

# Optimization Algorithms for Power Grid Planning and Operational Problems

by

Kathryn M. Schumacher

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Industrial and Operations Engineering)  
in The University of Michigan  
2014

Doctoral Committee:

Associate Professor Amy E.M. Cohn, Chair  
Richard Li-Yang Chen, Sandia National Laboratories  
Professor Mark S. Daskin  
Associate Professor Marina A. Epelman  
Professor Ian A. Hiskens  
Assistant Professor Siqian Shen

© Kathryn Schumacher 2014  

---

All Rights Reserved

## ACKNOWLEDGEMENTS

I would foremost like to thank my advisor Prof. Amy Cohn. Her continual encouragement, enthusiasm, and support made this work possible. She taught me about the art and science of formulating optimization models, advised me to never lose sight of the “big picture”, and pushed me to clearly communicate my ideas. I am eternally grateful for her mentorship.

Dr. Richard Chen has also been an incredibly helpful mentor. He has taught me a great deal about power systems optimization, and he has been a fantastic co-author. I very much appreciate his support and guidance over the years.

Many thanks are also due to my committee members Prof. Siqian Shen, Prof. Marina Epelman, Prof. Mark Daskin and Prof. Ian Hiskens. Their feedback and perspective has helped to shape this thesis.

I am grateful for the support of all my family and friends. I thank Katharina Ley Best for being an amazing officemate for over four years. I thank my parents for their love and encouragement, and for making me the nerd that I am. Last but certainly not least, I thank my husband Johannes Strom for helping with my programming problems, laughing at my jokes and believing in me.

# TABLE OF CONTENTS

<b>ACKNOWLEDGEMENTS</b> . . . . .	ii
<b>LIST OF FIGURES</b> . . . . .	vi
<b>LIST OF TABLES</b> . . . . .	vii
<b>LIST OF APPENDICES</b> . . . . .	viii
<b>ABSTRACT</b> . . . . .	ix
<b>CHAPTER</b>	
<b>I. Introduction</b> . . . . .	1
<b>II. Chance-Constrained Network Capacity Design with Stochastic Demands with Finite Support</b> . . . . .	5
2.1 Introduction . . . . .	5
2.2 Literature Review and Contributions . . . . .	7
2.3 Robust Capacity Design Problem . . . . .	10
2.3.1 Notation . . . . .	10
2.3.2 Network Flow Formulation . . . . .	11
2.3.3 Cut-set Solution Approach . . . . .	11
2.3.4 Implementation Details . . . . .	13
2.3.5 Run Time Results . . . . .	15
2.4 $\alpha$ -Satisfied Capacity Design Problem . . . . .	17
2.4.1 MIP Formulation . . . . .	18
2.4.2 Combinatorial Tree Algorithm . . . . .	19
2.4.3 Greedy Algorithm . . . . .	27
2.4.4 Heuristic Computational Results . . . . .	28
2.5 Conclusion . . . . .	30
<b>III. N-<math>k</math> Secure Unit Commitment with Transmission Switching</b>	31

3.1	Introduction . . . . .	31
3.2	Literature Review . . . . .	34
3.3	Problem Definition . . . . .	37
	3.3.1 Assumptions . . . . .	37
	3.3.2 Explicit Formulation . . . . .	40
3.4	Problem Decomposition . . . . .	43
	3.4.1 Natural Two-Stage Decomposition . . . . .	43
	3.4.2 Reformulation and Cutting Plane Algorithm . . . . .	45
3.5	Contingency Oracle . . . . .	49
	3.5.1 Bilevel Program . . . . .	50
	3.5.2 Bilevel Program Without Lower Level Switching Decisions . . . . .	51
	3.5.3 Contingency Oracle Solution Routine . . . . .	54
3.6	Implementation Details . . . . .	56
	3.6.1 Identifying Unsurvivable Contingencies . . . . .	56
	3.6.2 Ordered Time Periods . . . . .	57
3.7	Computational Results . . . . .	57
	3.7.1 Run Times . . . . .	58
	3.7.2 Critical $\epsilon$ Analysis . . . . .	60
	3.7.3 Scaled-Load Analysis . . . . .	61
	3.7.4 Line Removal Analysis . . . . .	65
3.8	Conclusion . . . . .	66

**IV. Transmission Expansion with Smart Switching Under Demand Uncertainty and Line Failures . . . . . 68**

4.1	Introduction . . . . .	68
4.2	Literature Review . . . . .	70
4.3	Problem Definition . . . . .	75
	4.3.1 Assumptions . . . . .	75
	4.3.2 Deterministic Problem . . . . .	78
	4.3.3 Robust Counterpart . . . . .	80
4.4	Decomposition . . . . .	84
	4.4.1 Switching Variable Generation . . . . .	84
	4.4.2 Oracle Motivation . . . . .	86
	4.4.3 Cutting Plane Algorithm . . . . .	86
4.5	Oracle Development . . . . .	87
	4.5.1 Bilinear reformulation . . . . .	88
	4.5.2 Iterative Oracle Routine . . . . .	91
4.6	Implementation . . . . .	94
	4.6.1 Implementation Details . . . . .	94
	4.6.2 Complete Algorithm . . . . .	95
4.7	Computational Results . . . . .	97
4.8	Conclusion . . . . .	103

V. Conclusion . . . . .	105
APPENDICES . . . . .	109
BIBLIOGRAPHY . . . . .	126

## LIST OF FIGURES

### Figure

2.1	CSCG Algorithm Run Time for IEEE118 and exh30 test systems . . .	17
2.2	Combinatorial Tree . . . . .	20
3.1	Illustration of IEEE14 test system. Bold lines are buses, regular lines are transmission lines, and circles with “G” inside are generators. . .	38
3.2	Algorithm Overview for N− <i>k</i> UC with Transmission Switching . . .	47
3.3	Contingency Oracle Solution Routine . . . . .	55
3.4	RTS-96 test system, from <i>Kamwa et al. (2007)</i> . . . . .	62
3.5	IEEE24, <i>k</i> =1, Optimal UC Cost at Different Load Levels . . . . .	63
3.6	IEEE24, <i>k</i> =2, Optimal UC Cost at Different Load Levels . . . . .	63
3.7	IEEE24, <i>k</i> =1 & <i>k</i> =2, Optimal UC Cost at Different Load Levels . .	64
4.1	Algorithm Overview for Transmission Expansion with Switching . .	87
4.2	Oracle Routine . . . . .	92
4.3	Optimal Investment Solution for Garver6 without Switching. . . .	100
4.4	Optimal Investment Solution for Garver6 with Switching. . . . .	101
4.5	Optimal Investment Cost for IEEE24 with Scaled Demand . . . . .	102
4.6	Optimal Investment Cost for IEEE14 with Scaled Demand . . . . .	102
4.7	Optimal Investment Cost for IEEE24 with Scaled Line Capacities .	103

## LIST OF TABLES

### Table

2.1	Number of Reduced Combinatorial Tree Nodes . . . . .	26
2.2	$\alpha$ SCD Heuristic Evaluation . . . . .	29
2.3	Greedy and RCT Algorithm Run Times (sec) . . . . .	30
3.1	IEEE24 and RTS-96 System Characteristics . . . . .	58
3.2	IEEE24 and RTS-96 Critical $\epsilon_k$ Values . . . . .	59
3.3	$N-k$ UC Constraint Generation Algorithm Run Times . . . . .	59
4.1	Garver6, IEEE14 and IEEE24 System Characteristics . . . . .	98
4.2	TEP Algorithm Run Times and Performance Metrics . . . . .	99



**LIST OF APPENDICES**

**Appendix**

- A. Explicit Formulation of  $N-k$  Unit Commitment Model with Transmission Switching . . . . . 110
- B. Explicit Formulation of Deterministic Transmission Expansion Model with Transmission Switching . . . . . 119

# ABSTRACT

Optimization Algorithms for Power Grid Planning and Operational Problems

by

Kathryn Schumacher

Chair: Amy Cohn

The modern electrical grid is an engineering marvel. The power grid is an incredibly complex system that largely functions very reliably. However, aging infrastructure and changing power consumption and generation trends will necessitate that new investments be made and new operational regimes be explored to maintain this level of reliability. One of the primary difficulties in power grid planning is the presence of uncertainty. In this thesis, we address short-term (i.e., day-ahead) and long-term power system planning problems where there is uncertainty in the forecasted demand for power, future renewable generation levels, and/or possible component failures. We initially consider a network capacity design problem where there is uncertainty in the nodal supplies and demands. This robust single-commodity network design problem underlies several applications including power transmission networks. Minimum cost capacity expansion decisions are made to ensure that there exists a feasible network flow solution for  $\alpha\%$  of the demand scenarios in the given set, where  $\alpha$  is a parameter specified by the user. We next consider a day-ahead planning problem that is specifically applicable to the power grid. We present an extension of the traditional unit

commitment problem where we additionally consider (1) a more stringent security requirement and (2) a more flexible set of recovery actions. We require that feasible operation is possible for any simultaneous failure of  $k$  generators and/or transmission lines (i.e.,  $N-k$  security), and transmission switching may be used to recover from a failure event. Finally, we consider a transmission expansion planning problem where there is uncertainty in future loads, renewal generation outputs and line failures, and transmission switching is also allowed as a recovery action. We propose a robust optimization model where feasible operation is required for all loads and renewable generation levels within given ranges, and for all single transmission line failures. For all three of these problems, novel algorithms are presented that enable these problems to be solved even when straight-forward formulations are too large to be tractable. Computational results are presented for each algorithm to provide insight into the advantages and limitations of these algorithms in practice

# CHAPTER I

## Introduction

The modern electrical grid is an engineering marvel. Electrification was selected as the greatest engineering achievement of the 20th century by the National Academy of Engineering (*Constable and Somerville* (2003)). The power grid is an incredibly complex system that largely functions very reliably. However, aging infrastructure, a growing population, the possibility of a terrorist attack on the grid, increased renewable penetration, and tightening budgets may threaten the level of reliability that most of us have come to expect. From an operations research perspective, these challenges present many interesting and important optimization problems. Operations research techniques can be used to identify ways to intelligently invest in new infrastructure, operate the grid efficiently, and balance the tradeoff between risk and cost.

One of the primary difficulties in planning for the power grid is the presence of uncertainty in problem parameters. In particular, there is often uncertainty associated with the forecasted demand for power, renewable generation levels, and possible component failures. Blackouts are extremely disruptive, and so power system operators are highly motivated to ensure that sufficient excess capacity is available such that the demand for power can be met even when unexpected events occur. However, operators face a competing pressure to deliver power to consumers at low cost.

In this thesis, we address short-term (i.e., day-ahead) and long-term power system planning problems. We present methods to solve these problems that deal with parameter uncertainty. From a mathematical standpoint, one difficulty in developing solution approaches is the two-stage nature of these planning problems. Some decisions are made up front, before any uncertainty is realized, while other decisions are made in response to a particular realization of the uncertain parameters. In each of these problems, the set of all scenarios or all possible events is assumed to be very large. Decomposition procedures are presented that enable these problems to be solved even when straight-forward formulations are too large to be tractable. These algorithms are intended to be tools which may help planners evaluate the cost of ensuring different levels of reliability.

In Chapter II, we initially consider a network capacity design problem where there is uncertainty in the nodal supplies and demands. This robust single-commodity network design problem underlies several applications, including telecommunications and gas pipeline networks, as well as power transmission networks. We first consider the problem of assigning arc capacities such that installation costs are minimized and a feasible network flow solution is guaranteed to exist for all demand scenarios in the given set. We review a constraint generation algorithm to solve this problem and empirically demonstrate the scalability of this algorithm when the set of scenarios is large. This work lays the foundation for later chapters in which similar decomposition methodologies are employed to deal with uncertainty sets that contain a large number of elements. We next consider a chance-constrained problem in which minimum cost capacity expansion decisions are made to ensure feasibility for  $\alpha\%$  of the scenarios in the given set, where  $\alpha$  is a parameter specified by the user. We present a novel approach for solving this problem which embeds the previously presented constraint generation algorithm into a tree-based, parallelizable framework. Additionally, we explore a greedy extension of this algorithm that solves for a heuristic solution. We

present theoretical and computational analysis to evaluate the performance of the proposed optimal and heuristic algorithms for the chance-constrained problem.

In Chapter III, we consider a day-ahead planning problem that is specifically applicable to the operation of the power grid. We present a unit commitment problem where the generator schedule must be robust to the failure of any  $k$  generators and/or transmission lines, and transmission switching is allowed as a recovery action. Traditionally, power systems are operated to be N-1 secure, meaning that if any one component fails, the system can be operated normally. Protecting against  $k$  failures, i.e., N- $k$  secure, where  $k > 1$ , is a more stringent security standard under consideration in light of extreme weather events and the possibility of a terrorist attack. However, the combinatorial explosion in the number of failure events, i.e. contingencies, when  $k > 1$  results in an explicit formulation of the unit commitment problem which is too large to be solved directly. Additionally, we allow operators to use transmission switching to recover from a contingency. Transmission switching is the practice of temporarily removing transmission lines as a way to increase flexibility and ensure feasibility at lower cost.

The existence of binary second stage switching variables means that traditional decomposition procedures cannot be naively applied. To deal with the large number of contingencies, we propose a Contingency Oracle to identify unsurvivable contingencies for a given unit commitment solution. Furthermore, to address the challenge posed by the existence of switching variables, we present a reformulation in which switching decisions are treated as first stage variables. We propose a constraint generation algorithm to solve this problem where the Contingency Oracle is used to identify violated constraints, and where switching variables are dynamically generated for the master problem. We present computational results and analyze the algorithmic performance.

In Chapter IV, we address a long-term planning problem that is similar in struc-

ture to the unit commitment problem presented in Chapter III. We consider a transmission expansion planning problem where there is uncertainty, not only in possible equipment failures, but also in future loads and renewable generation outputs. Furthermore, as in Chapter III, transmission switching is considered as an allowable recovery action in response to a particular realization of the demand and a given line failure. We propose a robust optimization model where feasible operation is required for all loads and renewable generation levels within given ranges, and for all single transmission line failures. To deal with the more complex uncertainty set, we develop an oracle which can identify an unsurvivable contingency-demand pair for a given investment solution. We propose a novel constraint generation algorithm to solve this long-term planning problem which is procedurally similar to the algorithm presented in Chapter III. Computational results are presented that demonstrate tractability even when the uncertainty set contains a very large number of possible contingency and demand realizations.

This thesis concludes with Chapter V, in which Chapters II, III, and IV are summarized, and future work is discussed.

## CHAPTER II

# Chance-Constrained Network Capacity Design with Stochastic Demands with Finite Support

### 2.1 Introduction

In many real-world contexts, such as transportation systems, the power grid, telecommunications networks, and gas pipeline networks, planners need to determine the amount of capacity to build on the arcs in a network. Such decisions must often be made before nodal supplies and demands are known. Furthermore, these supplies and demands may not be static but rather vary over time while the network has to remain fixed. We consider the problem of determining the minimum-cost set of arc capacities to install in a single-commodity network when there is uncertainty in the nodal supplies and demands.

We represent uncertainty as a finite set of possible scenarios, where each scenario is a set of nodal supplies and demands. The set of scenarios could exactly represent a probability distribution with finite support. Alternatively, the scenarios could represent an approximation of a general probability distribution, where scenarios are generated by Monte Carlo sampling techniques. Using techniques such as *Sample Average Approximation*, it has been demonstrated that a finite set of scenarios can be used to find good solutions for the original distribution (*Luedtke and Ahmed (2008)*).



As is common in sampling based approaches for generating scenarios, we assume that all scenarios have equal probabilities of realization. If the set of scenarios represents a discrete probability distribution where the scenarios have different rational probabilities, the set of scenarios can be transformed into a set where all scenarios have equal probabilities by making copies of the scenarios in proportion to the probabilities of realization. It is assumed here that these scenarios have been given as an input.

In long term planning, it is common to require feasible operating conditions *almost* all the time, because requiring feasible conditions under absolutely all future scenarios is typically prohibitively expensive. Our goal is to find a minimum-cost set of arc capacities such that there exists a feasible set of flows for  $\alpha\%$  of all scenarios. In practice  $\alpha$  might be chosen to be a value like 99.5%.

We consider only the costs of installing capacity on arcs, as investment costs tend to be the dominant costs. Additionally, we assume the total cost is a linear function of the capacity installed. This cost structure might arise if the network user must lease arc capacities from the owner of a pre-existing network. Alternatively, this problem might arise when determining how to upgrade the arc capacities in an existing network. For the sake of exposition, we assume that there is no existing capacity on any arc, but the approach presented here can easily be modified to incorporate existing arc capacities.

The rest of this Chapter is structured as follows. In Section 2.2, we review the relevant literature on network design problems and approaches for dealing with parameter uncertainty. In Section 2.3, we describe the *Robust Capacity Design problem (RCD)*, in which a set of feasible flows is required to exist for every scenario in the known set. We review the solution approach for solving RCD first presented in *Go-mory and Hu (1962)*, and empirically demonstrate that a decomposition procedure can be used to solve this problem quickly, even when the number of scenarios is large. In Section 2.4, we consider a chance-constrained problem variant in which only some

fraction,  $\alpha\%$  (typically close to but less than 100%), of the scenarios are required to have a set of feasible flows; this is called the  $\alpha$ -Satisfied Capacity Design problem ( $\alpha$ SCD). We propose an algorithm for solving this  $\alpha$ SCD problem which embeds the constraint generation algorithm presented for the RCD problem into a tree-based framework, and we argue that a parallelized implementation can provide significant benefit. Furthermore, we present a greedy algorithm that can quickly solve for a good heuristic solution under certain conditions. In our computational experiments, the heuristic solution is within at most 0.3% of the optimal solution. Finally, in Section 2.5 we conclude with a summary and suggestions for future research.

## 2.2 Literature Review and Contributions

*Gomory and Hu* (1962) consider the problem of identifying the minimum cost set of arc capacities such that feasible flows are ensured for a set of different single-commodity demand requirements, each defined by a pair of origin-destination nodes and a demand amount. The authors present a constraint generation algorithm to solve this problem based on network cut-sets. An algorithm for solving the variant of this problem in which capacities are integer-valued is provided in *Sridhar and Chandrasekaran* (1992).

While the network design formulations presented in *Gomory and Hu* (1962) and *Sridhar and Chandrasekaran* (1992) were originally formulated to represent deterministic demand requirements in different time periods, the formulations are nearly identical to the RCD problem defined in Section 2.3.2, where the set of scenarios represent uncertainty in the future supplies and demands. The only difference is that we allow any number of nodes to act as sources or sinks in each scenario, as opposed to a single origin-destination pair for each scenario. The constraint generation algorithm presented in *Gomory and Hu* (1962) is reviewed in Section 2.3.3.

Many authors have considered variants of this RCD problem. *Minoux* (1981) ex-

tends the algorithm presented in *Gomory and Hu* (1962) to solve a multi-commodity variant of the same problem, called a *non-simultaneous flows* multi-commodity network design problem. A review of other related work is provided in *Minoux* (1989). More recently, *Ouorou* (2006) addresses the same multi-commodity design problem as *Minoux* (1981), and empirically compared the performance of two different types of cutting planes within a constraint generation algorithm.

A variety of other robust network design problems have been addressed in the literature. *Dahl and Stoer* (1998) present a cutting plane algorithm to solve a capacity design problem for a multicommodity network where demands are deterministic, investment costs are minimized, and feasibility is required in the event of any single arc or node failure. *Petrou et al.* (2007) consider a robust multi-commodity capacity design problem where the worst case value of a function which penalizes unsatisfied demand is minimized over the set of demand scenarios. A cutting plane solution approach is proposed. *Ouorou* (2012) consider a similar problem as *Petrou et al.* (2007) but instead use an *Affinely Adjustable Robust Counterpart* to compute tighter bounds. *Minoux* (2010) proves that the robust multi-commodity capacity design problem with a polyhedral uncertainty set for the demand is NP-hard.

*Pesenti et al.* (2004) consider the capacity design problem in which demands are expressed in terms of *bilateral contracts*, i.e., agreements between node pairs in which a supplier agrees to meet a customer's demand for any amount within a fixed range; to solve this problem the authors use a cutting plane algorithm. *Atamtürk and Zhang* (2007) consider the robust capacity design problem where demand is assumed to belong to a budget uncertainty set, and the goal is to minimize the worst case total of investment and routing costs. The authors present a procedure for obtaining bounds, and computational results for several special problem instances. *Mudchanatongsuk et al.* (2008) address a similar problem where routing costs are additionally assumed to be uncertain. The authors explore polyhedral and ellipsoidal uncertainty

sets, and present a column generation procedure to solve a path constrained problem variant. *Ordóñez and Zhao (2007)* present a tractable conic LP formulation of a robust capacity design problem in which demands and travel times belong to polyhedral uncertainty sets, worst case travel time is minimized, and investment costs are constrained to be less than a specified budget.

A challenge with robust network design problems in general is that it is difficult to define an uncertainty set that appropriately controls the conservatism of the optimal solution. The focus of this Chapter is the chance-constrained  $\alpha$ SCD problem, where a small subset of the scenarios are allowed to be infeasible. An advantage of this formulation is that it allows the conservatism of the optimal solution to be controlled with a single parameter  $\alpha$ .

There are many different approaches in the literature for solving chance-constrained problems. Several authors including *Ahmed and Shapiro (2008)*, *Küçükyavuz (2009)* and *Luedtke et al. (2010)* present mixed-integer formulations of chance-constrained problems and describe several types of valid inequalities that can be added to strengthen these formulations. *Luedtke (2013)* present a branch-and-cut decomposition algorithm for solving the mixed-integer formulation of the chance-constrained problem.

Other authors such as *Beraldi and Ruszczyński (2002)*, *Dentcheva et al. (2000)*, *Prékopa (1990)* and *Ruszczynski (2002)* suggest methods that utilize *p-level efficient points* of discrete distributions to develop equivalent formulations of probabilistic constraints. Alternatively, *Watson et al. (2010)* propose an algorithm based on *progressive hedging* which can solve for a heuristic solution quickly, relative to other methods, even for large problems. These published methods have each been developed to solve a fairly general class of chance-constrained problems. We propose a method for solving a very particular chance-constrained problem and we are able to exploit the problem structure to our advantage.

## 2.3 Robust Capacity Design Problem

In this section we review the problem of determining the minimum-cost set of arc capacities to install in a given network such that there exists feasible flows for all scenarios in the given set. We first present a traditional network-flow based formulation, and then we review the constraint generation approach proposed by *Gomory and Hu* (1962). We discuss implementation details to augment the theoretical discussions in *Gomory and Hu* (1962) and other works.

We assume that in each scenario the total supply equals the total demand. Flows are defined to be feasible for a particular network if they satisfy flow balance constraints and arc capacity constraints.

### 2.3.1 Notation

The following notation will be used in the model formulations in this Chapter.

#### Sets

$N$  set of all nodes in the network

$A$  set of all arcs in the network. For each element  $(i, j) \in A$ ,  $i, j \in N$ ,  $i \neq j$

$\Omega$  set of all scenarios, where each scenario is a set of supplies/demands for all nodes.

#### Parameters

$c_{ij}$  cost of installing a unit of capacity on arc  $(i, j)$ , for all  $(i, j) \in A$ .

$b_i^\omega$  supply at node  $i$  in scenario  $\omega$ , for all  $i \in N$ ,  $\omega \in \Omega$ .

Demand is represented as negative supply.

## Variables

$f_{ij}^\omega$  flow on arc  $(i, j)$  in scenario  $\omega$ , for all  $(i, j) \in A$ ,  $\omega \in \Omega$ .

$u_{ij}$  capacity to be installed on arc  $(i, j)$ , for all  $(i, j) \in A$ .

We present here formulations for a directed network, but these formulations could easily be modified to model an undirected network.

### 2.3.2 Network Flow Formulation

RCD can be modeled with a traditional network-flow formulation as follows:

$$\min \sum_{(i,j) \in A} c_{ij} u_{ij} \quad (2.1a)$$

$$\text{s.t.} \quad \sum_{j:(i,j) \in A} f_{ij}^\omega - \sum_{j:(j,i) \in A} f_{ji}^\omega = b_i^\omega \quad \forall i \in N, \omega \in \Omega \quad (2.1b)$$

$$f_{ij}^\omega - u_{ij} \leq 0 \quad \forall (i, j) \in A, \omega \in \Omega \quad (2.1c)$$

$$f_{ij}^\omega \geq 0 \quad \forall (i, j) \in A, \omega \in \Omega \quad (2.1d)$$

$$u_{ij} \geq 0 \quad \forall (i, j) \in A \quad (2.1e)$$

Objective (2.1a) states that the total cost of installing capacity is to be minimized. For each scenario  $\omega \in \Omega$ , constraint (2.1b) enforces that flow is conserved at each node and constraint (2.1c) enforces that arc flow cannot exceed capacity. Because the number of flow variables  $f$  and the number of constraints depend on the number of scenarios, this LP may be intractable if the number of scenarios  $|\Omega|$  is large. In our computational results, (2.1) is intractable for numbers of scenarios of 5,000 and above.

### 2.3.3 Cut-set Solution Approach

In the network design problem addressed here, we only consider the cost of investing in new capacity. This suggests the potential value of a formulation that depends

only on the arc capacity variables  $u$ , as these variables do not depend on the number of scenarios and are the only variables in the objective function.

Let  $\Theta$  be the set of all nonempty proper subsets of nodes in  $N$ , i.e.,  $\Theta = \{\theta \subset N : \theta \neq \emptyset\}$ .  $|\Theta| = 2^{|N|} - 2$ . The following LP can also be used to solve for the minimum-cost arc capacity assignment such that a set of feasible flows exists for all scenarios.

$$\min \sum_{(i,j) \in A} c_{ij} u_{ij} \quad (2.2a)$$

$$\text{s.t.} \quad \sum_{i \in \theta, j \notin \theta} u_{ij} \geq \sum_{i \in \theta} b_i^\omega \quad \forall \theta \in \Theta, \omega \in \Omega \quad (2.2b)$$

$$u_{ij} \geq 0 \quad \forall (i, j) \in A \quad (2.2c)$$

Constraint set (2.2b) states that for any network cut-set defined by the partition of nodes  $\theta$  and  $N \setminus \theta$ , the total capacity of the arcs contained in the cut-set is at least equal to the total net supply for the nodes in  $\theta$ , for each scenario. These cut-set constraints (2.2b) are both a necessary and a sufficient condition for the existence of a feasible flow for every scenario (Theorem 6.12, pp. 196, *Ahuja et al. (1993)*).

The number of constraints in (2.2b) can be reduced to exactly one constraint per node subset by recognizing that for all  $|\Omega|$  cut-set constraints for a given  $\theta$ , one constraint dominates all other constraints. For a given  $\theta$ , if the constraint with the greatest right hand side is satisfied, all other constraints will immediately be satisfied. Let  $M(\theta) \equiv \max_{\omega \in \Omega} \{\sum_{i \in \theta} b_i^\omega\}$ . Constraint set (2.2b) can be replaced by the following constraint set:

$$\sum_{i \in \theta, j \notin \theta} u_{ij} \geq M(\theta) \quad \forall \theta \in \Theta. \quad (2.3)$$

The number of constraints in (2.3) is determined entirely by the number of node

subsets  $|\Theta|$ , which is a function only of the number of nodes  $|N|$ . Thus, the size of this linear program is completely independent of the number of scenarios  $|\Omega|$ . However, there are an exponential number of constraints in the set (2.3).

As proposed in *Gomory and Hu* (1962), we suggest that the cut-set LP (2.2) with constraint set (2.3) be solved via a constraint generation procedure which will here be referred to as the Cut-set Constraint Generation (CSCG) algorithm.

Let the *Master Problem (MP)* be a relaxation of the cut-set LP. In each iteration, MP is solved, and for each scenario in the set  $\Omega$ , a subproblem is solved to identify constraints from the set (2.3) that are violated for the current MP solution. The subproblem identifies a minimum cut-set for the current capacity assignment and the given scenario. There exist several methods for formulating and solving the min cut-max flow problem, and any one of these methods may be used to identify the minimum cut-set. The node subset  $\theta$  corresponding to this minimum cut-set is used to identify a constraint from the set (2.3) that is not satisfied, and this constraint is then added to the MP. Note that while a particular scenario  $\omega$  is used to identify the minimum cut-set, the corresponding constraint added to the MP is valid for all scenarios. The right hand side of the cut-set constraint is  $M(\theta)$ , which ensures that sufficient capacity is installed on this cut-set for all scenarios.

The procedure of iteratively solving the MP and adding violated constraints is repeated until, for all scenarios, the subproblem identifies that all constraints in the set (2.3) are satisfied. When this occurs, the algorithm exits with the optimal capacity assignment that is feasible for all scenarios.

### 2.3.4 Implementation Details

To provide a practical assessment of the theoretical concepts provided by earlier authors, we present here an analysis of the algorithm's performance in implementation. In this section we discuss the implementation details which have a significant



impact on the algorithm’s rate of convergence. We found that the three most important issues are (i) the choice of initial MP constraints, (ii) the rules for selecting a scenario from the list  $\Omega$  to define the subproblem, and (iii) how many identified violated constraints are added to the MP per iteration.

On the first issue, we experienced our best run times when we initialized the MP with one constraint per node requiring that the total capacity on the arcs directed out of a node be at least the maximum supply at the node across all scenarios (if a supply node) or that the total capacity on the arcs directed into a node must be at least the maximum demand at the node (if a demand node). We assume that each node acts only as a supply node or as a demand node in all scenarios, but these initializing constraints could also be used if a single node acts both as a demand node and as a supply node in different scenarios.

On the second issue, we found that the best policy was to maintain two mutually exclusive and collectively exhaustive lists of scenarios: the current list and the set-aside list. The current list is a list of scenarios that is currently being screened. When a scenario is identified to be feasible for the current set of arc capacities, it is moved from the current list to the set-aside list. The current list is then shorter the next time it is looped through, and generally contains the more “difficult” scenarios, while “easy” scenarios get moved to the set-aside list. In order to guarantee that the algorithm exits with an optimal solution, when the current list becomes empty it is necessary to loop through the entire list of set-aside scenarios to confirm that all scenarios are feasible. If any scenarios are not feasible, the process is repeated. The idea is that this policy avoids needlessly and repeatedly checking “easy” scenarios that have been found to be feasible in an early iteration and are likely to remain feasible in later iterations. Instead, the algorithm can focus on the “difficult” scenarios. Our computational results indicate that this policy results in the fewest total subproblems solved on average, which results in the fastest run time.

Finally, on the third issue, we recommend that scenario subproblems from the current list are solved only until a cut-set with insufficient capacity is identified. At this time one violated constraint is added to the MP and the MP is solved again. We have found that this approach of adding one constraint to the MP per iteration causes the algorithm to solve faster than when multiple constraints are added to the MP during each iteration.

