.55	18.75	16.39	16.41	16.58
Ave. Height (m)	12.50	10.15	9.15	9.32	8.67	10.79
Hardening Cost	15467	14768	12198	19693	9566	9254
Buses	30	27	26	30	23	22
Units	156	152	122	173	90	80
$ per Meter	212.61	212.49	192.86	202.38	203.53	231.36
Ave. Height (m)	2.60	2.53	2.84	3.17	2.84	2.83
Total Costs ($)	47446	44274	41246	46309	36284	36722
Evaluation (%)	93.27%	91.43%	92.08%	66.28&	16.00%	1.8%

Table 2.2: Comparing STO and DET solutions

Results in Table 2.2 reveal the fundamental difference of how risks are interpreted between the two perspectives of uncertainty. The cost breakdown between expansion and hardening are fundamentally different. STO consistently suggests investing more in expansion due to the various load profiles caused by SLR scenarios. As ϵ increases, adaptation costs gradually decreases to trade-off for resilience. In cases when risk is allowed with $\epsilon = 5\%$ or 10%, STO compensates adaptation resilience with conservative expansion for more flexibility in hardening. This is clear on the left side

[3] Note that average costs per MW/meter and a median height of chosen buses are calculated based on a weighted and aggregated vector of each bus.

of Figure 2.5, where the blue line selects buses with higher elevations. As for DET, the single scenario estimate leads the decision to the low probability "trap" of a single severe scenario. Although the finite scenario set has a rather limited scope compared to the true uncertainty, it describes the uncertainty space through an empirical distribution rather than a point estimate. DET is more sensitive to scenario change during elevation as hardening sharply drops when risks are allowed, as indicated on the right side of Figure 2.5. This change also degrades the system's resilience which is reflected in the resilience evaluation.

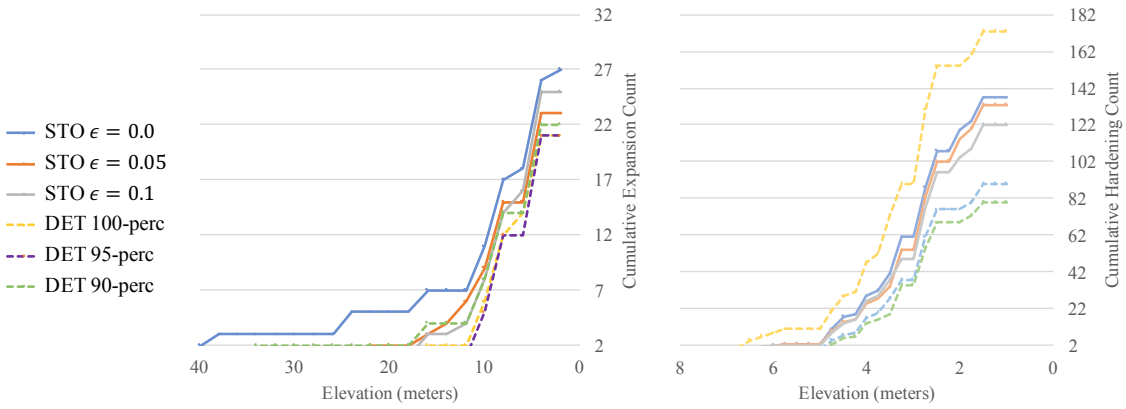


Figure 2.5: Solution representation using cumulative build count on different evaluations

These results suggest the fact that climate adaptation under uncertainty is not an easy problem. It is necessary to use a complex quantitative decision-making model, such as STO, to exploit the underlying complexity of developing better adaptation plans. Furthermore, the flexibility of SOCA to trade-off costs with resilience is valuable functionality for policymakers to learn and improve their decision-making process.

2.5 Conclusions and Future Research

Developing effective climate adaptations is challenging due to how nature evolves, how uncertainty is modeled and represented, and how electrical grids respond to exogenous impacts. These underlying complications suggest that it is inadequate to individually consider either climate studies or intuition-based decision-making methods for actionable adaptations. In this Chapter, we develop the SOCA decision framework to address these challenges by bridging state-of-the-art climate simulation models with optimization model to create economic, resilient climate adaptations.

Using a standard test system case study, we verify that SOCA is capable of providing high-quality solutions when compared to a number of deterministic approaches. Next, we discuss SOCA flexibility in providing trade-off analyses between cost and resilience. Finally, we provide an in-depth discussion of how a stochastic decision framework can benefit climate adaptation.

A critical next step in the future research is to address the computational tractability of SOCA for realistic-sized problems with a large number of scenarios. Although scenario approximation is a popular technique to formulate the STO, it imposes an NP-hard mathematical program that typically proves intractable for commercial solvers. To achieve improved tractability, a viable research direction, as suggested by related works by Luedtke [2014] and Ahmed et al. [2016], is to develop efficient algorithms that take advantage of the formulation's structural properties. It is also viable to consider an context-dedicated modeling techniques, such as Song and Luedtke [2013] and Song et al. [2014], to strengthen the current problem formulation, thereby, improving the STO's tractability. Furthermore, it is worthwhile to investigate how different climate simulation assumptions can affect the adaptation decisions using SOCA. For example, it is interesting to learn how adaptations responses to climate change when accelerated coastal basin erosion is considered.

Chapter 3

A Scenario-based Algorithm for Joint Chance-Constrained Programs with Finite support and Feasible, Discrete Recourse

3.1 Introduction

In modern society, many industrial, government, and commercial applications rely on optimization technologies and models to support decision making. These models are used to improve operation efficiency, ensure engineering constraints are enforced, and maintain safety and reliability standards. Unfortunately, many aspects of the underlying applications are uncertain or unknown in practice. To address this feature of such problems, stochastic optimization has been used extensively to model and analyze complex systems under uncertainty. Optimization models determine choices (decisions) that are robust or of low cost for most realizations of the uncertainty. Within the stochastic optimization literature, researchers have characterized uncertainties using a variety of approaches; no single approach is suitable for all problems (see Birge and Louveaux [2011]). Generally speaking, the approach used to model and solve a stochastic problem is based on structural

assumptions, such as problem size, solution region, and convexity.

In this section, we focus on problems that are modeled as joint chance-constrained programs with finite support and feasible integer recourse (JCC-FSFIR). A key feature of the JCC-FSFIR is its model of uncertainty: the uncertainty is modeled with a finitely supported scenario set that approximates all possible realizations of uncertainty. Most variations of the JCC-FSFIR (see Nemirovski and Shapiro [2006] and Bertsimas and Sim [2004] for examples) assume a full robustness criteria (i.e., that feasible recourse exists for all scenarios). In contrast, JCC problems define a risk parameter $\boldsymbol{\epsilon}$ that controls how often a feasible recourse is performed. As described in Luedtke and Ahmed [2008], our model of the FSFIR generalizes the robustness criteria to reflect the willingness of decision makers to assess the trade-off between nontrivial risks and effective cost savings.

Recently, there has been increased interest in problems that have a JCC-FSFIR structure. For example, the U.S. Department of Energy has identified resilient electric power system design as critically important (Ton and Wang [2015], Office of Electricity Delivery and Energy Reliability Smart Grid R&D Program [2014]). These problems have discrete decisions that model hardening and redundancy (design) options. The goal is to choose a minimum cost set of design options that meet a specified resiliency criteria during extreme events (see Yamangil et al. [2015] and Nagarajan et al. [2016]). JCC-FSFIR problems also arise in climate adaption applications. In these problems, the goal is to adapt engineered systems (such as electric power and natural gas) to survive the long-term impacts of climate change (United States Department of Energy [2013], Organization for Security and Co-operation in Europe [2016], Wang et al. [2017, submitted]). Given the computational challenges associated with solving JCC problems (discussed later), most existing solutions to these problems have used the full robustness (FSFIR) criteria and developed approaches to exploit this specific structure.

To address the JCC-FSFIR, we develop an algorithm that decomposes JCC-FSFIR by scenario. This algorithm, which we refer to as scenario-based heuristic configuration generation (SHCG), creates solution configurations by solving multiple smaller-sized tractable FSFIR problems. Without the loss of generality, we always consider the minimization problem where the maximization can be converted by nagating the objective function. The SHCG is tailored for those JCC-FSFIR that have the following properties:

1. Adding scenarios to an FSFIR always results in a monotonically increasing objective function.

2. Adding scenarios to an FSFIR that has a feasible solution always results in a JCC-FSFIR that also has a feasible solution.

It is important to note that the approach itself does not require that the JCC-FSFIR have these properties, only that the convergence and optimality guarantees of the proposed algorithm do. The SHCG algorithm makes the following key contributions:

1. It computes lower bounds for the JCC-FSFIR based on configurations generated from scenario-decomposed problems.

2. It computes upper bounds for the JCC-FSFIR based on configurations generated from scenario-decomposed problems.

3. It proposes a finite time convergence algorithm that solves stochastic optimization problems with integer recourse.

4. It has strong computational performance when compared to state-of-the-art commercial solvers.

3.2 Literature Review

The JCC-FSFIR falls into the general class of stochastic optimization problems that are modeled with two stages. The first stage captures non-anticipative decisions (here and now), and the second stage captures anticipative decisions (recourse). The first-stage decisions are made before realizations of uncertainty, and the second-stage decisions are made after uncertainties are revealed. Conceptually, the JCC-FSFIR is a two-stage stochastic optimization problem where the anticipative stage adjusts here-and-now decisions to achieve desirable feasibility on a set of scenarios (the support). In many stochastic optimization models, uncertainty is quantified via the objective function that comprises both here-and-now costs and expected recourse costs (see Shapiro et al. [2014]). This is in sharp contrast to the JCC-FSFIR, where the focus is obtaining a feasible recourse and the objective function is deterministic.

The JCC-FSFIR also assumes a finitely supported scenario set (i.e., a finite number of uncertainty realizations or scenarios). This is a common approach for modeling uncertainty in stochastic optimization because it can be challenging to analytically model arbitrary distribution functions (see Yuan et al. [2015] and Ahmed [2008]). These support sets are often constructed via sampling from

the corresponding distribution. In practice, the obtained scenario set can be representative of the majority outcomes of the random variables. Monte Carlo simulation is one important approach for constructing such sets because it has solution quality and convergence guarantees (see Pagnoncelli et al. [2009] and Luedtke and Ahmed [2008]). This method is typically used to approximate optimal decision making under an arbitrary uncertainty. This sampling method have been frequently applied to problems with expectation-based recourse in the anticipative stage. For problems with quantifiable anticipative stages, this method is often referred to as sample average approximation (SAA) and is discussed in detail in Kleywegt et al. [2002]. Here, we also assume that samples are used to approximate the uncertainty. However, because the scope of this section assumes that these samples are given, we do not focus on how to generate the samples.

Because the feasibility requirement is enforced via chance constraints, the support set introduces a disjunctive model that is atypical for stochastic optimization (see Pagnoncelli et al. [2009]). Generally, this disjunctive model is very hard to solve because of the nonconvex solution space. Some of the approaches used to solve disjunctive optimization models are (1) reformulation strategies (see Sen [1992]), (2) decomposition algorithms with specialized recourse (see Liu et al. [2016]), and (3) branch-and-cut algorithms of the scenario disjunctions (see Luedtke [2014] and Luedtke et al. [2010]). In most of this earlier work, it is assumed that the problem is convex, aside from the chance constraints. This assumption supports the use of strong duality to generate cutting planes which improve algorithmic performance.

In some cases, this convexity assumption can be too strong for real-world applications (i.e., Yamangil et al. [2015], Nagarajan et al. [2016], Bienstock and Shapiro [1988], Ralphs and Hassanzadeh [2014], Ahmed et al. [2004]). Some recent works extend the scope of this early work by assuming nonconvex here-and-now decisions. Song and Luedtke [2013] and Song et al. [2014] addressed applications through reformulation based on exploiting dedicated combinatorial structures. Zhang et al. [2014] focused on JCC, which integrates disjunctive structure in a multistage setup using a dynamic model. A study by Ahmed et al. [2016] focused on the generalized JCC problem. Here, nonconvex here-and-now decisions are allowed but the anticipatory-stage decision must be convex. This previous work provides some interesting ideas on how to reformulate the model by relaxing a part of the nonconvex space to handle stochastic integer problems. The reformulation penalizes deviations of duplicated first-stage decisions to captures the penalty them trying to differing themselves to satisfy each decomposed structure through Lagrangian dual. Using this method,

the algorithm attempts to conclude better relaxations. However, the recourse problem with fixed first-stage decisions is convex.

From a modeling perspective, the JCC-FSFIR also shares similarities with robust optimization (see Ben-Tal et al. [2009]). A key difference between robust optimization and the JCC-FSFIR is that the JCC-FSFIR requires the feasibility of recourse decisions to hold for a fraction of an uncertainty set (and the JCC-FSFIR gets to choose this fraction). Generally, as the risk parameter $\epsilon \to 0$, structurally the JCC-FSFIR looks more and more like a robust optimization problem where the support set is analogous to a finite uncertainty set. Indeed, our algorithm relies heavily on solving small robust optimization problems of this form.

The rest of this section is organized as follows. In Section 3.3, we introduce our notation, problem formulation, and the proposed decomposition scheme. Next, Section 3.4 discusses the structural properties of the JCC-FSFIR that are used to build our algorithm. The SHCG algorithm is presented in Section 3.5, followed by two example problem cases for evaluating the performance of SHCG in Section 3.6. Because these problems are computationally intractable when using state-of-the-art commercial solvers, numerical experiments are presented in Section 3.7. Finally, Section 3.8 summarizes our findings and conclusions and presents direction for future research.

3.3 General Formulation and Notations

In this section, we formally describe our formulation of the JCC-FSFIR and its assumptions, decomposition, and commonly used notation.

3.3.1 General Formulation

The JCC-FSFIR is formulated as a two-stage stochastic optimization problem with random variables $\dot{\omega}$:

$$\min\ c^\top x \tag{3.1a}$$

$$\text{s.t. } Ax \leq b \tag{3.1b}$$

$$\Pr\{y \in \xi(x, \dot{\omega}) \leq 0\} \geq 1 - \epsilon \tag{3.1c}$$

The here-and-now decisions, x, are made in response to any realization of random variables $\dot{\omega}$. For a realization of $\dot{\omega}$, a feasible recourse action, y, is determined based on function $\xi(x, \dot{\omega})$, which is a system of constraints. The probability of feasible recourse is modeled via the chance constraint specified in constraint (3.1c). An important distinction of this formulation is that it does not assume that x and y are continuous and hence that the feasible region imposed by $\xi(\cdot)$ is convex.

Formulation (3.1) assumes an arbitrary distribution over $\dot{\omega}$. Because arbitrary distributions are generally hard to formulate in closed form or via tractable optimization models, we use the SAA method to approximate the distribution using a finite support scenario set $\Omega := \{\omega_1, \omega_2, \cdots, \omega_S\}$ (see Pagnoncelli et al. [2009] and Luedtke and Ahmed [2008]), where $S = |\Omega|$ is the total scenario count. The binary indicator variables, $f_\omega, \forall \omega \in \Omega$, are used to select the scenarios with a feasible recourse action (constraint (3.2a)). With SAA and this notation, chance constraint (3.1c) is reformulated in (3.2) as

$$f_\omega = 1 \Leftrightarrow \{y_\omega \in \xi(x, \omega) \leq 0\} \quad \forall \omega \in \Omega \tag{3.2a}$$

$$\sum_{\omega \in \Omega} f_\omega \geq \lceil |\Omega|(1-\epsilon) \rceil \tag{3.2b}$$

$$f_\omega \in \{0, 1\} \quad \forall \omega \in \Omega \tag{3.2c}$$

We formally define a JCC-FSFIR problem with notation $\mathcal{P}(\Omega, \epsilon)$ in Formulation (3.3). Here, Ω and ϵ are the input parameters of a JCC-FSFIR.

$$\mathcal{P}(\Omega, \epsilon) = \min\ c^\top x \tag{3.3a}$$

$$\text{s.t (3.1b) and (3.2)} \tag{3.3b}$$

Given a $\mathcal{P}(\Omega, \epsilon)$, we use $\sigma_\epsilon^{\Omega *}$ and $h_\epsilon^{\Omega *}$ to denote an optimal solution and optimal objective value (i.e., $\mathcal{P}(\Omega, \epsilon) \rightarrow \langle h_\epsilon^{\Omega *}, \sigma_\epsilon^{\Omega *} \rangle$). We add a third argument to $\mathcal{P}(\Omega, \epsilon)$ when other constraints are included in the formulation.

For example, $\mathcal{P}(\Omega, \epsilon, x = \bar{x})$ constrains the assignments of variables x to take value $\bar{x}$. The functional usage of σ is denoted with $\sigma(\cdot)$, which returns the variable assignments of $\cdot$ in solution σ. For example, $\sigma_\epsilon^{\Omega *}(x)$ denotes the assignments of here-and-now variables x in the optimal solution of $\mathcal{P}(\Omega, \epsilon)$.

3.3.2 Special Cases of $\mathcal{P}$

$\mathcal{P}(\Omega, \epsilon)$ has a two-stage structure (i.e., for fixed x, constraint (3.2a) is separable on Ω). Note that no recourse of first-stage decisions is considered; the second stage is purely a feasibility problem. We now introduce special cases of $\mathcal{P}(\Omega, \epsilon)$ that are used later to exploit this separable structure.

Our first special case defines a *subproblem* $\mathcal{P}(\tilde{\Omega}, 0) \rightarrow \langle h_0^{\tilde{\Omega}*}, \sigma_0^{\tilde{\Omega}*} \rangle$, where $\tilde{\Omega} \subseteq \Omega$. This *subproblem* defines the case where a subset of scenarios must all have feasible recourse. This is a general robust optimization formulation.

In this special case, the formulation of $\mathcal{P}(\tilde{\Omega}, 0)$ is simplified by dropping the f_ω variables, i.e.,

$$\mathcal{P}(\tilde{\Omega}, 0) = \min\ c^\top x \tag{3.4a}$$

$$\text{s.t } Ax \leq b \tag{3.4b}$$

$$y_\omega \in \xi(x, \omega) \leq 0 \quad \forall \omega \in \tilde{\Omega} \tag{3.4c}$$

Our second special case defines solving $\mathcal{P}(\tilde{\Omega}, 0, x = \bar{x}) \rightarrow \langle h_0^{\tilde{\Omega}*}, \sigma_0^{\tilde{\Omega}*} \rangle$, where $\tilde{\Omega} \subseteq \Omega$ and the here-and-now variables x are set to be $\bar{x}$. Because this variant of the problem may not be feasible, we find it useful to define a new problem, $\mathcal{S}(\tilde{\Omega}, x = \bar{x}) \rightarrow \langle h_s^{\tilde{\Omega}*}, \sigma_s^{\tilde{\Omega}*} \rangle$, that minimizes the violation of recourse feasibility constraints for all included scenarios, given $x = \bar{x}$ (i.e., the violation of constraints (3.4c) and (3.4b)).

$$\mathcal{S}(\tilde{\Omega}, x = \bar{x}) = \min\ \lambda + \sum_{\omega \in \tilde{\Omega}} \lambda_\omega \tag{3.5a}$$

$$\text{s.t } A\bar{x} - \lambda \leq b \tag{3.5b}$$

$$y_\omega \in \xi(\bar{x}, \omega) - \lambda_\omega \leq 0 \quad \forall \omega \in \tilde{\Omega} \tag{3.5c}$$

$$\lambda \in \mathbb{R}^+ \tag{3.5d}$$

$$\lambda_\omega \in \mathbb{R}^+ \quad \forall \omega \in \tilde{\Omega} \tag{3.5e}$$

In this formulation, variables λ and λ_ω are slack variables that are used to allow violation of all constraints and ensure that $\mathcal{S}$ always has a feasible solution. It is useful to note that $h_s^{\tilde{\Omega}*} = 0$ implies that $\mathcal{P}(\tilde{\Omega}, 0, x = \bar{x})$ has a feasible solution (i.e., decision $\bar{x}$ is feasible for all scenarios $\omega \in \tilde{\Omega}$).

3.4 JCC-FSFIR Properties

In this section, some key structural properties of the JCC-FSFIR are described to motivate the algorithm proposed in Section 3.5.

3.4.1 Relations of $\mathcal{P}(\Omega, 0)$ and $\mathcal{P}(\Omega, \epsilon)$

I first focus on the properties of $\mathcal{P}(\Omega, 0)$. Let $\tilde{\Omega} \subseteq \Omega$, $\mathcal{P}(\tilde{\Omega}, 0) \rightarrow \langle h_0^{\tilde{\Omega}*}, \sigma_0^{\tilde{\Omega}*} \rangle$ and $\mathcal{P}(\Omega, 0) \rightarrow \langle h_0^{\Omega*}, \sigma_0^{\Omega*} \rangle$, then

Lemma 1. $h_0^{\tilde{\Omega}*} \leq h_0^{\Omega*}$.

Proof. By construction, $\mathcal{P}(\tilde{\Omega}, 0)$ contains a subset of the constraints in $\mathcal{P}(\Omega, 0)$ and is a relaxation. □

Let $\mathcal{P}(\Omega, \epsilon) \rightarrow \langle h_\epsilon^{\Omega*}, \sigma_\epsilon^{\Omega*} \rangle$ and $\alpha = \lceil |\Omega|(1-\epsilon) \rceil$, then

Lemma 2. *There exists an* $\tilde{\Omega} \subseteq \Omega$ *such that* $|\tilde{\Omega}| = \alpha$ *and* $h_0^{\tilde{\Omega}*} = h_\epsilon^{\Omega*}$.

Proof. Let $\Omega_a \subseteq \Omega$ be the subset of active scenarios in an optimal solution of $\mathcal{P}(\Omega, \epsilon)$. By construction $h_0^{\Omega_a*} = h_\epsilon^{\Omega*}$ and by (3.2b) we know that $|\Omega_a| \geq \alpha$. If $|\Omega_a| = \alpha$, then the existence of $\tilde{\Omega}$ is shown. If $|\Omega_a| > \alpha$, one can define a new scenario subset $\Omega_b \subset \Omega_a$ such that $|\Omega_b| = \alpha$. By construction, Ω_b satisfies (3.2b) and Lemma 1 ensures $h_0^{\Omega_b*} \leq h_0^{\Omega_a}$. Because $h_0^{\Omega_a*}$ is an optimal solution, $h_0^{\Omega_b*} = h_0^{\Omega_a*}$, which demonstrates that $h_0^{\Omega_b*} = h_\epsilon^{\Omega*}$ and the existence of $\tilde{\Omega}$. □

It is useful to know that every problem $\mathcal{P}(\Omega, \epsilon)$ can be converted into an equivalent $\mathcal{P}(\tilde{\Omega}, 0)$ problem. However, in practice, finding the correct $\tilde{\Omega}$ can be very challenging. This observation motivates the final relation presented. Let $\tilde{\Omega} \subseteq \Omega$, $\mathcal{P}(\tilde{\Omega}, 0) \rightarrow \langle h_0^{\tilde{\Omega}*}, \sigma_0^{\tilde{\Omega}*} \rangle$ and $\mathcal{S}(\Omega, x = \sigma_0^{\tilde{\Omega}*}(x)) \rightarrow \langle h_s^{\Omega*}, \sigma_s^{\Omega*} \rangle$, then

Lemma 3. *If* $h_s^{\Omega*} = 0$, *then* $\mathcal{P}(\Omega, 0) \Leftrightarrow \mathcal{P}(\Omega, 0, x = \sigma_0^{\tilde{\Omega}*}(x))$.

Proof. $\sigma_0^{\tilde{\Omega}*}$ is a feasible solution to $\mathcal{P}(\Omega, 0)$ because $h_s^{\Omega*} = 0$. $\mathcal{P}(\tilde{\Omega}, 0)$ is a relaxation of $\mathcal{P}(\Omega, 0)$ because $\tilde{\Omega} \subseteq \Omega$. Consequently, $\mathcal{P}(\Omega, 0, x = \sigma_0^{\tilde{\Omega}*}(x))$ yields an optimal solution of $\mathcal{P}(\Omega, 0)$. □

Lemma 3 yields the following insight: when the solution to $\mathcal{P}(\tilde{\Omega}, 0)$ happens to be feasible for all scenarios in Ω, then the equivalence between $\mathcal{P}(\tilde{\Omega}, 0)$ and $\mathcal{P}(\Omega, 0)$ is established. This is a key insight that motivates Algorithm 7 and was also considered by Yamangil et al. [2015].

3.4.2 An Upper Bound of $\mathcal{P}(\Omega, \epsilon)$

When $\epsilon > 0$, we cannot leverage Lemma 3 because $\mathcal{P}(\tilde{\Omega}, 0)$ is not necessarily a relaxation of $\mathcal{P}(\Omega, \epsilon)$. However, we show in Lemma 4 that for a sufficiently large scenario set $\tilde{\Omega}$, the solution to $\mathcal{P}(\tilde{\Omega}, 0)$ provides an upper bound of $\mathcal{P}(\Omega, \epsilon)$. Let $\tilde{\Omega} \subseteq \Omega$, $\mathcal{P}(\tilde{\Omega}, 0) \rightarrow \langle h_0^{\tilde{\Omega}*}, \sigma_0^{\tilde{\Omega}*} \rangle$ and $\mathcal{P}(\Omega, \epsilon) \rightarrow \langle h_\epsilon^{\Omega*}, \sigma_\epsilon^{\Omega*} \rangle$, then we have

Lemma 4. *If* $|\tilde{\Omega}| \geq \lceil |\Omega|(1-\epsilon) \rceil$*, then* $h_\epsilon^{\Omega*} \leq h_0^{\tilde{\Omega}*}$.

Proof. Because of the cardinality of $\tilde{\Omega}$ and the fact that all of the scenarios in $\tilde{\Omega}$ are satisfied by the definition of $\mathcal{P}(\tilde{\Omega}, 0)$, the assignment $\sigma_0^{\tilde{\Omega}*}$ satisfies constraints (3.1b) and (3.2) and hence is a feasible solution to $\mathcal{P}(\Omega, \epsilon)$. □

3.4.3 Lower Bounds of $\mathcal{P}(\Omega, \epsilon)$

The lower bound procedures presented in this section use sorted lists of a fixed cardinality of scenario subsets. We begin by defining some structural properties of these lists. Given a collection of scenarios Ω, there are $\gamma = \binom{|\Omega|}{n}$ possible subsets of cardinality n. Define $L_n = \langle \tilde{\Omega}_1, \tilde{\Omega}_2, \ldots \tilde{\Omega}_\gamma \rangle$ as the collection of all subsets of Ω where $|\tilde{\Omega}| = n$ sorted such that $h_0^{\tilde{\Omega}_i *} \leq h_0^{\tilde{\Omega}_{i+1} *}$. We can immediately observe two interesting properties of L_n lists:

1. Based on Lemma 2, without loss of generality, it is sufficient to consider L_n in the range of $1 \leq n \leq \alpha = \lceil |\Omega|(1-\epsilon) \rceil$.

2. For each scenario $\omega \in \Omega$, we can define L_n^ω as the first subset in L_n that includes ω, and by definition L_n^ω is the lowest-cost subset of size n that includes ω.

We now establish how L_n can be used to compute lower bounds for $\mathcal{P}(\Omega, \epsilon)$. Let $\Omega_o \subseteq \Omega$ be the subset of feasible scenarios in the optimal solution of $\mathcal{P}(\Omega, \epsilon)$. The first observation is that for any $n \leq \alpha$, the following property holds:

Lemma 5. $\max_{\omega \in \Omega_o} h_0^{L_n^\omega *} \leq h_\epsilon^{\Omega*}$.

Proof. By Lemma 2, it is sufficient to show that $\max_{\omega \in \Omega_o} h_0^{L_n^\omega *} \leq h_0^{\Omega_o *}$. For each $\omega \in \Omega_o$, there are two possibilities: (1) $L_n^\omega \subseteq \Omega_o$, in which case Lemma 1 applies; and (2) $L_n^\omega \not\subseteq \Omega_o$, in which case $h_0^{L_n^\omega *}$ is cheaper than all subsets of Ω_o of size n that include ω, which follows from the definition of L_n^ω. □

Note that the strength of this bound increases as n approaches α. In fact, when $n = \alpha$ this computation would compute the optimal solution of $\mathcal{P}(\Omega, \epsilon)$.