### 2.3.5 Run Time Results

Using the implementation options described in Section 2.3.4, we present computational experiments that indicate how the performance of the CSCG algorithm scales with the number of scenarios.

Our first test network is an IEEE 118 node test system (*Christie (1993)*), which is representative of a portion of the U.S. power grid. Supply/demand scenarios were generated by perturbing and scaling the nominal power generation and consumption levels provided for each of these nodes in the IEEE test system. The perturbation was done by adding a normally distributed random variable to the nominal supply/demand at each node, where the normal distribution has mean of 0 and standard deviation set so as to make the coefficient of variation (COV) equal to 0.25, if the nominal supply is positive, or equal to 0.75 if the nominal supply is negative. These COV values were chosen because, in a conventional power grid, the demand for power is typically more variable than the generation of power. Scaling was done by multiplying the perturbed supplies/demands by a random value, which is drawn from uniform distribution with minimum 0.1 and maximum 2.0. An additional dummy node was added to make the total supply equal the total demand in each scenario by supplying or demanding whatever is excess (i.e.,  $b_{119}^\omega = -\sum_{i=1}^{118} b_i^\omega \forall \omega \in \Omega$ ). An arc was added to and from each of the 118 nodes to this dummy node to ensure feasibility, changing the number of arcs from 358 to 594.

While the IEEE118 test network is a useful example of a real world network, it is a fairly sparse network. To test the performance of the CSCG algorithm on a dense network, for our second test instance we constructed a network with 30 nodes that is exhaustively dense, i.e., there exists an arc between every node pair. Let the test instance be called exh30. Supplies in each scenario were randomly generated from a continuous uniform distribution  $[-10, 10]$ . A dummy node was added and its supply/demand was set in each scenario to make the total supply equal the total demand.

For all test instances presented in this Chapter, the arc costs  $c_{ij}$  were randomly generated from a discrete uniform distribution  $[1, \dots, 50]$ . Algorithms were implemented in C++ which called CPLEX v12.4 to solve all linear programs, on a computer with a 2.3 GHz processor and 4G RAM. In Figure 2.1, the run time results shown are averaged over 5 trials and the error bars indicate one standard deviation in these run times.

Figure 2.1 shows the average run time of the CSCG algorithm over a range of different numbers of scenarios up to 40,000 for both the IEEE118 and the exh30 test systems. For both of these networks, the run time increases approximately linearly as the number of scenarios increases. The coefficient of determination ( $R^2$ ) of the linear regression is 0.99 for both test systems.

IEEE118 has more nodes than exh30, and so the number of cut-set constraints in constraint set (2.3) is larger. The number of cut-sets for a 118 node network is on the order of  $10^{35}$ , compared to  $10^9$  for a 30 node network. However, it seems that the dominant factor in determining the runtime is not the number of total cut-sets but the number of arcs. The denser exh30 has 870 arcs, compared to 594 arcs for the IEEE118. The number of variables in the master problem is equal to the number of arcs. This difference appears to drive the difference in runtimes shown in Figure 2.1. The CSCG algorithm solves IEEE118 with 40,000 scenarios in less than 4 min, and

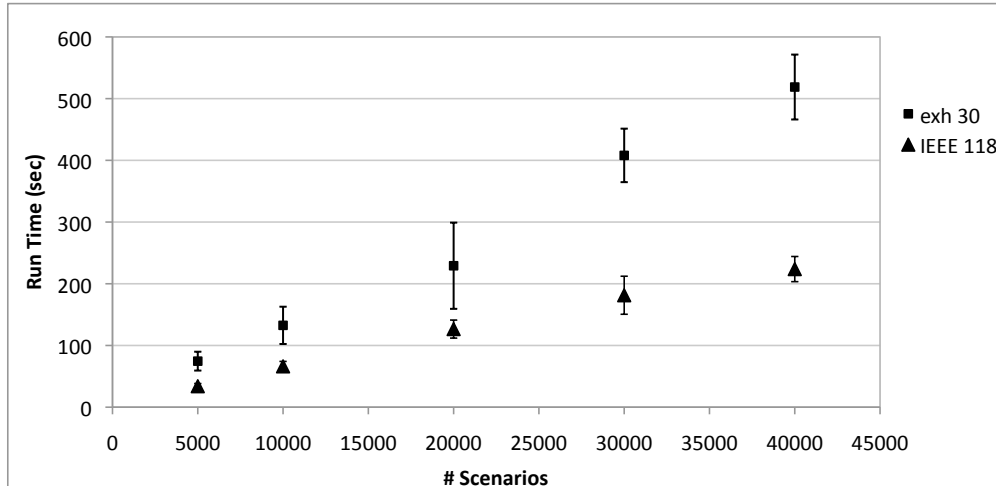


Figure 2.1: CSCG Algorithm Run Time for IEEE118 and exh30 test systems

exh30 with 40,000 scenarios in about 8 and a half minutes.

## 2.4 $\alpha$ -Satisfied Capacity Design Problem

Having reviewed a successful approach for solving the RCD problem, we now explore the more challenging chance-constrained problem where some percentage  $\alpha$  (typically close to but less than 100%) of all the scenarios in the set  $\Omega$  are required to have feasible flows.

We first consider a Mixed Integer Program (MIP) formulation of the  $\alpha$ SCD problem, and then discuss the difficulties of solving this formulation directly. The difference between the  $\alpha$ SCD problem and the RCD problem is that the set of scenarios for which feasibility is required is a decision for  $\alpha$ SCD, whereas it is given for RCD. The RCD problem is relatively easy to solve, so we propose an approach to solve  $\alpha$ SCD that employs a combinatorial tree-based framework for exploring subsets of scenarios for which feasibility could be required. We present an algorithm which embeds the CSCG algorithm within this tree to find the optimal solution. Finally, we present a greedy variation of this approach that can be used to solve for a heuristic solution, and we analyze the solution quality.

### 2.4.1 MIP Formulation

We introduce a set of binary indicator variables  $I_\omega$  for all  $\omega \in \Omega$  which enable us to determine which scenarios have feasible network flow solutions. Let  $I_\omega$  equal 1 if there exists a set of feasible flows for scenario  $\omega$ , and equal 0 otherwise. The MIP formulation of the  $\alpha$ SCD problem is as follows.

$$\min \sum_{(i,j) \in A} c_{ij} u_{ij} \quad (2.4a)$$

$$\sum_{i \in \theta, j \notin \theta} u_{ij} - \left( \sum_{i \in \theta} b_i^\omega \right) I_\omega \geq 0 \quad \forall \theta \in \Theta, \omega \in \Omega \quad (2.4b)$$

$$\sum_{\omega \in \Omega} I_\omega \geq \left( \frac{\alpha}{100} \right) |\Omega| \quad (2.4c)$$

$$u_{ij} \geq 0 \quad \forall (i, j) \in A \quad (2.4d)$$

$$I_\omega \in \{0, 1\} \quad \forall \omega \in \Omega \quad (2.4e)$$

In this formulation, constraint set (2.4b) is analogous to the cut-set constraint set (2.2b); if  $I_\omega = 1$ , the constraints require that demand must be fully satisfied in scenario  $\omega$ , or if  $I_\omega = 0$ , the constraint is not restrictive. Constraint (2.4c) states that at least  $\left(\frac{\alpha}{100}\right) |\Omega|$  binary variables must be equal to 1, indicating that at least  $\alpha\%$  of all scenarios in  $\Omega$  must be feasible.

Due to the binary variables, formulation (2.4) is difficult to solve. If the set of scenarios  $\Omega$  is large, (2.4) contains a large number of binary variables which have significant incentive to be fractional. For example, it is typically much cheaper to install half as much capacity as is needed on each cut-set for two different scenarios than to fully satisfy all cut-set constraints for one scenario. Thus, at the optimal solution of the LP relaxation of (2.4), the  $I_\omega$  variables will have fractional values and a branch-and-bound algorithm would require a lot of branching to fix the indicator variables to integer values.

Additionally, because the left hand side of each constraint in the set (2.4b) contains the scenario-specific variable  $I_\omega$ , the number of constraints cannot be reduced to 1 constraint per node subset  $\theta$  as was done in the formulation of the RCD problem. Every cut-set constraint is scenario-specific. Essentially, the Benders' feasibility cuts in the set (2.4b) are much weaker than the feasibility cuts for the RCD problem (2.3). A decomposition procedure for solving (2.4) is not promising due to the combination of weak Benders' cuts and a master problem which is difficult to solve.

There are several other approaches for solving mixed integer formulations of general chance-constrained programs, including enumerating  $p$ -efficient points as in *Beraldi and Ruszczyński* (2002), generating valid inequalities as in *Song and Luedtke* (2013), or the branch-and-cut algorithm described in *Luedtke* (2013). In this Chapter, we offer an alternative approach that leverages the specific structure of  $\alpha$ SCD, taking advantage of the fact that  $l$ , the number of scenarios allowed to be infeasible, is typically very small, and the fact that if the set of scenarios allowed to be infeasible were known, the resulting RCD problem could be solved quickly with the CSCG algorithm.

### 2.4.2 Combinatorial Tree Algorithm

There are a finite number of distinct subsets of  $\Omega$  of cardinality  $\lceil (\frac{\alpha}{100}) |\Omega| \rceil$ , so one approach to solve this problem is to enumerate all such subsets and apply the CSCG algorithm for each subset. One way of organizing all such subsets is to use a tree. At the root node, the set of scenarios required to be feasible is equal to the complete set  $\Omega$ . At depth  $d$  in the tree there are  $\binom{\Omega}{d}$  nodes, where at each node the set of scenarios required to be feasible is the complete set  $\Omega$  minus a unique set of  $d$  scenarios. At the bottom level at depth  $l = \lfloor (1 - \frac{\alpha}{100}) |\Omega| \rfloor$  there is a node for every distinct set of scenarios of cardinality  $\lceil (\frac{\alpha}{100}) |\Omega| \rceil$ . The optimal solution could be found by applying the CSCG algorithm to each of the nodes in the bottom level and identifying the

node with the overall minimum cost. A tree constructed as described is illustrated in Figure 2.2, where each tree node is labeled with the set of scenarios that are allowed to be infeasible.

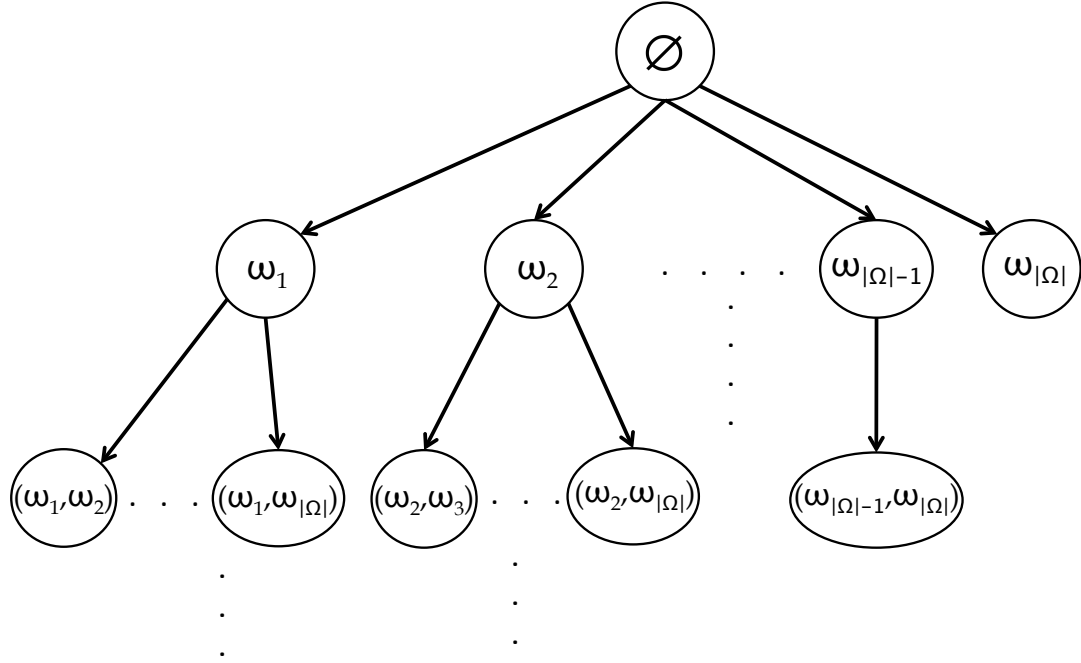


Figure 2.2: Combinatorial Tree

With the described tree, the number of nodes at bottom level  $\binom{\Omega}{l}$  could be very large even if  $l$  is relatively small. However, as we build the tree, we can identify some branches that will contain only suboptimal solutions and therefore we can prune these branches.

First we introduce useful terms and notation.

- When the CSCG algorithm terminates for a single RCD problem, the final MP includes a set of constraints that have been generated over the course of the algorithm which we will call *explicit constraints*.
- A subset of the explicit constraints will be tight at the optimal solution to a final MP; we call these *active explicit constraints*.

- For any cut-set constraint with corresponding node subset  $\theta$ , there are one or more scenarios that define the right hand side by having the greatest net supply over  $\theta$ . Let any scenario such that  $\bar{\omega} = \arg \max_{\omega \in \Omega} \{\sum_{i \in \theta} b_i^\omega\}$  be the *dominant scenario* for the given constraint.
- A *binding scenario* is a scenario that is dominant for at least one active explicit constraint.
- Each node in the combinatorial tree is a RCD problem where the set of scenarios required to be feasible is  $(\Omega \setminus E)$ , where  $E$  is the *exclusion set*.
- The set of binding scenarios for a node with exclusion set  $E$  is denoted  $B(E)$  and the optimal objective value for that node is  $V(E)$ .
- The goal of  $\alpha$ SCD is to both find an optimal set of arc capacities  $u$  and corresponding exclusion set  $E$  with cardinality  $l = \lfloor (1 - \frac{\alpha}{100}) |\Omega| \rfloor$ . Let this optimal set of excluded scenarios be denoted  $E_l^*$ .

We propose a branching rule that is based on the idea that, given a current node in the tree with exclusion set  $E$ , we can identify some branches that will contain only suboptimal solutions based on what scenarios are not in the set  $B(E)$ . In particular, a scenario that is not dominant for any active explicit constraints is implicitly dominated by one or more other scenarios that are contained in the set  $B(E)$ . Thus branching only on scenarios contained in the set  $B(E)$  is sufficient. Formal theorems and proofs will now be stated, and then the branching rule will be presented in more precise terms.

#### 2.4.2.1 Reduced Combinatorial Tree Theorems

**Lemma II.1.** *For any scenario  $\hat{\omega} \in \Omega \setminus B(E)$ ,  $V(E \cup \hat{\omega}) = V(E)$ .*



In other words, if a scenario is not binding, then excluding that scenario will not have any effect on the feasible region, and thus the optimal objective value will not change. The formal proof for this lemma is as follows.

*Proof.* The cut-set LP for RCD with exclusion set  $E$  could be appropriately modified to solve RCD with exclusion set  $(E \cup \hat{\omega})$  by relaxing every constraint in the set (2.3) whose dominant scenario is  $\hat{\omega}$ . The CSCG algorithm for solving RCD with exclusion set  $E$  exits when the MP contains a set of explicit constraints such that any solution  $u$  satisfying these constraints will also satisfy all cut-set constraints that were not explicitly added to MP. Relaxing any of these implicitly satisfied constraints has no effect on the optimal solution. Relaxing any explicit constraint in the final MP that is not tight at the optimal solution will not change the optimal solution. By the choice of  $\hat{\omega} \notin B(E)$ , there does not exist an active explicit constraint for which scenario  $\hat{\omega}$  is dominant. Thus  $V(E) = V(E \cup \hat{\omega})$ .  $\square$

**Lemma II.2.** *There exists an optimal exclusion set for  $\alpha$ SCD,  $E_l^*$ , that takes the form  $\{\omega_1, \omega_2, \dots, \omega_l\}$  where  $\omega_1 \in B(\emptyset)$  and  $\omega_i \in B(\{\omega_1, \dots, \omega_{i-1}\})$  for all  $i = 2, \dots, l$ .*

In other words, the optimal exclusion set can be ordered such that each scenario is binding for the exclusion set equal to all lower ordered scenarios. Binding scenarios are the only scenarios whose exclusion has the potential to improve the objective value. So when growing the optimal exclusion set from the empty set, only binding scenarios should be candidates for the next scenario to exclude. The formal proof for this lemma is as follows.

*Proof.* Suppose, in contradiction, that all optimal exclusion set(s) take the form  $E_l = \{\hat{\omega}_1, \dots, \hat{\omega}_l\}$  where scenarios  $\hat{\omega}_1, \dots, \hat{\omega}_l \in \Omega \setminus B(\emptyset)$ . If RCD is solved for exclusion set  $\emptyset$ , and then, one by one, each scenario in the set  $E_l$  is added to the exclusion set, by Lemma II.1, the optimal objective value will not change, i.e.,  $V(\emptyset) = V(E_l)$ . RCD with an exclusion set of cardinality  $l$  which includes a scenario  $\omega_i \in B(\emptyset)$  will have

an objective value less than or equal to the objective value of  $V(E_l)$ , thus it is not possible that all exclusion sets have the supposed form. There is a contradiction.

Suppose, for some  $2 \leq k \leq (l - 1)$ , that all optimal exclusion set(s) take the form  $E_l = \{\hat{\omega}_1, \dots, \hat{\omega}_l\}$  where  $\hat{\omega}_1 \in B(\emptyset)$  and for all  $i = 2, \dots, k$ ,  $\omega_i \in B(\{\hat{\omega}_1, \dots, \hat{\omega}_{i-1}\})$ , but all scenarios  $\hat{\omega}_{k+1}, \dots, \hat{\omega}_l \in \Omega \setminus B(\{\hat{\omega}_1, \dots, \hat{\omega}_k\})$ . If RCD is solved for  $E = \{\hat{\omega}_1, \dots, \hat{\omega}_k\}$ , and then, one by one, all other scenarios in the set  $E_l$  are also excluded, by Lemma II.1, the optimal objective value will not change, i.e.,  $V(\{\hat{\omega}_1, \dots, \hat{\omega}_k\}) = V(E_l)$ . RCD with an exclusion set of cardinality  $l$  which includes scenario  $\hat{\omega}_1, \dots, \hat{\omega}_k$  and includes a scenario  $\omega_i \in B(\{\hat{\omega}_1, \dots, \hat{\omega}_k\})$  will have an objective value less than or equal to the objective value of  $V(E_l)$ , thus it is not possible that all exclusion sets have the supposed form. There is a contradiction.  $\square$

Given the stated lemmas, we present the following algorithm which constructs a tree of RCD problems and finds the optimal set of arc capacities  $u$  for optimal exclusion set  $E_l^*$ . The tree nodes referred to in this algorithm each represent an RCD problems with different exclusion sets, as illustrated in Figure 2.2.

### Reduced Combinatorial Tree Algorithm:

1. Initialize the Last In First Out (LIFO) queue of tree nodes to contain only the root node, which is an RCD problem with exclusion set  $E = \emptyset$ . Initialize the current best objective value  $v = \infty$  and the current best arc capacities  $u = 0$ .
2. Pop off a tree node from the LIFO queue with exclusion set  $E$ . Solve this RCD problem. If  $|E| = l$  and  $V(E) < v$ , update  $v = V(E)$  and let  $u$  be the optimal solution to this RCD problem. Otherwise, if  $|E| < l$ , find the set of binding scenarios  $B(E)$ . For each scenario  $\omega \in B(E)$ , add a tree node to the LIFO queue which has exclusion set  $\{E \cup \omega\}$  if a tree node with this exclusion set does not already exist in the LIFO queue.

3. If the LIFO queue is not empty, go to step 2. Otherwise, exit with the optimal set of arc capacities  $u$ .

**Theorem II.3.** *The Reduced Combinatorial Tree Algorithm will exit with the optimal set of arc capacities  $u$  for optimal exclusion set  $E_l^*$ .*

*Proof.* The Reduced Combinatorial Tree Algorithm constructs a tree node for every exclusion set  $E$  of size  $l$  with form  $\{\omega_1, \omega_2, \dots, \omega_l\}$  where  $\omega_1 \in B(\emptyset)$  and  $\omega_i \in B(\{\omega_1, \dots, \omega_{i-1}\})$  for all  $i = 2, \dots, l$ . By Lemma II.2, the tree contains a node for every exclusion set that satisfies the necessary condition to be the optimal exclusion set, and thus includes the optimal exclusion set  $E_l^*$ . The algorithm finds the set of arc capacities that has the lowest cost for the RCD problem among these tree nodes with exclusion set of size  $l$ , which is the optimal set of arc capacities.  $\square$

#### 2.4.2.2 Reduced Combinatorial Tree Size

In our computational experiments with the Reduced Combinatorial Tree (RCT), we observe that the total number of tree nodes in each level to be significantly reduced from the number of tree nodes in the complete combinatorial tree. Instead of  $\binom{|\Omega|}{k}$  tree nodes in the  $k$ th level of the tree, the number of tree nodes in each level is a function of the number of binding scenarios in the tree nodes one level above. The set of binding scenarios  $B(E)$  for a tree node with exclusion set  $E$  is the set of dominant scenarios for the active explicit constraint. The size of this set  $|B(E)|$  depends on the number of active explicit constraints in MP, which depends on the number of variables,  $|A|$ , and on how close to pareto dominant the set of scenarios  $\Omega$  is. For example, if all the scenarios were scalar multiples of a base scenario, then the scenario with the largest scalar multiplier would be pareto dominant over all other scenarios, and would define the right hand side for every possible cut-set constraint. Then, regardless of how many active explicit constraints there were in the MP, there would be exactly one binding scenario, and exactly one tree node in each level. For any

other type of scenario set, the number of binding scenarios will likely be low if a few scenarios are dominant over all other scenarios, or will likely be higher if each cut-set has a different defining scenario.

To analyze the number of tree nodes required to be solved in the RCT for different problem instances, we performed computational experiments with an IEEE30 node test system (*Christie (1993)*). We generated scenarios using two different methods. One scenario generation method, labeled as “Scaled & Perturbed Supplies” in Table 2.1, is the same as was described in Section 2.3.5 for IEEE118. The other method, labeled as “Uniform Supplies” in Table 2.1, is the same as was described in Section 2.3.5 for exh30. For both ways of generating the scenarios, a dummy network node was used to make the supplies and demands net to 0 in each scenario, as was also described in Section 2.3.5.

The numbers of tree nodes created at each level of the RCT are presented in Table 2.1. From Table 2.1, it is evident that the scenario distribution has a significant impact on how many nodes must be generated in the tree. The number of tree nodes is significantly greater when the scenarios are drawn from a uniform distribution than when generated by scaling and perturbing a base scenario. When the scenarios are generated from a uniform distribution, the scenarios are very different from each other, and thus the active explicit cut-set constraints are likely to have different dominant scenarios, and the set of binding scenarios is larger. When the scenarios are generated by perturbing and scaling a base scenario, the scenarios are more similar to each other, and the active explicit cut-set constraints are more likely to have common dominant scenarios, and the set of binding scenarios is smaller.

We note that while the number of tree nodes in each level increases exponentially, all tree nodes in a particular level are independent of each other and could be solved in parallel. If all tree nodes per level were to be parallelized, the time to solve the reduced tree and find the optimal solution is equal to the time to run the CSCG

Num. Scenarios	$\alpha$	$l$	Uniform Supplies				Scaled & Perturbed Supplies			
			Total Tree Nodes	Level 1	Level 2	Level 3	Total Tree Nodes	Level 1	Level 2	Level 3
67	97%	2	696	34	661	-	250	21	228	-
100	97%	3	15,020	43	940	14,036	2,911	24	299	2,587
300	99%	3	28,396	50	1,383	26,962	5,200	29	441	4,789
600	99.5%	3	33,829	55	1,590	32,183	8,229	35	624	7,569

Table 2.1: Number of Reduced Combinatorial Tree Nodes

algorithm  $l$  times. Given a fixed number of processors  $p$ , the time required to solve the tree could be approximated by taking the maximum node solve time for each level, and multiplying this by the number of nodes in that level of the tree divided by  $p$ .

### 2.4.3 Greedy Algorithm

While utilizing parallelization may allow the RCT to be solved relatively quickly to find the optimal solution for certain problem instances, a greedy algorithm can alternatively be used to generate a heuristic solution more quickly without the need for a parallel implementation. The basic idea behind the algorithm is that scenarios are added one by one to the set of excluded scenarios, greedily choosing the scenario whose exclusion most improves the cost. A similar algorithm, which gradually excludes constraints to solve a chance-constrained problem, is presented in *Pagnoncelli et al.* (2012). We demonstrate that the heuristic is close to optimal for the problem instances we tested.

The greedy algorithm is as follows:

1. Solve the RCD problem with the exclusion set  $E = \emptyset$ . Initialize iteration  $d = 0$ .  
Let  $B(\emptyset)$  be the current set of binding scenarios.
2. Increase  $d$  by 1. If  $d < l$ , for each scenario  $\omega$  in the current set of binding scenarios  $B(E)$ , solve a RCD problem with exclusion set  $\{E \cup \omega\}$ .
3. Among these RCD problems, find the exclusion set  $E$  whose optimal objective value  $V(E)$  is smallest. Set the current set of binding scenarios equal to  $B(E)$ .  
Go to step 2.

The final solution  $u$  that this algorithm exits with is a feasible solution, but it is not guaranteed to be an optimal solution. However, computational results indicate that for certain types of problems the solution  $u$  is often optimal or close to optimal.

#### 2.4.4 Heuristic Computational Results

We tested the greedy algorithm on the IEEE30 test system and on an exhaustively dense 20 node test network, constructed in the same way as exh30 described in Section 2.3.5.

In Table 2.2 the column “Heuristic Obj. Value” is the heuristic objective value returned by the greedy algorithm, and these values are shown next to the optimal objective value. Note that “S&P” refers to the “Scaled&Perturbed” method of generating supply scenarios, as previously described in this section. For almost all of the computational experiments presented in Table 2.2 the greedy algorithm returns an optimal objective value. The only exception was for IEEE30 Uniform with  $\alpha = 97\%$ . As discussed in section 2.4.2.2, if the set of scenarios was pareto dominant, meaning that all scenarios were a scalar multiple of a base scenario, the greedy algorithm will exactly return the optimal solution. Performance should be close to optimal if the distribution of the scenarios is close to pareto dominant, and further from optimal if the scenarios are very different from each other.

In many real world systems, such as the power grid, supplies and demands among the nodes often have similar relationships across different scenarios, e.g. a larger generator will always have greater output than a smaller generator, though their absolute outputs may vary. With this type of scenario distribution, the heuristic should be close to optimal. Interestingly, in our computational experiments in which scenarios were generated from a uniform distribution where scenarios are different from each other and not at all close to pareto dominance, when we expect the heuristic to perform poorly, and the heuristic still returns the optimal objective in most cases.

The run times for the greedy algorithm and the RCT algorithm for this set of computational experiments are listed in Table 2.3. For all computational experiments, the greedy algorithm run time is dramatically faster than the RCT algorithm run time. The RCT run times presented here are for a serial implementation of the

Number Scenarios	$\alpha$	$l$	IEEE30 Uniform		IEEE30 S&P		exh20 Uniform	
			Optimal Obj. Value	Heuristic Obj. Value	Optimal Obj. Value	Heuristic Obj. Value	Optimal Obj. Value	Heuristic Obj. Value
67	97%	2	8706	8706	2607	2607	1420	1420
100	97%	3	8030	8053	2864	2864	1704	1704
300	99%	3	8674	8674	2867	2867	1360	1360
600	99.5%	3	8950	8950	3621	3621	1475	1475

Table 2.2:  $\alpha$ SCD Heuristic Evaluation



algorithm. As discussed in Section 2.4.2.2, the RCT algorithm lends itself well to a parallel implementation, as the tree nodes at each level could be solved in parallel. The serial RCT algorithm run times are an upper bound on the run times of a parallel implementation, and the greedy algorithm run times are a lower bound.

The greedy algorithm has relatively short run times with a simple serial implementation. Given how close to optimality the heuristic solution is for these computational experiments (within 0.3%, as shown in Table 2.2), the greedy algorithm is an attractive option for solving the  $\alpha$ SCD problem.

# Scenarios	$\alpha$	$l$	IEEE30 Uniform		IEEE30 S&P		exh20 Uniform	
			RCT	Greedy	RCT	Greedy	RCT	Greedy
67	97%	2	73	8	18	2	90	11
100	97%	3	1817	16	255	6	1333	21
300	99 %	3	8808	46	1166	14	3303	45
600	99.5 %	3	20050	72	2558	28	14672	110

Table 2.3: Greedy and RCT Algorithm Run Times (sec)

## 2.5 Conclusion

We have considered a robust network capacity design problem where there is uncertainty in the supplies and demands which is represented with a set of discrete scenarios. We review a constraint generation algorithm for this problem when all scenarios are required to have a feasible flow and present practical implementation details that empirically lead to improved algorithmic performance. We develop a novel algorithm to solve the chance-constrained problem when  $\alpha\%$  of all scenarios are required to be feasible. This tree-based combinatorial algorithm embeds the previously described constraint generation algorithm to find the optimal arc capacity assignment. Additionally, a greedy algorithm is presented that can often be used to solve for a high quality heuristic solution to this chance-constrained problem.

## CHAPTER III

# N- $k$ Secure Unit Commitment with Transmission Switching

### 3.1 Introduction

Recent blackout events have highlighted the need to have a power grid that is robust and reliable (*Liscouski and Elliot (2004), Srivastava et al. (2012)*). Currently, the North American Electricity Reliability Corporation (NERC) requires that the power grid be *N-1 secure*, meaning that load must be fully met in the event that any single component fails (*NERC (2011)*). The rationale for this policy is that the failure of a single component is considered to be a much more likely event than the near simultaneous failure of multiple components, and thus only single failures are considered when making operational planning decisions. However, given that component failures are not independent events, the probability of near-simultaneous failures may be higher than is currently estimated. The possibility of multiple failures is worth considering when making planning decisions.

Federal directives (*PPD-21 (2013)*) emphasize the importance of the security of the power grid as a critical infrastructure, and highlight the need to protect against major disruptions. Consequently there has been significant interest in considering reliability standards that are more stringent than N-1, such as N-2 or N-3, or more

generally,  $N-k$ , where the grid must be able to survive any simultaneous failure of  $k$  or fewer components. A failure event of one or more components is commonly called a *contingency*. The set of all contingencies under consideration greatly increases when multiple failures are included, and thus the task of making planning and operational decisions becomes much more challenging.

The *unit commitment problem (UC)* is the day-ahead planning problem in which generators are scheduled to be off or on for each hour in the coming day. The generators in the power grid have operational limits including constraints on their minimum up and down times and ramp rate limits. To meet the forecasted demand for a region, the on and off statuses of the generators must be planned ahead.