A key shortcoming of Lemma 5 is that we do not know the optimal scenario subset Ω_o a priori. To address this issue, we further relax Lemma 5 to consider only the α-cheapest scenarios in L_n. Given L_n, let $\Omega_\alpha \subseteq \Omega$ be the collection of the first $\alpha = \lceil |\Omega|(1-\epsilon) \rceil$ scenarios when sorting $\omega \in \Omega$ by $h_0^{L_n^\omega *}$, then

Lemma 6. $\max_{\omega \in \Omega_\alpha} h_0^{L_n^\omega *} \leq h_\epsilon^{\Omega *}$.

Proof. As before, by Lemma 2 it is sufficient to show that $\max_{\omega \in \Omega_\alpha} h_0^{L_n^\omega *} \leq h_0^{\Omega_o *}$. There are two possibilities: (1) $\Omega_\alpha = \Omega_o$, in which case Lemma 5 applies; and (2) $\Omega_\alpha \neq \Omega_o$, in which case Ω_α includes some ω that are cheaper than those in Ω_o and provides a lower bound to Lemma 5. This follows the construction of Ω_α, which sorts ω by increasing values of $h_0^{L_n^\omega *}$. □

Note that the special case of Lemma 6 for L_1 was proposed in Ahmed et al. [2016], where it is referred to as a quantile bound. A key feature of Lemmas 5 and 6 is that they do not rely on the strong duality theorem, which is commonly used for deriving lower bounds in the joint chance-constraint literature. In contrast, the lower bound described in Lemma 6 is valid for nonconvex structures.

Given Lemmas 4 and 6, we now have a procedure to compute upper and lower bounds to the JCC-FSFIR that will converge to the optimal solution of $\mathcal{P}(\Omega, \epsilon)$. We next derive valid inequalities for $\mathcal{P}(\Omega, \epsilon)$, which improve convergence of these bounding procedures.

3.4.4 Scenario Pruning

Let $\mathcal{P}(\Omega, \epsilon) \rightarrow \langle h_\epsilon^\Omega, \sigma_\epsilon^\Omega \rangle$ be a feasible solution and $\mathcal{P}(\tilde{\Omega}, 0) \rightarrow \langle h_0^{\tilde{\Omega}*}, \sigma_0^{\tilde{\Omega}*} \rangle$ be an optimal solution to a relaxed problem where $\tilde{\Omega} \subset \Omega$ and $|\tilde{\Omega}| \leq \lceil |\Omega|(1-\epsilon) \rceil$. Then the following inequality holds:

Lemma 7. *If $h_\epsilon^\Omega \leq h_0^{\tilde{\Omega}*}$, then $\sum_{\omega \in \tilde{\Omega}} f_\omega \leq |\tilde{\Omega}| - 1$ is a valid cut for $\mathcal{P}(\Omega, \epsilon)$.*

Proof. Because $h_\epsilon^\Omega \leq h_0^{\tilde{\Omega}*}$, there is no way that all of the scenarios in $\tilde{\Omega}$ are satisfied in the optimal solution of $\mathcal{P}(\Omega, \epsilon)$. At least one $\omega \in \tilde{\Omega}$ must be removed. □

In general, Lemma 7 provides valid cuts that eliminate some scenario combinations and make it easier to solve $\mathcal{P}(\Omega, \epsilon)$. However, the special case when $|\tilde{\Omega}| = 2$ is leveraged in the following scenario pruning procedure.

We next derive another set of inequalities by leveraging a counting argument to eliminate individual scenarios. Let $\mathcal{G} = (\Omega, \mathcal{E})$ be an undirected graph with one vertex for each scenario $i \in \Omega$. An edge exists between vertex i and vertex j if we cannot prove $f_i + f_j \leq 1$ is a valid inequality for $\mathcal{P}(\Omega, \epsilon)$ (i.e., using Lemma 7). Let $\deg(i)$ denote the degree of vertex $i \in \Omega$ and we show that for any vertex

Lemma 8. *If* $\deg(i) < \lceil |\Omega|(1-\epsilon) \rceil - 1$*, then* $f_i \leq 0$ *is a valid cut for* $\mathcal{P}(\Omega, \epsilon)$*.*

Proof. First observe that $\deg(i)$ indicates the maximum number of scenarios that can be paired with i in a solution of $\mathcal{P}(\Omega, \epsilon)$. By (3.2b), all feasible solutions to $\mathcal{P}(\Omega, \epsilon)$ satisfy $\lceil |\Omega|(1-\epsilon) \rceil$ scenarios. If $\deg(i) + 1 < \lceil |\Omega|(1-\epsilon) \rceil$, there is no feasible solution that includes i and $f_i \leq 0$ is valid. □

3.5 Scenario-Based Heuristic Configuration Generation Algorithm

The two-stage formulation of $\mathcal{P}(\Omega, \epsilon)$ is a deterministic mixed-integer linear program (MIP); thus, commercial MIP solvers, such as CPLEX or Gurobi, are a natural choice for solving such problems. However, as will be demonstrated in Section 3.7, these general-purpose solvers often struggle with instances of $\mathcal{P}(\Omega, \epsilon)$ that feature a large number of scenarios and/or complex recourse decisions. To help address these challenges, we develop an algorithm that leverages the structural properties of $\mathcal{P}(\Omega, \epsilon)$ developed in Section 3. The algorithm is inspired by formal two-stage decomposition approaches, such as Benders decomposition by Van Slyke and Wets [1969] and Dantzig-Wolfe decomposition by Dantzig and Wolfe [1960], and heuristic decomposition schemes, such as Coffrin et al. [2011] and Pillac et al. [2016]. A key advantage of this approach compared to previous works is that it does not rely on the strong duality theorem and is applicable to problems with integer recourse.

The algorithm, referred to as scenario-based heuristic configuration generation (SHCG), is designed to decouple the two core combinatorial challenges presented by $\mathcal{P}(\Omega, \epsilon)$:

1. The feasibility of any given scenario (i.e., constraint (3.2a))

2. The scenario subset selection needed to satisfy the chance constraint (i.e., (3.2b))

The first key idea of SHCG identifies a *first-stage assignment configuration*, which encodes a specific assignment of the first-stage variables that identifies the scenarios that are feasible given that assignment. The second key idea of SHCG identifies a *configuration union operator*, which encodes what happens to the first-stage variables if two or more configurations occur simultaneously. With these two ideas, at a high-level, the SHCG algorithm generates a collection of first-stage assignment configurations and uses the configuration union property to solve a set cover-like master problem that satisfies the chance constraint (3.2b). A key advantage of this configuration-based approach is that determining which scenarios are feasible requires solutions to only small $\mathcal{P}(\tilde{\Omega} \subseteq \Omega, 0)$ problems, which are assumed to be significantly easier to solve than $\mathcal{P}(\Omega, \epsilon)$. The rest of this section is organized as follows. First, an overview of the SHCG algorithm is described. This high-level description leverages subroutines for solving the set-covering master problem, generating new configurations, computing lower bounds, and pruning scenarios, each of which is described in detail in the subsequent sections.

3.5.1 SHCG Overview

The overall structure of SHCG is presented in Algorithm 1. SHCG takes a set of scenarios Ω and the chance-constraint parameter ϵ as inputs (line 2). The algorithm first initializes an upper bound, $\bar{h}_\epsilon^\Omega$, a lower bound, $\underline{h}_\epsilon^\Omega$, a best solution, $\bar{\sigma}_\epsilon^\Omega$, a configuration set, $\mathcal{C}$, and a cut set, Λ (line 1). With the current set of configurations, a new feasible solution is found by solving a master problem, $\mathcal{M}(\cdot)$ (line 4).[1] Leveraging the latest upper bound, valid cuts are computed using the GenerateCuts subroutine (line 5). A revised lower bound is determined with the UpdateLowerBound subroutine (line 6) and, lastly, new configurations are generated using the UpdateConfigurations subroutine (line 7).

This procedure is repeated until the upper and lower bounds converge and global optimality is proven or the algorithm times out (line 3). Upon completion, an assignment of all the decision variables is computed (line 9) and the upper bound, lower bound, and best solution are reported (line 10).

[1] In the interest of providing convergence guarantees, throughout this section we assume that Formulation (3.6) is solved to global optimality. However, in practice this formulation often benefits from a time limit.

Algorithm 1 SHCG Algorithm

Initialize $\bar{h}_\epsilon^\Omega \leftarrow \infty, \underline{h}_\epsilon^\Omega \leftarrow -\infty, \bar{\sigma}_\epsilon^\Omega \leftarrow \emptyset, \mathcal{C} \leftarrow \emptyset, \Lambda \leftarrow \emptyset$
function SHCG(Ω, ϵ)
  while $\bar{h}_\epsilon^\Omega > \underline{h}_\epsilon^\Omega$ & TIMEOUT = **false do**
    $\bar{h}_\epsilon^\Omega, \sigma \leftarrow \mathcal{M}(\mathcal{C}, \Lambda, \Omega, \epsilon)$
    $\Lambda \leftarrow$ GENERATECUTS($\mathcal{C}, \Lambda, \Omega, \epsilon, \bar{h}_\epsilon^\Omega$)
    $\underline{h}_\epsilon^\Omega \leftarrow$ UPDATELOWERBOUND($\mathcal{C}, \Lambda, \Omega, \epsilon$)
    $\mathcal{C} \leftarrow$ UPDATECONFIGURATIONS($\mathcal{C}, \Lambda, \Omega, \epsilon$)
  end while
  $\langle \bar{h}_\epsilon^\Omega, \bar{\sigma}_\epsilon^\Omega \rangle \leftarrow \mathcal{P}(\Omega, \epsilon, x = \sigma(x), f = \sigma(f))$
  return $\bar{h}_\epsilon^\Omega, \underline{h}_\epsilon^\Omega, \bar{\sigma}_\epsilon^\Omega$
end function

3.5.2 The Master Problem

Before introducing the master problem formulation, we must first formally define what a *configuration* is. A configuration, $c \in \mathcal{C}$, is associated with a subset of scenarios $\Omega^c \subseteq \Omega$ and a solution to the subproblem $\mathcal{P}(\Omega^c, 0) \rightarrow \langle h_0^{\Omega^c *}, \sigma_0^{\Omega^c *} \rangle$. To simplify computations across configurations, for a configuration c, we define the scenario feasibility variable $f_\omega \; \forall \omega \in \Omega$ as $f_\omega \Leftrightarrow \omega \in \Omega^c$.

The master problem, $\mathcal{M}$, takes a collection of configurations $\mathcal{C}$ and the chance-constraint parameter ϵ to compute the lowest-cost combination of configurations such that the chance constraint (3.2b) is satisfied. The method for combining configurations depends on the problem context, namely how the first-stage variables are affected when two configurations are considered simultaneously. In practical applications we have encountered, such as resilient design by Yamangil et al. [2014], power system network expansion by Gorenstin et al. [1993], ambulance location by Nickel et al. [2016], multi-commodity flow by Ruszczyński [2002], and optimal vaccination planning by Tanner et al. [2008] and Tanner and Ntaimo [2010], the max operation is sufficient for computing a configuration union and satisfies constraints (3.1b) and (3.2a). We use this operation in constraint (3.6c). A key feature of $\mathcal{M}$ is that scenario feasibility is captured in the configurations. Hence this formulation implicitly satisfies constraint (3.2a) and can focus on the combinatorics of satisfying constraint (3.2b).

We first reformulate $\mathcal{P}(\Omega, \epsilon)$ into a new optimization problem, $\mathcal{M}(\mathcal{C}, \Lambda, \Omega, \epsilon)$. Given configuration $c \in \mathcal{C}$, we use the shorthand $\bar{x}^c = \sigma_0^{\Omega^c}(x)$ to denote the first-stage assignment vector and $\bar{f}^c = \bar{f}_0^{\Omega^c}$ to denote the scenario feasibility vector. $\mathcal{M}$ is then defined as follows:

$$\mathcal{M}(\mathcal{C}, \Lambda, \Omega, \epsilon) = \min \; c^\top x \tag{3.6a}$$

$$\text{s.t } Ax \leq b \tag{3.6b}$$

$$x_i \geq \bar{x}_i^c z_c \quad \forall i = 1 \cdots |x|, c \in \mathcal{C} \tag{3.6c}$$

$$f_\omega \geq z_c \bar{f}_\omega^c \quad \forall\, c \in \mathcal{C}, \omega \in \Omega \tag{3.6d}$$

$$f_\omega \leq \sum_{c \in \mathcal{C}} z_c \bar{f}_\omega^c \quad \forall\, \omega \in \Omega \tag{3.6e}$$

$$\sum_{\omega \in \Omega} f_\omega \geq \lceil |\Omega|(1-\epsilon) \rceil \tag{3.6f}$$

$$f \in \Lambda \tag{3.6g}$$

$$z_c \in \{0,1\} \quad \forall c \in \mathcal{C} \tag{3.6h}$$

$$f_\omega \in \{0,1\} \quad \forall \omega \in \Omega \tag{3.6i}$$

In this model, constraints (3.6a) and (3.6b) preserve the first-stage structure of the original problem $\mathcal{P}$. Binary variables z control whether a configuration is selected to satisfy these constraints. Constraints (3.6c)–(3.6e) perform a union of the selected configurations based on the max operation. Constraint (3.6f) ensures that the union of configurations satisfies the chance constraint (i.e., constraint (3.2b)). The cuts are added in constraint (3.6g). In practice, if any knowledge of variables z exists, it is applicable to warm start the master problem with a supported MIP solver.

3.5.3 Configuration Generation

Line 7 of Algorithm 1 generates configurations to solve $\mathcal{P}(\Omega, \epsilon)$. In this section, we first introduce a heuristic method to generate an initial upper bound configuration. Then, we discuss two approaches for updating the configurations (the advantages and disadvantages of these two approaches are discussed in the numerical studies in Section 3.7). To generate these configurations we solve a specific collection of subproblems. For brevity, we define this procedure as $\textsc{Collect}(\Omega, d, \omega, \Lambda) := \{\tilde{\Omega} : \tilde{\Omega} \subseteq \Omega, \omega \notin \tilde{\Omega}, |\tilde{\Omega}| = d, (3.6g)\}$. This procedure defines scenario subsets of cardinality d that do not include scenario ω and satisfy constraint (3.6g) on Λ and $\tilde{\Omega}$.

3.5.3.1 Generate Initial Upper Bound Configuration

It is important to consider a scheme to prepare the master problem with at least one tangible solution. We consider a heuristic that solves a considerable-sized subproblem for an initial upper bound configuration. We define this heuristic as GenerateConfiguration-Initial and present

it in Algorithm 2. On lines 3–4, it sorts all single-scenario subproblems based on costs and selects the first $\lceil |\Omega|(1-\epsilon) \rceil$ scenarios to compose a scenario subset Ω^c. Line 5 solves the corresponding subproblem and the resulting configuration is returned. In the following discussions of configuration generation approaches, this heuristic is applied right after all single-scenario subproblems are solved. Therefore, the cost of line 3 is minor.

Algorithm 2 Generate Initial Upper Bound Configuration

1: **initialize** $\Omega^c \leftarrow \emptyset$
2: **function** GenerateConfiguration-Initial($\mathcal{C}, \Omega, \epsilon$)
3: $\quad \tilde{\omega} \leftarrow \text{Sort}(\omega \in \Omega$ **by** $h_0^{\omega *})$
4: $\quad \Omega^c \leftarrow \Omega^c \cup \tilde{\omega}_i$ **for** $i = 1 : \lceil |\Omega|(1-\epsilon) \rceil$
5: $\quad \langle h_0^{\Omega^c *}, \sigma_0^{\Omega^c *} \rangle \leftarrow \mathcal{P}(\Omega^c, 0)$
6: $\quad \mathcal{C} \leftarrow \mathcal{C} \cup \langle \Omega^c, \sigma_0^{\Omega^c *}, h_0^{\Omega^c *} \rangle$
7: $\quad$ **return** $\mathcal{C}$
8: **end function**

3.5.3.2 Enumeration

Our first implementation of UpdateConfigurations is based on an enumeration routine described in Algorithm 3. It generates configurations by exhaustively solving subproblems using scenario subsets with increasing cardinality. Line 3 maintains a cardinality counter that is initialized during the first call, and line 4 enumerates the scenario subsets with cardinality d that satisfy constraint (3.6g). Line 5 computes the solution to each subproblem, and line 6 stores that solution as a configuration. Coincidentally, solutions to a subproblem may be feasible for additional scenarios in Ω beyond those included in Ω^c. In lines 7–12 we build an extra *extended* configuration, $\Omega^{c'}$, that includes these additional scenarios. When $d = 1$, a heuristic is applied in line 14 to collect one additional configuration. Note that the convergence of Algorithm 1 is guaranteed when the solutions to all subproblems in this procedure are solved to optimality.

3.5.3.3 Incremental Enumeration

A key shortcoming of Algorithm 3 is that the number of additional subproblems that need to be solved in each iteration increases at a prohibitive rate. To address this challenge, we next propose an alternative approach for generating configurations. Algorithm 4 performs the enumeration incrementally on a scenario-by-scenario basis. In the first call, the algorithm creates all configurations for $d = 1$ (line 4) and sorts these configurations based on cost (line 5). It also generates and collects

Algorithm 3 Enumeration-based Configuration Generation

1: **initialize** $d = 0$
2: **function** UpdateConfigurations-Enumeration($\mathcal{C}, \Lambda, \Omega, \epsilon$)
3: $d \leftarrow d + 1$
4: **for** $\Omega^c \in$ Collect($\Omega, d, \emptyset, \Lambda$) **do**
5: $\langle h_0^{\Omega^c *}, \sigma_0^{\Omega^c *} \rangle \leftarrow \mathcal{P}(\Omega^c, 0)$
6: $\mathcal{C} \leftarrow \mathcal{C} \cup \langle \Omega^c, \sigma_0^{\Omega^c *}, h_0^{\Omega^c *} \rangle$
7: $\Omega^{c'} \leftarrow \Omega^c$
8: **for** $\omega \in \Omega \setminus \Omega^c$ **do**
9: $\langle h_s^{\omega *}, \sigma_s^{\omega *} \rangle \leftarrow \mathcal{S}(\omega, 0, x = \sigma_0^{\Omega^c *}(x))$
10: $\Omega^{c'} \leftarrow \Omega^{c'} \cup \omega$ **if** $h_s^{\omega *} = 0$
11: **end for**
12: $\mathcal{C} \leftarrow \mathcal{C} \cup \langle \Omega^{c'}, \sigma_0^{\Omega^c *}, h_0^{\Omega^c *} \rangle$
13: **end for**
14: **if** $d = 1$ **then** $\mathcal{C} \leftarrow$ GenerateConfiguration-Initial(Ω, ϵ)
15: **return** $\mathcal{C}$
16: **end function**

an initial upper bound configuration (line 1). In subsequent calls, one scenario is selected from the sorted list (line 9) and all combinations of other scenarios with this one are considered (line 10). For each of these combinations, both a configuration and an extended configuration are computed (lines 11–19). Finally, if all of the scenarios have been considered, the subset size is increased and the scenario counter is reset to start enumerating larger subsets (line 22). It is useful to note that, given a sufficient amount of time, this algorithm will enumerate all scenario subsets and has the same convergence properties as Algorithm 3.

3.5.4 Lower Bound Computation

The UpdateLowerBound subroutine leverages Lemma 6 to compute the lower bound of $\mathcal{P}(\Omega, \epsilon)$. This subroutine scans through sequences of sorted scenario subsets to calculate lower bounds; hence, we denote a sorted sequence as Sequence$(\Omega, d, \mathcal{C}, \Lambda) \rightarrow \langle \Omega_1, \Omega_2, \ldots, \Omega_{\binom{|\Omega|}{d}} \rangle$ such that $|\Omega_i| = d$ and $h_0^{\Omega_i *} \leq h_0^{\Omega_{i+1} *}$ $\forall i = 1 \ldots \binom{|\Omega|}{d}$. Because it is not practical to compute all of the $h_0^{\Omega_i *}$ values in this sequence, we relax the computation of $h_0^{\Omega_i *}$ to

$$h_0^{\Omega_i *} = \begin{cases} h_0^{\Omega_i *} & \text{if } \mathcal{P}(\Omega_i, 0) \rightarrow \langle h_0^{\Omega_i *} \sigma_0^{\Omega_i *} \rangle \text{ has been solved in } \mathcal{C} \\ \infty & \text{if } f_{\tilde{\omega}} = 1 \text{ for } \tilde{\omega} \in \Omega_i \text{ is infeasible for (3.6g)} \\ -\infty & \text{if } \mathcal{P}(\Omega_i, 0) \text{ has not been solved} \end{cases} \tag{3.7}$$

Algorithm 4 Incremental Enumeration-Based Configuration Generation

initialize $d \leftarrow 1$, $i \leftarrow 0$, $\tilde{\omega} \leftarrow \emptyset$
function UpdateConfigurations-Incremental($\mathcal{C}, \Lambda, \Omega, \epsilon$)
  if $d = 1$ **then**
    $\mathcal{C} \leftarrow$UpdateConfigurations-Enumeration($\mathcal{C}, \Lambda, \Omega, \epsilon$)
    $\tilde{\omega} \leftarrow$ Sort($\omega \in \Omega$ **by** $h_0^{\omega *} : \mathcal{P}(\omega, 0) \rightarrow \langle h_0^{\omega *}, \sigma_0^{\omega *} \rangle$)
    $d \leftarrow d + 1$
    return $\mathcal{C}$
  end if
  $i \leftarrow i + 1$
  for $\Omega^c \in$ Collect($\Omega, d - 1, \tilde{\omega}_i, \Lambda$) **do**
    $\Omega^c \leftarrow \Omega^c \cup \tilde{\omega}_i$
    $\langle h_0^{\Omega^c *}, \sigma_0^{\Omega^c *} \rangle \leftarrow \mathcal{P}(\Omega^c, 0)$
    $\mathcal{C} \leftarrow \mathcal{C} \cup \langle \Omega^c, \sigma_0^{\Omega^c *}, h_0^{\Omega^c *} \rangle$
    $\Omega^{c'} \leftarrow \Omega^c$
    for $\omega \in \Omega \setminus \Omega^c$ **do**
      $\langle h_s^{\omega *}, \sigma_s^{\omega *} \rangle \leftarrow \mathcal{S}(\omega, 0, x = \sigma_0^{\Omega^c *}(x))$
      $\Omega^{c'} \leftarrow \Omega^{c'} \cup \omega$ **if** $h_s^{\omega *} = 0$
    end for
    $\mathcal{C} \leftarrow \mathcal{C} \cup \langle \Omega^{c'}, \sigma_0^{\Omega^c *}, h_0^{\Omega^c *} \rangle$
  end for
  if $i = |\Omega|$ **then**
    $i \leftarrow 0$, $d \leftarrow d + 1$
  end if
  return $\mathcal{C}$
end function

The first case of (3.7) uses the actual value of $h_0^{\Omega_i *}$, which is available in $\mathcal{C}$. The second case of (3.7) eliminates known infeasible scenario combinations, and the last case of (3.7) provides a lower bound on unknown $h_0^{\Omega_i *}$ with $-\infty$. This function ensures the correctness of the lower bound computation while leveraging any information that is available as part of the already-computed configurations from $\mathcal{C}$.

Algorithm 5 presents the complete bounding procedure. The bound is initialized to $-\infty$ (line 2) and then strengthened by considering subsets of increasing cardinality (line 3). For each cardinality level, the sorted sequence of subsets is computed (line 5) and the bound is improved until Lemma 6's α-limit is reached (lines 6–9). Finally, the lower bound value is returned (line 12).

Algorithm 5 Update Lower Bound

```
1: function UpdateLowerBound(C, Λ, Ω, ε)
2:     h_ε^Ω ← −∞, α = ⌈(1 − ε)Ω⌉
3:     for d = 1 : α do
4:         γ ← (|Ω| choose d), Ω̂ ← ∅
5:         Let ⟨Ω_1, Ω_2, ..., Ω_γ⟩ ← Sequence(Ω, d, C, Λ)
6:         for i = 1 : γ do
7:             Ω̂ ← Ω̂ ∪ Ω_i
8:             break if |Ω̂| > α
9:             h_ε^Ω ← max(h_ε^Ω, h_0^{Ω_i *})
10:        end for
11:    end for
12:    return h_ε^Ω
13: end function
```

3.5.5 Generating Valid Inequalities

The GenerateCuts subroutine, presented in Algorithm 6, uses the upper bound, $\bar{h}_\epsilon^\Omega$, and configurations $\mathcal{C}$ to generate cuts using Lemmas 7 and 8. GenerateCuts is initialized with a fully connected graph $\mathcal{G}$ (see Lemma 8 for the definition of $\mathcal{G}$). The pruning process has two loops. The first loop (lines 3–6) generates cuts based on Lemma 7 (line 4) for each $c \in \mathcal{C}$. If a cut is generated on a scenario set of size two, the edge between the two scenario vertices is removed from the graph $\mathcal{G}$ (line 5). The second loop (lines 7-10) adds Lemma 8 cuts for all scenarios $\omega \in \Omega$ whose degree has become too small. Finally, the updated cut set Λ is returned in line 11.

Algorithm 6 Generate Cuts

Initialize complete graph $\mathcal{G} = (\Omega, \mathcal{E})$
function GenerateCuts($\mathcal{C}, \Lambda, \Omega, \epsilon, \bar{h}_\epsilon^\Omega$)
  for $c \in \mathcal{C}$ **do**
    $\Lambda \leftarrow \Lambda \ \cup \ \sum_{\omega \in \Omega^c} f_\omega \leq |\Omega^c| - 1$ **if** $h_0^{\Omega^c *} > \bar{h}_\epsilon^\Omega$
    Remove (ω_i, ω_j) from $\mathcal{G}$ **if** $\omega_i, \omega_j \in \Omega^c$, $|\Omega^c| = 2$, and $h_0^{\Omega^c *} > \bar{h}_\epsilon^\Omega$
  end for
  for $\omega \in \Omega$ **do**
    $\Lambda \leftarrow \Lambda \ \cup \ f_\omega \leq 0$ **if** $\deg(\omega) \leq \lceil |\Omega|(1-\epsilon) \rceil - 1$ for $\omega \in \Omega$
    Remove $\omega \in \mathcal{G}$ **if** $\deg(\omega) \leq \lceil |\Omega|(1-\epsilon) \rceil - 1$ for $\omega \in \Omega$
  end for
  return Λ
end function

3.5.6 Solving $\mathcal{P}(\Omega, 0)$

The SHCG algorithm relies heavily on solving subproblems of the form $\mathcal{P}(\Omega, 0)$, usually for $\tilde{\Omega} \subseteq \Omega$. For cases when $|\Omega|$ is large, the iterative algorithm described in Yamangil et al. [2015] has significant computational advantages. Hence, as a performance enhancement, SHCG utilizes this algorithm when solving $\mathcal{P}(\Omega, 0)$. In the interest of completeness, this procedure is presented in Algorithm 7.

Algorithm 7 is built on the premise that finding solutions for $\tilde{\Omega} \subseteq \Omega$ that are coincidentally feasible for all of Ω is faster than finding a solution for Ω directly. More formally, Algorithm 7 determines a solution to $\mathcal{P}(\tilde{\Omega}, 0)$ for a subset $\tilde{\Omega} \subseteq \Omega$ (line 4) and then checks whether this solution is feasible for all Ω (line 5). If this condition holds, by Lemma 3, this solution is returned as optimal (line 6). Otherwise, the most-infeasible scenario is found and added to $\tilde{\Omega}$ (line 8). The process repeats until feasibility for all scenarios is achieved.