One way of including security requirements in the unit commitment problem is by specifying operating reserve (*Read (2010)*) which requires, for example, that the excess capacity of the committed generators be at least as much as the capacity of the largest generator. However, a requirement of this type does not take into account transmission constraints. Excess generating capacity in the event of a failure is useless if the transmission constraints do not allow power to be transported to where it is needed. An  $N-k$  secure generator schedule specifically considers how the transmission network constraints impact the available recourse actions in the event of a failure.

As is common in most optimization literature on grid planning, the power flow model in this Chapter is based on steady state analysis. To model the steady-state transmission network constraints in a power system, the Alternating Current Optimal Power Flow (ACOPF) equations are the ideal way to represent the physical laws. When several assumptions are made regarding stable operation, the ACOPF equations reduce to the linear DC power flow (DCPF) equations. The ACOPF equations are highly nonlinear, and thus optimization models typically use the DCPF equations as a linear approximation of the ACOPF equations. The DCPF equations are commonly used both in the academic literature and in industry (*Hedman et al. (2011)*)

and are used in our model.

Researchers have been exploring new “smart grid” technologies that improve the flexibility and efficiency of the operation of the power grid. In addressing grid congestion, there is a paradox associated with the existence of transmission capacity. On one hand, an arc in the network, i.e., a transmission line, allows power to be transmitted from one node to another, and thus can be useful in transmitting power from the generators to the consumers. On the other hand, given the laws of physics that govern how power flows throughout the network (i.e., Kirchhoff’s circuit laws), the existence of an arc imposes a constraint on the system. In certain situations, removing a line can be advantageous in redirecting the flows in the network.

In a transmission model which uses DCPF constraints, removing a transmission line corresponds to removing the DCPF constraint for that line. Specifically, these situations arise in networks where there are cycles. Physical laws require that, when multiple paths exist between nodes, power must flow along all available paths. One path may be a bottleneck which constrains the flow on other paths, thus removing a transmission line may increase throughput. Cycles are often purposely designed into the power network to ensure redundancy, so there are often situations in which temporarily removing a line would be useful. An example of this phenomenon is presented in *Hedman et al.* (2011).

*Transmission switching* is a practice where operators may open circuit breakers to switch transmission lines out of service to redirect the flow of power. This additional degree of control over the network topology has the potential to reduce the costs of dispatching generators and improve survivability of a contingency event. However, this additional set of switching decisions also introduces algorithmic challenges by dramatically increasing the dimension of the problem.

In this Chapter, we consider a unit commitment problem where  $N-k$  security is required, and transmission switching is allowed. A problem with this structure could

naturally be decomposed into a two-stage program with mixed binary variables in both stages. However, such a formulation cannot be solved by traditional methods, due to the existence of integer variables in the second stage and the very large number of scenarios. We present novel models and methods to address these challenges.

The outline of this Chapter is as follows. We first review the literature on solving power system operational problems with  $N-k$  security and on using transmission switching to control the power flows in a network in Section 3.2. We then formally define the  $N-k$  unit commitment problem with transmission switching in Section 3.3. In Section 3.4, the natural two-stage decomposition and then an alternative decomposition are presented. In Section 3.5, the Contingency Oracle is derived, which is used to identify unsurvivable contingencies. In Section 3.6, the complete algorithm is defined, and implementation details are described which improve run time. Computational experiments are presented in Section 3.7 for the IEEE24 and RTS-96 test systems which demonstrate the value of switching, and the cost tradeoff of increasing reliability. Finally, we present conclusions and ideas for future work in Section 3.8.

## 3.2 Literature Review

There has been significant work on network interdiction problems, and on various other ways of analyzing vulnerabilities in the power grid. *Shen et al. (2012)* explore three different interdiction models in which nodes are deleted to maximize disconnectivity. *Bienstock and Verma (2010)*, *Salmeron et al. (2004)* and *Salmeron et al. (2009)* present theoretical and computational results on solving the worst-case power system interdiction bilevel program. *Pinar et al. (2010)* propose that the worst-case power grid interdiction problem can be accurately approximated as a network inhibition problem, whose mixed-integer formulation can be solved for realistically-sized networks. *Fan et al. (2011)* present a critical node detection method for solving the power grid interdiction problem, and an economic basis for evaluating the damage

caused by contingency events.

Several papers solve planning or operational problems with the  $N-k$  security standard. *Street et al.* (2011) present a robust optimization framework for solving the single bus unit commitment problem (i.e., where transmission constraints do not exist) when survivability is required for any simultaneous failure of up to  $k$  generators. *Wang et al.* (2012) formulate the  $N-k$  unit commitment problem, where generators or transmission lines may fail, as a two-stage program and propose a cutting plane algorithm that solves for an exact solution.

The potential of transmission switching to significantly reduce the cost of dispatching generators is explored in *Fisher et al.* (2008). Following this work, *Hedman et al.* (2008) address the drawbacks and explore further the benefits of transmission switching as a corrective mechanism. *Hedman et al.* (2009) consider how transmission switching affects the costs of dispatching generators  $N-1$  securely, and find that not only would it be possible for  $N-1$  security to be maintained when transmission switching is used, but that the economic dispatch cost savings due to transmission switching are sometimes greater with  $N-1$  security requirements than without. *Li et al.* (2012) use a constraint programming approach to solve for switching actions that enable the power system to recover from a contingency event without redistributing generators.

Analysis of how the worst-case power system interdiction models could be extended to include transmission switching is presented in *Delgadillo et al.* (2010) and in *Zhao and Zeng* (2011). *Delgadillo et al.* (2010) present a method for solving the worst-case electric grid interdiction bilevel program with transmission switching allowed in the lower level problem by using Benders' decomposition within a restart framework. *Zhao and Zeng* (2011) present a tri-level reformulation of the bilevel interdiction problem with transmission switching in the lower level, which has an equivalent single level form that can be solved with a cutting plane algorithm.

Modifications of the unit commitment problem to incorporate transmission switch-

ing are presented in *Hedman et al. (2010)* and *Khodaei and Shahidehpour (2010)*. *Khodaei and Shahidehpour (2010)* present a solution methodology that iterates between finding the best unit commitment decision and best transmission switching decisions, and apply this method for a handful of specific contingency events. *Hedman et al. (2010)* present a model for the N-1 secure unit commitment problem, where switching decisions are made for each time period, but the switching decisions are not changed in response to a contingency event. The authors present a heuristic method of solving this problem, which shows a cost savings of 3.7% in the unit commitment solution when transmission switching is employed, compared to when transmission switching is not used, for the RTS-96 test system. An economic analysis of the impact of transmission switching on the N-1 unit commitment problem is presented in *O'Neill et al. (2010)*. All of these papers on transmission switching and the unit commitment problem also use the DCPF equations to model power flows.

The N- $k$  secure unit commitment problem considered here could be classified as an adaptive robust problem (*Ben-Tal et al. (2004)*) with an uncertainty set defined as the set of all contingencies of size  $k$  or smaller. Herein we propose a formulation for the robust unit commitment problem which is similar in structure to a stochastic unit commitment problem with a finite number of contingency scenarios. However, our proposed algorithm does not require that all scenarios be explicitly enumerated, unlike other methods for solving two stage stochastic programs such as progressive hedging (*Watson and Woodruff (2011)*). When considering contingencies of size greater than 1, the number of contingencies is likely to be extremely large due to the combinatorial explosion, and thus it is necessary to develop a method which does not explicitly consider all scenarios. Our algorithm takes advantage of the specific structure of the problem and provides a tractable way of solving a problem that would otherwise be too large to solve with traditional methods. We also present several important implementation details which significantly impact the runtime of

the overall algorithm, as demonstrated by our computational results.

### 3.3 Problem Definition

Our ultimate goal is to solve for a set of unit commitment decisions and generator dispatch decisions for normal operating conditions. These decisions should minimize the total cost of normal operation but must also be able to survive any contingency of size  $k$  or smaller, where a contingency is defined as the simultaneous failure of one or more components. In response to a contingency event, operators have the opportunity to redispatch generators which are already committed, and to switch transmission lines out as needed.

#### 3.3.1 Assumptions

The following assumptions are made in order to construct a model that is realistic but also solvable.

- **Components fail completely or not at all.**

Partial failures are not considered.

- **Only transmission lines and/or generators may fail.**

We model only generators, transmission lines (i.e. arcs), and buses (i.e. nodes) in our representation of the power grid. An illustration of the IEEE14 test system is provided in Figure 3.1. One or more generators may be on a single bus. During a contingency event, it is assumed that only transmission lines and/or generators may fail. The failed elements that define a contingency event are said to be contained within the contingency.

- **For a contingency of size  $l$ , we define survival as meeting at least  $(1 - \epsilon_l)$  fraction of the total demand.**



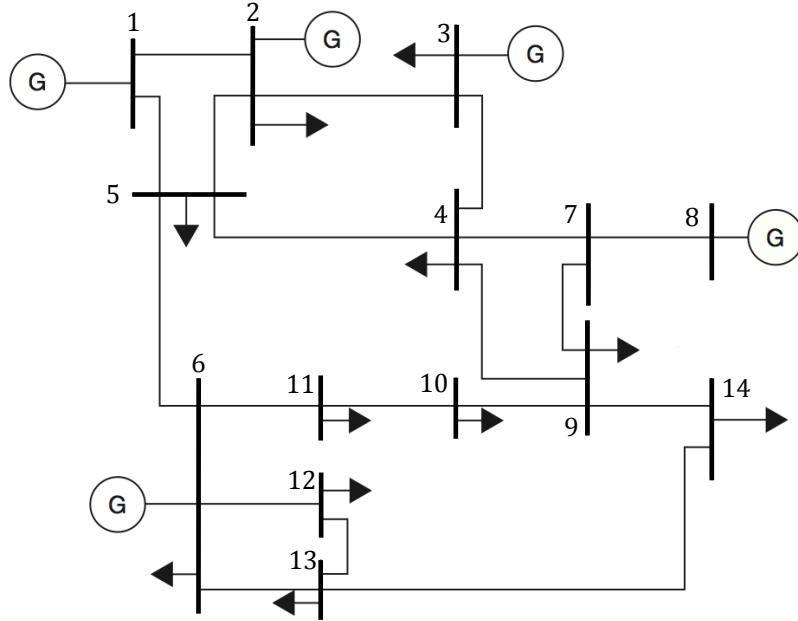


Figure 3.1: Illustration of IEEE14 test system. Bold lines are buses, regular lines are transmission lines, and circles with “G” inside are generators.

The parameter  $\epsilon_l$  is defined for  $l = 0, \dots, k$  such that  $0 \leq \epsilon_0 \leq \epsilon_1 \leq \dots \leq \epsilon_k \leq 1$ .

It is common to set  $\epsilon_0 = 0$  and  $\epsilon_1 = 0$ .

- **Time is discretized at one hour intervals.**

Multiple failures within the interval are considered simultaneous.

- **A contingency event is assumed to occur when the system is otherwise operating normally.**

Contingency events in sequential time periods, or cascading failures are not considered. When a contingency occurs in a particular time period, the generators that were already committed in that time period can be redispatched, but generators that were not committed cannot be turned on.

- **To respond to a contingency in a given time period, the generator’s output cannot be increased or decreased from the nominal output in that same time period by more than the ramp rate.**

- **For a single contingency, the post-contingency generator outputs are not linked across time periods.**

When a contingency occurs, the primary concern is immediately finding a feasible power flow solution so that a blackout event will not occur. Realistically, in subsequent time periods, the operator may take other actions to enable recovery including repairing broken components, bringing online generators that were previously uncommitted, etc. But for the purposes of the problem considered here, the only requirement is that a feasible power flow solution exists immediately following a contingency event. Subsequent time periods are not modeled, as it is assumed that once a stable solution has been found, the operator is able to recover using actions beyond just redispatching generators and switching transmission lines.

- **Transmission switching may be used in response to a contingency event but not during normal operation.**

Previous studies (*Fisher et al. (2008)*, *Hedman et al. (2010)*) have shown that transmission switching can be employed during normal operation to reduce the cost of dispatching generators by optimizing the network topology to allow the most efficient generators to meet demand. Other studies (*Hedman et al. (2011)*, *Li et al. (2012)*) have indicated that switching might also be used as recourse action, to help redirect flows in response to a contingency event to satisfy as much demand as possible. We focus here on the effect of transmission switching on system reliability, and thus we consider only the latter use case, in which transmission switching is used in response to a contingency event to improve the network's ability to survive the contingency. It is trivial to extend our model to allow switching during normal operation.

- **No cost is assigned to post-contingency response decisions.**

When a contingency occurs, the primary goal is to ensure feasibility, not to minimize the cost of operation. Thus, generator dispatch decisions under normal operation appear in the objective function, but post-contingency dispatch decisions do not. The decision to switch a line in or out of service is also not assigned any cost.

### 3.3.2 Explicit Formulation

Here we present the explicit formulation of the  $N-k$  unit commitment problem with transmission switching. For notational conciseness and clarity we present the explicit formulation of the problem using matrix notation. The full, detailed formulation is presented in Appendix A.

Let  $\mathcal{C}$  be the set of all contingencies of size  $k$  or smaller. Each contingency  $c \in \mathcal{C}$  is a binary vector of length equal to the number of generators and transmission lines, where  $c_e = 1$  indicates that element  $e$  has failed. More formally, let  $\mathcal{E}$  represent the set of transmission lines, and  $\mathcal{G}$  represent the set of generators. The set of all contingencies  $\mathcal{C} = \{c \in \{0, 1\}^{|\mathcal{E}|+|\mathcal{G}|} : e^T c \leq k\}$  where  $e$  is an appropriately sized unit vector.

Let  $i(c)$  be a function that maps the contingency  $c$  to its corresponding index in the set  $\mathcal{C}$ .  $i(c) \in \{0, 1, \dots, |\mathcal{C}|-1\}$  for any  $c \in \mathcal{C}$ . The set  $\mathcal{C}$  contains the 0-contingency,  $c = \mathbf{0}$ , is where no components have failed, i.e. normal operation. Let  $i(\mathbf{0}) = 0$ .  $\mathcal{T}$  is the set of 24 1-hr time periods, and  $b^t$  is the total load in time period  $t$ .

The vectors of variables used in this problem are:

- $x^t$  vector of binary unit commitment decisions including on/off and start-up/shut-down statuses of each generator in time period  $t$
- $p^{t,i(c)}$  vector of generator dispatch (i.e., output) decisions in time period  $t$  during contingency  $c$
- $f^{t,i(c)}$  vector of operational decisions in time period  $t$  during contingency  $c$  including line flows, node phase angles, and load shedding at each node
- $w^{t,i(c)}$  vector of binary transmission switching variables in time period  $t$  during contingency  $c$

Let  $p^0$  be a concatenation of  $p^{t,0}$  for all  $t \in \mathcal{T}$ , and  $x$  be a concatenation of  $x^t$  decisions for all  $t \in \mathcal{T}$ . The complete formulation is as follows:

$$\min_{x,p,f,w} d_x^T x + d_p^T p^0 \quad (3.1a)$$

$$\text{s.t.} \quad Ux + Qp^0 \leq q \quad (3.1b)$$

$$Af^{t,0} + Gp^{t,0} \leq r^t \quad \forall t \in \mathcal{T} \quad (3.1c)$$

$$Af^{t,i(c)} + (e - c)^T Bw^{t,i(c)} + Gp^{t,i(c)} \leq Hc + r^t \quad \forall c \in \mathcal{C} \setminus \{\mathbf{0}\}, t \in \mathcal{T} \quad (3.1d)$$

$$Yp^{t,i(c)} - (e - c)^T Dx^t \leq 0 \quad \forall c \in \mathcal{C}, t \in \mathcal{T} \quad (3.1e)$$

$$h^T f^{t,i(c)} \leq b^t \epsilon_{\epsilon^T c} \quad \forall c \in \mathcal{C}, t \in \mathcal{T} \quad (3.1f)$$

$$W(p^{t,i(c)} - p^{t,0}) \leq Vc + s \quad \forall c \in \mathcal{C} \setminus \{\mathbf{0}\}, t \in \mathcal{T} \quad (3.1g)$$

$$p \geq 0 \quad (3.1h)$$

$$x \text{ binary} \quad (3.1i)$$

$$w^{t,i(c)} \text{ binary } \forall c \in \mathcal{C} \setminus \{\mathbf{0}\}, t \in \mathcal{T} \quad (3.1j)$$

The objective function (3.1a) minimizes the total cost of operating the generators including start-up and shut-down costs and fuel costs under normal operating

conditions (i.e., the 0-contingency). We assume a linear fuel cost function, but a piecewise linear approximation of a quadratic cost curve could also be used, as is common with generator fuel costs (*Zhu (2009)*). Constraint set (3.1b) defines the requirements for the unit commitment variables including start-up, shut-down, and minimum up and down time, as well as the ramping constraints on the power dispatch variables under normal operation, which restrict the increase or decrease in the power output in consecutive time periods to obey limitations imposed by the equipment. Constraints (3.1c) and (3.1e) define the operational constraints in the 0-contingency, and constraints (3.1d) and (3.1e) define the operational constraints in contingency  $c$ . Constraint set (3.1d) includes power flow balance, DCPF constraints on available transmission lines, capacities on line flows, and bounds on node phase angles. When a line is contained in a contingency, the power flow on that line is forced to be 0, and the DCPF constraints for that line are not enforced. If a line is not contained in a contingency, but it is switched out, the power flow is similarly set to 0 and the DCPF constraints relaxed. Note that some constraints in this set depend on the particular time period (e.g., flow balance depends on time-dependent forecasted loads) and some constraints depend on the contingency (e.g., line capacities depend on whether the line is contained in a contingency). Constraint set (3.1c) contains the same operational constraints as in (3.1d) except that in the 0-contingency, all generators and transmission lines are available, and the switching variables are not included, because in this model switching is not allowed during normal operation. If switching were to be allowed during normal operation, this constraint set would be appropriately modified, and the same solution approach would be valid. Constraint set (3.1e) defines the bounds on the power output at each generator. The power output at a generator is restricted to be 0 if either the generator is not committed, or if the generator has failed in a particular contingency. Otherwise, the power output at a committed generator must be within the upper and lower output bounds. Constraint set (3.1f) requires

that the total loss-of-load be less than a specified threshold, where the threshold is a function of the size of the contingency. Constraint (3.1g) specifies that the redispatched power outputs must obey ramping limits relative to the 0-contingency power dispatch decisions prior to the contingency event, where the vector  $s$  contains the ramping limits, and the term  $Vc$  relaxes the limit on the post-contingency dispatch for a generator that is contained in a contingency.

### 3.4 Problem Decomposition

The full mixed-integer formulation (3.1) is typically very challenging to solve because the set of all contingencies  $\mathcal{C}$  is very large even for moderately sized networks if  $k > 1$ . The total number of contingencies is  $\sum_{l=1}^k \binom{|\mathcal{E}|+|\mathcal{G}|}{l}$ , which is on the order of  $(|\mathcal{E}| + |\mathcal{G}|)^k$ , assuming  $(|\mathcal{E}| + |\mathcal{G}|) \gg k$ . For example, for the RTS-96 test system used in our computational results, the number of transmission lines is 117 and the number of generators is 96. The number of contingencies of size 1 is 213, while the number of contingencies of size 2 is 22,578. Thus, we explore decomposition procedures that allow us to solve this MIP.

#### 3.4.1 Natural Two-Stage Decomposition

The natural decomposition of this problem follows from defining the set of scenarios to be all contingency-time period pairs, the first stage variables to be  $x$ ,  $p^0$ , and  $f^0$ , and the second stage variables to be  $p^{t,i(c)}$ ,  $f^{t,i(c)}$  and  $w^{t,i(c)}$ . The first stage

problem is then:

$$\begin{aligned}
& \min_{x, p^0, f^0} \quad d_x^T x + d_p^T p^0 \\
& \text{s.t.} \quad (3.1\text{b})\text{--}(3.1\text{c}) \\
& \quad Y p^{t,0} - e^T D x^t \leq 0 \quad \forall t \in \mathcal{T} \\
& \quad h^T f^{t,0} \leq 0 \quad \forall t \in \mathcal{T} \\
& \quad p^0 \geq 0, \quad x \text{ binary} \\
& \quad \mathcal{F}^{t,i(c)}(x, p^0) \text{ nonempty} \quad \forall t \in \mathcal{T}, c \in \mathcal{C} \setminus \{\mathbf{0}\}
\end{aligned} \tag{3.2}$$

The second stage feasibility problem for a particular contingency-time period pair is defined by the polyhedron  $\mathcal{F}^{t,i(c)}(x, p^0)$ . If this polyhedron is nonempty for a first stage solution  $(x, p^0)$ , for all contingencies and time periods, then all contingencies are survivable. This polyhedron will be referred to as the Unsurvivability Authenticator (UA).

$$(UA) \quad \mathcal{F}^{t,i(c)}(x, p^0) = \begin{cases} A f^{t,i(c)} + (e - c)^T B w^{t,i(c)} + G p^{t,i(c)} \leq H c + r^t \\ Y p^{t,i(c)} \leq (e - c)^T D x^t \\ h^T f^{t,i(c)} \leq b^t \epsilon_e \tau_c \\ W p^{t,i(c)} \leq V c + s + W p^{t,0} \\ p^{t,i(c)} \geq 0 \\ w^{t,i(c)} \text{ binary} \end{cases} \tag{3.3}$$

In this two-stage formulation, there exist binary variables  $w^{t,i(c)}$  in the second stage problem (3.3). Standard methods for solving stochastic programs cannot be used when there are integer variables in the second stage. There exist several methods for solving problems with second stage integer variables using disjunctive cuts or Fenchel cuts (*Ntaimo (2013), Sen and Sherali (2006), Sherali and Fraticelli (2002)*).

These approaches involve generating cutting planes for the second stage to define the convex hull, and thus tend to be computationally intensive and not scalable. Alternatively, other authors such as *Shen and Smith* (2013) have been successful in developing problem-specific decomposition approaches that use specific problem structure to develop valid procedures for solving problems with second stage integer variables. *Shen and Smith* (2013) propose a decomposition procedure to solve a broadcast domination network design problem, where the second stage broadcast domination decisions are required to be integer-valued.

We propose an approach in which we take advantage of the specific structure of our problem, and suggest a novel reformulation in which the binary switching variables  $w^{t,i(c)}$  are moved into the first stage. *Khodaei et al.* (2010) employ a similar technique in their model of a transmission expansion planning problem where transmission switching is used to reduce generator dispatch costs.

Once the switching variables have been moved into the first stage, the reformulated problem has a linear second stage problem, and thus a Benders' decomposition could be applied. However, this reformulation would result in a very large number of variables in the first stage problem (one switching variable for each transmission line for each contingency for each time period). We propose a procedure for solving this reformulation in which the switching variables are dynamically generated for the first stage problem on an as-needed basis. With this approach, the number of switching variables contained in the first stage problem is initially zero and grows slowly as cutting planes are added.

### 3.4.2 Reformulation and Cutting Plane Algorithm

In our reformulation, the master problem remains almost the same as (3.2) except that there exists a set of binary vectors of variables  $w^{t,i(c)}$  for all  $t \in \mathcal{T}$  and  $c \in \mathcal{C}$ ,



and second stage feasibility is enforced instead with:

$$\mathcal{F}^{t,i(c)}(x, p^0, w^{t,i(c)}) \text{ nonempty} \quad \forall t \in \mathcal{T}, c \in \mathcal{C} \setminus \{\mathbf{0}\} \quad (3.4)$$

where

$$\mathcal{F}^{t,i(c)}(x, p^0, w^{t,i(c)}) = \begin{cases} Af^{t,i(c)} + Gp^{t,i(c)} \leq Hc + r^t - (e - c)^T Bw^{t,i(c)} & (\pi) \\ Yp^{t,i(c)} \leq (e - c)^T Dx^t & (\beta) \\ h^T f^{t,i(c)} \leq b^t \epsilon_{e^T c} & (\gamma) \\ Wp^{t,i(c)} \leq Vc + s + Wp^{t,0} & (\rho) \\ p^{t,i(c)} \geq 0 & \end{cases} \quad (3.5)$$

We employ a Benders based approach where this second stage feasibility requirement (3.4) is initially relaxed, and then gradually enforced by adding Benders' feasibility cuts to the master problem.

In traditional Benders' decomposition, cutting planes would be generated for the master problem by solving a subproblem for each time period, for each contingency, in each iteration. Due to the large size of the set of contingencies when  $k > 1$ , this procedure is not viable because it would take an impractically long time to solve so many subproblems in each iteration. For example, for the RTS-96 test system used in the computational results, the number of contingencies considered when  $k = 2$  is 22,791. A subproblem would be solved for each contingency and each time period, so 546,984 subproblems would be solved in each iteration. If each subproblem took 0.1 seconds to solve, then it would take over 15 hours just to solve all of the subproblems for a single iteration.

To address this issue, we propose that a *Contingency Oracle* be used which identifies an unsurvivable contingency for the current unit commitment solution for a particular time period. The development of this oracle will be further explained in

the next section 3.5, but let us for now assume that such an oracle exists. The Contingency Oracle provides a means for identifying violated constraints for the master problem even when there is a very large number of contingencies.

The overall algorithm which incorporates the Contingency Oracle is illustrated in Figure 3.2. For every time period  $t$ , the unit commitment decisions  $x^t$  and 0-contingency economic dispatch decisions  $p^{t,0}$  are passed to the Contingency Oracle. If, in all time periods, the Contingency Oracle identifies that all contingencies are survivable, the overall algorithm exits with the optimal set of unit commitment decisions  $x$  and 0-contingency economic dispatch decisions  $p^0$ . If, for at least one time period, the Contingency Oracle identifies an unsurvivable contingency, this unsurvivable contingency is passed to a subproblem, along with the current master problem solution. The subproblem solution is then used to generate a feasibility cut to the master problem, and the procedure repeats.

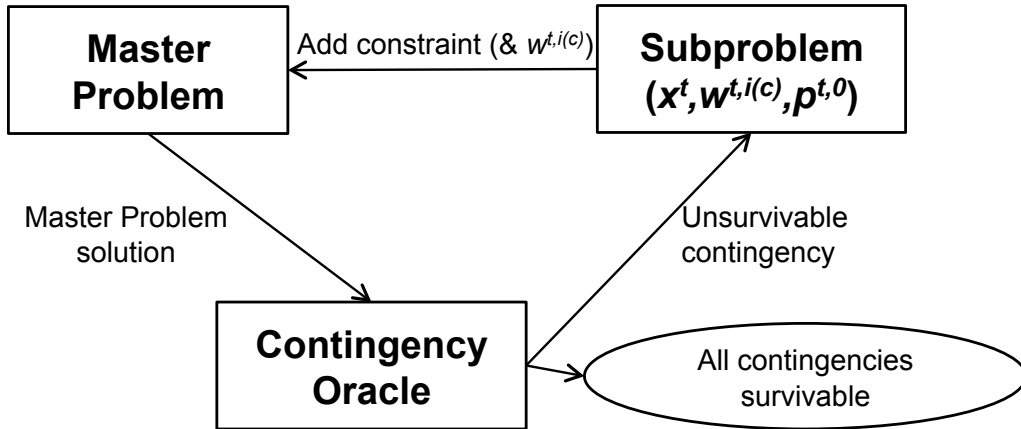


Figure 3.2: Algorithm Overview for N-k UC with Transmission Switching

Once an unsurvivable contingency has been identified for a particular time period, a feasibility cut is generated for the master problem by solving the dual of (3.5). The corresponding dual variables are denoted next to each constraint in (3.5).

For the unsurvivable contingency  $c$  in time period  $t$ , the feasibility cut takes the

following form, where  $(\bar{\pi}, \bar{\beta}, \bar{\gamma}, \bar{\rho})$  is the optimal dual subproblem solution.

$$\bar{\pi}^T(Hc + r^t - (e - c)^T Bw^{t,i(c)}) + \bar{\beta}^T((e - c)^T Dx^t) + b^t \epsilon_{ex_c} \bar{\gamma} + \bar{\rho}^T(Vc + s + Wp^{t,0}) \leq 0 \quad (3.6)$$

We recognize that initially, no feasibility cuts of the form (3.6) exist in the master problem, and all switching variables are unconstrained. Any first stage variables that are not contained in any constraints can effectively be ignored. As feasibility cuts of the form (3.6) are generated for the master problem, each of which contains a set of switching variables  $w^{t,i(c)}$ , we suggest that the relevant vector of variables  $w^{t,i(c)}$  be added to the formulation. Thus, the number of switching variables effectively in the master problem grows gradually as cutting planes are generated for the master problem.

Using this procedure of dynamically generating switching variables for the master problem, we note that there is a choice to make when passing the master problem solution to the subproblem. Once an unsurvivable contingency  $c$  has been identified for time period  $t$ , two cases are possible:

1. At least one feasibility cut (3.6) for the given time period  $t$  and contingency  $c$  has already been added to the master problem. Thus, the vector of variables  $w^{t,i(c)}$  is contained in at least one constraint in the current master problem and therefore has been assigned a value in the solution to the master problem. The master problem solution  $(x^t, p^{t,0}, w^{t,i(c)})$  should be passed to the subproblem (3.5).
2. No constraints from the set (3.6) for the given time period  $t$  and contingency  $c$  have yet been added to the master problem. The vector of variables  $w^{t,i(c)}$  is not yet contained in any master problem constraints, and thus any binary vector is a feasible solution for  $w^{t,i(c)}$ . The master problem solution  $(x^t, p^{t,0})$  is

passed to the subproblem (3.5), and any arbitrary binary vector can be set for  $w^{t,i(c)}$ .

### 3.5 Contingency Oracle

The purpose of the Contingency Oracle is to identify, for a particular time period  $t$  with unit commitment decisions  $x^t$  and 0-contingency economic dispatch decisions  $p^{t,0}$ , a contingency for which the minimum loss-of-load exceeds the allowable threshold, even when the network operator has the opportunity to redispatch generators and switch lines out of service in response to the contingency. If such a contingency does not exist, the oracle provides a certificate that all contingencies of size  $k$  or smaller are survivable.

We note that it is significant that the switching decisions determined in the master problem are *not* passed to the oracle. The overall goal is to determine unit commitment and 0-contingency dispatch decisions such that there is guaranteed to exist a feasible operating solution in the event of any contingency of size  $k$  or smaller. It is not necessary to know what the recovery solution is for every contingency, it is sufficient to just know that one exists. The oracle seeks a contingency that is unsurvivable even with the best switching configuration, not just with the switching configurations that are in the current master problem solution. This distinction means that the oracle will identify fewer unsurvivable contingencies than would be identified in a traditional Benders' approach. Fewer unsurvivable contingencies identified results in fewer feasibility cuts and fewer switching variables added to the master problem.

In this section, we first formulate a bilevel program that identifies the contingency that causes the maximum loss-of-load, even after the operator makes the optimal recovery decisions. We recognize that this bilevel program with mixed binary lower level decisions is difficult to solve (*DeNegre and Ralphs (2009)*, *Scaparra and Church (2008)*). But if transmission switching is ignored, the lower level problem becomes an

LP and the bilevel program can be reformulated as a relatively small single MIP.

We next present an iterative constraint generation algorithm that uses this observation to our advantage. Specifically, we observe that if a contingency can be survived without switching then clearly it can also be survived when switching is allowed. Thus, we use the no-switching bilevel program formulated as a single MIP to initially identify candidate unsurvivable contingencies; this MIP is referred to as the Candidate Contingency Identified (CCI). Once such a candidate has been identified we then verify whether the contingency is unsurvivable when switching is allowed. If the candidate unsurvivable contingency is survivable when switching is allowed, a constraint is generated for CCI. The constraint generation procedure continues until either an unsurvivable contingency is identified, or CCI certifies that no unsurvivable contingencies exist for the given  $(x^t, p^{t,0})$ . The unsurvivable contingency that is identified with this routine, if one exists, is used to generate a valid feasibility cut for the master problem.

### 3.5.1 Bilevel Program

For each time period  $t$ , we could pose a bilevel program in order to identify a maximally damaging contingency given the current first stage decisions  $x$  and  $p^0$ . This bilevel program could be thought of as an adversary's problem, where the adversary seeks to maximize the minimum loss-of-load. The adversary would decide which elements of the system to destroy, knowing that the system operator would have the opportunity to respond to the adversary's decision and would seek to minimize the loss-of-load.

Specifically, the upper level problem (i.e., the adversary) would determine which generators and/or transmission lines to destroy for a given time period. The lower level problem (i.e., the system operator) would determine how best to dispatch power and switch lines in response to this contingency event so as to minimize loss-of-load.

The optimal solution to the bilevel program would be a contingency which maximizes the minimum loss-of-load for the given time period.

We could attempt to solve the described bilevel program. However, the existence of binary switching variables in the lower level problem would mean the bilevel program has mixed-integer variables in both the upper and lower levels, which is known to be a very difficult class of problem (*DeNegre and Ralphs (2009), Scaparra and Church (2008)*). Furthermore, we would need to solve this difficult problem in each master problem iteration, for each time period. On the other hand, if transmission switching were not allowed in the lower level problem, the lower level problem would become an LP, and the bilevel program could be reformulated as a single, relatively small MIP, the CCI.

### 3.5.2 Bilevel Program Without Lower Level Switching Decisions

The bilevel program for identifying contingencies that maximize the minimum loss-of-load above the allowable threshold when switching is *not* permitted is formulated as follows.

$$\max_c L(x^t, p^{t,0}, c) \tag{3.7a}$$

$$\text{s.t. } e^T c \leq k \tag{3.7b}$$

$L(x^t, p^{t,0}, c)$  is the optimal objective value of the no-switching lower level problem, which minimizes the loss-of-load above the allowable threshold, given the inputs  $x^t, p^{t,0}, c$ . In this no-switching lower level problem, the operator has the option of redispatching generators in response to the contingency event  $c$ , but does *not* have the option of switching transmission lines out of service.

This no-switching lower level problem is as follows. Note that the indices for contingency and time period have been omitted from the variables  $f$  and  $p$  for the

sake of simplicity, but it should be understood that the lower level problem (3.8) is specific to a particular contingency-time period pair. The Contingency Oracle is called for a particular time period, and in that time period the upper level passes a contingency to the lower level problem.

$$L(x^t, p^{t,0}, c) = \min_{f,p} h^T f - b^t \epsilon_e^T c \quad (3.8a)$$

$$\text{s.t. } Af + Gp \leq Hc + r^t \quad (\pi) \quad (3.8b)$$

$$Yp \leq (e - c)^T D x^t \quad (\beta) \quad (3.8c)$$

$$Wp \leq Vc + s + Wp^{t,0} \quad (\rho) \quad (3.8d)$$

$$p \geq 0 \quad (3.8e)$$

Here we assume that the problem (3.8) has at least one feasible solution for any  $(x^t, p^{t,0}, c)$ . This assumption holds, for example, if the lower bound on committed generator output is equal to 0 for all generators because shedding all load is always a feasible solution (associated with setting all generation and power flows to zero). If there does not exist a feasible solution for any  $(x^t, p^{t,0}, c)$ , the formulation can be modified to ensure feasibility by defining slack variables for loss-of-load and excess generation, and appropriately modifying the objective such that the non-negative slack variables are minimized. It is assumed here that these slack variables are not needed to ensure feasibility.

A bilevel program with a linear lower level problem is traditionally solved by incorporating the upper level variables and constraints into the dual of the lower level problem. The upper level problem and the dual of the lower level problem have aligned objectives, and thus the bilevel program can be solved as a single optimization problem. The single optimization problem for solving the no-switching bilevel program is as follows, where the binary contingency variable vector  $c$  is added to the

dual of the no-switching lower level problem.

$$\max_{c, \pi, \beta, \rho} (Hc + r^t)^T \pi + ((e - c)^T Dx^t)^T \beta + (Vc + s + Wp^{t,0})^T \rho - b^t \epsilon_t \quad (3.9a)$$

$$\text{s.t. } \pi^T A = h^T \quad (f) \quad (3.9b)$$

$$\pi^T G + \beta^T Y + \rho^T W \leq 0^T \quad (p) \quad (3.9c)$$

$$\pi, \beta, \rho \leq 0, \quad c \text{ binary} \quad (3.9d)$$

$$e^T c = l \quad (3.9e)$$

Note that in constraint (3.9e), the size of the contingency is fixed to be of size  $l$ . If the size of the contingency were not fixed, and constraint (3.9e) were replaced with the requirement that the contingency be of size less than or equal to  $k$  ( $e^T c \leq k$ ), the size of the contingency would not be known *a priori*, and it would be unclear what value  $\epsilon$  should be used in the objective (3.9a). However, because  $k$  is typically a small value, such as 1, 2 or 3, we can use a procedure of fixing the size of the contingency  $l$  to progressively larger values, up to the value  $k$ . For example, we first restrict the size of contingency to be 1, and if no unsurvivable contingency of size 1 is found, then we change this constraint to consider contingencies of size 2, and so on, until an unsurvivable contingency is found, or it has been verified that there do not exist any contingencies of size  $k$  or smaller that are unsurvivable.

In its current form the objective (3.9a) contains bilinear terms, as the binary variables in  $c$  are multiplied by the dual variables  $\pi$ ,  $\beta$ , and  $\rho$ . However, the objective can be linearized by standard methods, which involve replacing these bilinear terms with auxiliary variables and adding appropriate constraints to enforce a relationship that the auxiliary variables take on the same value as the original bilinear terms. The formulation (3.9) in its linearized form will be referred to as the CCI.



### 3.5.3 Contingency Oracle Solution Routine

To identify a contingency that is unsurvivable given the unit commitment decisions  $x^t$  and 0-contingency economic dispatch decisions  $p^{t,0}$ , we propose an iterative constraint generation algorithm. This basic algorithm is illustrated in Figure 3.3.

For a particular contingency size  $l$ , CCI is first solved to identify an initial unsurvivable contingency candidate for the current unit commitment decisions  $x^t$  and 0-contingency economic dispatch decisions  $p^{t,0}$ . The unsurvivability of this contingency is checked by solving the Unsurvivability Authenticator (UA), the feasibility problem defined in (3.3). If UA is feasible, a constraint is added to CCI to make the current contingency solution infeasible. The procedure repeats until a contingency is identified that is unsurvivable, even for the optimal switching configuration, or a certification that no unsurvivable contingency of size  $l$  exists is returned. This certification is obtained if the optimal objective value of CCI is greater than 0. If this certification is returned for all  $l = 1, \dots, k$ , all contingencies of size  $k$  or smaller are survivable for the current first stage solution for the given time period. If, for all time periods, the Contingency Oracle verifies that no unsurvivable contingencies exist, the overall algorithm terminates with the optimal set of unit commitment and 0-contingency economic dispatch decisions.

When UA verifies that a particular contingency  $c$  is survivable, one valid inequality that could be added to CCI to make the current contingency solution  $c$  infeasible is to require that at least one element that is not included in the contingency  $c$  must be destructed. However, a tighter constraint is one that utilizes information about which elements were used in the feasible UA solution. Consider that for a survivable contingency  $c$  for time period  $t$ , the UA solution indicates a feasible set of edge flows and generator outputs. Let the vector of binary parameters  $u$  indicate which lines have nonzero flows and which generators have nonzero power outputs in the feasible solution to UA. In the next iteration, if none of the lines that had nonzero flows

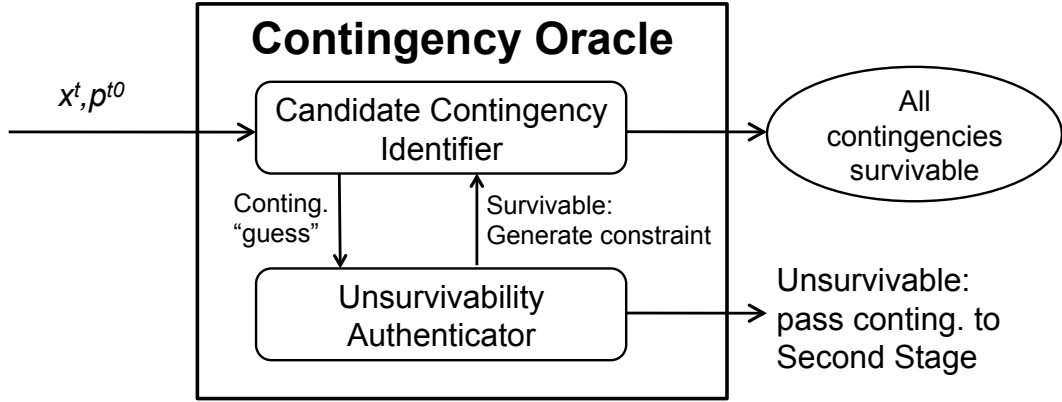


Figure 3.3: Contingency Oracle Solution Routine

and none of the generators that had nonzero power outputs are destructed, then the same solution to UA will be feasible. Thus, in order to identify an unsurvivable contingency, at least one line with nonzero flow or one generator with nonzero output must be destroyed, which is expressed in the following constraint:

$$u^T c \geq 1 \tag{3.10}$$

Thus we add constraint (3.10) to CCI based on the solution of UA to rule out survivable contingencies.

The Contingency Oracle takes in the unit commitment and 0-contingency dispatch decisions from the master problem  $(x, p^0)$  and returns an unsurvivable contingency  $c$ . The algorithm for solving the Contingency Oracle for a particular time period  $t$ , as illustrated in Figure 3.3, is summarized here.

1. Set  $l = 1$ .
2. Solve CCI with contingency size  $l$  to find solution  $c$ .
3. If the optimal objective value of CCI is greater than 0, increase  $l = l+1$ . If  $l > k$ , exit the Contingency Oracle for this time period  $t$  because all contingencies are

- survivable for the current  $x^t$  and  $p^{t,0}$ . Otherwise, if  $l \leq k$ , go to step 2.
4. Otherwise, pass the current first stage solution  $(x^t, p^{t,0})$  and current contingency  $c$  to UA (3.3), and solve for a feasible solution, if one exists.
  5. If a feasible solution for UA exists, compute the set of edges and generators used in the feasible solution and construct the indicator vector  $u$ . Add the following constraint to CCI:  $u^T c \geq 1$ . Go to step 2.
  6. If UA is infeasible, exit the Contingency Oracle with the unsurvivable contingency  $c$ .

## 3.6 Implementation Details

As described in section 3.4.2, the overall algorithm proceeds by iteratively solving the master problem, identifying unsurvivable contingencies, and solving the subproblem to generate feasibility cuts for the master problem. In this section, we discuss two algorithmic design decisions that have a significant impact on runtime, and we describe the implementation that we have empirically found to work well.

### 3.6.1 Identifying Unsurvivable Contingencies

Given the current master problem solution, the Contingency Oracle can identify an unsurvivable contingency for a particular time period, if one exists. However, the Contingency Oracle routine described in section 3.5.3 is an iterative procedure that involves generating constraints for CCI. The constraints generated for CCI are relatively weak, and so it is not uncommon for the routine to require many iterations, especially towards the end of the algorithm, when there do not exist many unsurvivable contingencies.

Rather than immediately calling the Contingency Oracle to identify an unsurvivable contingency, we suggest that a list of contingencies previously identified as

unsurvivable first be checked. For many contingencies, multiple feasibility cuts must be added to the master problem before survivability is achieved. Contingencies that have been previously identified as unsurvivable are thus good candidates for unsurvivability in future iterations. For a given master problem solution, we suggest checking all time periods. For each time period, we first check whether any contingencies in the list are unsurvivable. The Contingency Oracle is only called if all contingencies in the list are survivable for the time period *and* an unsurvivable contingency has not yet been identified for the current master problem solution. This routine reduces the frequency with which the Contingency Oracle is called while still ensuring that feasibility cuts are generated for the master problem in every iteration.

### 3.6.2 Ordered Time Periods

We also suggest that the time periods be ordered by decreasing total load. The time periods are checked in their ranked order. It is more likely that an unsurvivable contingency will exist for peak load time periods, so by checking these time periods early in the iteration, unsurvivable contingencies are identified sooner. A new unsurvivable contingency is immediately added to the list of contingencies. When the longer list of contingencies is checked for subsequent time periods, there is greater likelihood of generating a feasibility cut.

## 3.7 Computational Results

Our computational results were performed on a computer with 4GB RAM and a 2.3 GHz processor, using CPLEX v12.4. Computational tests were done with the IEEE24 and RTS-96 test systems, which are available online (*Grigg et al. (1999)*). In our test instances we modified the original network in the same way as described in *Hedman et al. (2010)*, with the intent of slightly increasing congestion. See *Hedman et al. (2010)* for details.

The characteristics of these networks are summarized in Table 3.1. Note that “# Conting.  $k = 1$ ” is the number of contingencies of size 1, which is the total number of transmission lines (i.e., arcs) and generators. Additionally, “# Conting.  $k = 2$ ”, is the number of contingencies of size 2 (total number of transmission lines and generators choose 2) plus the number of contingencies of size 1, because setting  $k = 2$  means protecting all contingencies of size 2 or smaller.

System	# Nodes	# Arcs	# Generators	# Loads	# Conting. $k = 1$	# Conting. $k = 2$
IEEE24	24	37	32	17	69	2,415
RTS-96	73	117	96	51	213	22,791

Table 3.1: IEEE24 and RTS-96 System Characteristics

### 3.7.1 Run Times

To perform our computational experiments, we needed to pick a value for  $\epsilon_k$ , the fraction of the allowable loss-of-load for contingencies of size  $k$ . In practice,  $\epsilon_0 = \epsilon_1 = 0$ , but there is not an established value for  $\epsilon_k$  for  $k > 1$ . To obtain the most meaningful results, we sought the tightest values of  $\epsilon_k$ , where the system is operating the closest to its limits. We refer to the smallest  $\epsilon_k$  value that yields a feasible  $N-k$  unit commitment solution as the *critical*  $\epsilon_k$ . For  $k = 1$  for the IEEE24 and RTS-96 systems, we initially set  $\epsilon_1 = 0$  and run our algorithm. If the  $N-k$  unit commitment problem was infeasible, we increased  $\epsilon_k$  by increments of 0.01 until a feasible  $N-k$  secure unit commitment solution was obtained. For  $k = 2$ , we held  $\epsilon_1$  at its critical value, and followed the same procedure to identify the critical  $\epsilon_2$ . The critical  $\epsilon_k$  values are for IEEE24 and RTS-96 for  $k = 1$  and  $k = 2$  are shown in Table 3.2.

We tried computing the critical  $\epsilon_k$  for  $k = 3$  for IEEE24, but no feasible  $N-k$  secure unit commitment solution could be obtained for any value of  $\epsilon_3 < 0.5$ . It is unrealistic to consider a 50% loss-of-load a feasible recovery solution, and so we did not perform computational experiments with  $k = 3$  for these test instances.  $N-3$

System	Critical $\epsilon_1$	Critical $\epsilon_2$
IEEE24, w/ switching	0	0.09
IEEE24, w/o switching	0.01	0.1
RTS-96, w/ switching	0	0.04
RTS-96, w/o switching	0	0.04

Table 3.2: IEEE24 and RTS-96 Critical  $\epsilon_k$  Values

security may make sense for larger systems where 3 components is a small fraction of the total number of components, but it does not make sense for these test instances.

Using these critical  $\epsilon_k$  values, we obtained the run time results for IEEE24 and RTS-96 test systems for  $k = 1$  and  $k = 2$  shown in Table 3.3.

Test Case	# Iterations	Run Time	% Run Time Spent in Last Iteration
IEEE24, $k = 1$ , $\epsilon_1 = 0$	60	4 min	10%
IEEE24, $k = 2$ , $\epsilon_1 = 0$ , $\epsilon_2 = 0.09$	99	18 min	7%
RTS-96, $k = 1$ , $\epsilon_1 = 0$	41	52 min	12%
RTS-96, $k = 2$ , $\epsilon_1 = 0$ , $\epsilon_2 = 0.04$	92	8.5 hrs	23%

Table 3.3: N- $k$  UC Constraint Generation Algorithm Run Times

We implemented the algorithm serially. However, one of the advantages of the proposed algorithm is that it could be easily parallelized. We will discuss the run-times of our serial implementation, and project how a parallel implementation could be performed.

For the the IEEE24 network with  $k = 1$ , the algorithm converged in 60 iterations, and the run time was a little under 4 minutes, of which about 10% of that time was spent solving the last iteration, taking about 23 seconds. The last iteration is the slowest because the Contingency Oracle must be called serially for each time period, to verify that no unsurvivable contingencies exist. The earlier 59 iterations average about 3 seconds each. In a parallelized implementation, if there were 24 processors, each of the 24 Contingency Oracle instances in the last iteration could be solved simultaneously, such that solving the Contingency Oracles in the last iteration would take about 1.5 seconds instead of 23 seconds. Additionally, the algorithm would not

require 60 iterations to converge. In the current implementation, the Contingency Oracle is only called if an unsurvivable contingency cannot be identified with the contingency list, and once an unsurvivable contingency is identified for this iteration, the Contingency Oracle is not called again. Thus, only a few constraints are added to the master problem in each iteration. However, in a parallelized implementation, the Contingency Oracle could be solved for multiple time periods in parallel in each iteration, potentially generating many more constraints for the master problem per iteration, and reducing the number of iterations necessary for convergence.

It is interesting to note that as  $k$  increased or as the network size increased, the number of iterations required did not dramatically increase. The main effect of increasing  $k$  and the network size is that the Contingency Oracle takes longer to solve. For example, the longest Contingency Oracle run time for RTS-96 with  $k = 1$  is 35 seconds. The longest Contingency Oracle run time for RTS-96 with  $k = 2$  is almost 10 minutes. The individual Contingency Oracle run times would not be reduced in a parallel implementation, but given that the bottleneck, the Contingency Oracle, could be parallelized, we would expect a substantial improvement in the overall runtime.

### 3.7.2 Critical $\epsilon$ Analysis

As previously mentioned, the minimum values of  $\epsilon_k$  for which there exists a feasible  $N-k$  secure unit commitment solution for IEEE24 and RTS-96 for  $k = 1$  and  $k = 2$  are shown in Table 3.2. These critical  $\epsilon_k$  values are a measure of how reliable the given power system is, and so it is interesting to analyze these  $\epsilon_k$  values.

For IEEE24, when  $k = 1$  and switching is used, it is possible to not shed any load for all contingencies, i.e., the critical  $\epsilon_1 = 0$  when switching is employed. When switching is not used in response to a contingency, it is necessary to shed 1% of the total load in the worst-case contingency in order for a feasible unit commitment solution to exist. The minimum loss-of-load that must be allowed for contingencies

of size 2 in order for there to exist a feasible unit commitment solution is 9%, when switching is allowed, but increases to 10% if switching is not employed.

For the RTS-96 network, the critical  $\epsilon_k$  values are the same with and without switching: for  $k = 1$ , the critical  $\epsilon_1 = 0$ , and for  $k = 2$ , the critical  $\epsilon_2 = 0.04$ . We believe the fact that switching reduces the critical  $\epsilon_k$  values in the IEEE24 system and not in the RTS-96 system demonstrates that switching is most valuable in dense systems. The RTS-96 system is constructed of three zones, where there is significant interconnection among the buses within a zone, but only minimal connection between different zones, whereas the IEEE24 system is equivalent to one of the zones in the RTS-96 system. The RTS-96 system is illustrated in Figure 3.4. The IEEE24 network is more dense overall and thus is more constrained, and switching is more likely to increase survivability in the worst-case contingency.

### 3.7.3 Scaled-Load Analysis

We analyzed how the value of switching changed as the system load levels varied. We defined the load levels used in Section 3.7.1 as the 100% baseline, and then scaled the load at each node and time period for the IEEE24 system from 85% to 102%. We observed that as the load increased, the difference between the cost of the optimal solution when transmission switching is allowed, and the cost of the optimal solution when transmission switching is not allowed increases. Essentially, in a more congested system, switching is more valuable. We observed this effect in the IEEE24 system both when  $k = 1$ , shown in Figure 3.5, and when  $k = 2$ , shown in Figure 3.6.

In these computational experiments, the  $\epsilon_k$  values were set equal to the critical  $\epsilon_k$  values with switching. Both when  $k = 1$  and when  $k = 2$ , the maximum load scaling at which there exists a feasible unit commitment solution without switching is 98%. For  $k = 1$ , there exists a feasible unit commitment solution with switching up to 102% scaled load, and for  $k = 2$  there exists a feasible unit commitment solution with



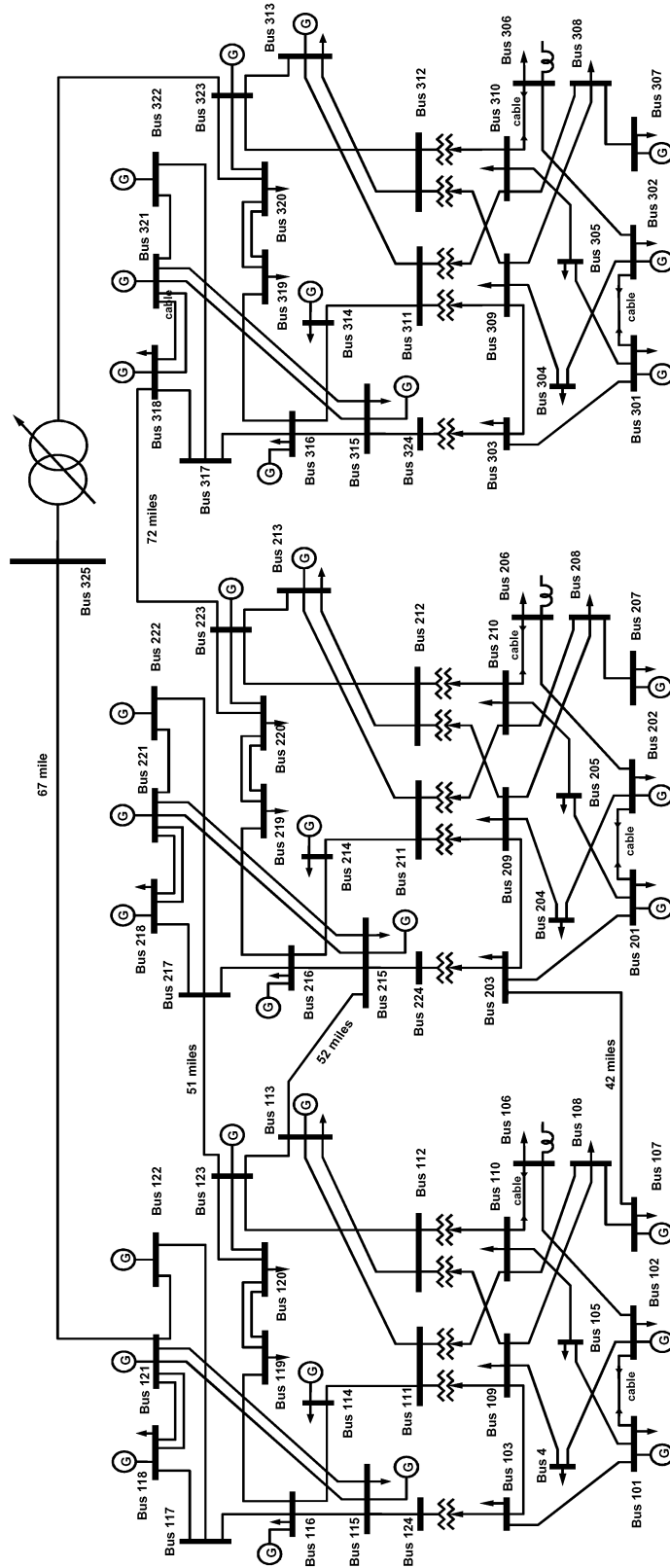


Figure 3.4: RTS-96 test system, from *Kamwa et al.* (2007)

switching up to 100% scaled load. At 98% scaled load, when there exists a feasible unit commitment solution both with and without switching, the optimal objective value of the unit commitment solution and 0-contingency dispatch is about 18% cheaper when switching is used, for  $k = 1$ , and about 13% cheaper with switching for  $k = 2$ .

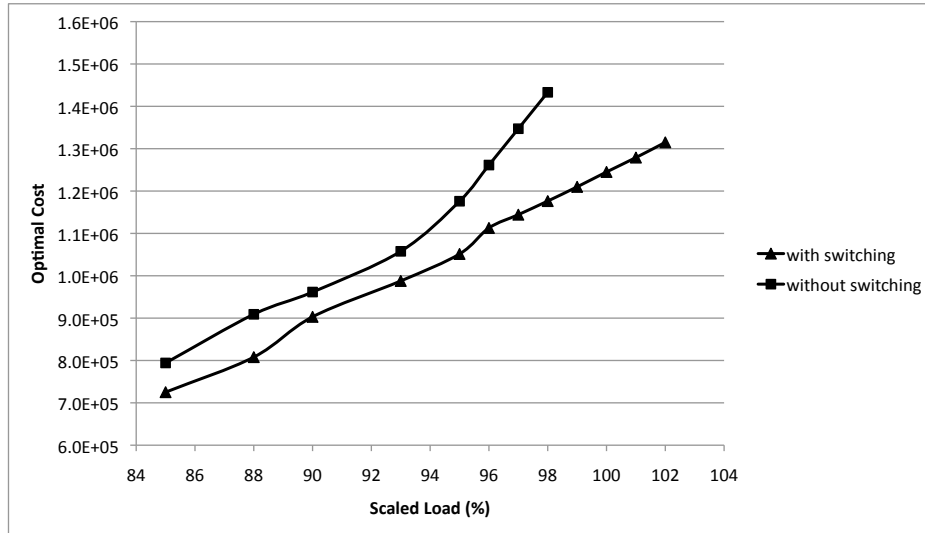


Figure 3.5: IEEE24, k=1, Optimal UC Cost at Different Load Levels

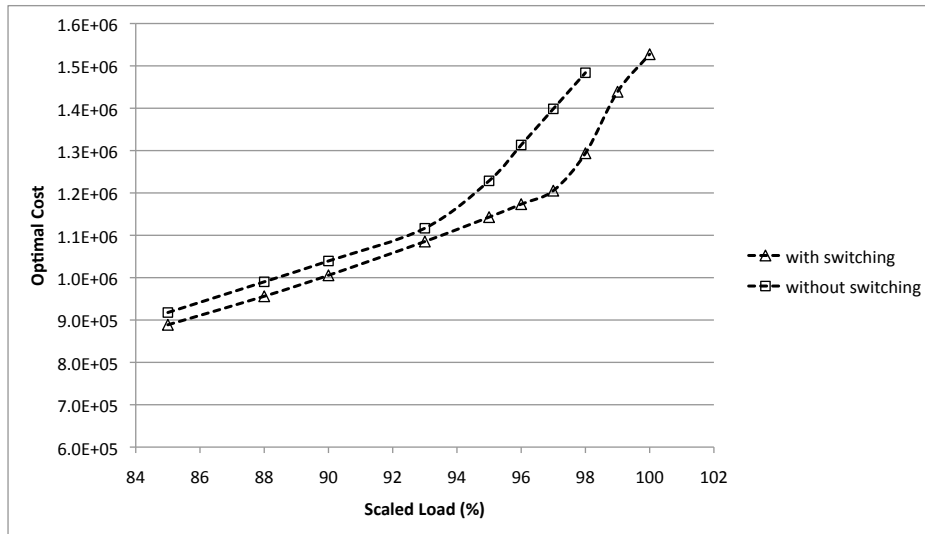


Figure 3.6: IEEE24, k=2, Optimal UC Cost at Different Load Levels

In Figure 3.7, the cost curves for IEEE24 from Figures 3.5 and 3.6 are overlaid on each other. In this plot, it can be seen that the cost curves for  $k = 1$  without

switching and  $k = 2$  with switching intersect at the scaled load level of 94%. This indicates that for demand levels above 94%, the decision to use transmission switching can allow the operator to achieve a higher level of reliability (N-2 security instead of N-1 security) at a lower cost. More generally a plot of this nature may help operators evaluate the cost of different levels of reliability, and determine the value of switching in their system.