Algorithm 7 Algorithm for Solving for $\mathcal{P}(\Omega, 0)$

1: **function** $\mathcal{P}(\Omega, 0)$
2: $\quad \tilde{\Omega} \leftarrow \{\arg\max_\omega \{h_0^{\omega *} : \mathcal{P}(\omega, 0) \rightarrow \langle h_0^{\omega *}, \sigma_0^{\omega *} \rangle, \omega \in \Omega\}\}$
3: $\quad$ **while** $\tilde{\Omega} \neq \Omega$ **do**
4: $\quad\quad h_0^{\tilde{\Omega} *}, \sigma_0^{\tilde{\Omega} *} \leftarrow \mathcal{P}(\tilde{\Omega}, 0)$
5: $\quad\quad$ **if** $\max(h_s^{\omega *} : \mathcal{S}(\omega, 0, x = \sigma_0^{\tilde{\Omega} *}(x)) \rightarrow \langle h_s^{\omega *}, \sigma_s^{\omega *} \rangle, \omega \in \Omega \setminus \tilde{\Omega}) = 0$ **then**
6: $\quad\quad\quad$ **return** $h_0^{\tilde{\Omega} *}, \sigma_0^{\tilde{\Omega} *}$
7: $\quad\quad$ **end if**
8: $\quad\quad \tilde{\Omega} \leftarrow \tilde{\Omega} \cup \arg\max_\omega \{h_s^{\omega *} : \mathcal{S}(\omega, 0, x = \sigma_0^{\tilde{\Omega} *}(x)) \rightarrow \langle h_s^{\omega *}, \sigma_s^{\omega *} \rangle, \omega \in \Omega \setminus \tilde{\Omega}\}$
9: $\quad$ **end while**
10: **end function**

3.6 Example Problem Cases

In this section, we test SHCG on two problems: (1) a stochastic knapsack problem (SKP) with feasibility recourse (Ross and Tsang [1989] and Abraham Flaxman [2011]), and (2) the climate adaptation problem (CAP) presented in Section 2.3. Although the problems have different recourse structure, they are both computationally intractable using commercial MIP solvers as the number of scenarios considered increases.

The SKP is a variation of the classic knapsack problem. Like the classic knapsack problem, the SKP finds the best subset of a collection of items, $\mathcal{N}$, subject to capacity constraints. However, unlike the classic knapsack problem, in the SKP each item has a random weight $(\dot{d}_i)_{i\in\mathcal{N}}$ and the knapsack has a random capacity, $\dot{B}$:

$$\max \sum_{i\in\mathcal{N}} c_i x_i \tag{3.8a}$$

$$\text{s.t. } \Pr\{\sum_{i\in\mathcal{N}} \dot{d}_i x_i \leq \dot{B}\} \geq 1-\epsilon, \tag{3.8b}$$

$$x_i \in \{0,1\} \quad \forall i \in \mathcal{N} \tag{3.8c}$$

Here, constraint (3.8a) maximizes the value of included items in the knapsack. Constraint (3.8b) constrains the probability of satisfying the knapsack capacity. For an arbitrary distribution of the random variables, scenarios $\Omega := \{(d^1, B^1), \cdots, (d^{|\Omega|}, B^{|\Omega|})\}$, are drawn using Monte Carlo sampling whereas the weight is assumed to be positive (i.e., $d^i \in \mathbb{R}^+ \; \forall i = 1 \ldots |\Omega|$). Hence, (3.8b) is reformulated as its deterministic problem (3.9).

$$\sum_{\omega\in\Omega} f_\omega \geq \lceil |\Omega|(1-\epsilon) \rceil, \tag{3.9a}$$

$$\sum_{i\in\mathcal{N}} d_i^\omega x_i - B^\omega \leq (1-f_\omega) M^\omega \quad \omega \in \Omega \tag{3.9b}$$

The recourse function in constraint (3.9b) checks the capacity violation for each scenario. In this model, $M^\omega = \sum_{i=1}^{N} d_i^\omega - B^\omega$. Therefore, the JCC-FSFIR formulation of problem (3.8) is formally defined in (3.10):

$$\mathcal{P}(\Omega, \epsilon) = \max \sum_{i\in\mathcal{N}} c_i x_i \tag{3.10a}$$

$$\text{s.t (3.9) and (3.8c)} \tag{3.10b}$$

Furthermore, constraint (3.11) is used in SHCG for the configuration union operator (i.e., constraint (3.6c)). This union operator will pick the lesser option when union configurations.

$$x_i \leq z_c \bar{x}_i^c + 2(1 - z_c) \quad \forall c \in \mathcal{C}, i \in \mathcal{N}. \tag{3.11}$$

3.7 Numerical Experiments

In this section we provide empirical results that evaluate the performance of SHCG. We first show the general applicability of SHCG by comparing SHCG with a general-purpose MIP solver. Second, we compare different column generation methods embedded within the SHCG framework.

3.7.1 Experiment Configuration

Our numerical experiments were conducted using the high-performance computing clusters at Los Alamos National Laboratory. Each experiment was performed on a single dedicated Intel(R) Xeon(R) CPU E3-2660-v3 processor with 22 Dual Threads Cores and 251 GB RAM. CPLEX 12.7 was used as the benchmark commercial solver for MIP test cases (32 threads). Our SHCG algorithm was implemented in Julia 0.5 and the mathematical program model package JuMP 0.15.1 (see Lubin and Dunning [2015]). All mixed-integer subproblems of SHCG (i.e., Algorithm 7) were solved using CPLEX. For SKP, a total time limit of 3600 seconds was used, and for CAP a time limit of 21600 seconds (i.e., 6 hours) was used. Any subproblem of SHCG was limited to 1800 seconds. We note that the lower bound of suboptimal solutions is used to calculate lower bound. We also note that SHCG can run Algorithms 3 and 4 in parallel. In this case, we limit SHCG to at most 16 parallel computations, each with 2 threads. The results in Table 3.1 are with sequential SHCG. The results in Tables 3.2, 3.3, 3.4 and 3.5 use parallel implementation of SHCG.

The SKPs have 8000 items. The costs of items were generated from a uniform distribution between 1 and 100. Each item's weight was generated using a normal distribution with a mean of 0.5 times the cost and a variance of 0.2 times the cost. All the weights were ensured to be positive, with a minimum threshold of 0.3 times the cost. Scenario sets of size 1000 were created (experiments with $|\Omega| \leq 1000$ scenarios use the first $|\Omega|$ of the scenario sets). The SPSAPs are based on the NESTA

IEEE-118 bus transmission system from Coffrin et al. [2014]. Scenario sets of size 200 were created (experiments with $|\Omega| \leq 200$ scenarios use the first $|\Omega|$ of the scenarios).

3.7.2 Analysis of the SKP

Table 3.1 compares CPLEX and SHCG on the SKP. The rows of the table show results for different ϵ values between 0.05 and 0.8. The results are grouped by $|\Omega| = (200, 500, 1000)$. In these results, SHCG uses Algorithm 3. The best solution for both methods is reported in the columns labeled "$\bar{h}$". The associated relative optimality gaps are recorded in the columns labeled "Gap(%)". The column labeled "$T^*(s)$" reports the CPU time (in seconds) when the best solution was found. To clearly show the benefits of SHCG, two additional columns explicitly compare SHCG with CPLEX. The column labeled "CPU Time for $\overline{h}(S) \geq \overline{h}(C)$" shows the CPU time (in seconds) when SHCG first finds a solution that is the same or better than the best solution found by CPLEX in 1 hour. The column labeled "Gap(S) $\leq$ Gap(C)" indicates whether the optimality gap of SHCG is smaller than that of CPLEX.

The SKP is difficult to solve for both CPLEX and SHCG. Neither approach is able to prove optimality within 1 hour. However, from Table 3.1, it is clear that SHCG consistently finds solutions that are better than those found by CPLEX. Moreover, the time required for SHCG to find a solution that is as good as CPLEX's best solution is much shorter in almost all cases. This suggests that SHCG has a natural advantage as a heuristic for finding high-quality solutions. Finally, SHCG has better optimality gaps than CPLEX in 22 of 27 test runs, which also suggests that SHCG has better convergence properties than CPLEX.

3.7.3 Analysis of the CAP

Table 3.2 and 3.3 summarizes our results on the CAP using Algorithm 3 for generating configurations in SHCG. The CAP is much more complicated than the SKP because it has integer recourse. Table 3.2 and 3.3 is organized by grouping choices of ϵ with sizes of Ω. CPLEX and SHCG are compared at four time intervals: 10 minutes, 1 hour, 2 hours, and 6 hours. The best results are highlighted in bold.

In Table 3.2 and 3.3, CPLEX is generally the best approach for finding a solution within 10 minutes when $|\Omega| \leq 20$. However, after an hour of computation, the advantages of SHCG are

Table 3.1: Performance summary of SKP with a 1-hour time limit

	CPLEX (C)			SHCG (S)			CPU Time for	Gap(S) ≤
ϵ	$\bar{h}$	Gap(%)	$T^*(s)$	$\bar{h}$	Gap(%)	$T^*(s)$	$\overline{h}$(S) ≥ $\overline{h}$(C)	Gap(C)
				$\|\Omega\| = 200$				
0.05	299	39.4	499	304	73.0	89	50	N
0.10	310	96.6	3375	318	72.6	270	41	**Y**
0.20	327	101.9	1385	343	72.6	407	28	**Y**
0.30	346	105.3	117	371	74.7	158	35	**Y**
0.40	377	88.5	1134	399	71.4	328	28	**Y**
0.50	401	86.4	1154	428	74.2	331	28	**Y**
0.60	429	79.2	850	459	74.7	263	28	**Y**
0.70	471	66.5	3345	482	77.6	284	36	N
0.80	504	53.5	2719	518	73.9	130	28	N
				$\|\Omega\| = 500$				
0.05	284	134.1	1548	290	73.0	1490	156	**Y**
0.10	292	157.8	2823	304	83.2	1225	103	**Y**
0.20	309	159.4	1425	329	80.6	1266	82.4	**Y**
0.30	334	147.6	3268	353	86.1	195	196	**Y**
0.40	359	148.2	809	381	84.3	2195	76.3	**Y**
0.50	392	131.5	1425	408	85.5	1982	110	**Y**
0.60	413	-	2546	434	74.7	1785	62	**Y**
0.70	471	102.4	2005	456	87.1	1785	77	N
0.80	473	53.5	1826	485	84.7	854	83	N
				$\|\Omega\| = 1000$				
0.05	268	-	TO	277	91.7	679	200	**Y**
0.10	285	-	TO	290	92.7	582	214	**Y**
0.20	305	198	219	316	91.5	712	195	**Y**
0.30	332	-	3336	341	92.7	1848	182	**Y**
0.40	358	-	349	366	91.8	2466	261	**Y**
0.50	382	-	350	390	94.1	692	299	**Y**
0.60	406	-	441	410	95.4	1286	269	**Y**
0.70	431	-	928	439	92.9	2276	177	**Y**
0.80	460	-	502	463	50.2	2138	1115	**Y**

Y=Yes; N=No; TO=Timeout; (-)=no gap is reported

clear because SHCG consistently outperforms CPLEX, especially when $|\Omega| > 20$. Once $|\Omega| = 200$, CPLEX is not able to find a feasible solution, whereas SHCG continues to improve. Table 3.2 and 3.3 also shows that the bounds of SHCG are better than those of CPLEX. There are only a few instances where CPLEX outperforms SHCG.

All the results in Table 3.2 and 3.3 SHCG use Algorithm 3 for generating configurations. Although it has convergence guarantees, the convergence rate slows considerably as $|\Omega|$ grows. Slow convergence is a combination of the number subproblems that need to be solved and the resulting increased computational time of solving the master problem (i.e., (3.6)).

In our experience, we noticed that a large number of configurations and subproblems are unproductive and do not contribute much to the optimal solution. In Section 3.7.4, we leverage this

Table 3.2: Performance summary on CAP using enumeration-based configuration generation (Algorithm 3): part 1

	$\leq 600s$				$\leq 3600s$				$\leq 7200s$				$\leq 21600s$			
	CPLEX		SHCG		CPLEX		SHCG		CPLEX		SHCG		CPLEX		SHCG	
ϵ	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g
$\lvert\Omega\rvert = 20$																
0.05	**32794**	**10.22**	NS	∞	**32794**	**10.22**	31952	13.17	**30824**	**5.48**	31952	13.17	29274	2.82	**29087**	**Opt**
0.1	**31542**	**8.68**	NS	∞	**31542**	**7.99**	31906	13.13	**31542**	**7.99**	31906	13.13	31542	7.99	**29500**	**6.05**
0.2	**105713**	**79.00**	NS	∞	34852	54.71	**28447**	**7.59**	34852	54.51	**28447**	**7.59**	28094	48.53	**26795**	**1.89**
0.3	**454864**	**94.22**	NS	∞	36693	56.04	**27465**	**5.19**	36693	56.04	**27465**	**5.19**	28156	49.22	**26817**	**2.91**
0.4	**1305130**	**97.95**	NS	∞	1305130	97.95	**27583**	**6.06**	31476	53.49	**26443**	**6.06**	31476	52.45	**26443**	**6.06**
0.5	**1305130**	**97.95**	NS	∞	33307	54.95	**26559**	**3.38**	29466	51.90	**26222**	**3.38**	27359	49.96	**26222**	**3.38**
0.6	**1305130**	**97.95**	NS	∞	1305130	97.95	**26161**	**2.24**	29573	51.99	**26161**	**2.24**	28251	50.12	**26161**	**2.24**
0.7	1294276	97.93	**25428**	**Opt**	34144	55.56	*	*	28249	50.83	*	*	27547	49.66	*	*
$\lvert\Omega\rvert = 50$																
0.05	**624097**	**95.38**	NS	∞	624097	95.38	**46528**	**34.93**	624097	95.38	**39516**	**23.38**	624097	95.38	**37233**	**18.69**
0.1	**120191**	**80.67**	NS	∞	58143	66.57	**42632**	**33.11**	58143	66.57	**38908**	**26.70**	58143	66.57	**34097**	**16.32**
0.2	**120593**	**81.18**	NS	∞	64473	69.57	**38985**	**29.59**	64473	69.57	**32360**	**15.18**	64473	69.57	**32321**	**15.0**
0.3	**134770**	**79.57**	NS	∞	43145	60.84	**37180**	**27.79**	43145	60.84	**31937**	**15.88**	43145	60.84	**31385**	**14.41**
0.4	NS	∞	NS	∞	51052	64.92	**34598**	**24.32**	51052	64.92	**27852**	**5.99**	51052	64.92	**27457**	**4.64**
0.5	NS	∞	NS	∞	117436	81.13	**31332**	**17.44**	117436	81.13	**26611**	**2.79**	46840	63.17	**26596**	**2.74**
0.6	**623777**	**95.81**	NS	∞	98384	78.27	**29265**	**12.30**	98384	78.27	**26488**	**3.11**	30317	52.61	**26465**	**3.03**
0.7	**623725**	**95.81**	NS	∞	43827	61.06	**27070**	**5.54**	43827	61.06	**26053**	**1.86**	28103	50.72	**26053**	**1.86**

NS=No Solution; $\overline{h}$ = upper bound; g=relative gap; Opt = Optimal; (*)= omitted since optimality already proved

Table 3.3: Performance summary on CAP using enumeration-based configuration generation (Algorithm 3): part 2

	$\leq 600s$				$\leq 3600s$				$\leq 7200s$				$\leq 21600s$			
	CPLEX		SHCG		CPLEX		SHCG		CPLEX		SHCG		CPLEX		SHCG	
ϵ	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g
$\lvert\Omega\rvert = 100$																
0.05	NS	∞	NS	∞	NS	∞	**50909**	**40.53**	NS	∞	**50909**	**40.53**	NS	∞	**50909**	**40.53**
0.1	NS	∞	NS	∞	NS	∞	**46149**	**37.92**	NS	∞	**46149**	**37.92**	724601	96.27	**46149**	**37.92**
0.2	NS	∞	NS	∞	NS	∞	**39933**	**30.70**	NS	∞	**39933**	**30.70**	137734	83.21	**39933**	**30.70**
0.3	NS	∞	NS	∞	NS	∞	**37260**	**28.02**	NS	∞	**37260**	**28.02**	50891	64.73	**36023**	**25.55**
0.4	NS	∞	NS	∞	NS	∞	**35125**	**25.17**	NS	∞	**35125**	**25.17**	109957	80.00	**34205**	**23.15**
0.5	NS	∞	NS	∞	NS	∞	**32631**	**20.47**	NS	∞	**32631**	**20.47**	724548	96.37	**29591**	**12.3**
0.6	NS	∞	NS	∞	754248	96.37	**30639**	**16.19**	754248	96.37	**30639**	**16.19**	754248	96.37	**28916**	**11.19**
0.7	NS	∞	NS	∞	754523	96.37	**27923**	**8.17**	754523	96.37	**27923**	**8.17**	754523	96.37	**26101**	**1.77**
$\lvert\Omega\rvert = 200$																
0.05	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**52943**	**41.78**	NS	∞	**52943**	**41.78**
0.1	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**47666**	**39.90**	NS	∞	**47666**	**39.90**
0.2	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**39814**	**31.45**	NS	∞	**39814**	**31.45**
0.3	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**36104**	**26.31**	NS	∞	**36104**	**26.31**
0.4	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**33658**	**22.13**	NS	∞	**33658**	**22.13**
0.5	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**31331**	**17.26**	NS	∞	**31331**	**17.26**
0.6	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**28171**	**8.85**	NS	∞	**28171**	**8.85**
0.7	NS	∞	NS	∞	NS	∞	NS	∞	NS	∞	**26720**	**4.46**	726555	96.38	**26720**	**4.46**

NS=No Solution; $\overline{h}$ = upper bound; g=relative gap; Opt = Optimal; (*)= omitted since optimality already proved

observation and introduce a more incremental approach to generating configurations (Algorithm 4).

3.7.4 Comparisons of Configuration Generation Approaches on the CAP

In Table 3.4 and 3.5, the two proposed approaches for generating configurations (i.e., Algorithm 3 and Algorithm 4) are evaluated. This table's layout is similar to that of Table 3.2 and 3.3.

Based on these results, it is clear that Algorithm 4 outperforms Algorithm 3 if $|\Omega| \geq 50$. This behavior is largely a product of Algorithm 4's ability to focus on promising scenario subsets and use those results to tighten bounds and eliminate the need to generate some of the configurations that are generated by Algorithm 3.

3.8 Conclusions and Future Research

This chapter considers JCC-FSFIR problems where uncertainty is modeled through a finite support set and risk is managed via a chance constraint that provides flexibility in the feasibility of the scenarios in the support set. A key property of the JCC-FSFIR problems considered herein is that they feature discrete decision variables in both the anticipative and recourse stages, which precludes the application of many established algorithms Luedtke [2014], Ahmed et al. [2016]. To address these types of JCC-FSFIR problems, we develops a scenario-based heuristic configuration generation algorithm (SHCG) which supports discrete variables. The convergence of the SHCG algorithm to the global optimum, given sufficient time, is ensured by leveraging relaxations and upper bounds of the generated configurations. An experimental evaluation of SHCG demonstrates how the algorithm outperforms CPLEX 12.7, a state-of-the-art commercial solver, on two problems from the literature.

The SHCG algorithm represents a significant first step in increasing the size and complexity of JCC-FSFIR problems that can be solved. However, there are a number of interesting points to explore in future research. For example, the upper bound procedure could be improved by extending Lemma 8 to consider on multiple scenarios at a time, or the results of Ahmed et al. [2016] could be leveraged to improve the runtime performance of the SHCG subproblems. It would also be worthwhile to consider a wider range of heuristics for generating high-quality configurations, or to develop specialized, problem-specific configuration generation schemes for the SHCG. Overall, the

Table 3.4: Performance summary on CAP when using different configuration generation algorithms: part 1

	$\leq 600s$				$\leq 3600s$				$\leq 7200s$				$\leq 21600s$			
	Algo 3		Algo 4		Algo 3		Algo 4		Algo 3		Algo 4		Algo 3		Algo 4	
ϵ	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g
$\lvert\Omega\rvert = 20$																
0.05	NS	∞	NS	∞	31952	13.17	**31128**	**10.86**	31952	13.17	**31128**	**10.86**	29087	Opt	28768	Opt
0.1*	NS	∞	NS	∞	**31906**	**13.13**	31994	13.37	**31906**	**13.13**	31994	13.37	**29500**	**6.05**	31994	13.37
0.2	NS	∞	NS	∞	**28447**	**7.59**	**28311**	**7.14**	28447	7.59	**28311**	**7.14**	**26795**	**1.89**	27102	3.00
0.3	NS	∞	NS	∞	**27465**	**5.19**	27600	5.87	**27465**	**5.19**	27600	5.87	**26817**	2.91	26962	3.43
0.4*	NS	∞	NS	∞	**27583**	**6.06**	27583	6.06	**26443**	**2.01**	27583	6.06	**26443**	**2.01**	27583	6.06
0.5*	NS	∞	NS	∞	**26559**	**3.38**	26528	3.26	**26222**	**2.14**	26528	3.26	**26222**	**2.14**	26528	3.26
0.6*	NS	∞	NS	∞	**26161**	**2.24**	26166	2.26	**26161**	**2.24**	26166	2.26	**26161**	**2.24**	26166	2.26
0.7	**25428**	**Opt**	**25428**	**Opt**	*	*	*	*	*	*	*	*	*	*	*	*
$\lvert\Omega\rvert = 50$																
0.05	NS	∞	NS	∞	46528	34.93	**39869**	**24.06**	39516	23.38	**38911**	**22.19**	37233	18.69	**37158**	**18.52**
0.1	NS	∞	NS	∞	42632	33.11	**35803**	**20.34**	38908	26.70	**34659**	**17.71**	34097	16.32	**33209**	**14.12**
0.2	NS	∞	NS	∞	38985	29.59	**32814**	**16.35**	32360	15.18	**32143**	**14.60**	32321	15.0	**31618**	**13.18**
0.3	NS	∞	NS	∞	37180	27.79	**31378**	**14.79**	31937	15.88	**30329**	**11.84**	31385	14.41	**29840**	**10.40**
0.4	NS	∞	NS	∞	34598	24.32	**28049**	**6.80**	27852	5.99	**27384**	**4.38**	27457	4.64	**27262**	**3.95**
0.5	NS	∞	NS	∞	31332	17.44	**27062**	**3.83**	**26611**	**2.79**	26920	3.81	26596	2.74	**26595**	**2.64**
0.6	NS	∞	NS	∞	29265	12.30	**26515**	**3.20**	26488	3.11	**26394**	**2.76**	26465	3.03	**26394**	**2.76**
0.7	NS	∞	NS	∞	27070	5.54	**26045**	**1.83**	26053	1.86	**25992**	**1.63**	26053	1.86	**25992**	**1.63**

NS=No Solution; $\overline{h}$ = upper bound; g=relative gap; Opt = Optimal; (*)= omitted since optimality already proved

Table 3.5: Performance summary on CAP when using different configuration generation algorithms: part 2

	$\leq 600s$				$\leq 3600s$				$\leq 7200s$				$\leq 21600s$			
	Algo 3		Algo 4		Algo 3		Algo 4		Algo 3		Algo 4		Algo 3		Algo 4	
ϵ	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g	$\overline{h}$	g
$\lvert\Omega\rvert = 100$																
0.05	NS	∞	NS	∞	50909	40.53	**46703**	**35.18**	50909	40.53	**43379**	**30.21**	50909	40.53	**41989**	**27.89**
0.1	NS	∞	NS	∞	46149	37.92	**42651**	**32.83**	46149	37.92	**40333**	**28.97**	46149	37.92	**38660**	**25.90**
0.2	NS	∞	NS	∞	39933	30.70	**38236**	**27.51**	39933	30.70	**35897**	**22.79**	39933	30.70	**35387**	**21.68**
0.3	NS	∞	NS	∞	37260	28.02	**35742**	**28.02**	37260	28.02	**33071**	**18.90**	36023	25.55	**32446**	**17.35**
0.4	NS	∞	NS	∞	35125	25.17	**32931**	**20.18**	35125	25.17	**30681**	**14.33**	34205	23.15	**29847**	**11.93**
0.5	NS	∞	NS	∞	32631	20.47	**29215**	**11.14**	32631	20.47	**27946**	**7.10**	29591	12.3	**27798**	**6.61**
0.6	NS	∞	NS	∞	30639	16.19	**27239**	**16.19**	30639	16.19	**26759**	**4.03**	28916	11.19	**26387**	**2.68**
0.7	NS	∞	NS	∞	27923	8.17	**26326**	2.61	27923	8.17	**26237**	2.28	**26101**	**1.77**	26129	1.87
$\lvert\Omega\rvert = 200$																
0.05	NS	∞	NS	∞	NS	∞	NS	∞	52943	41.78	**45332**	**32.11**	52943	41.78	**42350**	**27.33**
0.1	NS	∞	NS	∞	NS	∞	NS	∞	47666	39.90	**42307**	**32.38**	47666	39.90	**39515**	**27.50**
0.2	NS	∞	NS	∞	NS	∞	NS	∞	39814	31.45	**37709**	**27.63**	39814	31.45	**35093**	**22.23**
0.3	NS	∞	NS	∞	NS	∞	NS	∞	36104	26.31	**34448**	**22.76**	36104	26.31	**32714**	**18.67**
0.4	NS	∞	NS	∞	NS	∞	NS	∞	33658	22.13	**31512**	**15.67**	33658	22.13	**29759**	**11.92**
0.5	NS	∞	NS	∞	NS	∞	NS	∞	31331	17.26	**28020**	**7.68**	31331	17.26	**27208**	**4.71**
0.6	NS	∞	NS	∞	NS	∞	NS	∞	28171	8.85	**26800**	**4.18**	28171	8.85	**26403**	**2.74**
0.7	NS	∞	NS	∞	NS	∞	NS	∞	26720	4.46	**26253**	**2.76**	26720	4.46	**25985**	**1.76**

NS=No Solution; $\overline{h}$ = upper bound; g=relative gap; Opt = Optimal; (*)= omitted since optimality already proved

SHCG algorithm provides a flexible solution approach for JCC-FSFIR problems, and a number of other opportunities remain to leverage problem structure and improve the algorithm's performance.

Chapter 4

A Global Optimization Algorithm for Proving Rectangular AC Power Flow Infeasibility

4.1 Introduction

Based on our previous research, it is clear that any effective analysis of the climate adaptation problem requires careful consideration of electrical grid physics. When proposing the stochastic optimization model in Chapter 2, it was mentioned that capacity-based approximation techniques are commonly used in practice. For research purposes, more conservative approximations such as capacitated network flow or DC-approximation power flow (DCPF) are tested given their tractability for large size problems. In Chapter 2, we reviewed these approximations as to how each captures the electricity demand fulfillment physics for climate adaptation. The numerical experiment in Section 2.4.1.1 suggested the necessity to apply a more realistic power flow model for accurate cost estimation as well as to generate actionable plans. However, for computational tractability, a DC-approximation model was used for previous studies.

The use of DC-approximations is justified by its design and network operations (Coffrin and Van Hentenryck [2014]). The practice of embedding DC-approximations for tractable power flow problems in more complex problem systems is common, e.g. see works by Stott et al. [2009],

Gangammanavar et al. [2016], and Papavasiliou et al. [2011]. However, this method draws criticisms for its potential to produce unreliable solutions that violate true power flow physics during dispatching operations. DC-approximation omits imaginary power flow by assuming relatively small phase angle differences between connected buses, which changes the focus from a complex voltage representation to a linear, real voltage component. As suggested by Stott et al. [2009], the accuracy of DC-approximation varies enormously given different system profiles as well as different system conditions, especially when the system is under contingency. Hence, to better understand system operating conditions under scenarios such as hurricanes, it is not wise to consider DC-approximation alone without further AC-based analyses. In the context of this dissertation, developing adaptations using DC-approximations can underestimate costs and overestimate power system feasibility in delivering electricity under contingencies, which could result in unexpected load shedding and additional investments in practice.

Ever since it was first proposed by Carpentier [1962], the AC optimal power flow (ACOPF) problem has been one of the most interesting optimization problems in power systems research. The goal of ACOPF is to seek the optimal operating point of a transmission system under both power flow and physical network constraints. Local solution approaches for ACOPF have been extensively investigated and shown to be useful for large-scale systems (see the survey by Castillo and ONeill [2013] for more details). More recent research focuses on addressing the global optimal solution (see Molzahn et al. [2014] and Gopalakrishnan et al. [2012] for examples). Past studies of ACOPF have a clear focus on global optimality conditions. However, this focus may not be relevant when the system is under the severe impacts of exogenous uncertainty since the system operator's primary concern is to maintain feasible operations rather than efficient operations. In this Chapter, we look into the focus on AC power flow (ACPF) system feasibility (i.e., the objective function in the ACOPF problem is omitted). Note that ACPF feasibility is a prerequisite problem to ACOPF (i.e., if there no solution exists for ACPF then no solutions exist for the ACOPF problem). Realistically, ACPF serves as the "last-line" problem for system operations. Under specific extreme conditions, where there exists only one operations solution, ACPF is equivalent to ACOPF. Power systems applications can reach different conclusions when applying power flow solutions as the solutions profiles of both ACPF and ACOPF can vary drastically.