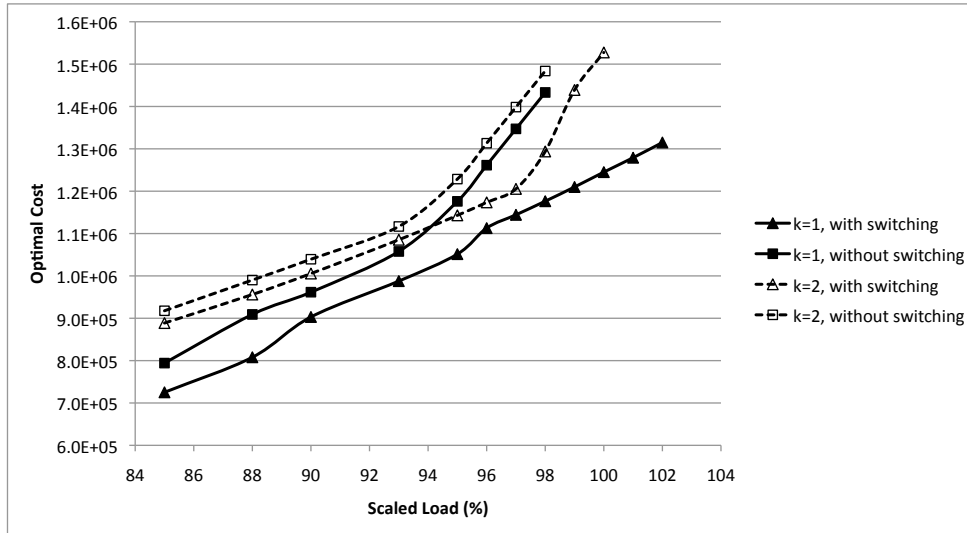


Figure 3.7: IEEE24,  $k=1$  &  $k=2$ , Optimal UC Cost at Different Load Levels

We note that with switching, the optimal cost of the unit commitment and 0-contingency dispatch for IEEE24 with 100% scaled load, where  $k = 1$  and  $\epsilon_1 = 0$ , is \$1.25 million. When  $k$  is increased to 2, with  $\epsilon_1 = 0$  and  $\epsilon_2 = 0.09$ , the optimal cost increases to \$1.53 million, an increase of 22.7%. For the RTS-96 network, the optimal cost of the unit commitment and 0-contingency dispatch when  $k = 1$  with 100% scaled load and  $\epsilon_1 = 0$  is \$2.98 million. When  $k$  is increased to 2, with  $\epsilon_1 = 0$  and  $\epsilon_2 = 0.09$ , the optimal cost increases to \$3.05 million, an increase of 2.5%. The difference between the optimal cost at  $k = 1$  and  $k = 2$  obviously is heavily dependent on the particular system costs. However, how much cost will increase when  $k$  is increased is difficult to predict without the use of tools like the algorithm presented here. The generators costs and characteristics used for the IEEE24 and

RTS-96 networks were quite similar, and yet the cost increase seen when increasing  $k$  from 1 to 2 was quite different. This result indicates the difficulty of estimating the cost of providing different reliability levels based on system characteristics alone, and highlights the need for tools such as the algorithm presented here.

#### 3.7.4 Line Removal Analysis

Another interesting observation concerns which lines are frequently switched out in the optimal switching solution for various contingencies. Given the optimal unit commitment solution returned by the cutting plane algorithm, the optimal switching problem was solved for each contingency-time period pair, determining the generator dispatch and switching decisions that minimize the total loss-of-load given the available components. There appear to be multiple optimal switching configurations for many of the contingency-time period pairs. However, among these different optimal solutions, there is a pattern; a small subset of lines are switched out in the optimal solution a significant percentage of the time while most other lines are hardly ever switched out. Most optimal switching solutions for the IEEE24 network with  $k = 1$  have 1-3 lines switched out in the optimal solution, and these lines generally belong to this subset of candidate switchable lines.

One might conclude that a line that is frequently switched out in the optimal solution should be permanently switched out. We tested this hypothesis by individually removing the six most frequently switched out lines, and computing the optimal unit commitment solution for each. In most cases, the objective value was nearly the same, within 1%. However, for two instances, the optimal unit commitment cost was 15% and 20% worse. In these cases, there was a line that was important for dispatching generators efficiently under normal operating conditions, but under contingency conditions it is useful to remove this line to minimize the loss-of-load. This result highlights the value of switching dynamically; the presence of a line can be valuable

under one set of conditions, while the absence of that same line is valuable under a different set of conditions.

### 3.8 Conclusion

We have presented models and algorithms for solving the  $N-k$  secure unit commitment problem when switching is allowed as a recovery action. We first presented a natural two-stage decomposition of the problem with mixed-integer variables in both stages. We then offered a novel reformulation where the second stage integer switching decisions are moved to the first stage. The resulting two-stage formulation has a linear second stage and, using a procedure for dynamically generating first stage switching variables, can be solved via a cutting plane algorithm inspired by Benders' decomposition.

We formulate a Contingency Oracle, an optimization problem which identifies an unsurvivable contingency for the current unit commitment and 0-contingency dispatch decisions. In each iteration of the overall algorithm, all contingencies do not have to be explicitly considered because the Contingency Oracle is used to identify unsurvivable contingencies, for which feasibility cuts can be generated for the first stage. We demonstrate that the effective number of variables in the first stage is modest, as the switching decisions are added to the first stage only when a feasibility cut is added for the corresponding contingency-time period pair. Thus, this approach may be used when the number of contingencies is extremely large.

We have also presented several implementation details which have a significant impact on the total runtime. In particular, maintaining a list of contingencies that have been unsurvivable in any previous iteration can be used to quickly identify unsurvivable contingencies in the current iteration.

We have shown computational results for the IEEE24 and RTS-96 systems, when  $k = 1$  and  $k = 2$ . Our results indicate that transmission switching is significantly

valuable in reducing the cost of an  $N-k$  unit commitment solution. Additionally, our results indicate that the ability to dynamically switch lines in and out as needed has significant value, as opposed to statically removing a line. Further, these results suggest that this algorithmic framework could be implemented in parallel, and used to solve problems of larger size, and larger values of  $k$ .

## CHAPTER IV

# Transmission Expansion with Smart Switching Under Demand Uncertainty and Line Failures

### 4.1 Introduction

Environmental concerns have motivated many governments to require that an increased amount of power be supplied by renewable sources. In the United States, most states have enacted renewable portfolio standards legislation which mandate the fraction of energy generation which must come from renewable sources (*Center for the New Energy Economy* (2013)). Renewable generation has many environmental benefits, but these non-dispatchable sources of power pose a challenge for planners due to the uncertainty in their power output. Additionally, new trends in the areas of demand response, plug-in hybrid electric vehicles and distributed generation are changing the profile of electricity demand. There is uncertainty in the future demand levels, especially when planning over a long-time horizon. Methods for dealing with this uncertainty must be used when planning where to build new transmission lines.

Furthermore, as a society that is increasingly reliant on digital technologies, an uninterrupted power supply is as important as ever. Designing a system that is resilient to failures is critical. However, building new transmission lines to provide redundancy is very expensive. It is important that transmission expansion decisions

be made intelligently so as to minimize the total investment costs while also ensuring that the system is robust to failure events. In response to a failure event, i.e., a contingency, it is important that a set of feasible actions be available to the operator that allow demand to be met, to prevent a blackout event.

Traditionally, the recovery actions available to the operator include the ability to change generator dispatch levels and influence transmission line power flows. To realize the goal of operating a system that is both reliable and efficient, as described in Chapter III, transmission switching has been proposed as an additional recovery action. Especially when considering where to build additional transmission lines, it is important to consider the paradox associated with the existence of any transmission line, that it can provide capacity or impose a bottleneck. Allowing transmission switching as a recovery action may have a significant impact on the optimal investment solution.

As some critics of transmission switching have noted, existing circuit breakers are intended to be used rarely, primarily to de-energize a line that must be repaired. The practice of using circuit breakers as a controllable element may require additional investment in equipment that is designed to be used repeatedly and is remotely controllable. The problem we consider here is how to make decisions about where to build new transmission lines *and* where to build new transmission switching equipment. We seek to solve a robust version of the problem where the total investment cost is minimized, and feasible operation is guaranteed for all contingencies and demands within the defined uncertainty set, given that transmission switching may be used as a recovery action.

The structure of this Chapter is as follows. In Section 4.2 we review the relevant literature. In Section 4.3 we formally define the deterministic transmission expansion problem (TEP) and develop the robust counterpart. We describe how a cutting plane algorithm could be used to solve the robust formulation in Section 4.4. In Section 4.5,



the development of an *oracle* is described which returns an unsurvivable contingency-demand pair given an investment solution. This oracle is utilized in the cutting plane algorithm described in Section 4.6 to solve the robust transmission expansion problem. Computational results are presented in Section 4.7. Finally, Section 4.8 contains concluding remarks.

## 4.2 Literature Review

Transmission expansion planning has been a rich area of research for several decades. In most early works, only dispatchable conventional generation is considered (i.e., uncertain renewable energy is not included) and demand forecasts are assumed to be accurate. *Latorre et al.* (2003) and *Romero et al.* (2002) review several types of deterministic models for the transmission expansion planning problem. More recently, there has been interest in incorporating uncertainty into the transmission expansion optimization models. A review of the transmission expansion area in general, including a presentation of a few models which incorporate uncertainty, is provided in *Sorokin et al.* (2012).

Several studies have used stochastic methods to deal with uncertainty in renewable generation and/or the demand for power. *Hemmati et al.* (2014) and *Yu et al.* (2009) consider a transmission expansion planning problem where there is uncertainty in both the demand and the power generated at wind farms. In both works, the authors assume that demand is normally distributed and that wind speeds are distributed according to a Weibull distribution, and they use a Monte Carlo simulation to approximate the probability distribution for the power output at the wind turbine generators. *Yu et al.* (2009) present a chance constrained formulation in which the model seeks a minimum cost expansion plan where the probability of meeting demand is at least equal to a specified threshold. They suggest a genetic algorithm which can return a heuristic solution to the chance constrained formulation. *Hemmati et al.*

(2014) propose a multi-objective model to solve for a transmission expansion solution that simultaneously minimizes investment cost, maximizes social welfare, and minimizes loss-of-load. They suggest a particle swarm algorithm to solve the proposed model. One downside of the approaches proposed in *Hemmati et al.* (2014) and *Yu et al.* (2009) are that the Monte Carlo simulations required to generate the wind power probability distribution are computationally intensive.

*López et al.* (2007) present a model that solves for both transmission and generation expansion decisions when there is uncertainty in demand. This model seeks to minimize the expected cost of both investment costs and operational costs. A set of possible demand scenarios and their probabilities are assumed to be given.

In contrast to stochastic optimization methods that require knowledge of probability distributions, which are generally difficult to ascertain, robust optimization has been used to solve for transmission expansion solutions that are feasible for a variety of demand and/or renewable generation conditions.

*Wu et al.* (2008) propose a robust model of transmission expansion where only uncertainty in demand is considered. The authors use a box uncertainty set for the demand (i.e., an ‘interval model’). They propose a branch-and-bound procedure to solve for the worst case demand for a given expansion plan, and use this routine within a Greedy Randomized Adaptive Search Procedure (GRASP) to solve for a heuristic transmission expansion solution.

*Yu et al.* (2011) apply the Taguchi’s Orthogonal Array Testing (TOAT) method to the transmission expansion planning problem where there is uncertainty in both renewable energy output and demand. The authors use a box uncertainty set for both demand and renewable generation. TOAT is used to identify a subset of scenarios which are representative of the set of all possible scenarios, as defined by the extreme points of the box uncertainty set. The authors demonstrate that by using only these representative scenarios within a genetic algorithm, they can identify an expansion

solution that is robust for most values. However, the transmission expansion solution obtained is not guaranteed to be feasible for all demand and renewable generation in the uncertainty set.

*Jabr* (2013) proposes a traditional robust model for transmission expansion where there is uncertainty in both renewable generation and loads using two different types of uncertainty sets: a box uncertainty set and a budget uncertainty set. The author proposes a Benders' decomposition procedure which is similar in spirit to what we propose here, although we additionally include uncertainty in line failures and transmission switching as a recovery action.

Outside of traditional stochastic programming or robust optimization frameworks, *Silva et al.* (2006) capture the tradeoff between cost and reliability in a transmission expansion model with demand uncertainty by setting objective coefficients that weight these opposing goals. The authors propose a genetic algorithm solution to find the optimal expansion plan with respect to these weights. Similar to these other robust optimization papers, the authors allow demand to vary within a range defined by upper and lower bounds. A limitation with the approach proposed in *Silva et al.* (2006) is that it may be difficult in practice to assign appropriate weighting coefficients that allow cost and reliability to be compared in the same units.

An alternative source of uncertainty that has been considered in the transmission expansion literature is the possibility of component failures. *Alguacil et al.* (2010) propose a model for the transmission expansion problem which is robust to intentional line failures. *Romero et al.* (2012) propose a tabu search algorithm to determine where to add line capacities, as well as generation capacities and spare transformers, to ensure that feasible operation is possible in response to a terrorist attack. *Choi et al.* (2005) employ network cut-set constraints to relate probability distributions on the availability of individual components to measures of system-wide reliability. The authors use this relationship to formulate constraints in a model which seeks a min-

imum cost transmission expansion which satisfies reliability criteria. In these works, demand and renewable generation are assumed to be deterministic, and transmission switching is not allowed. In a more general setting, *Shen (2013)* present two network design models which seek to minimize investment costs and expected recovery costs given a set of scenarios representing stochastic arc disruptions.

As discussed in Chapter III, the value of transmission switching has been demonstrated in several papers. *Fisher et al. (2008)*, *Hedman et al. (2010)*, *Hedman et al. (2009)*, *Khanabadi et al. (2013)* and *Khodaei and Shahidehpour (2010)* show how transmission switching might be used to reduce the cost of committing or dispatching generators. *Shirokikh et al. (2013)* present a method of choosing transmission switching actions that minimize generator dispatch costs while ensuring that conditional value at risk constraints are met which would limit the losses in response to contingency event. The authors assume that switching decisions are made prior to the realization of a contingency event and cannot be changed in response to a contingency.

In other works, transmission switching has been shown to be valuable as a corrective action to improve response to a contingency event. In addition to discussing the market implications of transmission switching, *Hedman et al. (2011)* explore how transmission switching might be used to improve reliability. *Li et al. (2012)* propose a method for determining the optimal switching actions for the sole purpose of ensuring reliable operation in response to a contingency event.

Several authors have investigated how the transmission expansion problem might be modified to incorporate transmission switching. *Khodaei et al. (2010)* present an algorithm for solving for the minimum cost transmission and generator expansion decisions where transmission switching is employed to reduce dispatch costs. The authors require that the investment solution be feasible for a small set of contingencies, where switching decisions cannot be changed in response to a contingency.

The authors use a Benders' decomposition procedure where transmission switching decisions are in the master problem, which is similar to the approach we propose. However, we employ a procedure for dynamically generating switching variables for the master problem on an as-needed basis which allows us to consider a larger set of contingencies, and to additionally consider uncertainty in demand.

*Villumsen and Philpott* (2012) propose a column generation approach to solving the transmission expansion and switching equipment investment problem when transmission switching is allowed and demands, generator capacities and generator costs are stochastic. The authors in *Villumsen et al.* (2013) propose a model of the transmission expansion problem when transmission switching is used in response to high wind penetration scenarios.

A problem related to the robust transmission expansion planning problem is that of identifying a worst case event from within the defined uncertainty set for a given expansion solution. This problem is especially interesting when it is assumed that the operator has the ability to react optimally to the event once it has occurred. Neglecting the demand uncertainty and considering only uncertainty in possible line failures, this type of optimization problem is an interdiction problem. *Arroyo and Fernández* (2009), *Delgado et al.* (2010) and *Zhao and Zeng* (2011) propose methods for solving this power grid interdiction problem where transmission switching is used as a recovery action. Our proposed approach for solving the oracle described in Section 4.5 extends these previously published methods by identifying a worst-case combination of contingency *and* demand events for a given investment solution. Additionally, the network expansion problem adds a level of complexity to the already hard power grid interdiction problem since interdiction analysis is a prerequisite to the network optimization problem.

In summary, other authors have considered the transmission expansion planning problem with transmission switching, *or* with uncertainty due to contingency events,

*or* uncertainty in demand or renewable generation, but our contribution is to explore novel solution methodologies when all of these complicating factors are considered simultaneously.

### **4.3 Problem Definition**

We seek an optimal investment solution which determines where new transmission lines should be built and on which lines transmission switching equipment should be installed. The objective is to minimize the total investment cost while ensuring that it is possible to recover from any single transmission line failure and any set of instantaneous demands and renewable generation levels in the defined box uncertainty set.

In this section we formally define the robust transmission expansion and switching equipment investment problem. In Section 4.3.1 we first explain the assumptions that we make in constructing our model. In Section 4.3.2 we define the deterministic problem, where the transmission line failures (i.e., contingency) and demand vector are fixed to a nominal value. In Section 4.3.3 we present the robust counterpart of the deterministic problem, where the contingency and demands/renewable generation vectors may take on any value within their respective uncertainty sets. We also derive the formulation of the robust counterpart as a linear mixed-integer program (MIP) with an exponential number of constraints. In the next section, we describe how this MIP formulation can be decomposed and solved via a constraint generation procedure.

#### **4.3.1 Assumptions**

To manage the tradeoff between accuracy and solvability, we make the following assumptions when formulating our model.

- **A set of candidate transmission lines is given.**

Our investment decisions are binary; for each line in the set of candidate transmission lines, we decide whether or not that line should be built.

- **Transmission switching equipment may be installed on any line.**

We assume there is a binary decision of whether or not transmission switching equipment should be installed for each transmission line, including both existing and candidate lines. We assume that switching equipment is not currently installed on any line, but this assumption could easily be modified by fixing the values of certain binary variables.

- **Transmission lines are the only components which may fail.**

Given the critical nature of transmission lines and the exposure of these lines to weather events, fallen trees, etc., we only consider transmission line failures in contingency events. However, the model presented here may be generalized to include failures of generators as well. Failures in both existing and new transmission lines are considered.

- **Renewable generation is treated as negative demand.**

Renewable generation is non-dispatchable, meaning that the generation output cannot be fully controlled by the operator, as availability depends on weather conditions. Typically the only control that the operator has over the renewable generation sources is that excess generation can be curtailed. In our model we assume that renewable generation is always used and never curtailed, but this assumption could easily be relaxed by adding a curtailment decision variable for each renewable generator in the operator's response to a contingency-demand event. The changes to the model necessary to include curtailment is described in detail in Appendix B.3. The methods presented here are still valid if curtailment is modeled. From the point of view of the operator, renewable generation

behaves the same way as the demand, in the sense that the operator must find a way to deal with whatever renewable output level is realized. Thus, in our model renewable generation is treated the same as negative demand. In the remainder of this Chapter, the term demand is used to refer to both true demand and renewable generation.

- **Demand values belong to a *box uncertainty set*.**

This uncertainty set on the demand parameters is defined by a lower bound and an upper bound for each node. Our goal is to ensure feasible operation in the event that any demand value within this range is realized.

- **We use the direct current power flow (DCPF) approximation.**

As in Chapter III, we employ steady-state operational assumptions. We use a linear approximation of the power flow equations which govern how power flows through the transmission network.

- **We seek to minimize investment cost, and neglect operational costs.**

Our goal is primarily to understand where new transmission lines and transmission switching equipment should be installed to ensure that feasible operation is possible under all events in our defined uncertainty set. Therefore, in our objective function we include the investment costs of building new transmission lines or switching equipment, but neglect the operational costs. Investment costs tend to be large relative to operational costs, so it is common in the transmission expansion planning literature to neglect operational costs (*Da Silva et al. (2001)*, *Binato et al. (2001)*, *Romero et al. (1996)*).

- **Transmission is the dominant limitation, not generator commitments.**

In our robust formulation, we are primarily interested in making transmission investments so as to ensure feasible operation for any realization of demand and contingency within the uncertainty set. We assume that during these extreme



events, generators are committed appropriately, and lower bounds on generator outputs are not constraining. This assumption is commonly made in long-term transmission expansion problems (*Romero et al. (2002)*). Transmission is assumed to be the dominant limitation, and so ramping, startup/shutdown, and other constraints on generator operation described in Section 3.3.2 are relaxed.

### 4.3.2 Deterministic Problem

We first formulate the deterministic problem in which there is no uncertainty in the parameter values; the failure state of all transmission lines is known and the set of nodal demands is known. In this formulation, the binary vector  $\bar{c}$  indicates which transmission lines are contained in the given contingency.  $\bar{c}_e = 1$  indicates that transmission line  $e$  has failed and is *not* available, and  $\bar{c}_e = 0$  indicates that the transmission line is available.

Additionally, the vector  $\bar{d}$  indicates the known demands. Demand  $\bar{d}_i$  at node  $i$  could either be positive, indicating true demand, or negative, indicating the level of renewable generation at the node. These vectors  $\bar{c}$  and  $\bar{d}$  will later be allowed to vary within a defined uncertainty set, but for now we assume that these vectors are known.

We note that realistically, the investment decisions must be made before the uncertain contingency and demands are known, and the operating decisions (which we represent by variables  $y$  and  $w$ ) are made in response to the realization of the contingency-demand event. However, in this initial deterministic model where the contingency and demands are known, this distinction of decisions made before and after the realized uncertainty is irrelevant.

The full explicit formulation of (4.1) is defined in Appendix B. The compact formulation is defined here using the following vector variable definitions:

- $x$  vector of binary transmission expansion and switching equipment decisions. Transmission expansion decisions are made for each line in a set of candidate transmission lines, and transmission switching equipment investment decisions are made for all transmission lines, both existing and candidate.
- $y$  vector of operating decisions including generator outputs, line flows, nodal phase angles, and net power injection at each node.
- $w$  vector of binary transmission switching decisions.

The compact deterministic problem is as follows:

$$\min_{x,y,w} b^T x \tag{4.1a}$$

$$\text{s.t. } Fx \leq f \tag{4.1b}$$

$$Ay + Bw + Cx \leq h + H\bar{c} \tag{4.1c}$$

$$Ry = E\bar{d} \tag{4.1d}$$

$$x, w \text{ binary} \tag{4.1e}$$

The objective (4.1a) minimizes the total investment cost of both building new transmission lines and installing new transmission switching equipment. Constraint set (4.1b) represents constraints on only the investment decisions. These constraints might include a limit on the number of transmission lines that can be built in total or on any particular right-of-way. Constraint set (4.1c) represents the operational constraints including limits on generator outputs and line capacities, DCPF equations, and power flow conservation. Constraint set (4.1d) requires that the net power flow out of any particular node is equal to the demand at that node.

Note that in our model we consider only transmission line failure as indicated by the vector  $\bar{c}$ . To extend this model to additionally consider generator failure, the dimension of  $\bar{c}$  could be adjusted to include binary indicator parameters for generators

as well, and constraint (4.1c) could be modified to account for the fact that destroyed generators cannot be dispatched.

### 4.3.3 Robust Counterpart

The robust counterpart of the proposed deterministic problem (4.1) treats the contingency vector  $c$  and the demand vector  $d$  as uncertain parameters. The vectors  $c$  and  $d$  are known to belong to uncertainty sets  $\mathcal{C}$  and  $\mathcal{D}$ , respectively. The goal is to solve for a transmission investment solution  $x$  such that there exists a nonempty set of feasible recovery actions for any  $c \in \mathcal{C}$  and  $d \in \mathcal{D}$ .

We assume that the uncertainty set of contingencies  $\mathcal{C}$  contains all contingencies of size 1 or 0. That is,  $\mathcal{C} = \{c \in \{0, 1\}^{|\mathcal{E}|} : e^T c \leq 1\}$ , where  $\mathcal{E}$  represents all existing and candidate transmission lines and  $e$  is an appropriately sized unit vector.

For the demand uncertainty set  $\mathcal{D}$ , we use a box uncertainty set. That is, the demand (and/or renewable generation) at each node is allowed to vary within predefined upper and lower bounds. Let  $\mathcal{N}$  be the set of all nodes, and  $L_i$  and  $U_i$  be the lower and upper bounds on the demand for node  $i$ , respectively. Thus,  $\mathcal{D} = \{d \in \mathbb{R}^{|\mathcal{N}|} : L_i \leq d_i \leq U_i \forall i \in \mathcal{N}\}$ .

We show that the robust counterpart of (4.1) can be formulated as a single-level MIP with an exponential number of variables and constraints.

Given a particular investment solution  $x$ , contingency  $c$  and demand vector  $d$ , the operator may choose a set of operational decisions  $y$  and a switching configuration  $w$  to best respond to the particular contingency-demand event. However, the existence of binary switching variables  $w$  makes the robust formulation much more complex, so for the moment let us assume that the switching configuration  $w$  is fixed *a priori*. The operator then must choose a set of operational decisions  $y$  that are feasible for

the following *fixed-switching recovery problem*:

$$S^P(x, w, c, d) = \min_y 0 \quad (4.2a)$$

$$\text{s.t. } Ay \leq h + Hc - Bw - Cx \quad (\phi) \quad (4.2b)$$

$$Ry = Ed \quad (\eta) \quad (4.2c)$$

Alternatively, the fixed-switching recovery problem could be formulated using a set of slack variables, and feasibility could be enforced by minimizing the sum of the slack variables in the objective function. This formulation was used in our implementation, but for the sake of clarity, we present the derivation here in terms of the feasibility problem (4.2) without slack variables.

The dual of (4.2) is as follows:

$$S^D(x, w, c, d) = \max_{\phi, \eta} \phi^T(h + Hc - Bw - Cx) + \eta^T Ed \quad (4.3a)$$

$$\text{s.t. } A^T \phi + R^T \eta = 0 \quad (y) \quad (4.3b)$$

$$\phi \leq 0 \quad (4.3c)$$

The solution  $\phi = 0, \eta = 0$  is feasible for (4.3) for any  $A$  and  $R$ , thus (4.3) is feasible for any inputs  $x, w, c$  and  $d$ .

By strong duality, if (4.3) has an optimal objective value equal to 0, then a feasible solution exists for the fixed-switching recovery problem (4.2). Otherwise, if the optimal objective value of (4.3) is unbounded, (4.2) does not have a feasible solution.

For a given investment solution  $x$  and switching configuration  $w$ , formulation (4.3) can be modified to find contingency and demand vectors that make the fixed-switching recovery problem infeasible by letting  $c$  and  $d$  become variables which may take any value within their respective uncertainty sets. If the optimal objective value If  $c$  and

$d$  become variables, (4.3) becomes the following optimization problem.

$$R(x, w) = \max_{d, c, \phi, \eta} \phi^T(h + Hc - Bw - Cx) + \eta^T Ed \quad (4.4a)$$

$$\text{s.t. } A^T \phi + R^T \eta = 0 \quad (4.4b)$$

$$\phi \leq 0 \quad (4.4c)$$

$$c \in \mathcal{C} \quad (4.4d)$$

$$d \in \mathcal{D} \quad (4.4e)$$

If the optimal objective value  $R(x, w)$  is unbounded, then a contingency-demand pair has been identified which causes (4.2) to be infeasible.

Since the uncertainty set  $\mathcal{D}$  can be expressed with a linear system of constraints which are disjoint with the other constraints (4.4b)-(4.4d), the optimal solution for  $d$  for (4.4) will be an extreme point of the polyhedron  $\mathcal{D}$  (*Jabr (2013)*). Furthermore, the optimal solution for  $(\phi, \eta)$  for (4.4) must be an extreme point or extreme ray of the feasible region defined by constraints (4.4b)-(4.4c), for the same reason.

Let  $\text{ext}(\mathcal{D})$  represent the set of extreme points of the polyhedron  $\mathcal{D}$ . Additionally, let  $\mathcal{V}$  represent the set of extreme rays of the feasible region of (4.3), and let  $\mathcal{X}$  represent the set of extreme points of the feasible region of (4.3). Given that the constraints (4.4b)-(4.4c), (4.4d) and (4.4e) are disjoint from each other in the sense that they do not contain any common variables, and that the sets  $\mathcal{X}$ ,  $\mathcal{V}$ ,  $\text{ext}(\mathcal{D})$  and  $\mathcal{C}$  all contain a finite number of elements, (4.4) can be rewritten as the following combinatorial optimization problem:

$$R(x, w) = \max_{(\phi, \eta) \in \mathcal{X} \cup \mathcal{V}, d \in \text{ext}(\mathcal{D}), c \in \mathcal{C}} \{\phi^T(h + Hc - Bw - Cx) + \eta^T Ed\} \quad (4.5)$$

If the optimal objective value  $R(x, w) = 0$ , then there exists a feasible solution to the fixed-switching recovery problem (4.2) for all  $d \in \text{ext}(\mathcal{D})$  and  $c \in \mathcal{C}$  for the particular

investment decisions  $x$  and switching recovery decisions  $w$ .

However, what we are really interested in is whether, for all  $d \in \text{ext}(\mathcal{D})$  and  $c \in \mathcal{C}$ , there exists a feasible solution to the recovery problem where switching is not fixed but allowed to be chosen in response to a particular  $(c, d)$  pair. Or, put another way, whether there exists at least one switching configuration for each contingency-demand pair for which there exists a feasible solution to the fixed-switching recovery problem.

Let  $i(c)$  be a function that maps the contingency  $c$  to its corresponding index in the set  $\mathcal{C}$ . That is, for any  $c \in \mathcal{C}$ ,  $i(c) \in \{1, 2, \dots, |\mathcal{C}|\}$ . Similarly, let  $j(d)$  be a function that maps a demand vector  $d$  which is an extreme point of the set  $\mathcal{D}$  to its corresponding index in  $\text{ext}(\mathcal{D})$ . That is, for any  $d \in \text{ext}(\mathcal{D})$ ,  $j(d) \in \{1, 2, \dots, |\text{ext}(\mathcal{D})|\}$

For a given  $x$ , the requirement that there there must exist a switching configuration  $w^{i(c),j(d)}$  for all  $d \in \text{ext}(\mathcal{D})$  and  $c \in \mathcal{C}$  such that there exists a feasible solution to the fixed-switching recovery problem can be expressed as follows:

$$\exists w^{i(c),j(d)} \forall c \in \mathcal{C}, d \in \text{ext}(\mathcal{D}) : R(x, w^{i(c),j(d)}) = 0$$

Thus, the formulation of the robust counterpart of the nominal transmission expansion problem (4.1) is as follows:

$$\min_{x,w} b^T x \tag{4.6a}$$

$$\text{s.t. } Fx \leq f \tag{4.6b}$$

$$\phi^T (h + Hc - Bw^{i(c),j(d)} - Cx) + \eta^T Ed \leq 0 \tag{4.6c}$$

$$\forall (\phi, \eta) \in \mathcal{X} \cup \mathcal{V}, d \in \text{ext}(\mathcal{D}), c \in \mathcal{C}$$

$$x \text{ binary} \tag{4.6d}$$

$$w^{i(c),j(d)} \text{ binary } \quad \forall d \in \text{ext}(\mathcal{D}), c \in \mathcal{C} \tag{4.6e}$$

Constraint (4.6c) enforces that for any feasible investment solution  $x$ , there must

exist a switching configuration  $w^{i(c),j(d)}$  such that the optimal objective function of the combinatorial optimization program (4.5) is less than or equal to 0. This requirement ensures the existence of a feasible recovery solution in response to every event in the uncertainty set  $\mathcal{C} \times \text{ext}(\mathcal{D})$ .

## 4.4 Decomposition

Formulation (4.6) is a linear MIP which can be used to find an investment solution  $x$  that minimizes investment cost and ensures that feasible operation is possible in response to any event in the defined uncertainty set. However, (4.6) contains an exponential number of constraints and variables, as the sets  $\mathcal{X} \cup \mathcal{V}$  and  $\text{ext}(\mathcal{D})$  both contains an exponential number of elements. Thus, to solve this MIP in practice, we employ a decomposition procedure. The basic idea is to relax (4.6c), and then generate violated constraints from the set (4.6c) iteratively, as needed, until a feasible investment solution  $x$  is identified.