Although it is desirable to incorporate ACPF in the CAP, it is not easy to solve the ACPF problem due to its nonlinear, nonconvex solution space induced by complex power laws. Proving of

linear DC-approximation feasibility can be easily accomplished with Farkas's lemma (see CPLEX manual Manual [1987] for more details). For both ACPF and ACOPF, each problem is often evaluated using a local solver (e.g. Ipopt by Wallace and Fleten [2003] and KNITRO by Byrd et al. [2006]) that attempts to find a local stationary point. There also exist heuristics techniques for general nonlinear, nonconvex programs, such as feasibility pump (Fischetti et al. [2005]) and particle swarm optimization (AlRashidi and El-Hawary [2009]), which can effectively search feasible solutions. In practice, we observe local solution approaches can effectively handle standard test transmission systems, as tested in Coffrin et al. [2014] and Coffrin et al. [2017a], without requiring any additional assumptions. However, this is not the case when the system is further constrained by contingency events such as congestion, topology change, or other physical restrictions locally. Given the potentially conclusive in nature of local approaches, the ACPF can be solved with no feasible solution retrieved. In fact, for nonconvex problems, Horst and Tuy [2013] suggest that a conclusive proof of general nonlinear program feasibility requires a globally exhaustive search of solution space.

In cases when a local stationary point is infeasible, it is unclear whether the problem is truly infeasible or not, and the ACPF problem is not solved. The counterpart of solving the ACPF problem is to prove the problem is infeasible. To do so, one can apply infeasibility diagnostics tools which are useful optimization techniques based on domain reduction and conflict analysis that provide a conclusion on the infeasible subsystem. Such studies are quite mature for mixed-integer programs and are often addressed for general optimization programs in constraint programming (Puranik and Sahinidis [2017a] and Guieu and Chinneck [1999]). Existing state-of-the-art infeasibility diagnostics tools are implemented as pre-solvers in global solver implementations (see Optimization [2014], Puranik and Sahinidis [2017b], and Tawarmalani and Sahinidis [2005] for more details). Although these tools have been validated to be effective in use, unfortunately they have no guarantee of proving infeasibility, especially for challenging ACPF problem instances.

Theoretical work conducted by Lehmann et al. [2016] and Bienstock and Verma [2015] prove that the ACPF problem is strongly NP-hard. This motivates additional focus on algorithmic development for solving the ACPF problem. Global optimization tools (e.g., BARON by Tawarmalani and Sahinidis [2005], COUENNE by Belotti et al. [2009]) provide a viable choice for proving infeasibility using spatial branch and bound (SB&B) until all branches are proven to be infeasible. A common criticism of SB&B is its relaxation applied at each branch can be weak, resulting in computational inefficiency during searches of the SB&B tree with extensive breadth and width. Work by

Gopalakrishnan et al. [2012] tried to address such issues by applying tighter semi-definite relaxations with no assumption of system restrictions or topology changes. A more recent work by Molzahn [2017] tries to compute the feasible region globally using and SB&B-based algorithm through fully discretized domains. In their paper, it was observed that it can take > 10 days to fully exploit the feasible region for a 9-bus system. In cases when system feasibility is non-trivial to assess for local solvers, proving system infeasibility can require an exhaustive exploration of the variable domain.

In recent years, convex relaxation (CR) of the ACPF problem has attracted more attention to provide optimal information in large-scale systems. The relaxation formulations varies over a wide range of optimization techniques, such as second-order cone relaxation (Jabr [2006]), quadratic convex relaxation (Coffrin et al. [2016]), semi-definite programming (Bai et al. [2008]), and moment-based relaxation (Molzahn and Hiskens [2014] and Molzahn [2017]). As indicated in Coffrin et al. [2016], Low [2014a], and Low [2014b], the strength of CRs can vary under different system profiles or operating conditions when it is used to prove ACOPF optimality conditions. In the meanwhile, the computational burden of solving these CRs is significantly less than that required by global methods given the recent progress of convex optimization solution tools (e.g. Gurobi by Optimization [2014], CPLEX by Manual [1987], KNITRO by Byrd et al. [2006], and Ipopt by Wächter and Biegler [2006]). As the strength of CRs on ACOPF has been validated, little research focus has been given to the use of CRs for ACPF problems.

A recent global optimization study by Nagarajan et al. [2016] proposes the adaptive multivariate partitioning (AMP) algorithm that focuses on constructing converging relaxation models using adaptive partitioning schemes. Unlike traditional SB&B algorithms, this algorithm is based on the idea of limited discrepancy search (LDS) by Harvey and Ginsberg [1995]. It first creates multiple SB&B leaves without revealing the entire tree using a heuristic partitioning scheme on variable domains. Then, it constructs a piece-wise McCormick-based relaxation (PMR) over the partitioned domains and selects the least relaxation as the best lower bound through state-of-the-art mixed-integer linear program solvers. The proposed algorithm also considers sequential optimality-based bound-tightening (OBBT) scheme as a pre-solver to improve global optimization computational performance. The OBBT applied can obtain the tightest bound possible when given a specific relaxation. This method is also considered by Narimani et al. [2018]. The global algorithm ensures global convergence as the heuristic applied can eventually achieve full domain discretization. Although the method can be very complex in the worst case, the numerical performance observed is more efficient

when compared to some state-of-the-art global solvers, namely BARON and COUENNE.

The AMP method is designed to address the optimality of general, mixed-integer nonlinear programs. The idea of combining LDS, PMR, OBBT, and adaptive partitioning motivates new research questions under the context of infeasibility proof. In contrast to the optimality problem, any infeasibility proof requires the separation between the system constrained space and relaxed region of model nonconvexity. These facts lead us to three research questions of interests. First, as previous studies clearly indicate that the effort required for obtaining global optimality can be lessened by incorporating some generally designated heuristic methods, is there a better algorithmic design for the purpose of effectively conducting an infeasibility proof? Second, if there exists such an algorithm, can infeasibility proof convergence be guaranteed or will the algorithm eventually find a non-trivial feasible solution to prove the problem infeasible? Finally, what insights from analyzing the ACPF problem can gleamed for general nonlinear, nonconvex programs infeasibility proofs?

In the light of these questions, we propose an iterative infeasibility proof algorithm based on a tight piece-wise relaxation for the ACPF problem. Our proposed algorithm flexibly adapts the algorithmic structure developed in AMP and creatively strengthens its approach based on observations of a numerical pattern in piece-wise ACPF relaxations. Furthermore, we theoretically demonstrate that our algorithm is applicable for proving general nonconvex nonlinear programs globally (i.e., it is guaranteed to find a feasible solution or to prove a problem infeasible). In the algorithm presentation, we show observed patterns of the numerical information about ACPF that help shaped the algorithm. We demonstrate the efficiency of our approach by comparing it to both state-of-the-art solvers and AMP. Lastly, we use experimental algorithm results to provide insights for both general nonlinear, nonconvex program infeasibility proofs and ACOPF problems.

The rest of this chapter is organized as follows. In Section 4.2, we review the rectangular ACPF formulation and define the problem formulation of interest. Then, in Section 4.3, we conduct some observatory experiments with congested transmission systems to demonstrate the non-triviality of detecting nonconvex ACPF infeasibility. To address these challenges, we first propose a partition-based, piece-wise convex relaxation for the nonconvex ACPF problem in Section 4.4. After considering the proposed relaxation and other inherent properties of the ACPF problem, we construct a global algorithm in Section 4.5 for proving the infeasibility of the ACPF problem. In Section 4.6, we compare our proposed algorithm with SB&B-based global solvers and the AMP algorithm through numerical experiments. The result of these experiments demonstrates our method's

computationally superiority. Finally, we conclude this study and provide insights for future research in Section 4.7.

4.2 Review of the Rectangular ACPF Formulation

In this section, we review the rectangular ACPF formulation based on the work of Carpentier [1962]. We first list the notation required for the formulation. General ACPF principles are introduced with complex numbers, and then we present the rectangular ACPF, which is a real number formulation. For readability, parameters are always shown in boldface font to differentiate them other notation.

4.2.1 Notation

	Sets
$\mathcal{N}$	Set of buses
$\mathcal{L}$	Set of directed transmission lines
$\mathcal{G}_i$	Set of generators located at bus $i \in \mathcal{N}$
	Variables
v_i^p, v_i^q	real and imaginary voltage at bus $i \in \mathcal{N}$
g_k^p, g_k^q	real and imaginary power generated by generator $k \in \mathcal{G}_i$ at bus $i \in \mathcal{N}$
f_{ij}^p, f_{ij}^q	active and reactive power flow on line $(i,j) \in \mathcal{N}$
	Parameters
$\boldsymbol{S_i^d}$	complex power demand at bus $i \in \mathcal{N}$
$\boldsymbol{Y_{ij}}$	complex admittance on line $(i,j) \in \mathcal{L}$
$\boldsymbol{d_i^p}, \boldsymbol{d_i^q}$	real and imaginary power angle on line $(i,j) \in \mathcal{L}$
$\boldsymbol{g_{ij}^\varepsilon}, \boldsymbol{b_{ij}^\varepsilon}$	conductance and susceptance on line $(i,j) \in \mathcal{L}$
$\boldsymbol{g_i^s}, \boldsymbol{b_i^s}$	shunt conductance and susceptance at bus $i \in \mathcal{N}$
$\boldsymbol{b_{ij}^C}$	charging susceptance on line $(i,j) \in \mathcal{L}$
$\underline{\boldsymbol{\phi}}_{\boldsymbol{ij}}, \overline{\boldsymbol{\phi}}_{\boldsymbol{ij}}$	lower and upper bounds of angle differences on line $(i,j) \in \mathcal{L}$
$\underline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{p}}, \overline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{p}}$	lower and upper bounds of real power generated by generator k at bus $i \in \mathcal{N}$
$\underline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{q}}, \overline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{q}}$	lower and upper bounds of imaginary power generated by generator k at bus $i \in \mathcal{N}$
$\underline{\boldsymbol{v}}^{\boldsymbol{i}}, \overline{\boldsymbol{v}}^{\boldsymbol{i}}$	lower and upper bounds of voltage magnitude at bus $i \in \mathcal{N}$

$\bar{\boldsymbol{l}}_{\boldsymbol{ij}}$	thermal limits on line $(i,j) \in \mathcal{L}$

4.2.2 Power Flow Laws

In its most general form, ACPF is defined within a transmission network, under the constraints of Kirchhoff's Current Law, Ohm's Law, and necessary physical and security requirements. A transmission network can be seen as a graph $G = (\mathcal{N}, \mathcal{L})$ comprised of buses, lines, generators, and demands. We first define the nodes $\mathcal{N}$ as a set of buses and the arcs $\mathcal{L}$ as be a set of directed transmission lines. A bus serves as the electricity bridge that connects flowing power with local generations and demands. At each bus $i \in \mathcal{N}$, there exists a set of generators $\mathcal{G}_i$, where each generator $k \in \mathcal{G}_i$ injects complex power S_k into the network. On the demand side, we use $\boldsymbol{S_i^d}$ to denote the complex power demand at bus $i \in \mathcal{N}$. To transmit power flow, each bus $i \in \mathcal{N}$ carries a variable complex voltage V_i that serves as the force to push power flowing. Transmission lines serve to transmit power flow from one bus to another under the guidance of complex admittance $\boldsymbol{Y_{ij}}$. For each directed line $(i,j) \in \mathcal{L}$, we denote the variable complex power flow F_{ij}. The general power flow law is formulated in (4.1) [1]:

$$F_{ij} = \boldsymbol{Y_{ij}^*} V_i V_i^* + \boldsymbol{Y_{ij}^*} V_i V_j^* \quad \forall (i,j) \in \mathcal{L} \tag{4.1a}$$

$$S_i^g - \boldsymbol{S}_i^d = \sum_{j:(j,i)\in\mathcal{L}} F_{ij} - \sum_{j:(i,j)\in\mathcal{L}} F_{ij} \quad \forall i \in \mathcal{N} \tag{4.1b}$$

4.2.3 Rectangular ACPF Formulation

Formulation (4.1) contains a series of complex numbers: complex voltage V, power generation S^g, power flow F, and demand $\boldsymbol{S^d}$. The real number interpretation of a complex number c can be of two forms: rectangular form $c = a + i\theta$ or polar form $c = |c|\angle\theta$. Comparison studies of different real-number formulations can be found in Cain et al. [2012] and Park et al. [2017], where each paper has a different perspective in addressing the advantages and disadvantages of different formulation methods. It remains unclear as to which formulation is better for ACPF problems given system, network profile, etc. Now, we consider a naive rectangular ACPF (4.2). We note an important future

[1] Superscript $*$ used in (4.1) to represent the conjugate of complex numbers.

direction of this study is to compare different formulations and understand how each formulation impacts ACPF problem tractability.

The rectangular representation separates complex numbers into two components: real (active) components and imaginary (or reactive) components. For each bus $i \in \mathcal{N}$, voltage variables $V_i = v_i + i\theta_i$, where v_i is the voltage magnitude and θ_i is voltage angle, are modeled with real voltage components $v_i^p \in \mathbb{R}$ and imaginary voltage components $v_i^q \in \mathbb{R}$ through $\theta_i = \arctan^{-1}(\frac{v_i^q}{v_i^p})$. Similarly, deterministic demand $\boldsymbol{S_i^d}$ is separated into real power demand $\boldsymbol{d_i^p} \in \mathbb{R}^+$ and imaginary power demand $\boldsymbol{d_i^q} \in \mathbb{R}^+$. For each generator $k \in \mathcal{G}_i$ at bus $i \in \mathcal{N}$, the power generated S_k^g incorporates both a real component $g_k^p \in \mathbb{R}^+$ and an imaginary component $g_k^q \in \mathbb{R}^+$. For each line $(i,j) \in \mathcal{L}$, real power flow $f_{ij}^p \in \mathbb{R}$ and imaginary power flow $f_{ij}^q \in \mathbb{R}$ are the two components that represent F_{ij}. Note that power at both line ends can be different due to the physical resistance of transmission lines. This creates asymmetric power transmission, i.e., for any line $(i,j) \in \mathcal{L}$, the values of f_{ij}^p, f_{ij}^q are not necessarily the same as the values of $-f_{ji}^p, -f_{ji}^q$, respectively. As the rectangular ACPF formulation is a notation-heavy formulation with complicated parameters calculations, we refer the reader to Section 4.2.1 for reference notation. The rectangular ACPF formulation of interest is presented in (4.2):

$$\forall (i,j) \in \mathcal{L}$$

$$\begin{aligned} f_{ij}^p =& \frac{1}{\boldsymbol{\tau_{ij}^2}} \boldsymbol{g_{ij}^\varepsilon}((v_i^p)^2 + (v_i^q)^2) - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{g_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{b_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_i^p v_j^p + v_i^q v_j^q) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{b_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{g_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_j^p v_i^q - v_i^p v_j^q) \end{aligned} \tag{4.2a}$$

$$\begin{aligned} f_{ji}^p =& \boldsymbol{g_{ij}^\varepsilon}((v_j^p)^2 + (v_j^q)^2) - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{g_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{b_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_j^p v_i^p + v_j^q v_i^q) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{b_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{g_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_i^p v_j^q - v_j^p v_i^q) \end{aligned} \tag{4.2b}$$

$$\begin{aligned} f_{ij}^q =& -\frac{1}{\boldsymbol{\tau_{ij}^2}}(\boldsymbol{b_{ij}^\varepsilon} + \frac{\boldsymbol{b_{ij}^C}}{2})((v_i^p)^2 + (v_i^q)^2) - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{g_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{b_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_j^p v_i^q - v_i^q v_j^p) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{b_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{g_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_i^p v_j^p - v_i^q v_j^q) \end{aligned} \tag{4.2c}$$

$$\begin{aligned} f_{ji}^q =& -(\boldsymbol{b_{ij}^\varepsilon} + \frac{\boldsymbol{b_{ij}^C}}{2})((v_i^p)^2 + (v_i^q)^2) - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{g_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{b_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_i^p v_j^q - v_j^p v_i^q) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}}(\boldsymbol{b_{ij}^\varepsilon}\cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{g_{ij}^\varepsilon}\sin(\boldsymbol{\phi_{ij}}))(v_j^p v_i^p + v_j^q v_i^q) \end{aligned} \tag{4.2d}$$

$$v_i^q v_j^p - v_i^p v_j^q \geq \tan(\underline{\boldsymbol{\phi}}_{\boldsymbol{ij}})(v_i^p v_j^p + v_i^q v_j^p) \tag{4.2e}$$

$$v_i^q v_j^p - v_i^p v_j^q \leq \tan(\overline{\boldsymbol{\phi}}_{\boldsymbol{ij}})(v_i^p v_j^p + v_i^q v_j^p) \tag{4.2f}$$

$$(f_{ij}^p)^2 + (f_{ij}^q)^2 \leq \bar{\boldsymbol{l}}_{\boldsymbol{ij}}^{\boldsymbol{2}} \tag{4.2g}$$

$\forall i \in \mathcal{N}$

$$\sum_{k \in \mathcal{G}_i} g_k^p - \sum_{j:(i,j)\in\mathcal{L}} f_{ij}^p + \sum_{j:(j,i)\in\mathcal{L}} f_{ji}^p - \boldsymbol{d}_{\boldsymbol{i}}^{\boldsymbol{p}} = \boldsymbol{g}_{\boldsymbol{i}}^{\boldsymbol{s}}((v_i^p)^2 + (v_i^q)^2) \tag{4.2h}$$

$$\sum_{k \in \mathcal{G}_i} g_k^q - \sum_{j:(i,j)\in\mathcal{L}} f_{ij}^q + \sum_{j:(j,i)\in\mathcal{L}} f_{ji}^q - \boldsymbol{d}_{\boldsymbol{i}}^{\boldsymbol{p}} = -\boldsymbol{b}_{\boldsymbol{i}}^{\boldsymbol{s}}((v_i^p)^2 + (v_i^q)^2) \tag{4.2i}$$

$$(v_i^p)^2 + (v_i^q)^2 \geq \underline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{2}} \tag{4.2j}$$

$$(v_i^p)^2 + (v_i^q)^2 \leq \overline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{2}} \tag{4.2k}$$

$$\underline{\boldsymbol{v}}_{\boldsymbol{i}} \leq v_i^p \leq \overline{\boldsymbol{v}}_{\boldsymbol{i}} \tag{4.2l}$$

$$\underline{\boldsymbol{v}}_{\boldsymbol{i}} \leq v_i^q \leq \overline{\boldsymbol{v}}_{\boldsymbol{i}} \tag{4.2m}$$

$$\underline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{p}} \leq g_k^p \leq \overline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{p}} \quad \forall k \in \mathcal{G}_i \tag{4.2n}$$

$$\underline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{q}} \leq g_k^q \leq \overline{\boldsymbol{g}}_{\boldsymbol{k}}^{\boldsymbol{q}} \quad \forall k \in \mathcal{G}_i \tag{4.2o}$$

Constraints (4.2a)-(4.2d) are real-number representations for (4.1a), where real power flow f_{ij}^p, f_{ji}^p and imaginary power flow f_{ij}^q, f_{ji}^q are calculated based on voltage variables $v_i^p, v_j^p, v_i^q, v_j^q$, and line admittance parameters. The angle difference limitation between connected buses is enforced with constraints (4.2e) and (4.2f). Note that, in practice, angle limitations are often set to be less than 10 degrees (Purchala et al. [2005]), which is a relatively small magnitude. Thermal limits of transmission lines are critical physical restrictions for security concerns, which is measured using the current on line (i, j). Constraints (4.2g) regulate the thermal limits on line (i, j) by measuring $|F_{ij}| = \sqrt{(f_{ij}^p)^2 + (f_{ij}^q)^2}$. At each bus, the network requires flow balance to be maintained (4.1b). This is enforced for both real and imaginary power components in constraints (4.2h) and (4.2i), respectively. Bus voltage limitations are enforced in constraints (4.2j) and (4.2k) by measuring the magnitude of the complex voltage through $|V_i| = \sqrt{(v_i^p)^2 + (v_i^q)^2}$. In constraints (4.2l) and (4.2m), tighter bounds are enforced on voltage components given practical operating conditions. Finally, generator capacities are regulated in constraints (4.2n) and (4.2o).

4.3 Preliminary Experiment with ACPF

The formulation presented in (4.2) is a challenging optimization problem due to its nonlinear, nonconvex continuous variable products. As elaborated in Section 4.1, recent research on ACPF or ACOPF applies CRs. Optimization models relaxations serve many purposes, such as proving dual bounds, guiding SB&B tree search, and generating cutting-planes, to name a few. In this study, we focus on an infeasibility proof. CRs are becoming more popular in ACOPF studies as they provide tight formulations and reliable solutions for a variety of applications with computational efficiency. Theoretical studies and practiced applications provide comprehensive overview on the strengths and weaknesses of existing CRs relaxation approaches (Coffrin et al. [2016], Low [2014a], Low [2014b]).

It is important to note that most CRs are static optimization problems. As we focus on systems under exogenous influences, transmission network components can be disabled to create congestion and/or imbalance. When the system is influenced in this way, a narrower operating conditions can hinder the local optimization solver from finding a feasible solution, while a relaxation may still be feasible. In these circumstances, assuming a system is feasible while it is actually not can potentially bias decision processes by overestimating the network resilience and compromising security. On the other hand, assuming a system is infeasible while it is feasible introduce overestimation on capability in fulfilling power demand and result in unnecessary costs.

We conduct an experiment to show the existence of situations when a rectangular ACPF problem cannot be solved by either a local solver, or some CRs, namely McCormick-based relaxation, QC relaxation, and SOC relaxation. Sixteen test systems from NESTA cases (Coffrin et al. [2014]) are selected for this experiment. Each instance is gradually congested by decreasing the thermal limits $\boldsymbol{l_{ij}}$ for every line $(i, j) \in \mathcal{L}$.

The results are summarized in Figure 4.1. From left to right, the thermal limits capacity gradually increases as congestion decreases. In the green area, local solver Ipopt is able to obtain a feasible solution, while the grey region denotes areas of proven infeasibility by the tested relaxation approaches. The red region shows the congestion range for which the system's feasibility in undetermined. While for some instances there can be a relatively small red region, the necessity to address the remaining red regions is clear.

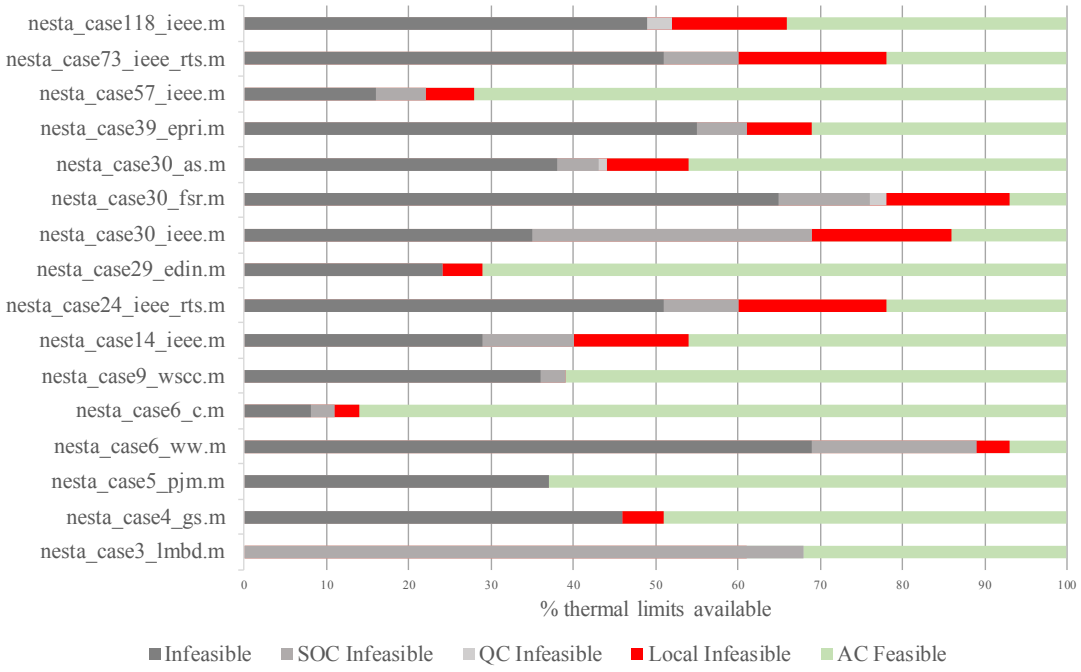


Figure 4.1: Initial experiment with congested transmission networks

4.4 Piece-wise Convex Relaxation for ACPF Formulation

In this section, we construct a piece-wise convex relaxation for the nonlinear, nonconvex ACPF (PCR-ACPF). This piece-wise convex relaxation (PCR) is constructed based on spatial discretization of variable domains, which means the tightness of the PCR-ACPF is a function of domain discretization. Our method is motivated by the work of Sundar et al. [2018], which provide theoretical analyses on tight PCR for general multilinear products. A brief review of theoretical developments on different PCR formulations is presented here, and then we propose some additional valid inequalities for the PCR-ACPF and show they are not necessarily redundant to its linear relaxation.

4.4.1 General Representation of ACPF and PCR-ACPF

For ease of readability, we abstract the ACPF and PCR-ACPF with general formulations. We first define a general ACPF in (4.3) using $\mathcal{F}$. Let x define all general decision variables. The notation $\boldsymbol{x^l}$ and $\boldsymbol{x^u}$ are variable lower and upper bound vectors, respectively.

$$\mathcal{F} : \{\min \ \mathbf{0} \mid g(x) \leqslant \mathbf{0}, \boldsymbol{x^l} \leqslant x \leqslant \boldsymbol{x^u}\} \tag{4.3}$$

In (4.3), $g(x)$ is a system of ACPF constraints (4.2).

To systematically construct our PCR-ACPF, we first apply a reformulation scheme proposed by Smith and Pantelides [1999] that symbolically lifts nonconvex terms in $g(x)$ with relaxed variables. Let $\hat{x} \in \mathbb{R}$ denote all lifted variables which are bounded by lower vector $\hat{\boldsymbol{x}}^{\boldsymbol{l}}$ and upper bound vectors $\hat{\boldsymbol{x}}^{\boldsymbol{u}}$. To constructed PCR, an additional continuous variable λ and binary variables z are required which will be later discussed in this section. The general form of PCR-ACPF $\mathcal{R}$ is defined in (4.4):

$$\mathcal{R} : \{\min\ \mathbf{0} \mid g'(x, \hat{x}) \leqslant \mathbf{0}, h(x, \hat{x}, \lambda, z) \leqslant \mathbf{0}, \boldsymbol{x}^{\boldsymbol{l}} \leqslant x \leqslant \boldsymbol{x}^{\boldsymbol{u}}, \hat{\boldsymbol{x}}^{\boldsymbol{l}} \leqslant \hat{x} \leqslant \hat{\boldsymbol{x}}^{\boldsymbol{u}}\} \tag{4.4}$$

In (4.4), $g'(x, \hat{x})$ is a system of reformulated linear constraints based on $g(x)$ with all nonconvex terms lifted as $\hat{x}$, and $h(x, \hat{x}, \lambda, z)$ is a system of linear constraints required for PCR.