### 4.4.1 Switching Variable Generation

Formulation (4.6) has the form of a master problem in a two-stage stochastic program in which the first stage variables are  $x$  and  $w$ , and the second stage problem is (4.2) with second stage variables  $y$ . The set of scenarios is the set of all contingency-demand pairs in the set  $\mathcal{C} \times \text{ext}(\mathcal{D})$ . Constraint set (4.6c) represents the set of Benders' feasibility cuts corresponding to all extreme points and extreme rays of the feasible region of the dual of the second stage problem for all scenarios.

We note that the switching variables are naturally second stage variables, because in practice the switching decisions can be chosen in response to particular contingency-demand event. However, we have effectively moved the switching variables into the first stage to alleviate the difficulty of solving a problem with second stage integer variables.

This reformulation is similar to the reformulation presented in Section 3.4.2 for the unit commitment problem. The  $x$  variables here represent transmission investment decisions rather than unit commitment decisions, but the second stage recovery decisions  $y$  and switching decisions  $w$  are very similar. The main difference is that in Chapter III, the operational decisions were made in response to contingency of size  $k$  or smaller in a particular time period, and in this Chapter, the operator responds to a particular demand realization and a single line failure. The different uncertainty sets defined in these two Chapters results in different scenario and subproblem definitions. In both Chapters we have made the reformulation decision to effectively treat the switching decisions  $w$  as first stage variables.

One challenge with first stage switching decisions in this formulation is that it results in a very large number of binary variables in the master problem. There exists a switching vector  $w^{i(c),j(d)}$  in the master problem for every contingency-demand pair  $(c, d)$ . As discussed, the set  $\mathcal{C} \times \text{ext}(\mathcal{D})$  contains an exponential number of elements, and thus there will exist a very large number of switching variables in the master problem even for relatively small systems.

Similar to what was proposed in Chapter III, to address this challenge we propose that switching variables be generated iteratively as needed as (4.6) is solved via a constraint generation procedure. The feasibility cuts in the set (4.6c) are initially relaxed and are then incrementally added as violations are identified. Any first stage variables that are not contained in any constraints can effectively be ignored. Switching variables are only contained in the constraints (4.6c), so initially all switching variables can be ignored. As violated constraints from (4.6c) are identified iteratively, each of which corresponds to a particular contingency-demand pair  $(c, d)$ , we generate the corresponding switching variables  $w^{i(c),j(d)}$ . Thus, the number of switching variables effectively in the master problem grows gradually as cutting planes are generated for the master problem.



In practice, we have found that switching variables are generated for only a small subset of all contingency-demand pairs. This observation will be further discussed in Section 4.7.

#### 4.4.2 Oracle Motivation

A problem with the structure of (4.6) would traditionally be solved with Benders' decomposition in which feasibility cuts in the set (4.6c) are generated by solving subproblems (4.2) for every contingency-demand pair in each iteration. Given that the set of all demand extreme points  $\text{ext}(\mathcal{D})$  contains an exponential number of elements, it would take an impractically long time to solve a subproblem for every  $(c, d) \in \mathcal{C} \times \text{ext}(\mathcal{D})$  in each iteration.

To address this challenge, we employ a similar approach to what was presented in Chapter III; we develop an *oracle*. The goal of the oracle is to identify a contingency-demand pair that does *not* have a feasible recovery solution for the current investment solution  $x$ , even with the best possible switching configuration. The development of the oracle will be explained further in Section 4.5, but for now let us assume that such an oracle exists. This oracle eliminates the need to explicitly screen all contingencies and demand pairs in the uncertainty set to identify an unsurvivable contingency-demand pair.

#### 4.4.3 Cutting Plane Algorithm

Assuming that there exists an oracle for identifying unsurvivable contingency-demand pairs, the proposed algorithm for finding the minimum cost robust investment solution proceeds as follows. An illustration of the algorithm is presented in Figure 4.1.

We note the similarity of the algorithm outlined in Figure 4.1, and that outlined in Figure 3.2 to solve the unit commitment problem. The details of the two problems

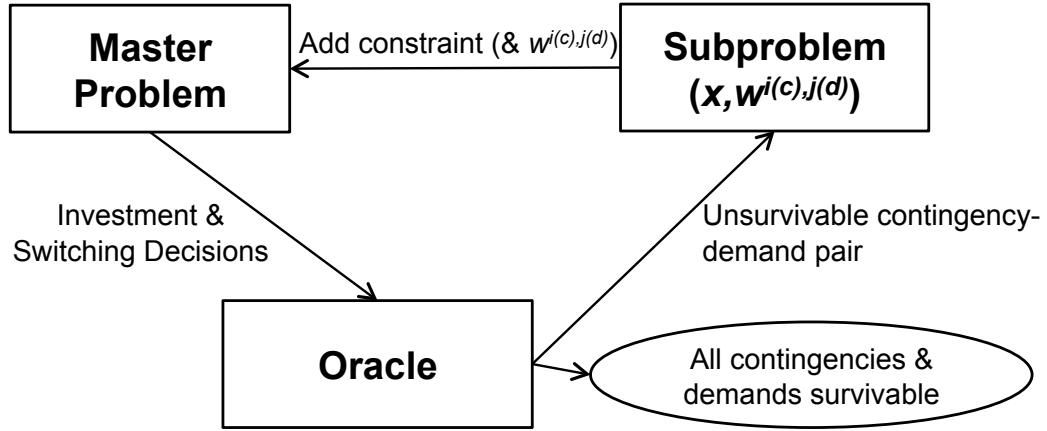


Figure 4.1: Algorithm Overview for Transmission Expansion with Switching

are different, but the overall algorithmic structure is similar.

In each iteration, the master problem is solved. Initially, the master problem is (4.6) where all constraints in set (4.6c) are relaxed, and all switching variables  $w$  are ignored. The oracle is called to identify an unsurvivable contingency-demand pair  $(\bar{c}, \bar{d})$ . A subproblem for this  $(\bar{c}, \bar{d})$  pair is solved, and the dual subproblem solution  $(\bar{\phi}, \bar{\eta})$  is used to generate a feasibility cut for the master problem. The form of the feasibility cut is as follows:

$$\bar{\phi}^T (h + H\bar{c} - Bw^{i(c),j(d)} - Cx) + \bar{\eta}^T E\bar{d} \leq 0$$

As feasibility cuts in the set (4.6c) are generated for the master problem, corresponding sets of switching variables are added as well. The procedure of generating feasibility cuts repeats until the oracle identifies that all contingencies and demands in the uncertainty set are survivable for the current investment solution  $x$ .

## 4.5 Oracle Development

The role of the oracle is to identify a contingency-demand pair for which feasible operation is not possible given the current investment solution  $x$ , even when transmis-

sion switching is available as a recovery action. If no such unsurvivable contingency-demand pair exists, the oracle should return a certification to indicate that the current investment decision  $x$  is optimal. This type of problem can be thought of from the perspective of a fictional adversary who seeks to identify a transmission line to disrupt and a particular demand scenario whose combination would maximize damage.

The optimal solution to (4.4) identifies a contingency-demand pair that would be unsurvivable for a given set of investment decisions  $x$  *if the recovery switching configuration were fixed*. If  $(c, d)$  is unsurvivable with this particular fixed switching configuration, this is *not* a certification that  $(c, d)$  would also be unsurvivable under a different, better switching configuration. However, the optimal solution  $(c, d)$  to (4.4) is a good candidate for unsurvivability. We use this optimization problem (4.4) within an iterative constraint generation routine which alternately identifies  $(c, d)$  pairs which are candidates for unsurvivability, and verifies whether a given  $(c, d)$  pair is actually unsurvivable when any switching configuration is allowed as a recovery action.

Before this constraint generation routine is presented, we first present a reformulation of (4.4) which eliminates bilinear terms in the objective and is a linear MIP.

#### 4.5.1 Bilinear reformulation

To transform formulation (4.4) into a linear MIP, the two bilinear terms in the objective function,  $\phi^T Hc$  and  $\eta^T Ed$ , must be linearized.

The linearization of the first term is fairly simple because the contingency variables are binary, and so the bilinear term is a product of a binary variable and a continuous variable. There exist standard methods for linearizing this type of bilinear term. The same type of linearization procedure was mentioned and used in Section 3.5.2. Here we explicitly explain the linearization.

A new set of auxiliary variables can be defined,  $\gamma_e$ , to represent the bilinear quan-

tity  $(\phi^T H)_e c_e$ . In the objective, the bilinear term  $\phi^T H c$  is replaced with the linear term  $e^T \gamma$ , where  $e$  is an appropriately sized unit vector. To enforce the relationship between  $\gamma_e$  and the original bilinear terms, the following set of constraints is added for each transmission line  $e \in \mathcal{E}$ .

$$\gamma_e \leq M c_e \quad (4.7a)$$

$$\gamma_e \geq -M c_e \quad (4.7b)$$

$$\gamma_e \leq (\phi^T H)_e + M(1 - c_e) \quad (4.7c)$$

$$\gamma_e \geq (\phi^T H)_e - M(1 - c_e) \quad (4.7d)$$

Let  $M$  be a parameter defined such that  $M \geq \max_{(\phi, \eta) \in \mathcal{X} \cup \mathcal{V}, e \in \mathcal{E}} \{-(\phi^T H)_e, (\phi^T H)_e\}$ . Constraints (4.7a)-(4.7b) enforce that  $\gamma_e = 0$  when  $c_e = 0$ , and constraints (4.7c)-(4.7d) enforce that  $\gamma_e = (\phi^T H)_e$  when  $c_e = 1$ . The set of equations (4.7a)-(4.7d) effectively enforce the original bilinear relationship that  $\gamma_e = (\phi^T H)_e c_e$  for each  $e \in \mathcal{E}$ . In compact form, let the constraints (4.7a)-(4.7d) for all  $e \in \mathcal{E}$  be represented by the constraint  $G\gamma \leq g + Jc + Q\phi$ .

The linearization of the second term  $\eta^T E d$  is more complex, as the demand  $d_i$  is a continuous variable which may take on any value within the specified upper and lower bounds. We propose an alternative representation of the demand  $d_i$  in terms of binary variables.

As discussed in Section 4.3.3,  $\mathcal{D} = \{d \in \mathbb{R}^{|\mathcal{N}|} : L_i \leq d_i \leq U_i \forall i \in \mathcal{N}\}$ , and the optimal solution for  $d$  to the optimization problem (4.4) will always be one of the extreme points of the polyhedral uncertainty set  $\mathcal{D}$ . Given the definition of the box uncertainty set, at an extreme point of  $\mathcal{D}$ , the demand  $d_i$  is either equal to its upper bound  $U_i$  or equal to its lower bound  $L_i$ . Thus, at any extreme point, the demand  $d_i$  can be represented in terms of a binary variable  $z_i$ . Let  $z_i$  equal 1 if  $d_i = U_i$ , or equal 0 if  $d_i = L_i$ . Thus, the demand  $d_i$  at an extreme point of  $\mathcal{D}$  can be expressed

as follows:

$$d_i = L_i + (U_i - L_i)z_i$$

$z$  binary

In the objective of the bilevel program, the demand variable  $d_i$  is multiplied by  $(E^T \eta)_i$ . For each element  $i$ , the term in the objective is rewritten as:

$$(E^T \eta)_i d_i = (E^T \eta)_i L_i + (E^T \eta)_i (U_i - L_i)z_i$$

Note that this expression contains bilinear terms, as  $(E^T \eta)_i$  is a continuous variable and  $z_i$  is a binary variable. However, as this bilinear term is the product of a binary variable and a continuous variable, it can be linearized in the same way as was  $(\phi^T H)_e c_e$ . Let  $\lambda_i$  be the auxiliary variable which represents the bilinear term  $(E^T \eta)_i z_i$ . Let constraints analogous to (4.7a)-(4.7d) enforce the relationship that  $\lambda_i = (E^T \eta)_i z_i$ , and let the compact representation of these constraints be  $S\lambda \leq s + Tz + V\eta$ .

Let the term  $\eta^T E d$  in the objective (4.4a) be replaced by  $\eta^T E L + (U - L)^T \lambda$ , which represents the linearized expression.

Thus, the nonlinear optimization problem (4.4) can be reformulated as a linear MIP

as follows:

$$R(x, w) = \max_{\phi, \eta, c, \gamma, z, \lambda} \phi^T h + e^T \gamma + \phi^T (-Bw) + \phi^T (-Cx) + \eta^T EL + (U - L)^T \lambda \quad (4.8a)$$

$$\text{s.t. } H^T \phi + R^T \eta = 0^T \quad (4.8b)$$

$$G\gamma \leq g + Jc + \phi \quad (4.8c)$$

$$S\lambda \leq s + Tz + V\eta \quad (4.8d)$$

$$\phi \leq 0 \quad (4.8e)$$

$$e^T c \leq 1 \quad (4.8f)$$

$$c, z \text{ binary} \quad (4.8g)$$

This linearized program (4.8) can be solved directly to identify a contingency-demand pair for which feasible recourse is not possible for the given investment  $x$  and a fixed switching configuration  $w$ .

There exist several other special types of uncertainty sets for which the extreme points can be expressed in terms of binary variables. Other authors have used this type of uncertainty set representation in robust power system optimization problems for a polyhedral uncertainty set (*Jiang et al. (2010)*), a budget uncertainty set (*Jabr (2013)*) and multiple budget uncertainty sets (*Zhao and Zeng (2012)*). For these types of uncertainty sets, the basic procedure presented in this section for developing the oracle is applicable.

#### 4.5.2 Iterative Oracle Routine

The reformulated program (4.8) can identify a contingency-demand pair which is unsurvivable for a given investment decision  $x$  when recovery switching actions are fixed to a given  $w$ . However, what we are more interested in is a contingency-demand pair which is unsurvivable when the set of switching actions  $w$  is not fixed *a priori*,

but is allowed to be chosen in response to particular  $(c, d)$  event. An iterative routine is proposed to identify a contingency-demand pair that unsurvivable even with the best case switching configuration.

The routine for solving the oracle is similar in structure to that presented for solving the Contingency Oracle in Section 3.5.3. The upper level problem identifies a contingency-demand pair  $(c, d)$  which is unsurvivable for the current investment solution for a given fixed switching configuration. That  $(c, d)$  pair is passed to the lower level to definitively determine whether a particular  $(c, d)$  pair is unsurvivable when *any* switching configuration may be chosen in response to that event. The routine for solving the oracle is illustrated in Figure 4.2.

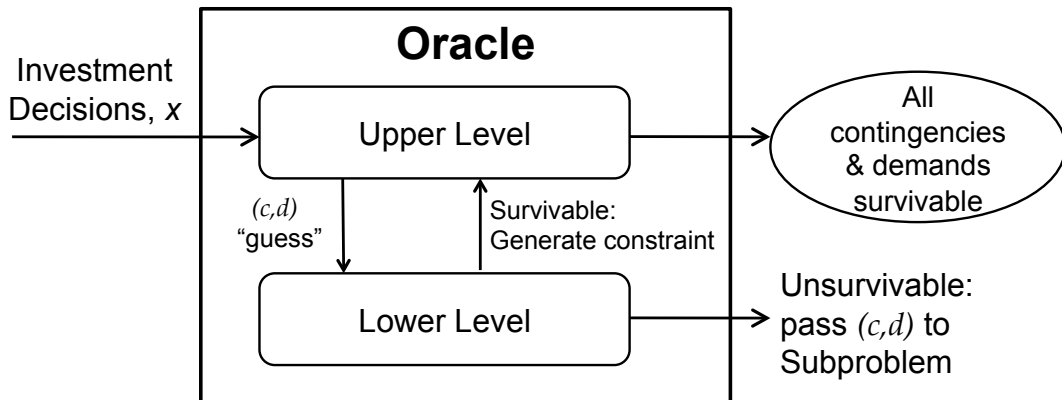


Figure 4.2: Oracle Routine

More specifically, the routine for solving the oracle proceeds as follows. Let  $\bar{x}$  be the current investment solution.

First, the initial upper level problem (4.8) is formulated where the switching vector is fixed to  $w = 0$  (i.e., no switching). If the optimal objective value of the upper level problem is unbounded, a contingency-demand pair  $(\bar{c}, \bar{d})$  has been identified which is unsurvivable when switching is fixed to  $w = 0$ . This  $(\bar{c}, \bar{d})$  pair is passed to the lower

level problem to check whether there exists a different switching configuration that would enable survivability.

The lower level problem is formulated as follows.

$$S(x, c, d) = \min_{y, w} 0 \quad (4.9a)$$

$$\text{s.t. } Ay + Bw \leq h + Hc - Cx \quad (4.9b)$$

$$Ry = Ed \quad (4.9c)$$

$$w \text{ binary} \quad (4.9d)$$

If the lower level problem is infeasible, then there does not exist any switching configuration that would allow there to exist a feasible recovery solution for this  $(\bar{c}, \bar{d})$ , meaning that  $(\bar{c}, \bar{d})$  is unsurvivable. The oracle routine can be exited, and this  $(\bar{c}, \bar{d})$  pair can then be passed to the subproblem (4.2). Otherwise, the lower level problem is feasible, indicating that there exists a switching configuration that enables survivability. A constraint is generated for the upper level problem to make the current contingency-demand solution infeasible.

While the overall routine for this oracle, illustrated in Figure 4.2, is similar to the routine for solving the Contingency oracle in Chapter III, illustrated in Figure 3.3, the primary difference between the two procedures is the form of the constraint generated for the upper level problem (analogously, CCI for the unit commitment problem). In Chapter III, the constraint added to CCI specifies what elements may be disrupted in the contingency specified by the next CCI solution. However, here the constraint generated for the upper level problem must be in terms of both the contingency variables  $c$ , as well as the demand variables  $d$ .

Let the optimal switching configuration in the feasible lower level solution be  $\hat{w}$ . The constraint added to the upper level problem requires that the dual objective value  $S^D(\bar{x}, \hat{w}, \bar{c}, \bar{d})$  be greater than 0. A dual objective that is greater than 0 indicates that



the fixed-switching recovery problem is infeasible. An unsurvivable  $(c, d)$  pair must be unsurvivable for all possible switching configurations, so requiring that the dual objective is greater than 0 for any particular switching configuration is valid.

Let  $\epsilon$  be a very small value which is the threshold at which a value is considered to be “greater than 0”. The constraint generated for the upper level problem is as follows:

$$\phi^T h + e^T \gamma + \phi^T(-B\hat{w}) + \phi^T(Cx) + \eta^T EL + (U - L)^T \lambda \geq \epsilon$$

The upper level problem is solved again, and in the next iteration a new contingency-demand pair will be identified. The procedure repeats until the lower level becomes infeasible, indicating that an unsurvivable  $(c, d)$  pair has been found, or the upper level becomes infeasible or has an optimal objective value equal to 0, indicating that there does not exist an unsurvivable  $(c, d)$  pair in the uncertainty set.

## 4.6 Implementation

### 4.6.1 Implementation Details

Here we will discuss two details of the implementation of cutting plane algorithm described in Section 4.4.3 which ensure convergence and improve performance.

The first issue deals with the inputs to the subproblem (4.2), which are  $x, w, c, d$ . The contingency  $c$  and demand  $d$  are set according to the unsurvivable contingency-demand pair identified by the oracle. The investment  $x$  is set according to the master problem solution, and the switching vector  $w$  may or may not be set according to the master problem solution, depending upon whether the switching vector  $w^{i(c),j(d)}$  exists in the master problem. If  $w^{i(c),j(d)}$  exists in the master problem, then to guarantee convergence,  $w$  must be set equal to the master problem solution for  $w^{i(c),j(d)}$ . However, if  $w^{i(c),j(d)}$  has not been added to the master problem, then any arbitrary binary vector may be set which is feasible for the current investment solution  $x$ . That

is, the switching configuration chosen may only switch lines out which have switching equipment installed on them according to the investment solution  $x$ . For simplicity, we choose to set  $w = 0$ .

The second issue concerns the manner in which unsurvivable contingency-demand pairs are identified. Drawing on the success of an implementation detail that was used in Chapter III, as described in Section 3.6.1, the following procedure is recommended. To ensure feasibility of a given  $(c, d)$  pair, multiple feasibility cuts are often necessary. Thus, rather than calling the oracle to identify an unsurvivable  $(c, d)$  pair in every iteration, we suggest that a *Critical  $(c, d)$  List* instead be checked for unsurvivable  $(c, d)$  pairs. This critical list is a list of all  $(c, d)$  pairs that have previously been identified as unsurvivable by the oracle, which are good candidates for unsurvivability.

The procedure for identifying unsurvivable contingency-demand pairs from the critical list is as follows. Given the current investment solution  $x$ , for each contingency-demand pair in the list, the with-switching recovery problem (4.9) is solved. If (4.9) is infeasible for any  $(c, d)$  pair, then an unsurvivable  $(c, d)$  pair has been identified. If (4.9) is feasible for all contingency-demand pairs in the critical list, the oracle is called to identify an unsurvivable contingency-demand pair, if one exists.

In our computational tests, we have found that there tends to exist a small set of “dominant” contingency-demand pairs, in the sense that once sufficient investments are made to ensure the survivability of these pairs, all other pairs are also survivable. Thus, checking this Critical List seems to be an efficient way to identify unsurvivable contingency-demand pairs.

#### 4.6.2 Complete Algorithm

Using these implementation details, the complete cutting plane algorithm is presented here.

1. Solve the initial master problem, which is (4.6) where all constraints in the set

(4.6c) are relaxed, and there are no switching variables  $w$ . Get the optimal solution  $\hat{x}$ .

## 2. Oracle Routine

- (a) Set  $w = 0$  and solve the initial upper level problem (4.8) for the optimal solution  $(\hat{\phi}, \hat{\eta}, \hat{c}, \hat{\gamma}, \hat{z}, \hat{\lambda})$ . If  $R(x, 0) \leq 0$ , then exit with the optimal investment solution  $x$ , which is robust to all contingency-demand pairs in the uncertainty set. Otherwise, use the optimal solution to generate the candidate contingency-demand pair  $(\hat{c}, \hat{d})$ , where  $\hat{d} = L + (U - L)^T \hat{z}$ .
- (b) Pass  $(\hat{x}, \hat{c}, \hat{d})$  to the lower level problem (4.9) and solve. If (4.9) is feasible, then  $(\hat{c}, \hat{d})$  is survivable. Let the optimal switching configuration be  $\hat{w}$ . Otherwise, go to step 3.
- (c) Add the following constraint to the upper level problem.

$$\phi^T h + e^T \gamma + \phi^T (-B\hat{w}) + \phi^T (C\hat{x}) + \eta^T EL + (U - L)^T \lambda \geq \epsilon$$

- (d) Solve the upper level problem. If the upper level problem is infeasible or if the optimal objective value is 0, then exit with the optimal investment solution  $\hat{x}$ , which is robust to all contingency-demand pairs in the uncertainty set. Otherwise, use the optimal solution  $(\hat{\phi}, \hat{\eta}, \hat{c}, \hat{\gamma}, \hat{z}, \hat{\lambda})$  to generate the candidate  $(\hat{c}, \hat{d})$  pair. Continue to step 2b.
3. Add  $(\hat{c}, \hat{d})$  to the critical list if it does not already exist.
  4. If the switching vector  $w^{i(\hat{c}), j(\hat{d})}$  exists in the master problem, pass  $(\hat{x}, \hat{w}^{i(\hat{c}), j(\hat{d})}, \hat{c}, \hat{d})$  to the fixed-switching recovery dual subproblem (4.3) and solve for the optimal solution  $(\tilde{\phi}, \tilde{\eta})$ .

5. Otherwise, if the switching vector  $w^{i(\hat{c}),j(\hat{d})}$  does not yet exist in the master problem, pass  $(x, 0, \hat{c}, \hat{d})$  to the fixed-switching recovery dual subproblem (4.3) and solve for the optimal solution  $(\tilde{\phi}, \tilde{\eta})$ .
6. Generate the following feasibility cut for the master problem:

$$\tilde{\phi}^T(h + H\hat{c} - Bw^{i(\hat{c}),j(\hat{d})} - Cx) + \tilde{\eta}^T E\hat{d} \leq 0$$

If  $w^{i(\hat{c}),j(\hat{d})}$  did not previously exist in the master problem, it is now added to the master problem.

7. Solve the master problem for the optimal solution  $(\hat{x}, \hat{w})$ .
  8. For each  $(c, d)$  pair in the critical list, solve the with-switching recovery problem (4.9). If (4.9) is infeasible for any  $(\hat{c}, \hat{d})$ , stop looping through the critical list. Pass  $(\hat{x}, \hat{w}^{i(\hat{c}),j(\hat{d})}, \hat{c}, \hat{d})$  to the dual subproblem (4.3). Let the optimal dual solution be  $(\tilde{\phi}, \tilde{\eta})$ . Go to step 6.
- Otherwise, if (4.9) is feasible for all  $(c, d)$  pairs in the critical list, go to step 2.

## 4.7 Computational Results

The proposed algorithm was implemented in C++ using CPLEX v12.4 Concert Technology. Our computational results were performed on a computer with 4GB RAM and a 2.3 GHz processor.

Results for three different test cases are presented here. The original sources for these test cases (*Freris and Sasson (1968)*, *Garver (1970)*, *Grigg et al. (1999)*) include descriptions of the topology and system characteristics, but do not however include sets of candidate lines. We use sets of candidate lines from relevant transmission expansion literature. For the IEEE24 test system and Garver test system, candidate lines from *Alguacil et al. (2010)* are used. The set of candidate lines and the capacities

for the existing lines for the IEEE14 test case are from *Xu et al. (2006)*. Costs of installing transmission switching equipment was not available in the references, so we chose switching costs to approximately match the relative cost of switching equipment and new transmission lines defined in *Villumssen et al. (2013)*. The essential characteristics of the test cases are summarized in Table 4.1

Test Case	# Nodes	# Loads	# Generators	# Existing Lines	# Candidate Lines
Garver6	6	5	3	6	39
IEEE14	14	11	5	20	10
IEEE24	24	17	32	35	10

Table 4.1: Garver6, IEEE14 and IEEE24 System Characteristics

The runtime ranges for each of the test cases are shown in Table 4.2 for a variety of different demand uncertainty sets, as defined later in this section. Additionally, the column titled “#(c,d) pairs considered” in Table 4.2 refers to the number of contingency-demand pairs explicitly considered when solving the algorithm over the same range of demand uncertainty sets. This refers to the number of unique contingency-demand pairs identified by the oracle while solving the algorithm, whose corresponding switching variables are effectively added to the master problem. The column “Total # of (c,d) pairs” indicates the number of elements in the set  $\mathcal{C} \times \text{ext}(\mathcal{D})$ , which is determined by the number of existing and candidate transmission lines and the number of demand nodes. Our intention is to demonstrate the order of magnitude of the number of contingency-demand pairs that are explicitly considered while solving the algorithm relative to the total number in the uncertainty set. As shown in Table 4.2, for these test cases only a handful of contingency-demand pairs were explicitly considered, despite the thousands or millions of contingency-demand pairs in the uncertainty set.

We note that for these three test instance, the larger networks do not necessarily have longer run times. The number of candidate lines, and the total number of new

Test Case	Run Time	# (c,d) pairs considered	Total # (c,d) pairs
Garver6	17-94 sec.	7-11	1.4E3
IEEE14	1-9 sec.	1-2	6.1E4
IEEE24	13-99 sec.	2-5	5.9E6

Table 4.2: TEP Algorithm Run Times and Performance Metrics

investments that must be made to ensure reliability, seem to be a more important indicator for how long the algorithm will take to converge. If more investments are needed, the algorithm will require more iterations. More conservative demand uncertainty sets require more investments, so longer run times are correlated with wider demand uncertainty sets. For these test instances, the algorithm’s run time was dominated by the time required to solve the master problem.

For test instances larger than these, we found that the upper level problem became the bottleneck. The upper level problem has binary variables for every demand node and for every transmission line, and the LP relaxation of the upper level problem is not particularly tight, so a lot of branching was necessary to identify the optimal MIP solution. For future work, we suggest utilizing inequalities that tighten the LP relaxation of the upper level problem. For our purposes, we focus on solving these three test instances to develop intuition as to how switching can improve the investment cost, and how the optimal investment cost varies with the choice of demand uncertainty set.

For the Garver 6 bus test instance, we first explore how the use of transmission switching as a recovery action changes the investment solution. Figure 4.3 represents the optimal investment solution when transmission switching is *not* an allowable recovery action, and Figure 4.4 represents the optimal investment solution when transmission switching is allowed. The dashed lines represent new transmission lines. In Figure 4.4, the circuit breaker image on the transmission line between nodes 2 and 3 represents new switching equipment. The solutions illustrated in Figures 4.3 and 4.4 share many of the same investments. However, the optimal cost with switching is

\$184 compared to \$200, when transmission switching is not allowed. This is due to the fact that when transmission switching is allowed, 1 fewer transmission lines are built, and transmission switching equipment is built on one line. The transmission switching equipment is much cheaper than building a new transmission line.

Note that by switching line (2-3) out, the cycle between nodes 1, 2, 3 and 5 is broken. Transmission switching is most likely to be useful in transmission networks which contain cycles. In a network which more resembles a tree or a line, there is a lot of flexibility to find phase angle values that would support whatever power flow patterns are desired. However, in a dense network with cycles, the DCPF constraints are likely to be limiting, as phase angle values are more constrained. Thus, these are the systems where transmission switching is most likely to be useful.

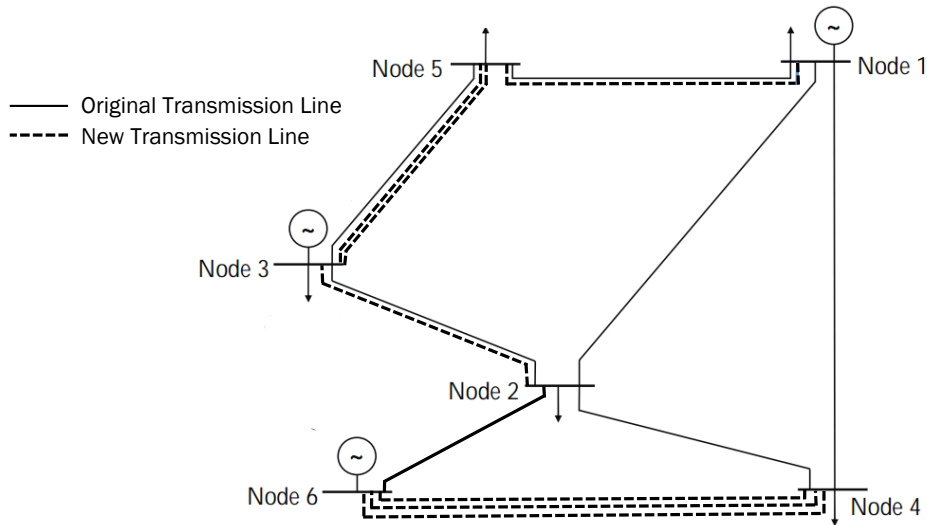


Figure 4.3: Optimal Investment Solution for Garver6 without Switching.