4.4.2 PCR for General Bilinear Terms

We use subscripts of x to denote a specific variable, i.e., $x_k \in x$. Let $\boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{l}}$ and $\boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{u}}$ denote the lower and upper bound of variable x_k, respectively. A bilinear term $x_m x_n \to \mathbb{R}$, which is considered as a subset nonconvex terms, is generally defined as

$$x_m x_n, \text{ where } x_m, x_n \in x, \boldsymbol{x}_{\boldsymbol{m}}^{\boldsymbol{l}} \leq x_m \leq \boldsymbol{x}_{\boldsymbol{m}}^{\boldsymbol{u}}, \boldsymbol{x}_{\boldsymbol{n}}^{\boldsymbol{l}} \leq x_n \leq \boldsymbol{x}_{\boldsymbol{n}}^{\boldsymbol{u}} \tag{4.5}$$

To construct a PCR for a bilinear term, the domains of variables $x_m, x_n \in x$ are spatially discretized into $\boldsymbol{P}^{\boldsymbol{m}}$ and $\boldsymbol{P}^{\boldsymbol{n}}$ partitions, which are represented through sorted sets of partition points $\boldsymbol{B}^{\boldsymbol{m}} = \{\boldsymbol{b}_{\boldsymbol{i}}^{\boldsymbol{m}} \mid i = 0 \cdots \boldsymbol{P}^{\boldsymbol{m}}\}$ and $\boldsymbol{B}^{\boldsymbol{n}} = \{\boldsymbol{b}_{\boldsymbol{i}}^{\boldsymbol{n}} \mid i = 0 \cdots \boldsymbol{P}^{\boldsymbol{n}}\}$, respectively. For ease in denoting the discretized domains, we define two index sets. First, let index set $\mathcal{Q}^m := \{i \mid i = 0 \cdots \boldsymbol{P}^{\boldsymbol{m}}\}$ associate each partition point in $\boldsymbol{B}^{\boldsymbol{m}}$, i.e., $i \to \boldsymbol{b}_{\boldsymbol{i}}^{\boldsymbol{m}} \in \boldsymbol{B}^{\boldsymbol{m}}\ \forall i \in \mathcal{Q}^m$. Second, let index set $\mathcal{P}^m := \{i | i = 1 \cdots \boldsymbol{P}^{\boldsymbol{m}}\}$ be associated each partition, i.e., $i \to p_i^m = [\boldsymbol{b}_{\boldsymbol{i-1}}^{\boldsymbol{m}}, \boldsymbol{b}_{\boldsymbol{i}}^{\boldsymbol{m}}]\ \forall i \in \mathcal{P}^m$. It is obvious that $\mathcal{Q}^m = \mathcal{P}^m \cup \{0\}$. The purpose of discretization is to regulate a variable's value to a tighter sub-region as a "piece" of the PCR. Any partition p_i^m, where $i \in \mathcal{P}^m$, is considered to be active if $x_m \in p_i^m$. Clearly, there can exist one and only one active partition, i.e., $x_m \in \exists! p_i^m$, where $i \in \mathcal{P}^m$. Let binary variables $z_i^m \in \{0, 1\}$, where $i \in \mathcal{P}^m$ denotes whether the i-th partition of variable x_m is active or not. This notation can be applied for any variable $x_n \in x$, yielding the corresponding $\mathcal{Q}^n$, $\mathcal{P}^n$, z^n.

The discretized domains of the variables in bilinear terms forms a two-dimensional lattice

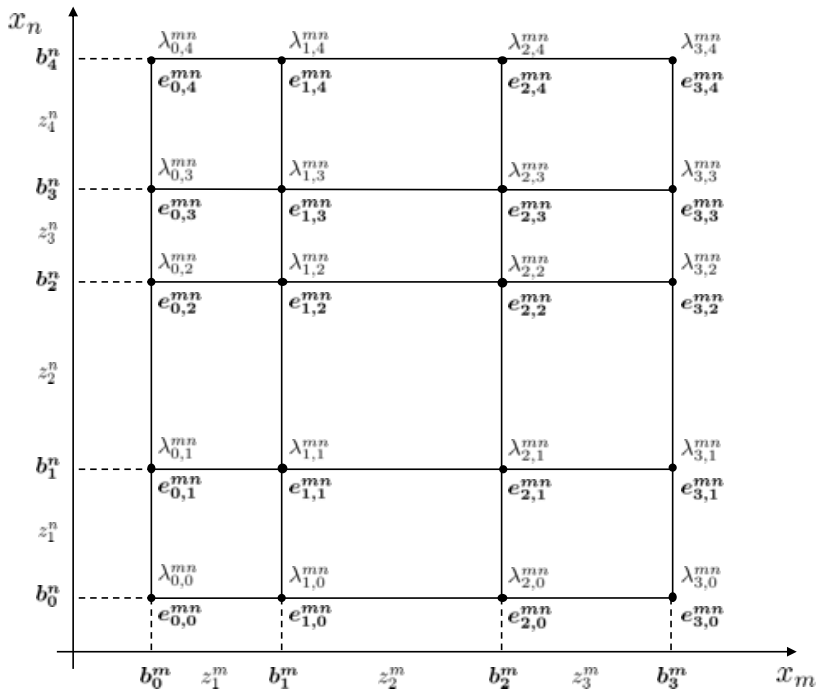


Figure 4.2: Example discretized domain and the resulting lattice for bilinear term $x_m x_n$, where there are three partitions for variable x_m and four partitions for variable n.

with crossing partitions. Each crossing is defined with two partitions, one from each variable. A crossing is also defined by four corner lattice points, where each point is associated with a locally extreme value of the bilinear product based on the associated partition points. We encode each lattice point through (i, j), where $i \in \mathcal{Q}^m$ and $j \in \mathcal{Q}^n$. For each lattice point (i, j), let $\boldsymbol{e_{ij}^{mn}} = \boldsymbol{b_i^m b_j^n}$ denote the corresponding extreme values. Let continuous variable $\lambda_{ij}^{mn} \in [0, 1]$, where $\lambda^{mn} \in \lambda$, $i \in \mathcal{Q}^m$ and $j \in \mathcal{Q}^n$, be the scalar variables associated with each lattice point (i, j) for term $x_m x_n$. Figure 4.2 provides an example of this notation system for PCR.

With all variables and parameters defined, we formulate $h(x, \hat{x}, \lambda, z)$ in (4.4) for bilinear term $x_m x_n$:

$$\widehat{x_m x_n} = \sum_{i \in \mathcal{Q}^m} \sum_{j \in \mathcal{Q}^n} \boldsymbol{e_{ij}^{mn}} \lambda_{ij}^{mn} \tag{4.6a}$$

$$\sum_{i \in \mathcal{Q}^m} \sum_{j \in \mathcal{Q}^n} \lambda_{ij}^{mn} = 1 \tag{4.6b}$$

$$\sum_{i \in \mathcal{P}^m} z_i^m = 1 \tag{4.6c}$$

$$\sum_{j \in \mathcal{P}^n} z_j^n = 1 \tag{4.6d}$$

$$\sum_{j \in \mathcal{Q}^n} \lambda_{0,j}^{mn} \leq z_i^m \tag{4.6e}$$

$$\sum_{j \in \mathcal{Q}^n} \lambda_{ij}^{mn} \leq z_{i-1}^m + z_i^m \quad \forall i \in \mathcal{Q}^m \setminus \{0, \boldsymbol{P^m}\} \tag{4.6f}$$

$$\sum_{j \in \mathcal{Q}^n} \lambda_{\boldsymbol{P^m},j}^{mn} \leq z_{\boldsymbol{P^m}}^m \tag{4.6g}$$

$$\sum_{i \in \mathcal{Q}^m} \lambda_{i,0}^{mn} \leq z_j^n \tag{4.6h}$$

$$\sum_{i \in \mathcal{Q}^m} \lambda_{ij}^{mn} \leq z_{j-1}^n + z_j^n \quad \forall i \in \mathcal{Q}^n \setminus \{0, \boldsymbol{P^n}\} \tag{4.6i}$$

$$\sum_{i \in \mathcal{Q}^m} \lambda_{i,\boldsymbol{P^n}}^{mn} \leq z_{\boldsymbol{P^n}}^n \tag{4.6j}$$

$$x_m = \sum_{i \in \mathcal{Q}^m} \boldsymbol{b_i^m} \sum_{j \in \mathcal{Q}^n} \lambda_{ij}^{mn} \tag{4.6k}$$

$$x_n = \sum_{j \in \mathcal{Q}^n} \boldsymbol{b_j^n} \sum_{i \in \mathcal{Q}^m} \lambda_{ij}^{mn} \tag{4.6l}$$

$$\lambda_{ij}^{mn} \geq 0 \quad \forall i \in \mathcal{Q}^m, j \in \mathcal{Q}^n \tag{4.6m}$$

$$z_i^m, z_j^n \in \{0, 1\} \quad \forall i \in \mathcal{P}^m, j \in \mathcal{P}^n \tag{4.6n}$$

First, a convex combination is constructed with all extreme points $\boldsymbol{e^{mn}}$ from the lattice and their associated scalar variables λ^{mn} to regulate lifted variable $\widehat{x_m x_n}$ through constraints (4.6a) and (4.6b). Constraints (4.6c) and (4.6d) controls the logic that value of variable x_m or x_n must fall into a partition. The purpose of PCR is to regulate a tighter region of lifted variable $\widehat{x_m x_n}$ when x_m, x_n fall in some active partitions. That means the convex combination must be constructed using only four extreme points associated with partition crossing. Constraints (4.6e)-(4.6j) are used for this purpose, where binary variables z^m and z^n are used to control the value of λ^{mn} under the form of special-order-set type-2 constraints. Furthermore, the value of x_m and x_n are regulated through a convex combination in constraints (4.6k) and (4.6l). Hence, variables z^m, z^n, and λ^{mn} link how lifted variable $\widehat{x_m x_n}$ reacts to the value of x_m and x_n. Finally, variable bounds and integral requirements are enforced in constraints (4.6m) and (4.6n).

4.4.2.1 Special Case for Quadratic Terms

Consider a special form of the defined bilinear term where $m = n$ which are referred to as quadratic terms. For quadratic terms, variable domain partitioning is necessary for a single dimension. Although formulation (4.6) can be applied directly a more compact formulation for quadratic term $x_m x_m$ is given in (4.7). Here, variable λ^{mm} is reduced to a single dimension. This formulation is based on (4.6) with additional convex, quadratic constraints (4.7b), and is readily solvable using state-of-the-art convex solvers:

$$\widehat{x_m^2} = \sum_{i \in \mathcal{Q}^m} \boldsymbol{b_i^m} \lambda_i^{mm} \tag{4.7a}$$

$$\widehat{x_m^2} >= x_m^2 \tag{4.7b}$$

$$\sum_{i \in \mathcal{Q}^m} \lambda_i^{mm} = 1 \tag{4.7c}$$

$$\lambda_0^{mm} \leq z_i^m \tag{4.7d}$$

$$\lambda_i^{mm} \leq z_{i-1}^m + z_i^m \quad \forall i \in \mathcal{Q}^m \setminus \{0, \boldsymbol{P^m}\} \tag{4.7e}$$

$$\lambda_{\boldsymbol{P^m}}^{mm} \leq z_{\boldsymbol{P^m}}^m \tag{4.7f}$$

$$\sum_{i \in \mathcal{P}^m} z_i^m = 1 \tag{4.7g}$$

$$x_m = \sum_{i \in \mathcal{Q}^m} \boldsymbol{b_i^m} \lambda_i \tag{4.7h}$$

$$\lambda_i^{mm} \geq 0 \quad \forall i \in \mathcal{Q}^m \tag{4.7i}$$

$$z_i^m \in \{0, 1\} \quad \forall i \in \mathcal{P}^m \tag{4.7j}$$

4.4.3 Comparing PCR and PMR

PCR for bilinear terms (4.6) is a special case of the method proposed by Sundar et al. [2018]. Theoretical analysis suggests that (4.6) is not locally ideal for $h(x, \hat{x}, \lambda, z)$, where ideal is defined as all extreme points of the mixed integer relaxation describe the convex hull of the mixed-integer formulation. An alternative to PCR, which we refer to as PCR-C (4.8), is locally ideal. The piece-wise McCormick-based relaxation (PMR) is also locally ideal. Note that this local property is designated for the subsystem $h(x, \hat{x}, \lambda, z)$. It is not proven for PCR, PCR-C, or PMR whether

known local properties still holds or not when system $g(x, \hat{x})$ is incorporated.

$$(4.6a), (4.6b), (4.6c), (4.6d), (4.6k), (4.6l), (4.6m), (4.6n)$$

$$\sum_{i=0}^{k} z_i^m \geq \sum_{i=0}^{k} \sum_{j \in \mathcal{Q}^n} \lambda_{ij} \quad \forall k \in \mathcal{Q}^m \setminus \{|\mathcal{Q}^m|\} \tag{4.8a}$$

$$\sum_{i=0}^{k} z_i^m \leq \sum_{i=0}^{k+1} \sum_{j \in \mathcal{Q}^n} \lambda_{ij} \quad \forall k \in \mathcal{Q}^m \setminus \{|\mathcal{Q}^m|\} \tag{4.8b}$$

$$\sum_{i=0}^{k} z_i^n \geq \sum_{j=0}^{k} \sum_{i \in \mathcal{Q}^m} \lambda_{ij} \quad \forall k \in \mathcal{Q}^n \setminus \{|\mathcal{Q}^n|\} \tag{4.8c}$$

$$\sum_{i=0}^{k} z_i^n \leq \sum_{j=0}^{k+1} \sum_{i \in \mathcal{Q}^m} \lambda_{ij} \quad \forall k \in \mathcal{Q}^n \setminus \{|\mathcal{Q}^n|\} \tag{4.8d}$$

To differentiate PCR, PCR-C, and PMR, we summarize some basic properties in Table 4.2. We list the dimensionality of system $h(x, \hat{x}, \lambda, z)$ in terms of "# of additional variables" and "# of additional constraints" required for constructing the relaxation. Furthermore, we show local property of each formulation in the row labeled "Local property." Note that a tighter formulation does not necessarily imply better numerical performance given the potential under-exploitation of its structural properties.

	PCR	PCR-C	PMR
# of additional variables	$1 + M + N + MN$	$1 + M + N + MN$	$1 + 2M + 2N + MN$
# of additional constraints	$8 + M + N$	$6 + M + N$	$6 + 4M + 4N + 3MN$
Local property	Locally sharp	Convex hull	Convex hull
Performance rank	1	2	3

Table 4.2: Comparison of PCR, PCR-C, and PMR on bilinear term $x_m x_n$, where x_m's domain has M partitions and x_n's domain has N partitions

We evaluate each relaxation method using test instances from Nagarajan et al. [2017] and rank general performance in row "Performance rank" where 1 =best (numerical tests results are provided in the APPENDIX A). Given these results we choose PCR for relaxing the ACPF problem in this study given its formulation compactness and superior performance.

4.4.4 Bounding Cuts

As shown in Section 4.4.3, PCR is not locally idea, which means the linear relaxation is not the convex hull of all integer solutions to $h(x, \hat{x}, \lambda, z)$. We propose additional bounding cuts in

(4.9) to strengthen the PCR. These cuts use the binary variable z to alternatively enforce the active partition onto x. Cuts (4.9) can be generated for any general variable x_m or x_n in bilinear term $x_m x_n$. However, these cuts will not be added if the associated count of partitions is less or equal than two [2].

$$\sum_{i \in \mathcal{Q}^m \setminus \{\boldsymbol{P^m}\}} \boldsymbol{b^m_{i-1}} z^m_i \leq x_k \leq \sum_{i \in \mathcal{Q}^m \setminus \{0\}} \boldsymbol{b^m_i} z^m_i \tag{4.9}$$

As shown in (4.6), the value of x is directly controlled through the variable λ, which is then constrained by z. The bounding cuts are meant to directly address this indirect relationship. Next, we discuss how bounding cuts (4.9) are valid to (4.6), but are not necessarily redundant to the linear relaxation of (4.6).

First, inequalities (4.9) do not eliminate any feasible integer solution. As there exists one and only one partition active for x_m, there exist P^m possible solution for z^m. Note that z^m has no impact on z^n through (4.6) since the lifted variable $\widehat{x_m x_n}$ is unbounded. Thus, we only focus on the possible solution on z^m. For each partition $i \in \mathcal{P}^m$, inequalities (4.9) can be rewritten as $\boldsymbol{b^m_{i-1}} z^m_i \leq x_k \leq \boldsymbol{b^m_i} z^m_i$, which is redundant to the subsystem comprised of constraints (4.6e), (4.6f), (4.6g), (4.6b), and (4.6k) with $z^m_i = 1$. Hence, (4.9) are valid inequalities. Next, we show that (4.9) is not necessarily redundant to the linear relaxation of (4.6) via counter examples Let x_m have three partitions: $[1, 10], [10, 100], [100, 1000]$ (i.e., $\boldsymbol{B^m} =: \{1, 10, 100, 1000\}$), and let x_n have only one partition $[1, 2]$ (i.e., $\boldsymbol{B^n} : \{1, 2\}$). The linear relaxation solution to (4.6) $\lambda^{mn}_{00} + \lambda^{mn}_{01} = 0$, $\lambda^{mn}_{10} + \lambda^{mn}_{11} = \frac{2}{8}$, $\lambda^{mn}_{20} + \lambda^{mn}_{31} = \frac{3}{8}$, $\lambda^{mn}_{40} + \lambda^{mn}_{41} = \frac{3}{8}$, $z^m_1 = \frac{5}{8}$, $z^m_2 = 0$, $z^m_3 = \frac{3}{8}$, $x^m = 415$, $x^n = 1.5$ is feasible for (4.6) but violates (4.9) on the right side, which requires $z^m_1 \leq 10 z^m_1 + 100 z^m_2 + 1000 z^m_3$. Also, consider the solution to $\lambda^{mn}_{00} + \lambda^{mn}_{01} = \frac{3}{8}$, $\lambda^{mn}_{10} + \lambda^{mn}_{11} = \frac{3}{8}$, $\lambda^{mn}_{20} + \lambda^{mn}_{31} = \frac{2}{8}$, $\lambda^{mn}_{40} + \lambda^{mn}_{41} = 0$, $z^m_1 = \frac{3}{8}$, $z^m_2 = 0$, $z^m_3 = \frac{5}{8}$, $x^m = 650.375$, $x^n = 1.5$ is feasible for (4.6) but violates constraints (4.9) on the left side.

4.4.5 PCR-ACPF

With PCR for bilinear terms discussed in Section 4.4.2, we denote the relaxation procedures on a bilinear term $x_m x_n$ as $\langle x_m x_n \rangle^R$. This procedure incorporates two steps: 1) lifting bilinear term $x_m x_n$ with a relaxed variable $\widehat{x_m x_n}$, and 2) adding additional variables and constraints as in (4.6)

[2] When partition count is less or equal than two, cuts (4.9) are trivially redundant.

or (4.7) to $h(x, \hat{x}, \lambda, z)$. PCR-ACPF $\mathcal{R}$ is then formulated in (4.10). We note that in practice it is viable to recursively parse ACPF constraints $g(x)$ for any bilinear terms and construct PCR term by term instead of directly using (4.10). It follows that PCR-ACPF $\mathcal{R}$ is a mixed-integer quadratic convex program (MIQCP).

$$\forall (i,j) \in \mathcal{L}$$

$$\begin{aligned} f_{ij}^p =& \frac{1}{\boldsymbol{\tau_{ij}^2}} \boldsymbol{g_{ij}^{\varepsilon}} (\langle (v_i^p)^2 \rangle^R + \langle (v_i^q)^2 \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{g_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{b_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_i^p v_j^p \rangle^R + \langle v_i^q v_j^q \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{b_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{g_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_j^p v_i^q \rangle^R - \langle v_i^p v_j^q \rangle^R) \end{aligned} \tag{4.10a}$$

$$\begin{aligned} f_{ji}^p =& \boldsymbol{g_{ij}^{\varepsilon}} (\langle (v_j^p)^2 \rangle^R + \langle (v_j^q)^2 \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{g_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{b_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_j^p v_i^p \rangle^R + \langle v_j^q v_i^q \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{b_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{g_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_i^p v_j^q \rangle^R - \langle v_j^p v_i^q \rangle^R) \end{aligned} \tag{4.10b}$$

$$\begin{aligned} f_{ij}^q =& - \frac{1}{\boldsymbol{\tau_{ij}^2}} (\boldsymbol{b_{ij}^{\varepsilon}} + \frac{\boldsymbol{b_{ij}^C}}{2}) (\langle (v_i^p)^2 \rangle^R + \langle (v_i^q)^2 \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{g_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{b_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_j^p v_i^q \rangle^R - \langle v_i^q v_j^p \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{b_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{g_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_i^p v_j^p \rangle^R - \langle v_i^q v_j^q \rangle^R) \end{aligned} \tag{4.10c}$$

$$\begin{aligned} f_{ji}^q =& - (\boldsymbol{b_{ij}^{\varepsilon}} + \frac{\boldsymbol{b_{ij}^C}}{2}) (\langle (v_i^p)^2 \rangle^R + \langle (v_i^q)^2 \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{g_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) + \boldsymbol{b_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_i^p v_j^q \rangle^R - \langle v_j^p v_i^q \rangle^R) \\ & - \frac{1}{\boldsymbol{\tau_{ij}}} (\boldsymbol{b_{ij}^{\varepsilon}} \cos(\boldsymbol{\phi_{ij}}) - \boldsymbol{g_{ij}^{\varepsilon}} \sin(\boldsymbol{\phi_{ij}})) (\langle v_j^p v_i^p \rangle^R + \langle v_j^q v_i^q \rangle^R) \end{aligned} \tag{4.10d}$$

$$\langle v_i^q v_j^p \rangle^R - \langle v_i^p v_j^q \rangle^R \geq \tan(\underline{\boldsymbol{\phi}}_{\boldsymbol{ij}}) (\langle v_i^p v_j^p \rangle^R + \langle v_i^q v_j^q \rangle^R) \tag{4.10e}$$

$$\langle v_i^q v_j^p \rangle^R - \langle v_i^p v_j^q \rangle^R \leq \tan(\overline{\boldsymbol{\phi}}_{\boldsymbol{ij}}) (\langle v_i^p v_j^p \rangle^R + \langle v_i^q v_j^q \rangle^R) \tag{4.10f}$$

(4.2g)

$$\forall i \in \mathcal{N}$$

$$\sum_{k \in \mathcal{G}_i} g_k^p - \sum_{j:(i,j) \in \mathcal{L}} f_{ij}^p + \sum_{j:(j,i) \in \mathcal{L}} f_{ji}^p - \boldsymbol{d_i^p} = \boldsymbol{g_i^s} (\langle (v_i^p)^2 \rangle^R + \langle (v_i^q)^2 \rangle^R) \tag{4.10g}$$

$$\sum_{k \in \mathcal{G}_i} g_k^q - \sum_{j:(i,j) \in \mathcal{E}} f_{ij}^q + \sum_{j:(j,i) \in \mathcal{E}} f_{ji}^q - \boldsymbol{d_i^p} = \boldsymbol{-b_i^s} (\langle (v_i^p)^2 \rangle^R + \langle (v_i^q)^2 \rangle^R) \tag{4.10h}$$

$$\langle (v_i^p)^2 \rangle^R + \langle (v_i^q)^2 \rangle^R \geq \underline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{2}} \tag{4.10i}$$

(4.2k), (4.2l), (4.2m), (4.2n), (4.2o)

Furthermore, some additional cutting planes are added based on the work by Coffrin et al. [2017b] to regulate the nonconvex constraints (4.2j). To construct the cutting planes, we define the following parameters:

$$\begin{aligned}
\boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{b}} &= \boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{l}} + \boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{u}} \quad \forall i \in \mathcal{N} \\
\overline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}} &= \max((\underline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{p}})^2, (\overline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{p}})^2) \\
\underline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}} &= \min((\underline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{p}})^2, (\overline{\boldsymbol{v}}_{\boldsymbol{i}}^{\boldsymbol{p}})^2) \\
\boldsymbol{\eta}_{\boldsymbol{ij}} &= \frac{\overline{\boldsymbol{\theta}}_{\boldsymbol{ij}} + \underline{\boldsymbol{\theta}}_{\boldsymbol{ij}}}{2} \quad \forall (i,j) \in \mathcal{L} \\
\boldsymbol{\psi}_{\boldsymbol{ij}} &= \frac{\overline{\boldsymbol{\theta}}_{\boldsymbol{ij}} + \underline{\boldsymbol{\theta}}_{\boldsymbol{ij}}}{2} \quad \forall (i,j) \in \mathcal{L}
\end{aligned}$$

Next, we append constraints (4.12a) and (4.12b) to PCR-ACPF (4.10) for a tighter relaxation:

$$\begin{aligned}
&\boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{b}} \boldsymbol{v}_{\boldsymbol{j}}^{\boldsymbol{b}} (\cos(\boldsymbol{\eta}_{\boldsymbol{ij}}) \widehat{v_i^p v_j^p} + \sin(\boldsymbol{\eta}_{\boldsymbol{ij}}) \widehat{v_i^q v_j^q}) - (\overline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}} \cos(\boldsymbol{\psi}_{\boldsymbol{ij}}) \boldsymbol{v}_{\boldsymbol{j}}^{\boldsymbol{b}} \widehat{(v_i^p)^2} \\
&\quad - (\overline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} \cos(\boldsymbol{\psi}_{\boldsymbol{ij}}) \boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{b}} \widehat{(v_j^p)^2} \geq (\overline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} (\overline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}} \cos(\boldsymbol{\psi}_{\boldsymbol{ij}}) ((\underline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} (\underline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}} - (\overline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} (\overline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}})
\end{aligned} \tag{4.12a}$$

$$\begin{aligned}
&\boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{b}} \boldsymbol{v}_{\boldsymbol{j}}^{\boldsymbol{b}} (\cos(\boldsymbol{\eta}_{\boldsymbol{ij}}) \widehat{v_i^p v_j^p} + \sin(\boldsymbol{\eta}_{\boldsymbol{ij}}) \widehat{v_i^q v_j^q}) - (\underline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}} \cos(\boldsymbol{\psi}_{\boldsymbol{ij}}) \boldsymbol{v}_{\boldsymbol{j}}^{\boldsymbol{b}} \widehat{(v_i^p)^2} \\
&\quad - (\underline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} \cos(\boldsymbol{\psi}_{\boldsymbol{ij}}) \boldsymbol{v}_{\boldsymbol{i}}^{\boldsymbol{b}} \widehat{(v_j^p)^2} \geq (\underline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} (\underline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}} \cos(\boldsymbol{\psi}_{\boldsymbol{ij}}) ((\underline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} (\underline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}} - (\overline{\boldsymbol{\omega}}_{\boldsymbol{i}}^{\boldsymbol{p}})^{\frac{1}{2}} (\overline{\boldsymbol{\omega}}_{\boldsymbol{j}}^{\boldsymbol{p}})^{\frac{1}{2}})
\end{aligned} \tag{4.12b}$$

4.5 ACPF Infeasibility Proof Algorithm

In this section, we propose an ACPF infeasibility proof algorithm algorithm (ACPF-IP) for ACPF problem. ACPF-IP iteratively tightens PCR-ACPF with the assistance of OBBT, automatic adaptive partitioning, and network-based heuristics. Moreover, ACPF-IP also incorporates common optimization techniques, including symbolic reformulation, bound propagation, and conflict analysis. The notation system is expanded for the ease of algorithm introduction. The ACPF-IP algorithm is presented through a function-based structure in Algorithm 8. Each function is discussed in a separate subsection with detailed pseudocode blocks. Finally, we discuss the convergence of ACPF-IP and demonstrate to argue that it either finds a feasible solution or proves the ACPF problem

infeasible.

4.5.1 Notation and Usage

Let s denote a general solution to a PCR-ACPF problem instance. The functional usage of s is denoted as $s(x_i)$, which returns the solution of variable $x_i \in x$. For example, $s(v_i^p)$ returns the solution of the real voltage variable at bus $i \in \mathcal{N}$, which is a real number. This usage is also applied when the input argument is a variable vector (e.g., $s(x)$ returns the solution vector of all variables x). The tightness of PCR-ACPF is dependent on the discretized variable domains, i.e., partitions information stored in B^i for some variables $x_i \in x$. Let $\mathcal{B}$ denote a partition information collection (PIC), which is a dictionary-like data structure indexed by variable reference $x_i \in x$. We assume each entry in PIC $\mathcal{B}$ maintains a sorted array data structure (i.e., partition information B^i of variable x_i is sorted when stored in $\mathcal{B}$). Let $B \leftarrow \mathcal{B}(x_i)$ denote the use of $\mathcal{B}$ to fetch variable x_i's partition information B. Further, let $\mathcal{B}(x_i) \leftarrow B$ denote the assignment of partition information B to the PIC $\mathcal{B}$ on variable x_i.

Recall that $\mathcal{R}$ is used to denote the general PCR-ACPF from Section 4.4.1. We define the usage $\mathcal{R}^l \leftarrow \mathcal{R}(\mathcal{B})$ as a formulation constructor, where PCR-ACPF $\mathcal{R}^l$ is constructed based on the PIC $\mathcal{B}$. By default, the constructed $\mathcal{R}^l$ is a feasibility problem since no objective function exists. Next, we define an additional usage of $\mathcal{R}$ with an optional argument. Let $\mathcal{R}^l \leftarrow \mathcal{R}(\mathcal{B},\ \min\ x_i)$ denote the construction of PCR-ACPF $\mathcal{R}^l$ with the objective function of minimize x_i. For example, $\mathcal{R}^l \leftarrow \mathcal{R}(\mathcal{B},\ \min \sum_{i\in\mathcal{N}} \sum_{k\in\mathcal{G}_i} c_k^p g_k^p)$ constructs a PCR-ACPF with PIC $\mathcal{B}$ while minimizing the total cost of real power generation, where c_k^p is defined to be the real power generation cost of generator $k \in \mathcal{G}_i$ at bus $i \in \mathcal{N}$.

In addition, we define function $\langle L, s\rangle \leftarrow \textsc{Solve}(\mathcal{R}(\cdot))$ to solve a PCR-ACPF using a MIQCP solver. This function allows one input argument, which is the returned problem of a PCR-ACPF constructor $\mathcal{R}(\cdot)$. The output of this function is the best objective bound L and the best solution s reported from the MIQCP solver. In the case of an infeasible problem, the return values are $\langle -\infty, \emptyset\rangle$. Function $U \leftarrow \textsc{LocalSearch}(\mathcal{F}, x = s(x))$ defines a local solve of ACPF problem $\mathcal{F}$ using a nonlinear program (NLP) solver with $s(x)$ as the starting solution point. Given our focus on ACPF feasibility, the solution to LocalSearch is not of interest. The best objective value U is sufficient to solve ACPF when $L < \infty$. In cases when the NLP solver finds an infeasible stationary point, we assume the returned value is ∞. Note that the dimension of the solution s can be larger

than x, as s comes from a PCR-ACPF solution which contains additional variables (e.g., $\hat{x}.\lambda, z$).

4.5.2 ACPF-IP Algorithm Overview

Algorithm 8 ACPF-IP Algorithm

1: **function** ACPF-IP
2: $\langle \mathcal{B}, \mathcal{T}, s, U, L \rangle \leftarrow$ PreProcess($\mathcal{F}$)
3: **while** $U = \infty$ and $L > -\infty$ **do**
4: $\mathcal{V} \leftarrow$ UpdatePartitioningVars($\mathcal{T}, s$)
5: $\mathcal{B} \leftarrow$ BoundTightening($\mathcal{V}, \mathcal{B}$)
6: $\mathcal{B} \leftarrow$ AddPartitions($\mathcal{B}, \mathcal{V}, s$)
7: $\mathcal{R}^l \leftarrow \mathcal{R}(\mathcal{B})$
8: $\langle L, s \rangle \leftarrow$ Solve($\mathcal{R}^l$)
9: $U \leftarrow$ LocalSearch($\mathcal{F}, x = s(x)$)
10: **end while**
11: **return** $\langle L, U \rangle$
12: **end function**

The proposed ACPF-IP is presented in Algorithm 8. ACPF-IP iteratively solves tighter PCR-ACPF problems by further partitioning variable domains until the termination criteria is met. At Step 2, Algorithm 8 first performs function PreProcess (see Section 4.5.3) to initialize essential information, propagate variable bounds, and perform a root relaxation without discretizing variable domains. The main loop begins at Step 3. At each iteration, ACPF-IP first selects a subset of partitioning variables using function UpdatePartitioningVars at Step 4 (see Section 4.5.5). The selected variables' domain will have new partitions added, which specify the direction of subsequent discretization based on the most recent relaxation solution s. OBBT is performed sequentially at Step 5 with function BoundTightening (see Section 4.5.4). OBBT produces tighter bounds for the upcoming PCR-ACPF at Step 8. Based on the relaxation solution s, PIC $\mathcal{B}$ is updated with new partitions added to selective variable domains in function AddPartitions at Step 6 (see Section 4.5.6). A PCR-ACPF $\mathcal{R}^l$ is constructed based on the updated PIC $\mathcal{B}$ at Step 7. This relaxation is solved at Step 8 using a MIQCP solver which then updates the relaxation solution s and best lower bound L. Function LocalSearch is then applied at Step 9 to detect any feasible solution using the most recent relaxation solution s as the starting point. Finally, the algorithm terminates at Step 3 if: 1) a feasible solution is detected with $U < \infty$ or 2) an infeasible relaxation is detected with $L = -\infty$ [3]

[3] In our implementation, ACPF-IP can also be terminated given parameters such as time limits, maximum iterations, etc. For clarity, these practical termination criteria are not listed in Algorithm 8.

4.5.3 ACPF-IP Pre-processor

The function PreProcess, presented in Algorithm 9, prepares ACPF-IP by parsing ACPF formulation $\mathcal{F}$ for bilinear terms, performing basic bound propagation, and initializing algorithmic variables.

Algorithm 9 ACPF-IP Pre-Processor

function PreProcess($\mathcal{F}$)
  $U, L \leftarrow \infty, -\infty$
  $\mathcal{T} \leftarrow$ Parse($\mathcal{F}$)
  for $j = 1 \cdots N$ **do**
    for $x_k \in x$ **If** $a_{kj} \neq 0$ **do**
      $\boldsymbol{x_k^l} = \max(\boldsymbol{x_k^l}, \frac{1}{\boldsymbol{a_{kj}}}(\boldsymbol{b_j} - \sum_{i=1:j\neq k}^{N} \max(\boldsymbol{a_{ij}x_i^l}, \boldsymbol{a_{ij}x_i^u}) - \sum_{i=1}^{D} \max(\boldsymbol{a'_{ij}\hat{x}_i^l}, \boldsymbol{a'_{ij}\hat{x}_i^u})))$
      $\boldsymbol{x_k^u} = \min(\boldsymbol{x_k^u}, \frac{1}{\boldsymbol{a_{kj}}}(\boldsymbol{b_j} - \sum_{i=1:j\neq k}^{N} \min(\boldsymbol{a_{ji}x_i^l}, \boldsymbol{a_{ji}x_i^u}) - \sum_{i=1}^{D} \min(\boldsymbol{a'_{ij}\hat{x}_i^l}, \boldsymbol{a'_{ij}\hat{x}_i^u})))$
    end for
  end for
  $\mathcal{B}^0 \leftarrow \{x_i : \{\boldsymbol{x_i^l}, \boldsymbol{x_i^u}\}\ \forall x_i \in \boldsymbol{x}\}$
  $\langle L, s^0 \rangle \leftarrow$ Solve($\mathcal{R}(\mathcal{B}^0)$)
  return $\langle \mathcal{B}^0, \mathcal{T}, s^0, U, L \rangle$
end function

Algorithm 9 takes ACPF problem $\mathcal{F}$ as input. The first step in Algorithm 9 is to initialize the problem's best upper bound U to $+\infty$ and its best lower bound L to $-\infty$. Function Parse is applied to ACPF problem $\mathcal{F}$ in order to collect a distinct set $\mathcal{T}$ of bilinear terms. This function is based on the work by Smith and Pantelides [1999], who observed that general algebraic expression is comprised of five basic operations of arithmetic, limited transcendental functions, and unary operators. The symbolic expressions of $\mathcal{F}$ can be parsed recursively to obtain all algebraic terms that are responsible for problem nonconvexities, i.e., terms that fit (4.5). For brevity, we use this method to collect $\mathcal{T} := \{\langle x_m x_n \rangle | x_m, x_n \in x, x_m x_n \in \mathcal{F}\}$ instead of exhaustively listing all bilinear terms as in (4.2). We also observe that reformulated constraints $g'(x, \hat{x})$ in PCR-ACPF $\mathcal{R}$ can also be created using this symbolic reformulation technique.

The next step of function PreProcess is a basic algebraic propagation for trivial variable bounds. Reformulated linear constraints $g'(x, \hat{x})$ take the form of (4.13) with N constraints, M variables of x, and D lifted variables $\hat{x}$:

$$\sum_{i=1}^{M} \boldsymbol{a_{ij}} x_i + \sum_{i=1}^{D} \boldsymbol{a'_{ij}} \hat{x}_i - \boldsymbol{r_j} \leq 0 \quad \forall j = 1 \cdots N \tag{4.13}$$

In (4.13), a_{ij} and $a_{ij}^{'}$ are coefficients of variable x_i and $\hat{x}_i$ at the j-th constraint, respectively. Parameter r_j is the right hand side of the j-th constraint. The bound propagation, as shown in Steps 4 to 9 is performed for all variables x.

At Step 10, function PrePROCESS initializes a root PIC $\mathcal{B}^0$ of all x using the propagated lower and upper bounds. A root relaxation $\mathcal{R}^0$ based on $\mathcal{B}^0$ is solved at Step 11. Given $\mathcal{B}$ has no discretization on variable domains, $\mathcal{R}^0$ is a tractable quadratic program (QP). At Step 11, a root lower bound L and the associated root relaxed solution s^0 are obtained. Algorithm 9 returns PIC $\mathcal{B}^0$, all bilinear terms $\mathcal{T}$, root relaxation solution s^0, and initialized best upper and lower bound U and L to ACPF-IP.

4.5.4 OBBT Algorithm

As surveyed in Puranik and Sahinidis [2017a], OBBT solves the optimization problem in (4.14) for each variable $x_i \in x$ by placing them in the objective function. This yields lower and upper limits that are considered the tightest lower and upper bounds of x_i, respectively.

$$\{\min \pm x_i \mid g(x) \leqslant \mathbf{0}, \boldsymbol{x^l} \leqslant x \leqslant \boldsymbol{x^u}\} \tag{4.14}$$

Applying OBBT in optimization algorithm is a common practice because it can effectively reduce the search space and improve convergence performance. At each iteration of ACPF-IP, a tighter PCR-ACPF is solved using a MIQCP solver based on a branch-and-bound algorithm. Therefore, tighter bounds on PCR-ACPF can effectively reduce the required solver effort and, thus, improve the performance of ACPF-IP. However, the efforts required to solve (4.14) can be very expensive given its nonconvex nature. In this study, we consider using PCR-ACPF for OBBT as in (4.15):

$$\mathcal{R}(\mathcal{B}, \min \pm x_i) \tag{4.15}$$

Function OBBT BoundTightening is called in Algorithm 8. The quality of bound tightening is dependent on the tightness of PCR-ACPF, which is dependent on PIC $\mathcal{B}$. The computational challenge of ACPF-IP lies in solving PCR-ACPF $\mathcal{R}^l$ through an exhaustive tree search to assess infeasibility. OBBT is applied at every iteration based on the updated PIC $\mathcal{B}$ to further narrow variable domains. Reducing variable domains can effectively eliminate binary variables in PCR-

ACPF to ease computational requirements. However, the required computational effort for OBBT increases as variable domains are further discretized. A trade-off exists between the computational efforts required for OBBT and PCR-ACPF. In our implementation, a time limit is enforced for each OBBT in ACPF-IP.

Algorithm 10 OBBT Algorithm

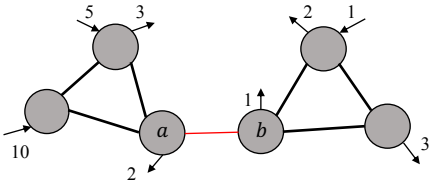
1: **function** BOUNDTIGHTENING($\mathcal{V}, \mathcal{B}$)
2: **for** $x_i \in \mathcal{V}$ **do**
3: $\langle x_i^{\min}, s\rangle \leftarrow$ SOLVE($\mathcal{R}(\mathcal{B},\ \min x_i)$)
4: $\mathcal{B}(x_i) \leftarrow \{b \mid b \in \mathcal{B}(x_i),\ b \geq x_i^{\min}\}$
5: **if** $|\mathcal{B}(x_i)| = 1$ **then**
6: $\mathcal{B}(x_i) \leftarrow \{1, -1\}$
7: **break**
8: **end if**
9: $\langle x_i^{\max}, s\rangle \leftarrow$ SOLVE($\mathcal{R}(\mathcal{B},\ \max x_i)$)
10: $\mathcal{B}(x_i) \leftarrow \{b \mid b \in \mathcal{B}(x_i),\ b \leq x_i^{\max}\}$
11: **if** $|\mathcal{B}(x_i)| = 1$ **then**
12: $\mathcal{B}(x_i) \leftarrow \{1, -1\}$
13: **break**
14: **end if**
15: **end for**
16: **return** $\mathcal{B}$
17: **end function**

OBBT is presented in Algorithm 10. It takes two inputs: a subset of variables $\mathcal{V} \subseteq x$ to perform OBBT and a PIC $\mathcal{B}$. For each variable $x_i \in \mathcal{V}$, we first tighten its lower bound by solving a PCR-ACPF based on $\mathcal{B}$ at Step 3. The resulting $x_i^{\min}$ is used to update the corresponding partition information $\mathcal{B}$ at Step 4. A conflict detection is performed at Step 5 to see if the updated lower bound x_i^{min} violates its upper bound, which results in $\mathcal{B}(x_i)$ with length of one. If detected, we inject a dummy infeasible partition set at Step 6 and exit OBBT since the problem is proven infeasible. This dummy set will make the next PCR-ACPF infeasible. If not, OBBT proceeds to the upper bound side and performs the same algorithm in the opposite direction from Step 9 to Step 14. Finally, the updated PIC $\mathcal{B}$ is returned to ACPF-IP.

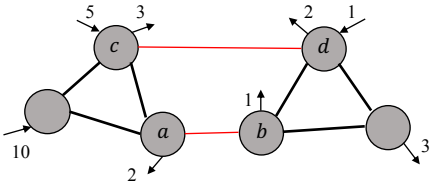
4.5.5 ACPF Partitioning Variable Selection

Partitioning variable selection has been rarely discussed in previous research. An adapted method proposed by Boukouvala et al. [2016] is commonly considered for global optimization methods. This method constructs a graph to represent nonconvexities in the formulation. In our study,

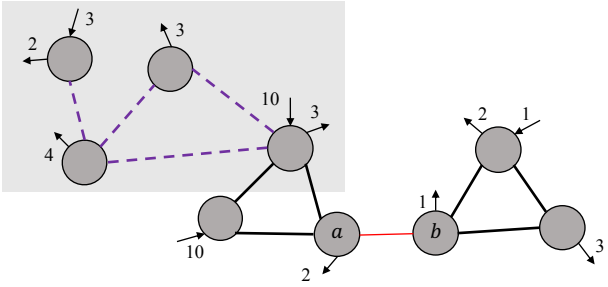
the graph is limited to focus on bilinear terms. In the graph, variables involved in bilinear terms are nodes while the bilinear product relationship are denoted as undirected arcs. Selecting all nodes in the graph for partitioning can be challenging for PCR-ACPF given that number of binary variables in PCR-ACPF can increase drastically when partitions are added. However, selecting more partitioning variables provides a tighter relaxation to reduce the total number of iterations required by ACPF-IP. Alternatively, a minimum vertex cover (MVC) problem can be solved on the graph to determine the partitioning variables with all arcs covered. On a sparse graph, this MVC-based method has been proven to be effective on the global convergence for general mixed-integer nonlinear programs as shown in Nagarajan et al. [2017].



(a) Example network where the thin line (a, b) has a tight thermal limit while other lines have relatively large thermal limits.



(b) Example network adapted from 4.3a with bypass line (c, d) constructed to alleviate congestion on line (a, b). The bypass line also limited thermal limits, which is reflected by the line's thickness.



(c) A more complex example network with non-trivial infeasibility from congested lines in relaxation solutions. The dashed lines indicate an intermediate thermal line in comparison to large thermal limits on lines.

Figure 4.3: Example networks to demonstrate the heuristic design for partitioning variable selection

MVC-based partitioning variable selection is not ideal for ACPF as the corresponding graph is dense. This motivates the need for a dedicated partitioning variable selection method for the ACPF problem. ACPF-IP is iteratively tightening PCR-ACPF to reduce the overlaps between the relaxed nonconvex space defined through $h(x, \hat{x}, \lambda, z)$ and the lifted linear space defined by system $g'(x, \hat{x})$. A dedicated method should consider the properties of the linearly lifted space $g'(x, \hat{x})$,

where all problem context is described with ACPF constraints. In Algorithm 11, we consider a hybrid heuristic that adaptively selects partitioning variables at every iteration of ACPF-IP based on network analysis with relaxed solution s. The algorithm takes PCR-ACPF solution s as input and the outputs a subset of variables $\mathcal{V}$ for further partitioning.

When a PCR-ACPF with few variable domain partitions is solved, the linear lifted system $g'(x, \hat{x})$ is likely to induce infeasible conflicts with trivially violated ACPF constraints. Consider the example in Figure 4.3a, where demand is marked as an exit arrow, while power generated is marked as an inflow arrow. The thin line has the most constrained thermal limits while the other lines' thermal limits are relatively large. When the nonconvexities are relaxed, line (a, b) is the most important line for a feasible relaxation. Further, line (a, b) is also the most congested line when pushing the flow through this relaxed problem (i.e., the relaxation is infeasible if line limits are less than 5). This observation that trivial infeasibility can be proven through conflicts even with a weak relaxation. If the thermal limits on line (a, b) exceeds 5, the true limitations for an infeasible ACPF is nontrivially limited through nonconvexities. In this case, the line (a, b) stays congested as the relaxation will try to push as much flow on this line as possible to fulfill demand. If the problem is actually infeasible, focusing on tightening the relaxation with line (a, b) (i.e., adding partitions to associated voltage variables) is a viable choice to quickly create an infeasible PCR-ACPF.

In example Figure 4.3b, if bypass line (c, d) is added to alleviate stress on line (a, b), the relaxed flow on (c, d) is likely to increase when tighter relaxation is imposed on the nonconvexities associated with line (a, b). This can result in shifting of focus from line (a, b) to line (c, d) when line (a, b) is fully tightened, which then selects variables associated with line (c, d) for further partitioning. Therefore, we consider a heuristic from Step 5 to Step 7 that collects partitioning variables based on line congestion given a PCR-ACPF solution. The congestion is measured through $|(\tilde{f}_{ij}^p)^2 + (\tilde{f}_{ij}^q)^2 - \boldsymbol{l_{ij}^2}|$, where $\tilde{f}_{ij}^p$ and $\tilde{f}_{ij}^q$ are obtained from a PCR-ACPF solution s at Step 4.

The analysis above can result in a local trap in a complex network since a partial knowledge of the network is revealed. One can continue to focus on partitioning variables associated with the congested lines while the infeasibility of ACPF exists elsewhere. For example in Figure 4.3c, focus on line (a, b) may not be effective while infeasibility exists in the shaded region, i.e., where relaxation on line thermal limits are not actually congested. In this case, line (a, b) is congested because the true thermal limits overlap with the tightest relaxation obtained while the rest of the network is relaxed. Hence, we need to direct the algorithm to escape the local trap.

Algorithm 11 Partitioning variable selection algorithm

1: **function** UPDATEPARTITIONINGVARS(s)
2: $\langle \mathcal{V}, d^* \rangle \leftarrow \emptyset, -\infty$
3: **for** $(i,j) \in \mathcal{L}$ **do**
4: $\langle \tilde{f}_{ij}^p, \tilde{f}_{ij}^q \rangle \leftarrow s(f_{ij}^p), s(f_{ij}^q)$
5: **if** $|(\tilde{f}_{ij}^p)^2 + (\tilde{f}_{ij}^q)^2 - \boldsymbol{l}_{\boldsymbol{ij}}^2| \leq \epsilon$ **then**
6: $\mathcal{V} \leftarrow \mathcal{V} \cup \{v_i^p, v_j^p, v_i^q, v_i^q\}$
7: **end if**
8: $\langle \tilde{v}_i^p, \tilde{v}_j^p, \tilde{v}_i^q, \tilde{v}_i^q \rangle \leftarrow s(v_i^p), s(v_j^p), s(v_i^q), s(v_j^q)$
9: $\langle \tilde{\omega}_i^p, \tilde{\omega}_j^p, \tilde{\omega}_i^q, \tilde{\omega}_j^q \rangle \leftarrow s(\widehat{v_i^p v_i^p}), s(\widehat{v_j^p v_j^p}), s(\widehat{v_i^q v_i^q}), s(\widehat{v_j^q v_j^q})$
10: $\langle \tilde{\omega}_{ij}^p, \tilde{\omega}_{ij}^q, \tilde{\omega}_{ji}^p, \tilde{\omega}_{ij}^{pq}, \tilde{\omega}_{ij}^{qp} \rangle \leftarrow s(\widehat{v_i^p v_j^p}), s(\widehat{v_i^q v_j^q}), s(\widehat{v_j^p v_i^p}), s(\widehat{v_i^p v_j^q}), s(\widehat{v_i^q v_j^p})$
11: $a = |\tilde{\omega}_i^p - (\tilde{v}_i^p)^2| + |\tilde{\omega}_j^p - (\tilde{v}_j^p)^2| + |\tilde{\omega}_i^q - (\tilde{v}_i^q)^2| + |\tilde{\omega}_j^q - (\tilde{v}_j^q)^2|$
12: $b = |\tilde{\omega}_{ij}^p - \tilde{v}_i^p \tilde{v}_j^p| + |\tilde{\omega}_{ij}^q - \tilde{v}_i^q \tilde{v}_j^q| + |\tilde{\omega}_{ji}^p - \tilde{v}_j^p \tilde{v}_i^p| + |\tilde{\omega}_{ji}^{pq} - \tilde{v}_i^p \tilde{v}_j^q| + |\tilde{\omega}_{ij}^{qp} - \tilde{v}_i^q \tilde{v}_j^p|$
13: **if** $a + b > d^*$ **then**
14: $\bar{V} \leftarrow \{v_i^p, v_j^p, v_i^q, v_i^q\}$
15: $d^* \leftarrow a + b$
16: **end if**
17: **end for**
18: $\mathcal{V} \leftarrow \mathcal{V} \cup \bar{V}$
19: **return** $\mathcal{V}$
20: **end function**

Given a PCR-ACPF solution s, we define the deviation d of bilinear term $\langle x_m x_n \rangle$ to be the distance between the solution of lifted variable $s(\widehat{x_m x_n})$ and variable solution product $s(x_m)s(x_n)$ (i.e., $d = |s(\widehat{x_m x_n}) - s(x_m)s(x_n)|$). This value reflects the magnitude lifted variable $\widehat{x_m x_n} \in x'$ is trying to "cheat" in relaxed space for a feasible solution to system $g'(x, \hat{x})$. Clearly, for any solution s^* of PCR-ACPF, if the maximum deviation of all bilinear terms is less than the feasibility tolerance, then s^* must be a feasible solution. Therefore, in additional to the line congestion heuristic, the total deviation of all bilinear terms associated with each line $(i,j) \in \mathcal{L}$ is measured at Step 11 and Step 12 [4], while the values are obtained from Steps 8 to 10. The variables associated with the most deviated line are selected for further domain discretization in Steps 13 to 16.

4.5.6 Constructing New Partitions

An effective method proposed by Nagarajan et al. [2017] discretizes variable domains by injecting a new partition pivoting around the relaxed solution, where the radius of the new partition is controlled through a constant parameter. Although this method is directly applicable to ACPF-IP, it requires a local search to obtain the best parameter.

[4] For brevity, the associate bilinear can be collected through ACPF formulation (4.2) or augment bilinear terms collection $\mathcal{T}$ with witch line $(i,j) \in \mathcal{L}$ each term is associated with.

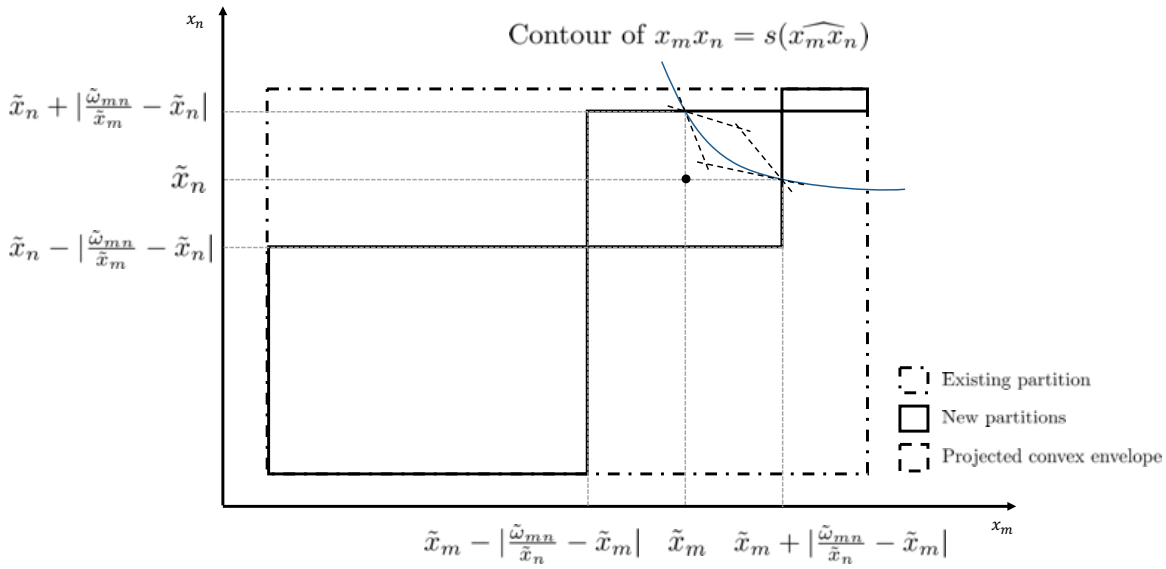


Figure 4.4: Graph for constructing new partitions

When a PCR-ACPF is feasible, the relaxed region still overlaps with the linear lifted space defined by $g'(x, \hat{x})$. Newly constructed partitions should be able to eliminate the known feasible solution based on the existing relaxation solution s. However, is it possible that using a constant-based partition construction may not tighten PCR-ACPF to eliminate the known feasible region. If it fails to do so, then the upcoming PCR-ACPF, which is harder to solve, can have a trivial feasible region which extends the required iterations of ACPF-IP.

We adapt the constant-driven method with a new scheme that automatically select parameters for creating new partitions in Algorithm 12. This algorithm uses the collection of bilinear terms $\mathcal{T}$, current PIC $\mathcal{B}$, selected partitioning variables $\mathcal{V}$, and the best PCR-ACPF relaxation solution s as inputs.