In an effort to explore how the conservatism of the defined demand uncertainty set impacts the optimal cost, for the IEEE14 and IEEE24 bus test cases, we have fixed the lower bound on the demand uncertainty set and scaled the upper bound. The lower bound is set equal to 70% of the nominal demand. The upper bound is set equal to the nominal demand times a scaling factor. High scaling levels for the demand

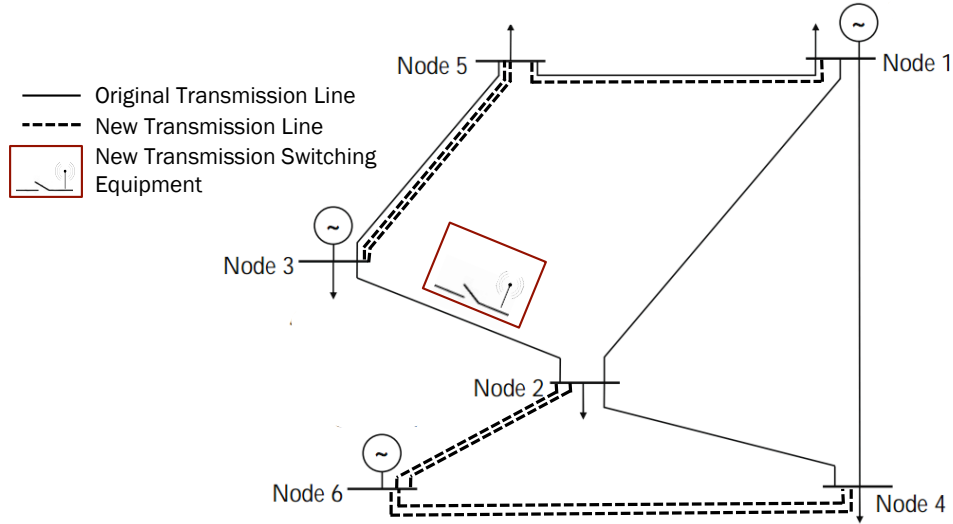


Figure 4.4: Optimal Investment Solution for Garver6 with Switching.

upper bound represent an increased level of conservatism in the defined uncertainty set. Figure 4.5 represents the optimal investment cost for different scaling factors for the demand upper bound for the IEEE24 test case. Similarly, Figure 4.5 represents the same quantities for the IEEE14 test case.

We note that for both of these test cases, the cost of the optimal investment solution is lower when switching is allowed as a recovery action than when transmission switching is not employed as a recovery action. Essentially, the flexibility introduced by transmission switching allows the same level of reliability to be achieved by installing new switching equipment rather than new transmission lines, as the cost of the switching equipment is small relative to the cost of new transmission lines. In Figure 4.6, when the demand upper bound is fixed to 100%, the optimal cost with transmission switching is shown, but without transmission switching, a feasible solution is not possible. Thus, in some instances allowing transmission switching may allow a level of reliability to be attained that would not be possible under any investment solution when transmission switching is not employed.

We note that another way to visualize optimal cost as a function of relative conges-



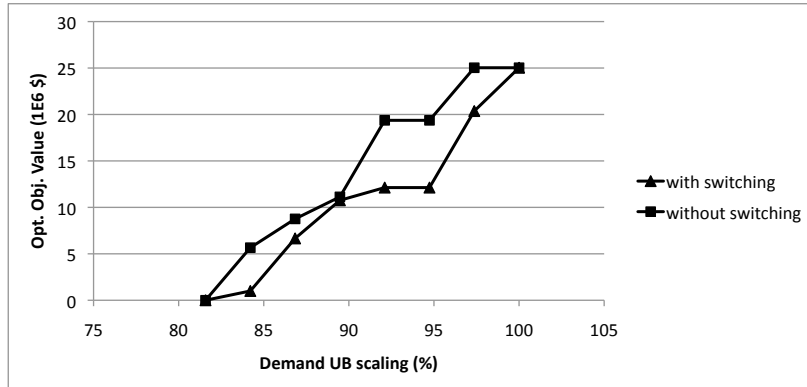


Figure 4.5: Optimal Investment Cost for IEEE24 with Scaled Demand

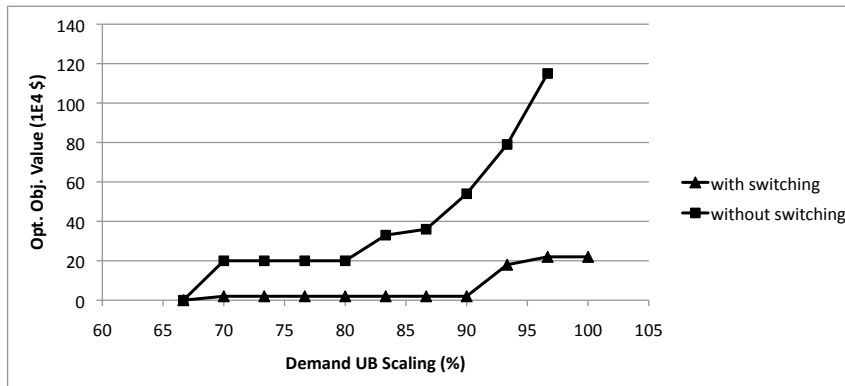


Figure 4.6: Optimal Investment Cost for IEEE14 with Scaled Demand

tion is to vary the transmission line capacities. In the IEEE24 test case, the existing transmission lines are divided among “low” and “high” capacity lines. We fixed the capacity on the low capacity lines, and scaled the capacity on the high capacity lines relative to their nominal capacity. The optimal cost as a function of the scaled line capacity is shown in Figure 4.7.

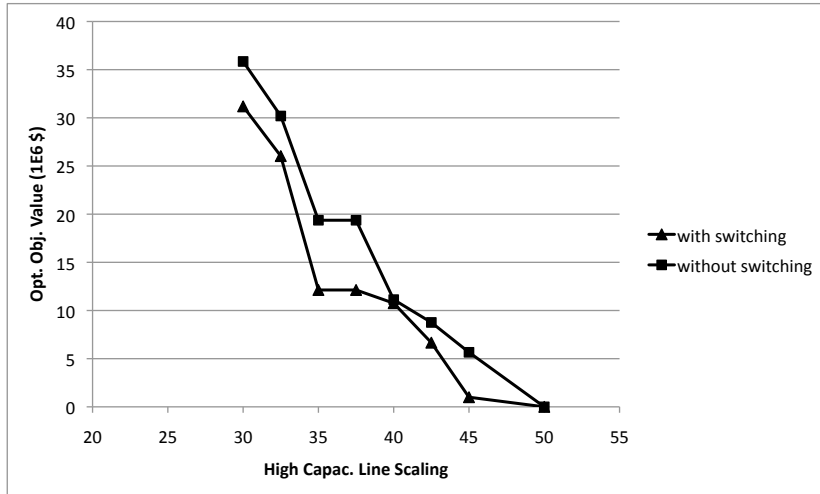


Figure 4.7: Optimal Investment Cost for IEEE24 with Scaled Line Capacities

It is interesting how similar the shape of the curves in the plot in Figure 4.7 are to the curves in the plot in Figure 4.5. Essentially, scaling the upper bound on the demand and scaling the transmission capacities are two different ways of controlling the congestion level in the network. Similar levels of congestion require similar levels of investment, regardless of the source of the congestion.

## 4.8 Conclusion

A robust model for the transmission expansion problem has been presented in which there is uncertainty in both possible line failures and nodal demands, and transmission switching is used as a recovery action. The box uncertainty set that we have chosen to model demand uncertainty can represent uncertainty in loads and uncertainty in renewable generation. This robust uncertainty model is appropriate

in the transmission expansion setting, as probabilities about possible failures or demand scenarios is typically not available, and avoiding blackout events is a critical priority. Within this conservative planning framework, it is shrewd to consider recovery actions such as transmission switching that would introduce flexibility, allowing the operator to achieve maintain reliable operation for a lower investment cost. The algorithm presented here could be used as a tool to evaluate the potential cost savings of allowing transmission switching as a recovery action for various ranges on the demand/renewable generation.

We have presented an algorithm that is based on the Benders' decomposition framework, but utilizes a novel oracle for identifying unsurvivable contingency-demand events. The development of the oracle allows enables the Benders' routine to be used when the number of all contingency-demand pairs is too large to practically use a naive Benders' decomposition.

## CHAPTER V

### Conclusion

This dissertation considers three problems related to the design or operation of the power grid under uncertainty. The unifying theme among these problems is that the goal is to identify a minimum cost solution that ensures feasible operation over a range of possible situations. We develop novel algorithms to solve each of the long or short term planning problems. Computational results are presented to provide proof of concept for each of the proposed solution approaches, and to provide insight into the advantages and limitations of the algorithms in practice.

In Chapter II, a network capacity design problem is presented where there is uncertainty in the nodal supplies and demands. This robust general network design problem lays the foundation for the later chapters which are specifically applicable to the power system domain. We first consider a minimum cost capacity assignment problem where feasible network flows are required for all demand scenarios in the given set. We review a constraint generation algorithm to solve this problem and present implementation details that empirically improve run time. This general decomposition methodology is also used in later chapters to deal with large uncertainty sets. We next present a minimum cost capacity assignment problem where  $\alpha\%$  of the demand scenarios in the given set are required to have feasible network flow solutions. We develop a combinatorial solution approach to solve this chance-constrained

problem in which the previously presented constraint generation algorithm is embedded into a tree-based framework. Based on this optimal algorithm, we develop a greedy algorithm and demonstrate that it identifies near-optimal heuristic solutions for several test cases.

In Chapter III, a short-term operational problem is addressed in which a day-ahead generator schedule is constructed. The traditional unit commitment problem identifies a generator schedule that can meet forecasted demand in the event of any single failure (i.e. N-1 security). We extend this traditional problem to additionally consider (1) a more stringent security requirement where feasible operation is required for any simultaneous failure of  $k$  generators and/or transmission lines (i.e. N- $k$  security), and (2) the addition of transmission switching as an action that operators can use to recover from a contingency.

The N- $k$  security standard significantly increases the difficulty of the problem because there is a combinatorial explosion in the number of contingencies that must be considered when  $k > 1$ . Furthermore, allowing switching as a recovery action greatly increases complexity because traditional decomposition approaches are not applicable when there are binary second stage switching variables. We present a novel algorithm for solving the unit commitment problem that simultaneously addresses both the challenges of the N- $k$  security requirement and the use of transmission switching. This algorithm utilizes a formulation of the problem in which switching decisions are treated as first stage variables, which makes it possible to apply a Benders'-like decomposition. Furthermore, the algorithm employs a Contingency Oracle that can identify an unsurvivable contingency for a given unit commitment solution, and thus eliminates the need to explicitly consider all contingencies of size  $k$  or smaller. As the algorithm proceeds, constraints and switching variables are dynamically generated for the master problem. We present computational results, and provide some analysis on the tradeoff between cost and reliability in the N- $k$  unit commitment problem.

Chapter II is inspired by a transmission expansion problem where there is uncertainty in the demand and renewable generation, and Chapter III considers a unit commitment problem where there is uncertainty in component failures. Chapter IV ties together these two projects and considers a transmission expansion problem where there is uncertainty in demand, renewable generation and component failures. Furthermore, as in Chapter III, transmission switching is allowed as a recovery action. The algorithm proposed to solve this transmission expansion problem is similar in structure to that proposed in Chapter III, but is significantly adapted to solve a long-term planning problem and to address an additional dimension of uncertainty. In the proposed constraint generation algorithm, an Oracle is utilized which can identify a combination of a line failure and a realization of demand that would be unsurvivable for the current investment solution. Computational results are presented that demonstrate that this algorithm can be used to solve for optimal robust transmission expansion solutions even when the set of all contingency-demand pairs in the uncertainty set is very large.

Future efforts to extend the work in this thesis may focus on exploring different types of uncertainty sets. The uncertainty sets for contingencies in Chapters III and IV are defined by contingency cardinality, and a box uncertainty set is used for demand in Chapter IV. The methods proposed here for solving the unit commitment and transmission expansion problems can be easily extended for any uncertainty set where the extreme points can be expressed in terms of binary variables. Other types of uncertainty sets, such as polyhedral or budget uncertainty sets, may satisfy this property and thus may be used within this algorithmic framework. Additionally, beyond component failures and demands, other uncertain parameters may be considered including costs, capacities, etc.

Efforts to improve the run time or scalability of the proposed algorithms may involve moving to a parallel implementation. Each of the decomposition procedures

presented in this thesis utilizes independent oracles and/or subproblems. Distributing these oracles and/or subproblems across multiple processors could result a reduced number of iterations necessary for convergence.

Additionally, more research is needed to fully understand how solutions obtained using DCPF approximations apply to AC systems. Authors including *Coffrin et al.* (2012) have explored ways of better approximating the ACOPF within a linear set of constraints. *Lipka et al.* (2013) explore an alternative linearization of the ACOPF constraints to identify optimal transmission switching decisions. Improved approximations of the power flow equations may be used to extend the models proposed here.

The power grid faces many challenges in the coming years. The congested grid is already being operated close to its limits, and aging infrastructure, and increasing demands for power and levels of renewable generation will necessitate that new investments be made and new operational regimes be explored. As power system operators balance the demand for cheap power and the desire for a reliable system, they will need tools to assist them in evaluating costs. The methods presented in this thesis for solving transmission expansion and unit commitment problems are intended to help operators decide how to plan and operate the grid under uncertainty.

## APPENDICES



## APPENDIX A

# Explicit Formulation of $N-k$ Unit Commitment Model with Transmission Switching

The explicit formulation of the  $N-k$  secure unit commitment problem with transmission switching (3.1) is as follows. The only difference is that in (3.1), feasibility is strictly enforced with the constraints, and in this formulation, there exist slack variables, and feasibility is enforced by strongly penalizing slack variables in the objective function. Both formulations are valid, but this formulation with the slack variables was more convenient to implement.

### A.1 Notation

#### Sets and indices

$N$  set of buses, i.e. nodes in the network.

$\mathcal{G}$  set of all generating units. Each generator  $g \in \mathcal{G}$  is located at exactly one bus  $n \in N$ .

$\mathcal{G}_n$  set of generating units at bus  $n \in N$ .

- $\mathcal{E}$  set of all transmission elements, i.e. arcs in the network. Power may flow in either direction on an arc, but an arbitrary direction is chosen for each arc for convenience of notation.
- $\mathcal{E}_n^{\text{out}}$  set of transmission lines directed out of bus  $n \in N$ .
- $\mathcal{E}_n^{\text{in}}$  set of transmission lines directed into bus  $n \in N$ .
- $h(e)$  bus that transmission element  $e$  is directed into, i.e. the head of  $e$ .
- $t(e)$  bus that transmission element  $e$  is directed out of, i.e. the tail of  $e$ .
- $\mathcal{C}$  set of all contingencies of size  $k$  or fewer, where a contingency is the simultaneous failure of generators and/or transmission lines. Each element  $c \in \mathcal{C}$  corresponds to a set of indicator parameters  $c_e \forall e \in \mathcal{E}$  and  $c_g \forall g \in \mathcal{G}$  which equal 1 if the respective element  $e$  or  $g$  is in the contingency, or 0 if it is not. Let  $i(c)$  be a function which maps contingency  $c$  to its index in the set  $\mathcal{C}$ . Let  $\mathbf{0} \in \mathcal{C}$  indicate the no contingency state (i.e. no elements fail), and let  $i(\mathbf{0}) = 0$ .

### Parameters

- $k$  contingency budget (i.e. at most  $k$  power system elements can fail).
- $T$  number of time periods in the planning horizon (e.g. 24 hrs). Time periods  $t$  indexed from  $1, \dots, T$ .
- $B_e$  electrical susceptance on line  $e \in E$ .
- $P_g^{\text{max}}$  upper bound on the power output at generator  $g \in \mathcal{G}$ .
- $P_g^{\text{min}}$  lower bound on the power output at generator  $g \in \mathcal{G}$  when  $g$  is committed.
- $b_n^t$  load (i.e. power demand) at bus  $i$  in time  $t$ .  $b_n^t \geq 0 \forall n \in N, t \in \{1, \dots, T\}$ .
- $\epsilon_l$  maximum acceptable fraction of total demand in any time  $t$  that is unsatisfied in a contingency  $c \in (\mathcal{C} \setminus \mathbf{0})$  where the contingency is of size  $l$ . Note that for the 0-contingency all load must be satisfied, i.e.  $\epsilon_0 = 0$ .

- $\theta^{\max}$  upper bound on phase angle values.
- $\theta^{\min}$  lower bound on phase angle values.
- $C_g^P$  marginal cost of producing power at generator  $g \in \mathcal{G}$ .
- $C_g^U$  fixed cost incurred whenever generator  $g \in \mathcal{G}$  is started up, i.e. switched to on from off.
- $C_g^D$  fixed cost incurred whenever generator  $g \in \mathcal{G}$  is shut down, i.e. switched to off from on.
- $F_e$  power flow capacity of transmission line  $e \in \mathcal{E}$ .
- $U_g$  minimum number of time periods for which the generator must remain on when generator  $g \in \mathcal{G}$  turned on from off.  $U_g \geq 1$  and  $T - U_g \geq 1 \forall g \in \mathcal{G}$ .
- $D_g$  minimum number of time periods for which generator must remain off when generator  $g \in \mathcal{G}$  turned off from on.  $D_g \geq 1$  and  $T - D_g \geq 1 \forall g \in \mathcal{G}$
- $R_g^U$  ramp-up limit: maximum amount that generator  $g$  can increase output from time  $t$  to  $t + 1$ , given that  $g \in \mathcal{G}$  is committed in both time  $t$  and  $t + 1$ .  
 $R_g^U \leq P_g^{\max}$ .
- $R_g^D$  ramp-down limit: maximum amount that generator  $g$  can decrease output from time  $t$  to  $t + 1$ , given that  $g \in \mathcal{G}$  is committed in both time  $t$  and  $t + 1$ .  
 $R_g^D \leq P_g^{\max}$ .
- $S_g^U$  start-up limit: maximum amount that generator  $g$  can increase output from time  $t$  to  $t + 1$ , given that  $g \in \mathcal{G}$  is *not* committed in  $t$  and *is* committed in  $t + 1$ .  $P_g^{\min} \leq S_g^U < R_g^U$ .
- $S_g^D$  shut-down limit: maximum amount that generator  $g$  can decrease output from time  $t$  to  $t + 1$ , given that  $g \in \mathcal{G}$  *is* committed in  $t$  and *is not* committed in  $t + 1$ .  $P_g^{\min} \leq S_g^D < R_g^D$ .

## Variables

- $x_g^t$  binary commitment variable, equals 1 if generator  $g$  is committed at time  $t$ .
- $y_g^{Ut}$  binary variable, equals 1 if generator  $g$  is switched to on at time  $t$  from being off at time  $(t - 1)$ , and is 0 otherwise
- $y_g^{Dt}$  binary variable, equals 1 if generator  $g$  is switched to off at time  $t$  from being on at time  $(t - 1)$ , and is 0 otherwise
- $p_g^{t,i(c)}$  power output at generator  $g$  in time  $t$  in contingency  $c$ .
- $f_e^{t,i(c)}$  power flow on transmission element  $e$  in time  $t$  in contingency  $c$ .
- $\theta_n^{t,i(c)}$  phase angle of bus  $n$  in time  $t$  in contingency  $c$ .
- $w_e^{t,i(c)}$  binary switching variable, equals 1 if transmission line  $e$  is switched out of service (i.e. if line  $e$  is effectively removed), in time  $t$  in the contingency  $c \in \mathcal{C}$ .
- $q_n^{t,i(c)}$  unsatisfied demand at bus  $n$  in time  $t$  in contingency  $c$ .
- $s_n^{t,i(c)}$  undelivered supply at bus  $n$  in time  $t$  in contingency  $c$ .
- $\hat{q}^{t,i(c)}$  amount by which the total unsatisfied demand exceeds the allowed amount of unsatisfied demand in time period  $t$  in contingency  $c$  (i.e.  $\sum_{i \in N} q_n^{t,i(c)} - \epsilon_l \sum_{n \in N} b_n^t$ )

The vector  $x^t$  in section 3.3 includes the variables  $x_g^t, y_g^{Ut}$  and  $y_g^{Dt}$  for all  $g \in \mathcal{G}$ . Similarly, the vector  $f^{t,i(c)}$  includes the variables  $f_e^{t,i(c)}$  for all  $e \in \mathcal{E}$ ,  $\theta_n^{t,i(c)}$ ,  $q_n^{t,i(c)}$ ,  $s_n^{t,i(c)}$  for all  $n \in N$ , and  $\hat{q}^{t,i(c)}$ . The vector  $w^{t,i(c)}$  includes  $w_e^{t,i(c)}$  for all  $e \in \mathcal{E}$ , and the vector  $p^{t,i(c)}$  includes  $p_g^{t,i(c)}$  for all  $g \in \mathcal{G}$ .

In the matrix notation formulation (3.1), the constraint block (3.1b) represents constraints (A.2)-(A.11). Constraint block (3.1g) represents constraints (A.12)-(A.13). The constraint blocks (3.1c) and (3.1d) represent constraints (A.14)-(A.23). Constraint block (3.1e) represents constraints (A.24). Constraint (3.1f) represents constraint (A.25).

## A.2 Explicit MIP Formulation

$$\min \sum_{t=2}^T \sum_{g \in \mathcal{G}} (C_g^U y_g^{Ut} + C_g^D y_g^{Dt}) + \sum_{t=1}^T \sum_{g \in \mathcal{G}} C_g^P p_g^{t,0} + M \sum_{c \in \mathcal{C}} \sum_{t=1}^T \left( \hat{q}^{t,i(c)} + \sum_{n \in N} s_n^{t,i(c)} \right) \quad (\text{A.1})$$

$$y_g^{Ut} \geq x_g^t - x_g^{t-1} \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.2})$$

$$y_g^{Dt} \geq x_g^{t-1} - x_g^t \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.3})$$

$$\sum_{t'=t}^{t+U_g-1} x_g^{t'} \geq U_g(x_g^t - x_g^{t-1}) \quad \forall g \in \mathcal{G}, t = 2, \dots, (T - U_g + 1) \quad (\text{A.4})$$

$$\sum_{t'=t}^{t+D_g-1} (1 - x_g^{t'}) \geq D_g(x_g^{t-1} - x_g^t) \quad \forall g \in \mathcal{G}, t = 2, \dots, (T - D_g + 1) \quad (\text{A.5})$$

$$y_g^{Ut} \leq 1 - x_g^{t-1} \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.6})$$

$$y_g^{Ut} \leq x_g^t \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.7})$$

$$y_g^{Dt} \leq 1 - x_g^t \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.8})$$

$$y_g^{Dt} \leq x_g^{t-1} \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.9})$$

$$p_g^{t,0} - p_g^{t-1,0} \leq R_g^U x_g^{t-1} + S_g^U (x_g^t - x_g^{t-1}) \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.10})$$

$$p_g^{t-1,0} - p_g^{t,0} \leq R_g^D x_g^t + S_g^D (x_g^{t-1} - x_g^t) \quad \forall g \in \mathcal{G}, t = 2, \dots, T \quad (\text{A.11})$$

$$p_g^{t,i(c)} - p_g^{t,0} \leq R_g^U \quad \forall g \in \mathcal{G}, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.12})$$

$$p_g^{t,0} - p_g^{t,i(c)} \leq R_g^D + P^{\max} c_g \quad \forall g \in \mathcal{G}, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.13})$$

$$\sum_{g \in \mathcal{G}_n} p_g^{t,i(c)} + \sum_{e \in E_n^{\text{in}}} f_e^{t,i(c)} - \sum_{e \in E_n^{\text{out}}} f_e^{t,i(c)} + q_n^{t,i(c)} - s_n^{t,i(c)} = b_n^t \quad \forall n \in N, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.14})$$

$$B_e(\theta_{t(e)}^{t,i(c)} - \theta_{h(e)}^{t,i(c)}) - f_e^{t,i(c)} + M(1 - c_e)w_e^{t,i(c)} \geq -M c_e \quad \forall e \in E, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.15})$$

$$B_e(\theta_{t(e)}^{t,i(c)} - \theta_{h(e)}^{t,i(c)}) - f_e^{t,i(c)} - M(1 - c_e)w_e^{t,i(c)} \leq M c_e \quad \forall e \in E, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.16})$$

$$f_e^{t,i(c)} - F_e(1 - c_e)w_e^{t,i(c)} \geq -F_e(1 - c_e) \quad \forall e \in \mathcal{E}, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.17})$$

$$f_e^{t,i(c)} + F_e(1 - c_e)w_e^{t,i(c)} \leq F_e(1 - c_e) \quad \forall e \in \mathcal{E}, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.18})$$

$$\theta^{\min} \leq \theta_n^{t,i(c)} \leq \theta^{\max} \quad \forall n \in N, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.19})$$

$$0 \leq q_n^{t,i(c)} \leq b_i^t \quad \forall n \in N, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.20})$$

$$s_n^{t,i(c)} - \sum_{g \in \mathcal{G}_n} p_g^{t,i(c)} \leq 0 \quad \forall n \in N, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.21})$$

$$\hat{q}^{t,i(c)} \geq 0 \quad \forall t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.22})$$

$$s_n^{t,i(c)} \geq 0 \quad \forall n \in N, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.23})$$

$$P_g^{\min}(1 - c_g)x_g^t \leq p_g^{t,i(c)} \leq P_g^{\max}(1 - c_g)x_g^t \quad \forall g \in \mathcal{G}, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.24})$$

$$- \left( \hat{q}^{t,i(c)} - \sum_{n \in N} q_n^{t,i(c)} \right) \leq \epsilon_l \sum_{n \in N} b_n^t \quad \forall t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.25})$$

$$x_g^t \in \{0, 1\} \quad \forall g \in G, t = 1, \dots, T \quad (\text{A.26})$$

$$y_g^{Ut}, y_g^{Dt} \in \{0, 1\} \quad \forall g \in G, t = 2, \dots, T \quad (\text{A.27})$$

$$w_e^{t,i(c)} \in \{0, 1\} \quad \forall e \in E, t = 1, \dots, T, c \in \mathcal{C} \quad (\text{A.28})$$

Constraint set (A.2) forces the variable  $y_g^{Ut}$  to take value 1 if the generator is turned on at time  $t$ , which occurs when  $x_g^t = 1$  and  $x_g^{t-1} = 0$ . Otherwise, this constraint allows  $y_g^{Ut} = 0$ . Constraint set (A.3) forces the variable  $y_g^{Dt}$  to take value 1 if the generator is turned off at time  $t$ , which occurs when  $x_g^t = 0$  and  $x_g^{t-1} = 1$ . Otherwise, this constraint allows  $y_g^{Dt} = 0$ .

Constraint set (A.4) defines the minimum up time constraints: if the generator is turned on at time  $t$ , i.e. if  $(x_g^t - x_g^{t-1}) = 1$ , then the generator must remain on for  $U_g$  time periods, enforced by requiring that  $\sum_{t'=t}^{t+U_g-1} x_g^{t'} \geq U_g$ . For any other configuration at time  $t$ , the constraint is not restrictive. Constraint set (A.5) defines

the minimum down time constraints: if the generator is turned off at time  $t$ , i.e. if  $(x_g^{t-1} - x_g^t) = 1$ , then the generator must remain off for  $D_g$  time periods, enforced by requiring that  $\sum_{t'=t}^{t+D_g-1} (1 - x_g^{t'}) \geq D_g$ . For any other configuration at time  $t$ , the constraint is not restrictive.

Constraints (A.6)-(A.9) aim to improve the LP relaxation by restricting the instances in which  $y^U$  or  $y^D$  may be nonzero.

Constraint set (A.10) defines the ramp-up and start-up limits in the no-contingency state. If the generator  $g$  is turned on at time  $t$ , i.e. if  $x_g^{t-1} = 0$  and  $x_g^t = 1$ , then the constraint becomes  $p_g^t - p_g^{t-1} \leq S_g^U$ , and because  $p_g^{t-1} = 0$ , then this constraint enforces that  $p_g^t \leq S_g^U$ . If the generator is on in both times  $t-1$  and  $t$ , i.e. if  $x_g^{t-1} = 1$  and  $x_g^t = 1$ , then the constraint becomes  $p_g^t - p_g^{t-1} \leq R_g^U$ , indicating that the generator output cannot increase by more  $R_g^U$ . For any other situation, this constraint is not restrictive. If  $x_g^{t-1} = 0$  and  $x_g^t = 0$ , the left hand side will be 0, and is bounded from above by 0. If  $x_g^{t-1} = 1$  and  $x_g^t = 0$ , the generator output will decrease and the left hand side  $p_g^t - p_g^{t-1} < 0$ , and is bounded above by  $R_g^U - S_g^U > 0$ .

Constraint set (A.11) defines the ramp-down and shut-down limits in the no-contingency state. If the generator  $g$  is turned off at time  $t$ , i.e. if  $x_g^{t-1} = 1$  and  $x_g^t = 0$ , then the constraint becomes  $p_g^{t-1} - p_g^t \leq S_g^D$ , and because  $p_g^t = 0$ , this constraint enforces  $p_g^{t-1} \leq S_g^D$ . If the generator  $g$  is on at both times  $t-1$  and  $t$ , i.e. if  $x_g^{t-1} = 1$  and  $x_g^t = 1$ , then the constraint becomes  $p_g^{t-1} - p_g^t \leq R_g^D$ , indicating that the generator output cannot decrease by more than  $R_g^D$ . For any other situation, this constraint is not restrictive. If  $x_g^{t-1} = 0$  and  $x_g^t = 0$ , the left hand side will be 0, and is bounded from above by 0. If  $x_g^{t-1} = 0$  and  $x_g^t = 1$ , the generator output will increase and the left hand side  $p_g^{t-1} - p_g^t < 0$ , and is bounded above by  $R_g^D - S_g^D > 0$ .

Constraints (A.12) restricts the increase in power output at generator  $g$  in contingency  $c$  in time  $t$  relative to output of the generator in the 0-contingency not to exceed the ramp-up limit. Similarly, constraints (A.13) restricts the decrease in power output at generator  $g$  in contingency  $c$  in time  $t$  relative to output of the generator in the 0-contingency not to exceed the ramp-down limit. However, the contingency ramp-down constraint has an additional term  $P^{\max}_{c_g}$  which effectively removes the constraint if generator  $g$  is contained in the contingency, wherein the generator output will be 0, and is not bound by ramp-down limitations.