For each partitioning variable x_i in $\mathcal{V}$, we first collect its solution value $\tilde{x}_i$ and the associated active partition $[b_l, b_u]$, such that $\tilde{x}_i \in [b_l, b_u]$, at Step 3 and 4. For each bilinear term that contains x_i, we obtain the solution to the lifted variable $\tilde{\omega}_{ij}$ and variable $\tilde{x}_j$ at Step 9. Step 10 measures the required radius for the new partition that guarantees the existing relaxed solution $(\tilde{x}_i, \tilde{x}_j, \tilde{\omega}_{ij})$ is left outside the convex envelope on the contour line as shown in Figure 4.4. The measured radius is regulated within an from 4 to 32 based on past experiences by Nagarajan et al. [2017]. It is important to observe that a variable may exist in multiple bilinear terms, which leads to different radius measurements. To resolve this, measured radius are collected into Δ at Step 10. The average

Algorithm 12 Construct New Partitions

function AddPartitions($\mathcal{T}, \mathcal{B}, \mathcal{V}, s, \Delta$)
  for $x_i \in \mathcal{V}$ **do**
    $\tilde{x}_i \leftarrow s(x_i)$
    $\langle b_l, b_u \rangle \leftarrow \max\{b_k | b_k \in \mathcal{B}(x_i), b_k \leq \tilde{x}_i\}, \min\{b_k | b_k \in \mathcal{B}(x_i), b_k \geq \tilde{x}_i\}$
    $\Delta \leftarrow \emptyset$
    for $\langle x_m x_n \rangle \in \mathcal{T}$ **do**
      if $i \in \{m, n\}$ **then**
        if $i = m$ **then** $j = n$ **else** $j = m$ **end if**
        $\langle \tilde{\omega}_{ij}, \tilde{x}_j \rangle \leftarrow s(\widehat{x_m x_n}), s(x_j)$
        $\Delta \leftarrow \Delta \cup \{\max(4, \min(32, \frac{|b_u - b_l|}{|\tilde{\omega}_{ij}/\tilde{x}_j - \tilde{x}_i|}))\}$
      end if
    end for
    $r \leftarrow$ Average(Δ)
    if $\tilde{x}_i - r > b_l$ **then**
      $\mathcal{B}(x_i) \leftarrow \mathcal{B}(x_i) \cup \{\tilde{x}_i - r\}$
    end if
    if $\tilde{x}_i + r < b_u$ **then**
      $\mathcal{B}(x_i) \leftarrow \mathcal{B}(x_i) \cup \{\tilde{x}_i + r\}$
    end if
    if $\tilde{x}_i - r \leq b_l$ and $\tilde{x}_i + r \geq b_u$ **then**
      $\bar{k} \leftarrow \text{argmax}_k |b_k - b_{k-1}| := \{b_k, b_{k-1} \in \mathcal{B}(x_i)\}$
      $\langle b_l, b_u \rangle \leftarrow \mathcal{B}(x_i)[k], \mathcal{B}(x_i)[k-1]$
      $\mathcal{B}(x_i) \leftarrow \mathcal{B}(x_i) \cup \{b_l + \frac{|b_u - b_l|}{2}\}$
    end if
    Sort($\mathcal{B}(x_i)$)
  end for
  return $\mathcal{B}$
end function

Δ is used as the selected radius r (at Step 13) to create new partitions around the pivot point $\tilde{x}_i$ within $[b_l, b_u]$ from Steps 14 to 19. In cases of failing to create new partitions at both Step 15 and 18, a scheme to avoid the local trap is in Steps 20 to 24 by dividing the largest partition in two. This construction appends new partition data to the PIC $\mathcal{B}$, where the order of partition information is rearranged at Step 25. Finally, the updated PIC is returned in Step 27.

4.5.7 Discussion on Algorithm Convergence

The convergence of ACPF-IP is defined as ACPF-IP can either find a feasible solution or prove infeasibility in a finite number of iterations. As suggested by Horst and Tuy [2013], the sufficient condition for global algorithm convergence is the exhaustiveness on a given mixed-integer nonlinear program. The exhaustiveness of ACPF-IP is constructed in Algorithm 8 Steps 4 and 6, where new partition are added for a strictly tighter PCR-ACPF in each iteration. Let s^*, u^* denotes the optimal solution and objective value for an ACPF, respectively. Let s_n, l_n denote the PCR-ACPF solution and corresponding bound detected at the n-th iteration of ACPF-IP. Given that ACPF is a subset of mixed-integer nonlinear programs, applying the proof in Nagarajan et al. [2017] shows that lower bound l_n monotonically increases to u^* as n increases for ACPF problems.

Next, we assert that s_n will converge to s^* when there exists a feasible tolerance $\epsilon > 0$ in a finite number of iterations. In ACPF-IP Steps 4 and 6, new partitions are constructed for variables with the largest deviation of bilinear term relaxation based on the previous iteration. In a finite number of iterations, there will exist enough partitions for variables in bilinear term $x_m x_n$, which means the maximum deviation of the relaxed variable $d^* = |\widehat{x_m x_n} - x_m x_n|$ converges to $d^*_{mn} \leq \epsilon$, where d^*_{mn} is defined by (4.16):

$$d^*_{mn} \leq \max\{\max([b^m_{i-1}, b^m_i] \times [b^n_{j-1}, b^n_j]^\top) - \min([b^m_{i-1}, b^m_i] \times [b^n_{j-1}, b^n_j]^\top) \mid \forall i \in \mathcal{P}^m, j \in \mathcal{P}^n\} \quad (4.16)$$

Equation (4.16) provides the least maximum magnitude of the relaxed region of any partition's on bilinear term variables. As ACPF-IP seeks to tighten the largest deviated bilinear term relaxations, the maximum deviation of all bilinear term relaxations eventually converges to below ϵ at some iteration n. In this circumstance, solution s_n from PCR-ACPF $\mathcal{R}$ is feasible for $g(x)$ in ACPF $\mathcal{F}$ within feasibility tolerance ϵ. Otherwise, ACPF-IP will terminate with ACPF being proven infeasible, which completes the assertion.

4.6 Experimental Results

In this section, we present experimental results from using the proposed ACPF-IP to analyze a few standard test systems. We compare the effectiveness of ACPF-IP to the commercial global solver BARON, an open-source global solver COUENNE, and AMP.

4.6.1 Experimental Setup

Four standard test systems from Coffrin et al. [2014] are tested within the congested thermal limit range (marked as the red region in Figure 4.1) as described in Section 4.3. For each congested range, we use a 2% step size in our experiments. Table 4.3 contains a summary of the test systems, including basic dimensional information and the thermal limit capacity range of interest.

Test System	# of buses	# of lines	Thermal limit capacity test range
`nesta_case6_c`	6	7	14% - 6%
`nesta_case6_ww`	6	11	94% - 86%
`nesta_case14_ieee`	14	20	54% - 32%
`nesta_case24_ieee_rts`	24	38	78% - 50%

Table 4.3: Summary of selected test systems

We implement the ACPF-IP in `Julia` 0.6.2 supported by the interface packages of `JuMP.jl` 0.18.0 and `MathProgBase.jl` 0.6.4. The implementation of the comparison algorithm AMP is supported by the same packages. Both ACPF-IP and AMP require dependent solvers, namely a MIQCP solver and an NLP solver. In this study, we use `Gurobi` 7.5.2 as the MIQCP solver and `Ipopt` 3.12.1 as the NLP solver (Note: these solvers are connected to `Julia` through solver interface packages `Gurobi.jl` 0.3.3 and `Ipopt.jl` 0.2.6). When solving test systems with either ACPF-IP or AMP, we limit `Gurobi` to using 8 threads for its parallel branch-and-bound algorithm. For the benchmark global solvers, we use `BARON` 17.10.16 and `COUENNE` 0.5 with their default settings.

All tests are conducted on the Palmetto high-performance cluster at Clemson University using compute nodes equipped with Intel(R) Xeon(R) CPU L5420 processor(s) @ 2.50GHz and 16 GB of RAM. Based on initial test's, we see that benchmark global solvers may not solve the ACPF problem in two hours. The experimental time limit is extended to enable the benchmark methods to fully realize their potentials. The maximum allowed solution wall clock time is set to be 6 hours for each experiment while overhead time for data loading and software compilation are omitted in the results comparison.

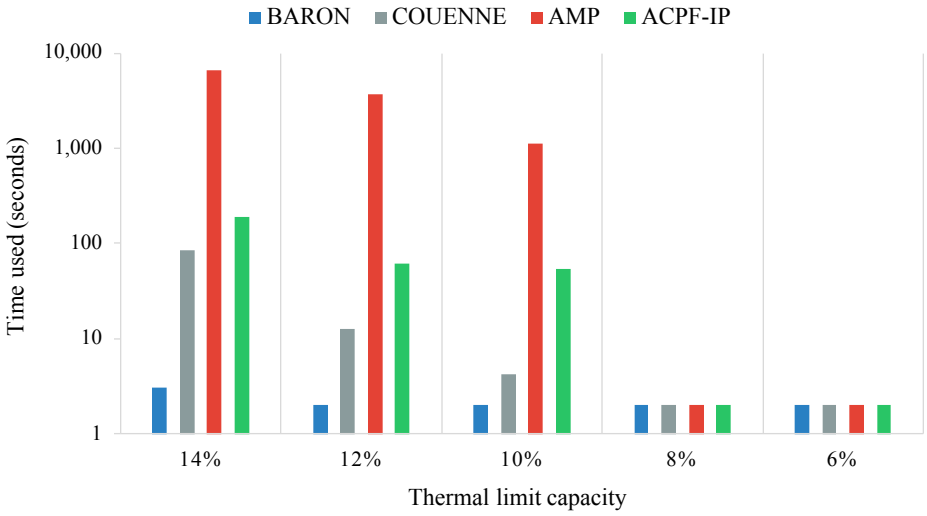
4.6.2 Results

The results for the four test systems in Table 4.3 are shown in Figure 4.5a to Figure 4.7. The experiment for each test system is conducted by gradually increasing the thermal limits capacity percentage. It is important to note that increasing the congestion percentage strictly tightens the ACPF problem, i.e., an ACPF problem on a test system is infeasible with 55% thermal limit capacity if it is proven infeasible with 60% thermal limit capacity. However, the choice of congestion level is not trivially known a prior. Therefore, we compare each step in the test range independently across different methods in our study. In Figures 4.5a to Figure 4.7, each step of the congestion percentage spreads horizontally into several blocks, where each block contains the results of all tested methods.

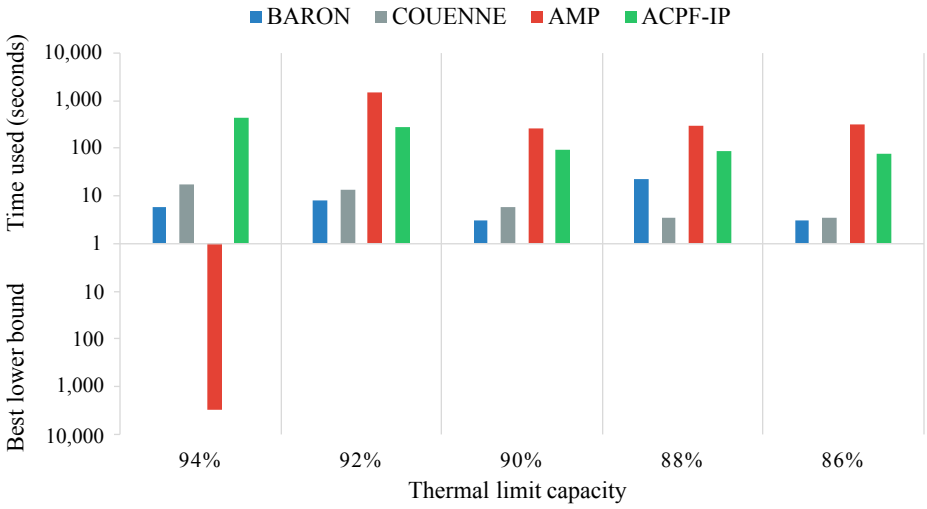
In comparing the competing methods, we first focus on proving an ACPF problem infeasible, which is reflected through two aspects: 1) whether the method is able to prove an infeasible ACPF problem is infeasible, and 2) the time required took to complete the proof. In each Figure in this Section, the top half of the Figure indicates such a comparison. When an algorithm is able to prove the system infeasible under a specific congestion percentage, the bars in the top half of a Figure indicate the time required to complete the proof. Otherwise, no bar is shown in the Figure's top half. The lower the bar, the faster the method is. The time required to complete the infeasible proof drastically reduces when the proof is completed through detected variable bounds conflicts. Hence, we show the solution time bars on a logarithmic scale.

When a method neither proves a problem infeasible nor finds a feasible solution within the allowed six-hour time limit, a best lower bound is returned that measures the minimum objective value if the ACPF problem ever becomes feasible. Before a problem is proven infeasible, a higher lower bound is preferred because it indicates the effectiveness of the global algorithm. We show this measurement in the lower half of each bar chart. The deeper the bar, the better the method is. Similar to the top half of each Figure, a logarithmic scale is applied to this side of the bar chart. When infeasibility is proven, there will be no bar shown in the lower half of the Figure with that method. To differentiate the performance of detecting a lower bound more clearly, we mark the best lower bound (lowest bars) with the symbol $\bigwedge$ whenever a comparison is necessary.

For the two small 6-bus test systems `nesta_case6_c` and `nesta_case6_ww` in Figure 4.5, we observe that the entire thermal limit capacity test range is proven to be infeasible. In both cases, we observe that infeasibility detection can be trivial for the tested global solvers as seen in the short



(a) Results summary of test system `nesta_case6_c`



(b) Results summary of test system `nesta_case6_ww`

Figure 4.5: Results summary of six-bus test systems `nesta_case6_c` and `nesta_case6_ww`

time required for the proof.

AMP is the worst method because extensive time can be required for proving infeasibility. With system `nesta_case6_ww` at its 94% thermal limit, AMP is unsuccessful in concluding the known infeasibility within six hours. From the detailed results, it is clear that AMP iterates eight times before the time limited is reached. Further, the last two iterations need $\sim 20,000$ seconds to solve the PMR. As an adapted algorithm of AMP, ACPF-IP is able to shorten. AMP's time by nearly 50% for each test case. For this particular example with 94% thermal limit capacity, ACPF-IP used just 426 seconds with 20 iterations in total. The improvements are due to the joint effect of the PCR-ACPF formulation, dynamic partitioning variable selection, and automatic partition construction. These results suggest that improvement can be obtained with dedicated attention to improve the general framework of AMP. However, the time required it is still not acceptable when compared to the global solvers.

As we scale the size of the test systems, the performance of the two global solvers gives little to no conclusion on the problem feasibility. For the 14-bus system `nesta_case14_ieee` (Figure 4.6), ACPF-IP is the only method that concludes the entire tested range of congestion percentage infeasible. The performance of ACPF-IP is followed by COUENNE, which is able to prove infeasibility from the 44% thermal limit capacity level and below, but with much more time being required. AMP is the third best method for 40% thermal limit capacity and below as it has comparable computation time to COUENNE. BARON, the least favored method, provides little to no conclusion on infeasibility. In the region where only ACPF-IP is able to address the ACPF problem (thermal limit capacity range from 54% to 46%), AMP is not able to detect higher lower bounds when compared to BARON or COUENNE.

Finally, for the 24-bus system `nesta_case24_ieee_rts` (Figure 4.7), ACPF-IP again shows its ability to prove infeasibility with 66% thermal limit capacity and below, while the other methods are only able to capture the tail at 52% and below. We also note that the time required for concluding infeasibility by ACPF-IP is often less than three hours, which is one-half of the time limit. In cases where ACPF-IP is not able to prove infeasibility, it detect the best lower bound with regular frequency (Figure 4.7).

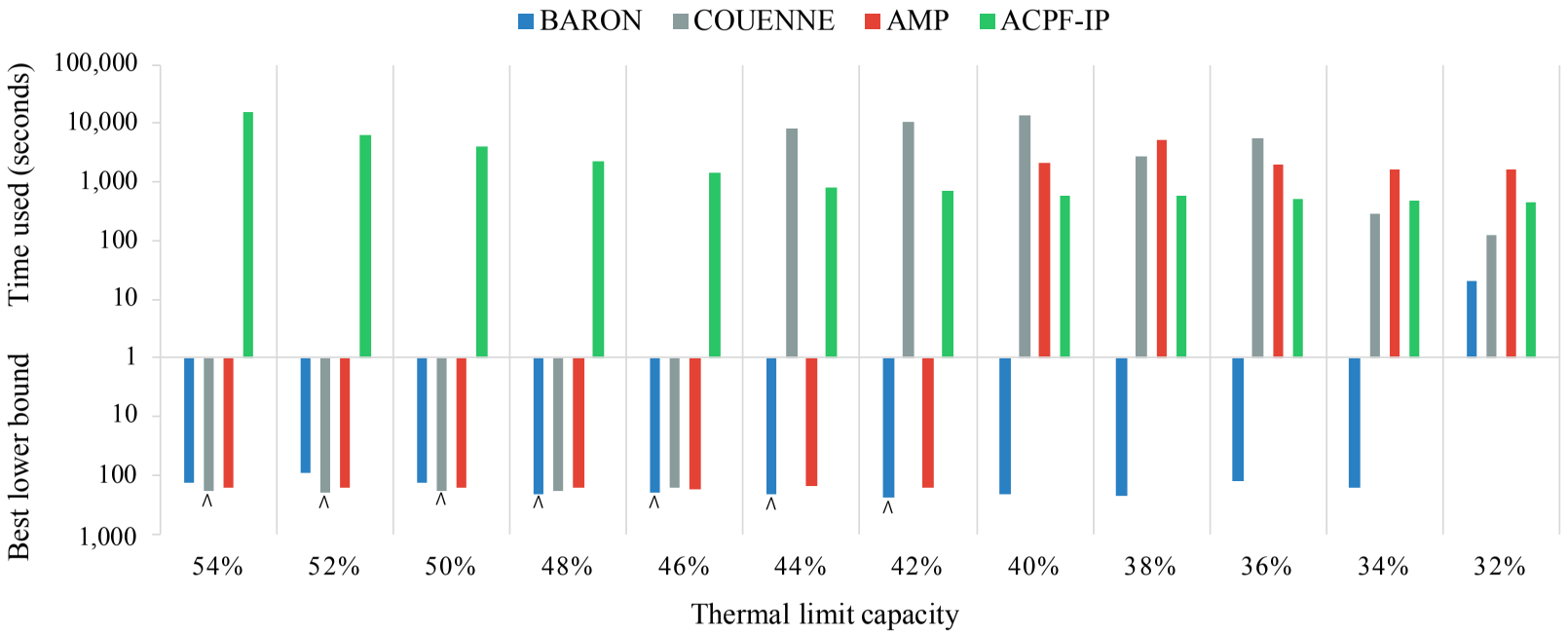


Figure 4.6: Results summary of a 14-bus test system `nesta_case14_ieee`

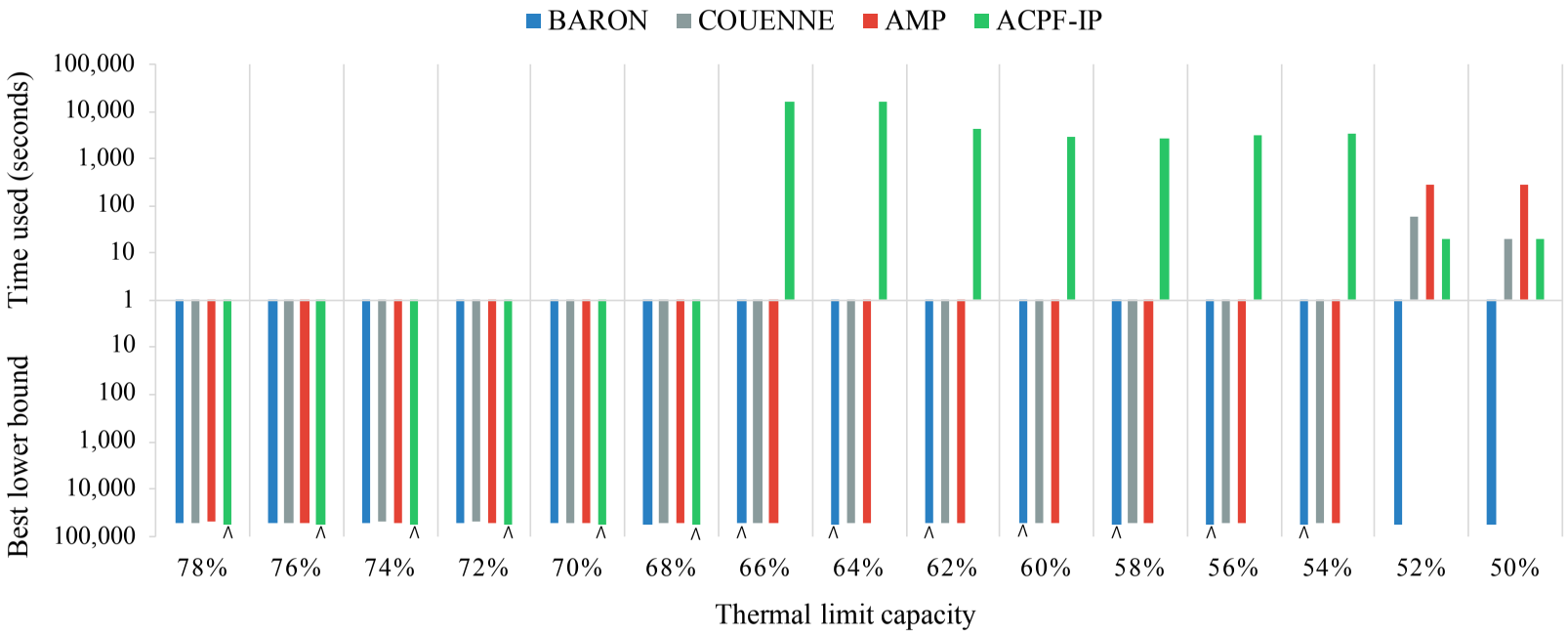


Figure 4.7: Results summary of a 24-bus test system `nesta_case24_ieee_rts`

4.7 Conclusions and Future Research

In this chapter, we propose an algorithm to analyze nonlinear, nonconvex rectangular ACPF problems where problem feasibility is not trivially revealed. A key idea to address ACPF problems is to prove the problem infeasible by tightening relaxations iteratively rather than exhaustively explore the solution space like most SB&B methods do. ACPF-IP applies a tight PCR method with dedicated valid inequalities to construct the PCR-ACPF that is dependent on discretized variable domains. Algorithm ACPF-IP also incorporates several novel schemes to construct discretized variable domains. The convergence of ACPF-IP is asserted as it will either find a feasible solution or terminate with a statement of ACPF infeasibility in finite number of iterations. Numerical experiments are conducted on test systems from the literature to compare the proposed ACPF-IP with state-of-the-art global solvers and AMP. Our results demonstrate the strength of ACPF-IP for analyzing challenging ACPF problems by showing ACPF-IP is able to prove the problem infeasible in a timely manner.

Although ACPF-IP is shown to outperform other methods as problem size scales, we recognize several interesting directions for future research. First, it is important to explore alternative formulations for ACPF problems, such as polar and current-voltage. This will involve a wider investigation on applying PCR for formulations that have general transcendental functions and making proper adjustments to ACPF-IP to fit different formulations appropriately. Second, we should investigate techniques to strengthen the PCR. Future direction is to incorporate semi-definite programming (SDP) techniques to improve algorithm tractability on large networks, given the promising results of Bai et al. [2008] and Gopalakrishnan et al. [2012]. Based on preliminary experiments, an effective SDP relaxation can help eliminate the red region in Section 4.3. However, as in this case with the 24-bus system `nesta_case24_ieee_rts`, SDP relaxations are still inconclusive for a 6%-wide test ranges. Finally, future research can expand the types of influences enforced on the system under study. Our experiments only considered congestion of the entire system, while in the real world, a variety of influences such as regional imbalances, network topology changes, tighter angle difference limitations, can and do exist.

Chapter 5

Conclusions and Future Research

Scientific evidence suggests there are evolving and uncertain risks to the electrical grid under a changing climate. Recognition of these emerging impacts can provide valuable insights in building a resilient system for the future. Climate adaptations can be made ahead of time to better prepare electrical grids for uncertain, extreme events. Developing effective climate adaptations is challenging due to how nature evolves, how uncertainty is modeled and represented, and how electrical grids respond to exogenous impacts. These underlying complications suggest that it is inadequate to individually consider either climate studies or intuition-based decision-making methods for actionable adaptations. In this dissertation, we developed a novel decision framework for climate adaptation to achieve a more resilient future.

5.1 Dissertation Summary

In the first phase of this dissertation, we focus on developing a stochastic optimization decision framework that links state-of-the-art climate simulation models with optimization models to plan climate adaptations that are resilient for an uncertain future. The decision framework is validated through extensive numerical experiments that show the proposed decision framework is capable of providing high-quality solutions when compared to a number of deterministic or heuristic approaches that are commonly considered in practice. Furthermore, we explore the flexibility of our decision framework to provide trade-off analyses between costs and uncertain risks.

In the second phase, we focus on addressing our decision framework's tractability through

an algorithmic approach. Our decision framework is essentially a class of discrete stochastic optimization models where the incorporated uncertainty is represented through a finite support set and risk is managed via a joint chance constraint that provides flexibility in the feasibility of the incorporated scenarios. The general form of the problem is decomposed into an anticipative and a recourse stage where the recourse stage is a pure feasibility problem; discrete decision variables exist in both stages. A scenario-based decomposition algorithm is proposed to solve this particular class of challenging optimization problems. The global convergence of the algorithm is ensured by leveraging relaxations and upper bounds of the generated configurations. Numerical experimental show that the algorithm can outperform CPLEX on two representative problems, including the climate adaptation problem in Chapter 2.

In the third phase of this dissertation, we focus on addressing the feasibility of an electrical transmission system under the influence of thermal limit congestion by proposing an algorithmic approach for nonlinear, nonconvex feasibility problems. These problems are frequently encountered if the proposed decision framework incorporates realistic modeling of system operations. As the challenge of the feasibility problem is well recognized, our main idea is to use tightening piecewise convex relaxations for an infeasibility proof, rather than pursuing an exhaustive search on solution space. Our proposed algorithm adaptively performs variable domain discretization and constructs tighter relaxations at every iteration. In addition, several novel algorithmic schemes are considered in the algorithm design that make further improvements. We assert the convergence of the algorithm as it either finds a feasible solution or terminates with the problem being proven as infeasible. Experimental results suggest the proposed algorithm is superior to two state-of-the-art global solvers, as well as a recently proposed global algorithm.

5.2 Future Research

A critical next research step is to explore more models for the climate adaptation problem. This incorporates three future research directions: 1) consider dedicated models to strengthen the existing problem formulation (see Song and Luedtke [2013], Song et al. [2014] as examples); 2) investigate analytical formulations that incorporate uncertainties and represent the concept of risk in a different manner; and 3) incorporate AC power flow in the climate adaptation problem in Chapter 2 for more realistic system operation modeling and investigate the applicability of the proposed

algorithm in Chapter 3 in combination with the infeasibility proof algorithm in Chapter 4. These three future research directions will help to address real-world problems and provide decision-makers with more actionable advice.

Additional experiments should be conducted to further investigate the role of uncertainty in the decision-making process. For example, it is curious to learn how adaptations respond when uncertain coastal basin erosion speed is considered in the simulation model. Moreover, it is important to understand how a long-term adaptation will form a closed-loop with the environment, which can further impact the decision process under a changing climate.

Finally, future research should pursue improvements in the proposed methodologies, which includes: 1) enlarging the scope of the algorithm in Chapter 3 by testing more problems with improved heuristics for high-quality configuration generation, or to develop specialized, problem-specific configuration generation schemes; 2) investigating the use of semi-definite programming techniques for ACPF-IP discussed in Chapter 4 for better tractability; and 3) performing experiments with a wider scope of system influences, such as regional imbalances, network topology changes, tighter angle difference limitations, etc., to find more interesting AC power flow problems to improve the overall applicability of algorithm in Chapter 4.

Appendices

Appendix A Comparison of Piece-wise Relaxation Formulations

In Chapter 4 Section 4.4.3, we compare the formulations of PCR, PCR-C, and PMR for the ACPF problem by conducting numerical experiments. The numerical performance of these formulations on general mixed-integer, nonlinear programs is evaluated in the experiments. The reader is referred to Nagarajan et al. [2017] for more details of the test instances under study. For each instance, we uniformly discretize the domains of a subset of variables into 20 partitions and construct the relaxation problem based on these formulations. The constructed problem is then solved using `Gurobi` solver with a one hour time limit.

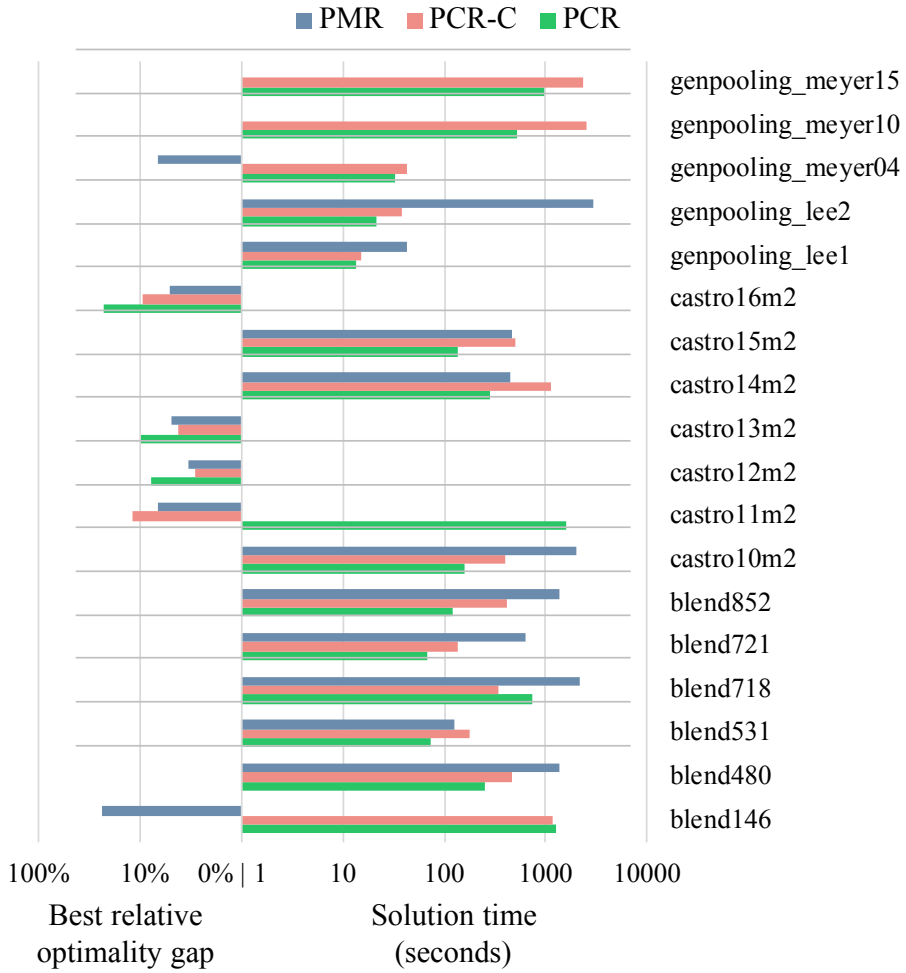


Figure A.1: Graph for constructing new partitions

The experimental setup is the same as described in Chapter 4 Section 4.6. The results are summarized in Figure A.1. From top to bottom, each block is associated with an instance while the experimental comparison is reflected through three bars representing the three formulations. The right side of the Figure A.1 measures the time required to solve the relaxation problem to optimality (a shorter bar is preferred). Given the time limit, it is possible for a problem to terminate without achieving optimality. In this case, we record the best optimality gap on the left side of Figure A.1 (shorter bar is preferred). Results show that PCR outperforms PCR-C and PMR in 16 of 18 instances, which leads to our decision to select PCR for ACPF problems.

Bibliography

N. Abi-Samra and W. Malcolm. Extreme weather effects on power systems. In *Power and Energy Society General Meeting, 2011 IEEE*, pages 1–5. IEEE, 2011.

Abraham Flaxman. Healthy algorithms: A simple optimization problem i dont know how to solve (from dcp), 2011. URL `https://healthyalgorithms.com/2011/06/02/a-simple-optimization-problem-i-dont-know-how-to-solve-from-dcp/`.

S. Ahmed. Solving chance-constrained stochastic programs via sampling and integer programming. In *State-of-the-Art Decision-Making Tools in the Information-Intensive Age*, pages 261–269. INFORMS, 2008.

S. Ahmed, A. J. King, and G. Parija. A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization*, 26(1):3–24, 2003.

S. Ahmed, M. Tawarmalani, and N. V. Sahinidis. A finite branch-and-bound algorithm for two-stage stochastic integer programs. *Mathematical Programming*, 100(2):355–377, 2004.

S. Ahmed, J. Luedtke, Y. Song, and W. Xie. Nonanticipative duality, relaxations, and formulations for chance-constrained stochastic programs. *Mathematical Programming*, pages 1–31, 2016.

N. Alguacil, A. L. Motto, and A. J. Conejo. Transmission expansion planning: A mixed-integer lp approach. *IEEE Transactions on Power Systems*, 18(3):1070–1077, 2003.

M. R. AlRashidi and M. E. El-Hawary. A survey of particle swarm optimization applications in electric power systems. *IEEE transactions on evolutionary computation*, 13(4):913–918, 2009.

X. Bai, H. Wei, K. Fujisawa, and Y. Wang. Semidefinite programming for optimal power flow problems. *International Journal of Electrical Power & Energy Systems*, 30(6-7):383–392, 2008.

P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter. Branching and bounds tighteningtechniques for non-convex minlp. *Optimization Methods & Software*, 24(4-5):597–634, 2009.

A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust optimization*. Princeton University Press, 2009.

D. Bertsimas and M. Sim. The price of robustness. *Operations research*, 52(1):35–53, 2004.

D. Bienstock and S. Mattia. Using mixed-integer programming to solve power grid blackout problems. *Discrete Optimization*, 4(1):115–141, 2007.

D. Bienstock and J. F. Shapiro. Optimizing resource acquisition decisions by stochastic programming. *Management Science*, 34(2):215–229, 1988.

D. Bienstock and A. Verma. Strong np-hardness of ac power flows feasibility. *arXiv preprint arXiv:1512.07315*, 2015.

R. Bierkandt, M. Auffhammer, and A. Levermann. Us power plant sites at risk of future sea-level rise. *Environmental Research Letters*, 10(12):124022, 2015.

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

M. Bloomberg. A stronger, more resilient new york. *City of New York, PlaNYC Report*, 2013.

F. Boukouvala, R. Misener, and C. A. Floudas. Global optimization advances in mixed-integer nonlinear programming, minlp, and constrained derivative-free optimization, cdfo. *European Journal of Operational Research*, 252(3):701–727, 2016.

R. H. Byrd, J. Nocedal, and R. A. Waltz. Knitro: An integrated package for nonlinear optimization. In *Large-scale nonlinear optimization*, pages 35–59. Springer, 2006.

M. B. Cain, R. P. Oneill, and A. Castillo. History of optimal power flow and formulations. *Federal Energy Regulatory Commission*, pages 1–36, 2012.

J. Carpentier. Contribution to the economic dispatch problem. *Bulletin de la Societe Francoise des Electriciens*, 3(8):431–447, 1962.

A. Castillo and R. P. ONeill. Survey of approaches to solving the acopf (opf paper 4). *US Federal Energy Regulatory Commission, Tech. Rep*, 2013.

C. Coffrin and P. Van Hentenryck. A linear-programming approximation of ac power flows. *INFORMS Journal on Computing*, 26(4):718–734, 2014.

C. Coffrin, P. Van Hentenryck, and R. Bent. Strategic stockpiling of power system supplies for disaster recovery. In *Power and Energy Society General Meeting, 2011 IEEE*, pages 1–8. IEEE, 2011.

C. Coffrin, D. Gordon, and P. Scott. Nesta, the nicta energy system test case archive. *arXiv preprint arXiv:1411.0359*, 2014.

C. Coffrin, H. L. Hijazi, and P. Van Hentenryck. The qc relaxation: A theoretical and computational study on optimal power flow. *IEEE Transactions on Power Systems*, 31(4):3008–3018, 2016.

C. Coffrin, R. Bent, K. Sundar, Y. Ng, and M. Lubin. Powermodels. jl: An open-source framework for exploring power flow formulations. *arXiv preprint arXiv:1711.01728*, 2017a.

C. Coffrin, H. L. Hijazi, and P. Van Hentenryck. Strengthening the sdp relaxation of ac power flows with convex envelopes, bound tightening, and valid inequalities. *IEEE Transactions on Power Systems*, 32(5):3549–3558, 2017b.

R. A. Collado, D. Papp, and A. Ruszczyński. Scenario decomposition of risk-averse multistage stochastic programming problems. *Annals of Operations Research*, 200(1):147–170, 2012.

E. M. Constantinescu, V. M. Zavala, M. Rocklin, S. Lee, and M. Anitescu. A computational framework for uncertainty quantification and stochastic optimization in unit commitment with wind power generation. *IEEE Transactions on Power Systems*, 26(1):431–441, 2011.

J. Cook, N. Oreskes, P. T. Doran, W. R. Anderegg, B. Verheggen, E. W. Maibach, J. S. Carlton, S. Lewandowsky, A. G. Skuce, S. A. Green, et al. Consensus on consensus: a synthesis of consensus estimates on human-caused global warming. *Environmental Research Letters*, 11(4):048002, 2016.

E. L. da Silva, H. A. Gil, and J. M. Areiza. Transmission network expansion planning under an improved genetic algorithm. In *Power Industry Computer Applications, 1999. PICA'99. Proceedings of the 21st 1999 IEEE International Conference*, pages 315–321. IEEE, 1999.

G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations research*, 8 (1):101–111, 1960.

R. M. DeConto and D. Pollard. Contribution of antarctica to past and future sea-level rise. *Nature*, 531(7596):591–597, 2016.

Department of Defense. 2014 climate change adaptation roadmap, 2014. URL `http://ppec.asme.org/wp-content/uploads/2014/10/CCARprint.pdf`.

A. H. Escobar, R. A. Gallego, and R. Romero. Multistage and coordinated planning of the expansion of transmission systems. *IEEE Transactions on Power Systems*, 19(2):735–744, 2004.

Executive Office of the President. Economic benefits of increasing electric grid resilience to weather outages, 2013. URL `https://energy.gov/sites/prod/files/2013/08/f2/Grid%20Resiliency%20Report_FINAL.pdf`.

M. Fischetti, F. Glover, and A. Lodi. The feasibility pump. *Mathematical Programming*, 104(1): 91–104, 2005.

R. A. Francis, S. M. Falconi, R. Nateghi, and S. D. Guikema. Probabilistic life cycle analysis model for evaluating electric power infrastructure risk mitigation investments. *Climatic change*, 106(1): 31–55, 2011.

H. Gangammanavar, S. Sen, and V. M. Zavala. Stochastic optimization of sub-hourly economic dispatch with wind energy. *IEEE Transactions on Power Systems*, 31(2):949–959, 2016.

A. Gopalakrishnan, A. U. Raghunathan, D. Nikovski, and L. T. Biegler. Global optimization of optimal power flow using a branch & bound algorithm. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pages 609–616. IEEE, 2012.

B. Gorenstin, N. Campodonico, J. Costa, and M. Pereira. Power system expansion planning under uncertainty. *IEEE Transactions on Power Systems*, 8(1):129–136, 1993.

D. G. Groves and C. Sharon. Planning tool to support planning the future of coastal louisiana. *Journal of Coastal Research*, 67(sp1):147–161, 2013.

O. Guieu and J. W. Chinneck. Analyzing infeasible mixed-integer and integer linear programs. *INFORMS Journal on Computing*, 11(1):63–77, 1999.

W. D. Harvey and M. L. Ginsberg. Limited discrepancy search. In *IJCAI (1)*, pages 607–615, 1995.

J. L. Higle. Stochastic programming: optimization when uncertainty matters. In *Emerging Theory, Methods, and Applications*, pages 30–53. Informs, 2005.

R. Horst and H. Tuy. *Global optimization: Deterministic approaches*. Springer Science & Business Media, 2013.

R. A. Jabr. Radial distribution load flow using conic programming. *IEEE transactions on power systems*, 21(3):1458–1459, 2006.

C. P. Jelesnianski, J. Chen, and W. A. Shaffer. *SLOSH: Sea, lake, and overland surges from hurricanes*, volume 48. US Department of Commerce, National Oceanic and Atmospheric Administration, National Weather Service, 1992.

P. Jirutitijaroen and C. Singh. Reliability constrained multi-area adequacy planning using stochastic programming with sample-average approximations. *IEEE Transactions on Power Systems*, 23(2): 504–513, 2008.

J. R. Kasprzyk, S. Nataraj, P. M. Reed, and R. J. Lempert. Many objective robust decision making for complex environmental systems undergoing change. *Environmental Modelling & Software*, 42: 55–71, 2013.

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

R. E. Kopp, A. C. Kemp, K. Bittermann, B. P. Horton, J. P. Donnelly, W. R. Gehrels, C. C. Hay, J. X. Mitrovica, E. D. Morrow, and S. Rahmstorf. Temperature-driven global sea-level variability in the common era. *Proceedings of the National Academy of Sciences*, 113(11):E1434–E1441, 2016.

K. Lehmann, A. Grastien, and P. Van Hentenryck. Ac-feasibility on tree networks is np-hard. *IEEE Transactions on Power Systems*, 31(1):798–801, 2016.

N. Lin, K. Emanuel, M. Oppenheimer, and E. Vanmarcke. Physically based assessment of hurricane surge threat under climate change. *Nature Climate Change*, 2(6):462–467, 2012.

N. Lin, R. E. Kopp, B. P. Horton, and J. P. Donnelly. Hurricane sandys flood frequency increasing from year 1800 to 2100. *Proceedings of the National Academy of Sciences*, page 201604386, 2016.

C. M. Little, R. M. Horton, R. E. Kopp, M. Oppenheimer, G. A. Vecchi, and G. Villarini. Joint projections of us east coast sea level and storm surge. *Nature Climate Change*, 5(12):1114–1120, 2015.

X. Liu, S. Küçükyavuz, and J. Luedtke. Decomposition algorithms for two-stage chance-constrained programs. *Mathematical Programming*, 157(1):219–243, 2016.

J. Á. López, K. Ponnambalam, and V. H. Quintana. Generation and transmission expansion under risk using stochastic programming. *IEEE Transactions on Power Systems*, 22(3):1369–1378, 2007.

S. H. Low. Convex relaxation of optimal power flowpart i: Formulations and equivalence. *IEEE Transactions on Control of Network Systems*, 1(1):15–27, 2014a.

S. H. Low. Convex relaxation of optimal power flowpart ii: Exactness. *IEEE Transactions on Control of Network Systems*, 1(2):177–189, 2014b.

M. Lubin and I. Dunning. Computing in operations research using julia. *INFORMS Journal on Computing*, 27(2):238–248, 2015. doi: 10.1287/ijoc.2014.0623.

J. Luedtke. A branch-and-cut decomposition algorithm for solving chance-constrained mathematical programs with finite support. *Mathematical Programming*, 146(1-2):219–244, 2014.

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

J. Luedtke, S. Ahmed, and G. L. Nemhauser. An integer programming approach for linear programs with probabilistic constraints. *Mathematical programming*, 122(2):247–272, 2010.

W.-K. Mak, D. P. Morton, and R. K. Wood. Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations research letters*, 24(1):47–56, 1999.

C. U. Manual. Ibm ilog cplex optimization studio, 1987.

D. McInerney, R. Lempert, and K. Keller. What are robust strategies in the face of uncertain climate threshold responses? *Climatic change*, 112(3-4):547–568, 2012.

D. K. Molzahn. Computing the feasible spaces of optimal power flow problems. *IEEE Transactions on Power Systems*, 32(6):4752–4763, 2017.

D. K. Molzahn and I. A. Hiskens. Moment-based relaxation of the optimal power flow problem. In *Power Systems Computation Conference (PSCC), 2014*, pages 1–7. IEEE, 2014.

D. K. Molzahn, B. C. Lesieutre, and C. L. DeMarco. A sufficient condition for global optimality of solutions to the optimal power flow problem. *IEEE Transactions on Power Systems*, 29(2): 978–979, 2014.

J. M. Morales, A. J. Conejo, and J. Pérez-Ruiz. Economic valuation of reserves in power systems with high penetration of wind power. *IEEE Transactions on Power Systems*, 24(2):900–910, 2009.

R. Moreno, D. Pudjianto, and G. Strbac. Transmission network investment with probabilistic security and corrective control. *IEEE Transactions on Power Systems*, 28(4):3935–3944, 2013.

H. Nagarajan, E. Yamangil, R. Bent, P. Van Hentenryck, and S. Backhaus. Optimal resilient transmission grid design. In *Power Systems Computation Conference (PSCC), 2016*, pages 1–7. IEEE, 2016.

H. Nagarajan, M. Lu, S. Wang, R. Bent, and K. Sundar. An adaptive, multivariate partitioning algorithm for global optimization of nonconvex programs. *arXiv preprint arXiv:1707.02514*, 2017.

M. R. Narimani, D. K. Molzahn, and M. L. Crow. Improving qc relaxations of opf problems via voltage magnitude difference constraints and envelopes for trilinear monomials. *arXiv preprint arXiv:1804.02735*, 2018.

R. Nateghi, S. D. Guikema, and S. M. Quiring. Comparison and validation of statistical methods for predicting power outage durations in the event of hurricanes. *Risk analysis*, 31(12):1897–1906, 2011.

R. Nateghi, S. Guikema, and S. M. Quiring. Power outage estimation for tropical cyclones: Improved accuracy with simpler models. *Risk analysis*, 34(6):1069–1078, 2014.

National Hurricane Center-Storm Surge Unit. Introduction to storm surge overview, 2013. URL `http://www.nhc.noaa.gov/surge/surge_intro.pdf`.

A. Nemirovski and A. Shapiro. Scenario approximations of chance constraints. In *Probabilistic and randomized methods for design under uncertainty*, pages 3–47. Springer, 2006.

R. J. Nicholls and A. Cazenave. Sea-level rise and its impact on coastal zones. *science*, 328(5985): 1517–1520, 2010.

S. Nickel, M. Reuter-Oppermann, and F. Saldanha-da Gama. Ambulance location under stochastic demand: A sampling approach. *Operations Research for Health Care*, 8:24–32, 2016.

S. of Louisiana. Louisianas comprehensive master plan for a sustainable coast, 2017. URL `http://coastal.la.gov/wp-content/uploads/2017/01/DRAFT-2017-Coastal-Master-Plan.pdf`.

Office of Electricity Delivery and Energy Reliability Smart Grid R&D Program. Summary Report : 2014 DOE Resilient Electric Distribution Grid R & D Workshop. Technical report, 2014.

G. Optimization. Inc.,gurobi optimizer reference manual, 2015. *Google Scholar*, 2014.

Organization for Security and Co-operation in Europe. Protecting Electricity Networks from Natural Hazards. Technical report, Vienna, Austria, 2016.

B. Pagnoncelli, S. Ahmed, and A. Shapiro. Sample average approximation method for chance constrained programming: theory and applications. *Journal of optimization theory and applications*, 142(2):399–416, 2009.

F. S. Paolo, H. A. Fricker, and L. Padman. Volume loss from antarctic ice shelves is accelerating. *Science*, 348(6232):327–331, 2015.

A. Papavasiliou, S. S. Oren, and R. P. O'Neill. Reserve requirements for wind power integration: A scenario-based stochastic programming framework. *IEEE Transactions on Power Systems*, 26(4): 2197–2206, 2011.

B. Park, L. Tang, M. C. Ferris, and C. L. DeMarco. Examination of three different acopf formulations with generator capability curves. *IEEE Transactions on Power Systems*, 32(4):2913–2923, 2017.

D. Pasqualini. A statistical approach for modeling tropical cyclones. synthetic hurricanes generator model. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2016.

V. Pillac, P. Van Hentenryck, and C. Even. A conflict-based path-generation heuristic for evacuation planning. *Transportation Research Part B: Methodological*, 83:136–150, 2016.

PJM. 2017 regional transmission expansion plan process scope and input assumptions, 2017. URL `http://www.pjm.com/-/media/library/reports-notices/2017-rtep/20170731-rtep-input-assumptions-and-scope-whitepaper.ashx?la=en`.

J. E. Price. Evaluation of stochastic unit commitment for renewable integration in california's energy markets. In *Power & Energy Society General Meeting, 2015 IEEE*, pages 1–5. IEEE, 2015.

Y. Puranik and N. V. Sahinidis. Domain reduction techniques for global NLP and MINLP optimization. *Constraints*, 22(3):338–376, 2017a. ISSN 15729354. doi: 10.1007/s10601-016-9267-5.

Y. Puranik and N. V. Sahinidis. Deletion presolve for accelerating infeasibility diagnosis in optimization models. *INFORMS Journal on Computing*, 29(4):754–766, 2017b.

K. Purchala, L. Meeus, D. Van Dommelen, and R. Belmans. Usefulness of dc power flow for active power flow analysis. In *Power Engineering Society General Meeting, 2005. IEEE*, pages 454–459. IEEE, 2005.

J. Qiu, Z. Y. Dong, J. Zhao, Y. Xu, F. Luo, and J. Yang. A risk-based approach to multi-stage probabilistic transmission network planning. *IEEE Transactions on Power Systems*, 31(6):4867–4876, 2016a.

J. Qiu, H. Yang, Z. Y. Dong, J. Zhao, F. Luo, M. Lai, and K. P. Wong. A probabilistic transmission planning framework for reducing network vulnerability to extreme events. *IEEE Transactions on Power Systems*, 31(5):3829–3839, 2016b.

M. Rahmani, R. Romero, and M. J. Rider. Strategies to reduce the number of variables and the combinatorial search space of the multistage transmission expansion planning problem. *IEEE Transactions on Power Systems*, 28(3):2164–2173, 2013.

T. K. Ralphs and A. Hassanzadeh. A generalization of benders algorithm for two-stage stochastic optimization problems with mixed integer recourse. Technical report, Tech. rep., Technical Report 14T-005, Department of Industrial and Systems Engineering, Lehigh University, 2014.

N. Ranger. Adaptation as a decision making under deep uncertainty. In *Climate*, pages 89–122. Springer, 2011.

K. W. Ross and D. H. Tsang. The stochastic knapsack problem. *IEEE Transactions on communications*, 37(7):740–747, 1989.

E. J. Russo, K. A. Burks-Copes, et al. Adaptive capacity and tipping points of coastal military installation sustainability with storms and rising seas. In *Coasts and Ports 2013: 21st Australasian Coastal and Ocean Engineering Conference and the 14th Australasian Port and Harbour Conference*, page 659. Engineers Australia, 2013.

A. Ruszczyński. Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra. *Mathematical Programming*, 93(2):195–215, 2002.

S. Schmidtko, K. J. Heywood, A. F. Thompson, and S. Aoki. Multidecadal warming of antarctic waters. *Science*, 346(6214):1227–1231, 2014.

S. Sen. Relaxations for probabilistically constrained programs with discrete random variables. *Operations Research Letters*, 11(2):81–86, 1992.

S. Sen, L. Yu, and T. Genc. A stochastic programming approach to power portfolio optimization. *Operations Research*, 54(1):55–72, 2006.

A. Shapiro, D. Dentcheva, et al. *Lectures on stochastic programming: modeling and theory*, volume 16. Siam, 2014.

J. Simm, A. Guise, D. Robbins, and J. Engle. Us north atlantic coast comprehensive study: resilient adaptation to increasing risk. Technical report, 2015.

E. M. Smith and C. C. Pantelides. A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex minlps. *Computers & Chemical Engineering*, 23(4-5): 457–478, 1999.

Y. Song and J. R. Luedtke. Branch-and-cut approaches for chance-constrained formulations of reliable network design problems. *Mathematical Programming Computation*, 5(4):397–432, 2013.

Y. Song, J. R. Luedtke, and S. Küçükyavuz. Chance-constrained binary packing problems. *INFORMS Journal on Computing*, 26(4):735–747, 2014.

A. Staid, S. D. Guikema, R. Nateghi, S. M. Quiring, and M. Z. Gao. Simulation of tropical cyclone impacts to the us power system under climate change scenarios. *Climatic change*, 127(3-4):535–546, 2014.

B. Stott, J. Jardim, and O. Alsaç. Dc power flow revisited. *IEEE Transactions on Power Systems*, 24(3):1290–1300, 2009.

K. Sundar, H. Nagarajan, S. Wang, J. Linderoth, and R. Bent. Tight piecewise polyhedral relaxations of multilinear terms. *To be submitted to Operations Research Letters*, 2018.

M. W. Tanner and L. Ntaimo. Iis branch-and-cut for joint chance-constrained stochastic programs and application to optimal vaccine allocation. *European Journal of Operational Research*, 207(1): 290–296, 2010.

M. W. Tanner, L. Sattenspiel, and L. Ntaimo. Finding optimal vaccination strategies under parameter uncertainty using stochastic programming. *Mathematical biosciences*, 215(2):144–151, 2008.

M. Tawarmalani and N. V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103(2):225–249, 2005.

C. Tebaldi, B. H. Strauss, and C. E. Zervas. Modelling sea level rise impacts on storm surges along us coasts. *Environmental Research Letters*, 7(1):014032, 2012.

S. Teimourzadeh and F. Aminifar. Milp formulation for transmission expansion planning with short-circuit level constraints. *IEEE Transactions on Power Systems*, 31(4):3109–3118, 2016.

D. Ton and W.-t. P. Wang. A More Resilient Grid. *IEEE Power & Energy Magazine*, (April):26–34, 2015.

A. Tsoukalas and A. Mitsos. Multivariate mccormick relaxations. *Journal of Global Optimization*, 59(2-3):633–662, 2014.

F. Ugranli and E. Karatepe. Transmission expansion planning for wind turbine integrated power systems considering contingency. *IEEE Transactions on Power Systems*, 31(2):1476–1485, 2016.

United States Department of Energy. U.S. Energy Sector Vulnerabilities to Climate Change and Extreme Weather. Technical report, 2013.

R. M. Van Slyke and R. Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.

M. T. Van Vliet, J. R. Yearsley, F. Ludwig, S. Vögele, D. P. Lettenmaier, and P. Kabat. Vulnerability of us and european electricity supply to climate change. *Nature Climate Change*, 2(9):676–681, 2012.

A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.

W. E. Walker, M. Haasnoot, and J. H. Kwakkel. Adapt or perish: a review of planning approaches for adaptation under deep uncertainty. *Sustainability*, 5(3):955–979, 2013.

S. W. Wallace and S.-E. Fleten. Stochastic programming models in energy. *Handbooks in operations research and management science*, 10:637–677, 2003.

S. Wang et al. Energy infrastructure adaptation framework toward storm surges under rising sea level. *Work in progress*, 2017, submitted.

P. J. Webster, G. J. Holland, J. A. Curry, and H.-R. Chang. Changes in tropical cyclone number, duration, and intensity in a warming environment. *Science*, 309(5742):1844–1846, 2005.

A. J. Wood and B. F. Wollenberg. *Power generation, operation, and control*. John Wiley & Sons, 2012.

J. D. Woodruff, J. L. Irish, and S. J. Camargo. Coastal flooding by tropical cyclones and sea-level rise. *Nature*, 504(7478):44–52, 2013.

E. Yamangil, R. Bent, and S. Backhaus. Designing resilient electrical distribution grids. *arXiv preprint arXiv:1409.4477*, 2014.

E. Yamangil, R. Bent, and S. Backhaus. Resilient upgrade of electrical distribution grids. In *AAAI*, pages 1233–1240, 2015.

D. Yates, B. Q. Luna, R. Rasmussen, D. Bratcher, L. Garre, F. Chen, M. Tewari, and P. Friis-Hansen. Stormy weather: Assessing climate change hazards to electric power infrastructure: A sandy case study. *IEEE Power and Energy Magazine*, 12(5):66–75, 2014.

H. Yu, C. Chung, K. Wong, and J. Zhang. A chance constrained transmission network expansion planning method with consideration of load and wind farm uncertainties. *IEEE Transactions on Power Systems*, 24(3):1568–1576, 2009.

Y. Yuan, Z. Li, and B. Huang. Robust optimization approximation for joint chance constrained optimization problem. *Journal of Global Optimization*, pages 1–23, 2015.

H. Zhang, G. T. Heydt, V. Vittal, and H. D. Mittelmann. Transmission expansion planning using an ac model: formulations and possible relaxations. In *Power and Energy Society General Meeting, 2012 IEEE*, pages 1–8. IEEE, 2012a.

H. Zhang, V. Vittal, G. T. Heydt, and J. Quintero. A mixed-integer linear programming approach for multi-stage security-constrained transmission expansion planning. *IEEE Transactions on Power Systems*, 27(2):1125–1133, 2012b.

H. Zhang, G. T. Heydt, V. Vittal, and J. Quintero. An improved network model for transmission expansion planning considering reactive power and network losses. *IEEE Transactions on Power Systems*, 28(3):3471–3479, 2013.

M. Zhang, S. Küçükyavuz, and S. Goel. A branch-and-cut method for dynamic decision making under joint chance constraints. *Management Science*, 60(5):1317–1333, 2014.