Constraints (A.14) are flow balance constraints. Constraints (A.15)-(A.16) enforce that the DC power flow constraints are enforced only on available transmission lines, which is any line  $e$  where both  $c_e = 0$  and  $w_e^{t,i(c)} = 0$ . If either  $c_e = 1$  or  $w_e^{t,i(c)} = 1$ , then these constraints reduce to  $-M \leq B_e(\theta_{t(e)}^t - \theta_{h(e)}^t) - f_e^t \leq M$ , which when  $M$  is sufficiently large, effectively means that these constraints impose no restriction.

Constraints (A.17)-(A.18) enforce the flow bounds. When line  $e$  is available, meaning that both  $c_e = 0$  and  $w_e^{t,i(c)} = 0$ , then the flow is bounded by  $-F_e$  and  $F_e$ . When line  $e$  is not available because either  $c_e = 1$  or  $w_e^{t,i(c)} = 1$ , then these constraints force the flow on line  $e$  to be 0. Constraints (A.19) restrict the phase angles to be within specified upper and lower bounds.

Constraint (A.20) enforces that the loss-of-load at a given node cannot exceed the total load at that node, and constraint (A.21) enforces that the undelivered supply at a given node cannot exceed the total power generated at that node.

Constraints (A.24) states that if generator  $g$  is committed in time  $t$  ( $x_g^t = 1$ ), its



power output is bounded from above by the upper operating limit  $P_g^{\max}$  and from below by the lower operating limit  $P_g^{\min}$ , and otherwise ( $x_g^t = 0$ ), the generator must have an output of 0. Constraint (A.25) defines the variable  $\hat{q}^{t,i(c)}$  to be lower bounded by the difference between the total loss-of-load in the current time period and the allowable loss-of-load. Because  $\hat{q}^{t,i(c)}$  is being minimized in the objective function, it will be forced to equal the difference, measuring the amount by which the total loss-of-load exceeds the allowable loss-of-load.

## APPENDIX B

# Explicit Formulation of Deterministic Transmission Expansion Model with Transmission Switching

The explicit formulation of the deterministic transmission expansion problem with transmission switching (4.1) is defined as follows. In this deterministic formulation, the uncertain parameters for the contingency  $\bar{c}$  and the demand vector  $\bar{d}$  are assumed to be known.

### B.1 Notation

#### Sets and indices

- $\mathcal{N}$  set of buses, i.e. nodes in the network.
- $\mathcal{G}$  set of all generating units. Each generator  $g \in \mathcal{G}$  is located at exactly one bus  $i \in \mathcal{N}$ .
- $\mathcal{G}_i$  set of generating units at bus  $i \in \mathcal{N}$ .
- $i(g)$  the bus  $i$  such that  $g \in \mathcal{G}_i$ .
- $\mathcal{E}^{\text{cand}}$  set of all candidate transmission elements

- $\mathcal{E}$  set of all existing and candidate transmission elements. Power may flow in either direction on an arc, but an arbitrary direction is chosen for each arc for convenience of notation.
- $\mathcal{E}_i^{\text{out}}$  set of existing and candidate transmission lines directed out of bus  $i \in \mathcal{N}$ .
- $\mathcal{E}_i^{\text{in}}$  set of existing and candidate transmission lines directed into bus  $i \in \mathcal{N}$ .
- $h(e)$  bus that transmission element  $e$  is directed into, i.e. the head of  $e$ .
- $t(e)$  bus that transmission element  $e$  is directed out of, i.e. the tail of  $e$ .

### Parameters

- $B_e$  electrical susceptance on line  $e \in \mathcal{E}$ .
- $\bar{c}_e$  binary parameter indicating the availability of transmission line  $e$  in the given contingency.  $\bar{c}_e = 1$  indicates that the transmission line  $e$  is contained in the contingency and is *not* available.
- $P_g^{\text{max}}$  upper bound on the power output at generator  $g \in \mathcal{G}$ .
- $\bar{d}_i$  load or renewable generation at bus  $i$ .  $\bar{d}_i > 0$  represents true demand, and  $\bar{d}_i < 0$  represents renewable generation.
- $\theta^{\text{min}}, \theta^{\text{max}}$  lower and upper bounds, respectively, on phase angle values.
- $b_e^{\text{line}}$  investment cost of building transmission line  $e \in \mathcal{E}^{\text{cand}}$ .
- $b_e^{\text{switch}}$  investment cost of building transmission switching equipment on line  $e \in \mathcal{E}$ .
- $F_e$  capacity on power flow on transmission line  $e \in \mathcal{E}$ .

## Variables

- $x_e^{\text{line}}$  binary transmission expansion variable, equals 1 if transmission line  $e$  is built, for all  $e \in \mathcal{E}^{\text{cand}}$ .
- $x_e^{\text{switch}}$  binary switching equipment investment variable, equals 1 if transmission switching equipment is built on line  $e$ , for all  $e \in \mathcal{E}$
- $p_g$  power output at generator  $g$ , for all  $g \in \mathcal{G}$ .
- $f_e$  power flow on transmission element  $e$ , for all  $e \in \mathcal{E}$ .
- $\theta_i$  phase angle of bus  $i$ , for all  $i \in \mathcal{N}$ .
- $r_i$  net power injection at node  $i$ , for all  $i \in \mathcal{N}$ .
- $w_e$  binary transmission switching variable, equals 1 if transmission line is switched out (i.e. effectively removed), for all  $e \in \mathcal{E}$ .

## B.2 Explicit MIP Formulation

The explicit deterministic transmission expansion problem is as follows:

$$\min \sum_{e \in \mathcal{E}^{\text{cand}}} b_e^{\text{line}} x_e^{\text{line}} + \sum_{e \in \mathcal{E}} b_e^{\text{switch}} x_e^{\text{switch}} \quad (\text{B.1})$$

$$\sum_{g \in G_i} p_g + \sum_{e \in \mathcal{E}_i^{\text{in}}} f_e - \sum_{e \in \mathcal{E}_i^{\text{out}}} f_e - r_i = 0 \quad \forall i \in N \quad (\text{B.2})$$

$$\theta^{\min} \leq \theta_i \leq \theta^{\max} \quad \forall i \in \mathcal{N} \quad (\text{B.3})$$

$$-F_e(1 - \bar{c}_e - w_e) \leq f_e \leq F_e(1 - \bar{c}_e - w_e) \quad \forall e \in \mathcal{E} \quad (\text{B.4})$$

$$w_e \leq 1 - \bar{c}_e \quad \forall e \in \mathcal{E} \quad (\text{B.5})$$

$$w_e \leq x_e^{\text{switch}} \quad \forall e \in \mathcal{E} \quad (\text{B.6})$$

$$0 \leq p_g \leq P_g^{\max} \quad \forall g \in \mathcal{G} \quad (\text{B.7})$$

$$B_e(\theta_{t(e)} - \theta_{h(e)}) - f_e \leq M(\bar{c}_e + w_e) \quad \forall e \in E \setminus \mathcal{E}^{\text{cand}} \quad (\text{B.8})$$

$$B_e(\theta_{t(e)} - \theta_{h(e)}) - f_e \geq -M(\bar{c}_e + w_e) \quad \forall e \in E \setminus \mathcal{E}^{\text{cand}} \quad (\text{B.9})$$

$$B_e(\theta_{t(e)} - \theta_{h(e)}) - f_e - M(1 - x_e^{\text{line}} + \bar{c}_e + w_e) \leq 0 \quad \forall e \in \mathcal{E}^{\text{cand}} \quad (\text{B.10})$$

$$B_e(\theta_{t(e)} - \theta_{h(e)}) - f_e + M(1 - x_e^{\text{line}} + \bar{c}_e + w_e) \geq 0 \quad \forall e \in \mathcal{E}^{\text{cand}} \quad (\text{B.11})$$

$$-F_e x_e^{\text{line}} \leq f_e \leq F_e x_e^{\text{line}} \quad \forall e \in \mathcal{E}^{\text{cand}} \quad (\text{B.12})$$

$$r_i = \bar{d}_i \quad \forall i \in \mathcal{N} \quad (\text{B.13})$$

$$x_e^{\text{line}} \in \{0, 1\} \quad \forall e \in \mathcal{E}^{\text{cand}} \quad (\text{B.14})$$

$$x_e^{\text{switch}} \in \{0, 1\} \quad \forall e \in \mathcal{E} \quad (\text{B.15})$$

$$w_e \in \{0, 1\} \quad \forall e \in \mathcal{E} \quad (\text{B.16})$$

The objective (B.1) minimizes the total investment cost of building new transmission lines and transmission switching equipment.

Constraint (B.2) requires that power flow balance must be met at each node. Con-

straint (B.3) requires that the node phase angles are within the upper and lower bounds. Constraint (B.4) requires that the line flows be within upper and lower bounds if the line is available. The power flow is forced to 0 if the power line is disrupted in the contingency, or if the transmission line is switched out. Constraint (B.5) requires that a line can only be switched out if that line is not disrupted in the contingency. Constraint set (B.6) requires that a line cannot be switched out unless transmission switching equipment has been installed on that line. Constraint (B.7) specifies that the power output at a generator must be less than its upper bound. The lower bound on the generator dispatch is set equal to 0 because it is assumed that the generator is allowed to be operated in regimes that are inefficient but allowable for short periods when the system is stressed. Constraints (B.8)-(B.9) specify that, for all existing transmission lines, the DC power flow constraints must be enforced if the transmission line is not contained in the contingency and is not switched out.

Constraints (B.10)-(B.11) specify that, for all candidate transmission lines, if the line is built, the DC power flow constraints must be enforced if the transmission line is not contained in the contingency and is not switched out. Constraint (B.12) specifies that, for all candidate transmission lines, the power flow must be 0 for all transmission lines that are not built. Constraint (B.13) specifies that net power flow injection at each node must be equal to the power demand or renewable generation at that node.

### **B.3 Allowing Curtailment**

In the model (B.1)-(B.16), it is assumed that all renewable generation must be used, i.e., curtailment is not allowed. To extend this model to allow curtailment, the following changes should be made.

A curtailment variable is defined: let  $s_i$  be the amount of renewable power that is curtailed at node  $i$ , defined for all  $i \in \mathcal{N}$ . This variable is defined to be non-

negative. It must be equal to 0 if there is no renewable generation at node  $i$ . If there is renewable generation at node  $i$ ,  $s_i$  can be at most equal to  $|\bar{d}_i|$ .

Additionally, let the binary indicator parameter  $\bar{u}_i$  be defined, which indicates whether there is renewable generation at node  $i$  (i.e.,  $\bar{d}_i < 0$ ).  $\bar{u}_i$  is explicitly determined by the demand  $\bar{d}_i$ .

The only constraint in the original model which is modified is constraint (B.13), which is changed to the following:

$$r_i - s_i = \bar{d}_i \quad \forall i \in \mathcal{N} \quad (\text{B.17})$$

Previously, this constraint said that the net injection of power into node  $i$  must exactly equal the demand or renewable generation at node  $i$ . Constraint (B.17) now says that net injection into node  $i$  must equal the demand or renewable generation plus the curtailment. We will force  $s_i$  to be 0 if there is no renewable generation at node  $i$ , in which case this constraint will be the same as the original. If there is renewable generation at node  $i$ , this constraint says that  $s_i$  units of the renewable generation will not leave node  $i$ . For example, suppose  $\bar{d}_i = -10$ , and 2 units of that renewable generation will be curtailed,  $s_i = 2$ . Then  $r_i = -10 + 2 = -8$ , meaning that 8 units of renewable generation originate at node  $i$ .

To enforce the requirement that  $s_i = 0$  if there is no renewable generation at node  $i$ , the following constraint is added:

$$0 \leq s_i \leq M\bar{u}_i \quad \forall i \in \mathcal{N} \quad (\text{B.18})$$

If  $\bar{u}_i = 1$ , this constraint non-restrictive. Otherwise if  $\bar{u}_i = 0$ , meaning that there is no renewable generation, then this constraint requires that  $s_i = 0$ .

To enforce the requirement that, when there is renewable generation at node  $i$ ,  $s_i$

cannot exceed  $|\bar{d}_i|$ , the following constraint is added:

$$s_i \leq -\bar{d}_i + M(1 - \bar{u}_i) \quad \forall i \in \mathcal{N} \quad (\text{B.19})$$

If  $\bar{u}_i = 0$ , this constraint is non-restrictive. Otherwise if  $\bar{u}_i = 1$ , this constraint requires that  $s_i \leq -\bar{d}_i = |\bar{d}_i|$ , because if  $\bar{u}_i = 1$  then  $\bar{d}_i < 0$ .

With these described changes, the nominal model (B.1)-(B.16) can be extended to allow the decision to curtail renewable generation. In the compact matrix formulation of the deterministic model (4.1),  $s_i$  decisions are included in the vector  $y$ , and the  $\bar{u}_i$  parameters are included in the vector  $\bar{d}$ . The constraint (4.1d) would be updated to include a constant vector on the right hand side in order to include constraint (B.19). With these changes, the formulation of the robust counterpart and decomposition can proceed in the same way.



## BIBLIOGRAPHY

## BIBLIOGRAPHY

- Ahmed, S., and A. Shapiro (2008), Solving chance-constrained stochastic programs via sampling and integer programming, in *Tutorials in Operations Research*, edited by Z. Chen and S. Raghavan, pp. 261–269, INFORMS.
- Ahuja, R., T. Magnanti, and J. Orlin (1993), *Network flows: theory, algorithms, and applications*, p. 179, Prentice hall.
- Alguacil, N., J. M. Arroyo, and M. Carrión (2010), Transmission network expansion planning under deliberate outages, in *Handbook of Power Systems I*, pp. 365–389, Springer.
- Arroyo, J., and F. Fernández (2009), A genetic algorithm approach for the analysis of electric grid interdiction with line switching, in *15th International Conference on Intelligent System Applications to Power Systems*, pp. 1–6, IEEE.
- Atamtürk, A., and M. Zhang (2007), Two-stage robust network flow and design under demand uncertainty, *Operations Research*, 55(4), 662–673.
- Ben-Tal, A., A. Goryashko, E. Guslitzer, and A. Nemirovski (2004), Adjustable robust solutions of uncertain linear programs, *Mathematical Programming*, 99(2), 351–376.
- Beraldi, P., and A. Ruszczyński (2002), A branch and bound method for stochastic integer problems under probabilistic constraints, *Optimization Methods and Software*, 17(3), 359–382.
- Bienstock, D., and A. Verma (2010), The  $N - k$  problem in power grids: New models, formulations, and numerical experiments, *SIAM Journal on Optimization*, 20(5), 2352–2380.
- Binato, S., G. C. De Oliveira, and J. L. De Araújo (2001), A greedy randomized adaptive search procedure for transmission expansion planning, *IEEE Transactions on Power Systems*, 16(2), 247–253.
- Center for the New Energy Economy (2013), State renewable portfolio standards hold steady or expand in 2013 session, Available at <http://www.aeltracker.org/graphics/uploads/2013-State-By-State-RPS-Analysis.pdf>.
- Choi, J., T. Tran, A. El-Keib, R. Thomas, H. Oh, and R. Billinton (2005), A method for transmission system expansion planning considering probabilistic reliability criteria, *IEEE Transactions on Power Systems*, 20(3), 1606–1615.

- Christie, R. (1993), UW power systems test archive, <http://www.ee.washington.edu/research/pstca/>.
- Coffrin, C., P. Van Hentenryck, and R. Bent (2012), Approximating line losses and apparent power in AC power flow linearizations, in *IEEE Power and Energy Society General Meeting*, pp. 1–8, IEEE.
- Constable, G., and B. Somerville (2003), *A century of innovation: twenty engineering achievements that transformed our lives*, Joseph Henry Press.
- Da Silva, E. L., J. A. Ortiz, G. C. de Oliveira, and S. Binato (2001), Transmission network expansion planning under a tabu search approach, *IEEE Transactions on Power Systems*, 16(1), 62–68.
- Dahl, G., and M. Stoer (1998), A cutting plane algorithm for multicommodity survivable network design problems, *INFORMS Journal on Computing*, 10, 1–11.
- Delgadillo, A., J. M. Arroyo, and N. Alguacil (2010), Analysis of electric grid interdiction with line switching, *IEEE Transactions on Power Systems*, 25(2), 633–641.
- DeNegre, S. T., and T. K. Ralphs (2009), A branch-and-cut algorithm for integer bilevel linear programs, in *Operations Research and Cyber-Infrastructure*, pp. 65–78, Springer.
- Dentcheva, D., A. Prékopa, and A. Ruszczyński (2000), Concavity and efficient points of discrete distributions in probabilistic programming, *Mathematical Programming*, 89(1), 55–77.
- Fan, N., H. Xu, F. Pan, and P. M. Pardalos (2011), Economic analysis of the  $N - k$  power grid contingency selection and evaluation by graph algorithms and interdiction methods, *Energy Systems*, pp. 1–12.
- Fisher, E. B., R. P. O’Neill, and M. C. Ferris (2008), Optimal transmission switching, *IEEE Transactions on Power Systems*, 23(3), 1346–1355.
- Freris, L. L., and A. M. Sasson (1968), Investigation of the load-flow problem, in *Proceedings of the Institution of Electrical Engineers*, vol. 115, pp. 1459–1470, IET.
- Garver, L. L. (1970), Transmission network estimation using linear programming, *IEEE Transactions on Power Apparatus and Systems*, PAS-89(7), 1688–1697.
- Gomory, R., and T. Hu (1962), An application of generalized linear programming to network flows, *Journal of the Society for Industrial & Applied Mathematics*, 10(2), 260–283.
- Grigg, C., et al. (1999), The IEEE reliability test system-1996, *IEEE Transactions on Power Systems*, 14(3), 1010–1020, <http://www.ee.washington.edu/research/pstca/>.

- Hedman, K. W., R. P. O'Neill, E. B. Fisher, and S. S. Oren (2008), Optimal transmission switching: sensitivity analysis and extensions, *IEEE Transactions on Power Systems*, 23(3), 1469–1479.
- Hedman, K. W., R. P. O'Neill, E. B. Fisher, and S. S. Oren (2009), Optimal transmission switching with contingency analysis, *IEEE Transactions on Power Systems*, 24(3), 1577–1586.
- Hedman, K. W., M. C. Ferris, R. P. O'Neill, E. B. Fisher, and S. S. Oren (2010), Co-optimization of generation unit commitment and transmission switching with N-1 reliability, *IEEE Transactions on Power Systems*, 25(2), 1052–1063.
- Hedman, K. W., S. S. Oren, and R. P. O'Neill (2011), Optimal transmission switching: economic efficiency and market implications, *Journal of Regulatory Economics*, 40(2), 111–140.
- Hemmati, R., R.-A. Hooshmand, and A. Khodabakhshian (2014), Market based transmission expansion and reactive power planning with consideration of wind and load uncertainties, *Renewable and Sustainable Energy Reviews*, 29, 1–10.
- Jabr, R. A. (2013), Robust transmission network expansion planning with uncertain renewable generation and loads, *IEEE Transactions on Power Systems*, 28(4), 4558–4567.
- Jiang, R., M. Zhang, G. Li, and Y. Guan (2010), Two-stage robust power grid optimization problem, Available at [http://www.optimization-online.org/DB\\_FILE/2010/10/2769.pdf](http://www.optimization-online.org/DB_FILE/2010/10/2769.pdf).
- Kamwa, I., A. K. Pradhan, and G. Joós (2007), Automatic segmentation of large power systems into fuzzy coherent areas for dynamic vulnerability assessment, *IEEE Transactions on Power Systems*, 22(4), 1974–1985.
- Khanabadi, M., H. Ghasemi, and M. Doostizadeh (2013), Optimal transmission switching considering voltage security and n-1 contingency analysis, *IEEE Transactions on Power Systems*, 28(1).
- Khodaei, A., and M. Shahidehpour (2010), Transmission switching in security-constrained unit commitment, *IEEE Transactions on Power Systems*, 25(4), 1937–1945.
- Khodaei, A., M. Shahidehpour, and S. Kamalinia (2010), Transmission switching in expansion planning, *IEEE Transactions on Power Systems*, 25(3), 1722–1733.
- Küçükyavuz, S. (2009), On mixing sets arising in chance-constrained programming, *Mathematical Programming*, pp. 1–26.
- Latorre, G., R. D. Cruz, J. M. Areiza, and A. Villegas (2003), Classification of publications and models on transmission expansion planning, *IEEE Transactions on Power Systems*, 18(2), 938–946.

- Li, M., P. B. Luh, L. D. Michel, Q. Zhao, and X. Luo (2012), Corrective line switching with security constraints for the base and contingency cases, *IEEE Transactions on Power Systems*, 27(1), 125–133.
- Lipka, P. A., R. P. O'Neill, S. Oren, A. Castillo, M. Pirnia, and C. Campaigne (2013), Optimal transmission switching using the IV-ACOPF linearization, in *FERC Technical Conference on Opportunities for Increasing Real-time and Day-ahead Market Efficiency Through Improved Software*.
- Liscouski, B., and W. Elliot (2004), Final report on the August 14, 2003 blackout in the United states and Canada: Causes and recommendations, *A report to US Department of Energy*, 40, 4.
- López, J., K. Ponnambalam, and V. H. Quintana (2007), Generation and transmission expansion under risk using stochastic programming, *IEEE Transactions on Power Systems*, 22(3), 1369–1378.
- Luedtke, J. (2013), A branch-and-cut decomposition algorithm for solving chance-constrained mathematical programs with finite support, *Mathematical Programming*, pp. 1–26.
- Luedtke, J., and S. Ahmed (2008), A sample approximation approach for optimization with probabilistic constraints, *SIAM Journal on Optimization*, 19(2), 674–699.
- Luedtke, J., S. Ahmed, and G. Nemhauser (2010), An integer programming approach for linear programs with probabilistic constraints, *Mathematical Programming*, 122(2), 247–272.
- Minoux, M. (1981), Optimum synthesis of a network with non-simultaneous multi-commodity flow requirements, in *Annals of Discrete Mathematics (11) Studies on Graphs and Discrete Programming*, vol. 59, pp. 269–277, North-Holland.
- Minoux, M. (1989), Network synthesis and optimum network design problems: Models, solution methods and applications, *Networks*, 19(3), 313–360.
- Minoux, M. (2010), Robust network optimization under polyhedral demand uncertainty is np-hard, *Discrete Applied Mathematics*, 158(5), 597–603.
- Mudchanatongsuk, S., F. Ordóñez, and J. Liu (2008), Robust solutions for network design under transportation cost and demand uncertainty, *Journal of the Operational Research Society*, 59(5), 652–662.
- NERC, N. (2011), Standard TPL-001-01, system performance under normal conditions, [http://www.nerc.com/docs/standards/sar/Project\\_2006-02\\_TPL-001-1.pdf](http://www.nerc.com/docs/standards/sar/Project_2006-02_TPL-001-1.pdf).
- Ntaimo, L. (2013), Fenchel decomposition for stochastic mixed-integer programming, *Journal of Global Optimization*, 55(1), 141–163.

- O'Neill, R. P., K. W. Hedman, E. A. Krall, A. Papavasiliou, and S. S. Oren (2010), Economic analysis of the N-1 reliable unit commitment and transmission switching problem using duality concepts, *Energy Systems*, 1(2), 165–195.
- Ordóñez, F., and J. Zhao (2007), Robust capacity expansion of network flows, *Networks*, 50(2), 136–145.
- Ouorou, A. (2006), Robust capacity assignment in telecommunications, *Computational Management Science*, 3(4), 285–305.
- Ouorou, A. (2012), Tractable approximations to a robust capacity assignment model in telecommunications under demand uncertainty, *Computers & Operations Research*, 40(1), 318–327.
- Pagnoncelli, B. K., D. Reich, and M. C. Campi (2012), Risk-return trade-off with the scenario approach in practice: a case study in portfolio selection, *Journal of Optimization Theory and Applications*, 155(2), 707–722.
- Pesenti, R., F. Rinaldi, and W. Ukovich (2004), An exact algorithm for the min-cost network containment problem, *Networks*, 43(2), 87–102.
- Petrou, G., C. Lemaréchal, and A. Ouorou (2007), An approach to robust network design in telecommunications, *RAIRO-Operations Research*, 41(04), 411–426.
- Pinar, A., J. Meza, V. Donde, and B. Lesieutre (2010), Optimization strategies for the vulnerability analysis of the electric power grid, *SIAM Journal on Optimization*, 20(4), 1786–1810.
- PPD-21 (2013), Presidential policy directive - critical infrastructure security and resilience, <http://www.whitehouse.gov/the-press-office/2013/02/12/presidential-policy-directive-critical-infrastructure-security-and-resil>.
- Prékopa, A. (1990), Dual method for the solution of a one-stage stochastic programming problem with random RHS obeying a discrete probability distribution, *Zeitschrift für Operations Research*, 34(6), 441–461.
- Read, E. G. (2010), Co-optimization of energy and ancillary service markets, in *Handbook of Power Systems I*, pp. 307–327, Springer.
- Romero, N., N. Xu, L. K. Nozick, I. Dobson, and D. Jones (2012), Investment planning for electric power systems under terrorist threat, *IEEE Transactions on Power Systems*, 27(1), 108–116.
- Romero, R., R. Gallego, and A. Monticelli (1996), Transmission system expansion planning by simulated annealing, *IEEE Transactions on Power Systems*, 11(1), 364–369.
- Romero, R., A. Monticelli, A. Garcia, and S. Haffner (2002), Test systems and mathematical models for transmission network expansion planning, *IEE Proceedings-Generation, Transmission and Distribution*, 149(1), 27–36.

- Ruszczynski, A. (2002), Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra, *Mathematical Programming*, 93(2), 195–215.
- Salmeron, J., K. Wood, and R. Baldick (2004), Analysis of electric grid security under terrorist threat, *IEEE Transactions on Power Systems*, 19(2), 905–912.
- Salmeron, J., K. Wood, and R. Baldick (2009), Worst-case interdiction analysis of large-scale electric power grids, *IEEE Transactions on Power Systems*, 24(1), 96–104.
- Scaparra, M. P., and R. L. Church (2008), A bilevel mixed-integer program for critical infrastructure protection planning, *Computers & Operations Research*, 35(6), 1905–1923.
- Sen, S., and H. D. Sherali (2006), Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming, *Mathematical Programming*, 106(2), 203–223.
- Shen, S. (2013), Optimizing designs and operations of a single network or multiple interdependent infrastructures under stochastic arc disruption, *Computers & Operations Research*, 40(11), 2677–2688.
- Shen, S., and J. C. Smith (2013), A decomposition approach for solving a broadcast domination network design problem, *Annals of Operations Research*, 210(1), 333–360.
- Shen, S., J. C. Smith, and R. Goli (2012), Exact interdiction models and algorithms for disconnecting networks via node deletions, *Discrete Optimization*, 9(3), 172–188.
- Sherali, H. D., and B. M. P. Fraticelli (2002), A modification of Benders’ decomposition algorithm for discrete subproblems: An approach for stochastic programs with integer recourse, *Journal of Global Optimization*, 22(1-4), 319–342.
- Shirokikh, O., A. Sorokin, and V. Boginski (2013), A note on transmission switching in electric grids with uncertain line failures, *Energy Systems*, 4(4), 419–430.
- Silva, I. J., M. J. Rider, R. Romero, and C. A. F. Murari (2006), Transmission network expansion planning considering uncertainty in demand, *IEEE Transactions on Power Systems*, 21(4), 1565–1573.
- Song, Y., and J. R. Luedtke (2013), Branch-and-cut approaches for chance-constrained formulations of reliable network design problems, *Mathematical Programming Computation*, 5(4), 397–432.
- Sorokin, A., J. Portela, and P. M. Pardalos (2012), Algorithms and models for transmission expansion planning, in *Handbook of Networks in Power Systems I*, pp. 395–433, Springer.

- Sridhar, S., and R. Chandrasekaran (1992), Integer solution to synthesis of communication networks, *Mathematics of Operations Research*, pp. 581–585.
- Srivastava, S. C., A. Velayutham, and A. S. Bakshi (2012), Report of the enquiry committee on grid disturbance in northern region on 30th July 2012 and in northern, eastern & north-eastern region on 31st July, 2012, Ministry of Power, Government of India, Available at [http://www.powermin.nic.in/pdf/GRID\\_ENQ\\_REP\\_16\\_8\\_12.pdf](http://www.powermin.nic.in/pdf/GRID_ENQ_REP_16_8_12.pdf).
- Street, A., F. Oliveira, and J. M. Arroyo (2011), Contingency-constrained unit commitment with  $n - K$  security criterion: A robust optimization approach, *IEEE Transactions on Power Systems*, *26*(3), 1581–1590.
- Villumsen, J. C., and A. B. Philpott (2012), Investment in electricity networks with transmission switching, *European Journal of Operational Research*, *222*(2), 377–385.
- Villumsen, J. C., G. Bronmo, and A. B. Philpott (2013), Line capacity expansion and transmission switching in power systems with large-scale wind power, *IEEE Transactions on Power Systems*, *28*(2), 731–739.
- Wang, Q., J.-P. Watson, and Y. Guan (2012), Two-stage robust optimization for  $N - k$  network and contingency-constrained unit commitment, Available at [http://www.optimization-online.org/DB\\_FILE/2012/01/3332.pdf](http://www.optimization-online.org/DB_FILE/2012/01/3332.pdf).
- Watson, J.-P., and D. L. Woodruff (2011), Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems, *Computational Management Science*, *8*(4), 355–370.
- Watson, J. P., R. J. B. Wets, and D. L. Woodruff (2010), Scalable Heuristics for a Class of Chance-Constrained Stochastic Programs, *INFORMS Journal on Computing*, *22*(4), 543–554.
- Wu, P., H. Cheng, and J. Xing (2008), The interval minimum load cutting problem in the process of transmission network expansion planning considering uncertainty in demand, *IEEE Transactions on Power Systems*, *23*(3), 1497–1506.
- Xu, Z., Z. Dong, and K. Wong (2006), Transmission planning in a deregulated environment, *IEE Proceedings-Generation, Transmission and Distribution*, *153*(3), 326–334.
- Yu, H., C. Y. Chung, K. P. Wong, and J. H. Zhang (2009), 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.
- Yu, H., C. Y. Chung, and K. P. Wong (2011), Robust transmission network expansion planning method with Taguchi’s orthogonal array testing, *IEEE Transactions on Power Systems*, *26*(3), 1573–1580.



- Zhao, L., and B. Zeng (2011), An exact algorithm for power grid interdiction problem with line switching, Available at [http://imse.eng.usf.edu/faculty/bzeng/MOChA\\_group/papers/energy/Power\\_Grid\\_Interdiction.pdf](http://imse.eng.usf.edu/faculty/bzeng/MOChA_group/papers/energy/Power_Grid_Interdiction.pdf).
- Zhao, L., and B. Zeng (2012), Robust unit commitment problem with demand response and wind energy, in *IEEE Power and Energy Society General Meeting*, pp. 1–8, IEEE.
- Zhu, J. (2009), *Optimization of power system operation*, vol. 49, Wiley-IEEE Press.