A DISTRIBUTED LINEAR PROGRAMMING MODEL IN A SMART GRID

A Dissertation
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Prakash Ranganathan

In Partial Fulfillment of the Requirements
for the Degree of
DOCTOR OF PHILOSOPHY

Major Department:
Computer Science and Operations Research

April 2013

Fargo, North Dakota

North Dakota State University

Graduate School

Title

A Distributed Linear Programming Model in a Smart grid

By

Prakash Ranganathan

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

DOCTOR OF PHILOSOPHY

SUPERVISORY COMMITTEE:

Dr. Kendall Nygard
Chair

Dr. Saeed Salem

Dr. Simone Ludwig

Dr. Joseph Szmerekovsky

Approved:

April 5, 2013 — Date

Dr. Brian M. Slator — Department Chair

ABSTRACT

Advances in computing and communication have resulted in large-scale distributed environments in recent years. They are capable of storing large volumes of data and, often, have multiple compute nodes. However, the inherent heterogeneity of data components, the dynamic nature of distributed systems, the need for information synchronization and data fusion over a network, and security and access-control issues makes the problem of resource management and monitoring a tremendous challenge in the context of a Smart grid. Unfortunately, the concept of cloud computing and the deployment of distributed algorithms have been overlooked in the electric grid sector. In particular, centralized methods for managing resources and data may not be sufficient to monitor a complex electric grid. Most of the electric grid management that includes generation, transmission, and distribution is, by and large, managed at a centralized control. In this dissertation, I present a distributed algorithm for resource management which builds on the traditional simplex algorithm used for solving large-scale linear optimization problems. The distributed algorithm is exact, meaning its results are identical if run in a centralized setting.

More specifically, in this dissertation, I discuss a distributed decision model, where a large-scale electric grid is decomposed into multiple sub models that can support the resource assignment, communication, computation, and control functions necessary to provide robustness and to prevent incidents such as cascading blackouts. The key contribution of this dissertation is to design, develop, and test a resource-allocation process through a decomposition principle in a Smart grid. I have implemented and tested the Dantzig-Wolfe decomposition process in standard IEEE 14-bus and 30-bus systems. The dissertation provides details about how to formulate, implement, and test such an LP-based design to study the dynamic behavior and impact of an electrical network while considering its failure and repair rates. The computational benefits of the Dantzig-Wolfe approach to find an optimal solution and its applicability to IEEE bus systems are presented.

ACKNOWLEDGEMENTS

A variety of people should be thanked for the completion of this work. First and foremost, I want to thank my major adviser, Dr. Kendall Nygard. It has been an honor to be his Ph.D. student. He has taught me, both consciously and unconsciously, how good software engineering, high performance computing, and operational research concepts can be applied in the area of a Smart grid. I appreciate all his contributions of time, ideas, and funding to make my Ph.D. experience very productive and stimulating. The joy and enthusiasm he has for his research were contagious and motivational for me, even during tough times in the Ph.D. pursuit. The doctoral study has been a long and challenging ride while balancing my full-time teaching at the University of North Dakota.

The members of the Smart grid research group have contributed immensely to my personal and professional time at NDSU. The group has been a source of friendships as well as good advice and collaboration. I am thankful to group members Steve Boughson and Minhaz Chowdhury. I am also thankful to the University of North Dakota for the support and release time to make this experience a reality.

I would also extend my special thanks to Dr. Joseph Szmerekovsky, Dr. Saeed Salem, and Dr. Simone Ludwig. I appreciate their enthusiasm, intensity, willingness to have frequent meetings, and amazing ability to cleave and manipulate linear programming concepts. Without their support, the completion of this long, extremely challenging, and personally changing endeavor would have been much more difficult.

Last, I would like to thank my family members, Jagan, Rohini, and Amma, for all their love and encouragement. Thanks, most of all, for my loving, supportive, encouraging wife, Roopa, whose faithful support during all stages of this Ph.D. is so appreciated.

Thank you, Prakash

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AMI……………………..……………………..…………...Automated Metering Infrastructure

AMPL ……………………………………..…...Algebraic Modeling and Programming Language

BB ……………..………………………………………………………..Branch and Bound

CC …………………...…………………………………………………….. Control Center

DLP ………………………………..…………………………..... Direct Linear Programming

DW ………………………………………………….....…….Dantzig-Wolfe decomposition

DWLG …………………...……………………………...… Dantzig-Wolfe Lagrangian Relaxation

GPS ………………………………..……………………………….. Global Positioning Systems

IEEE ……………………………………………Institute for Electric and Electronic Engineers

LP …………………….………...…………..………………...….….……Linear Programming

MG ……………………………………………………………………................ Micro Grid

MP ……………………………………………………………….……. Master Problem

PDC ……………………………...……………………….....…….. Phasor Data Concentrator

PMU ………………………………..……………………………..Phasor Measurement Unit

S.T ………………………………………………………………..…………….. Subjected to

SP …………………………………………………………………….....……..Sub Problem

WAMS ………………...…………..……………………..………….. Wide Area Monitoring System

CHAPTER 1. INTRODUCTION

The worldwide electric power industry is undergoing a transformation unlike anything it has seen in over a century. The entire supply chain for electricity, including how it is generated, transmitted, distributed, and consumed, is being overhauled with the goal of establishing a more sustainable energy future. Adopting new technologies and the associated market restructuring are a complex undertaking that requires understanding the many interacting variables and conflicting cost functions for various market participants, such as power producers, system operators, load-serving entities, regulators, aggregators, service providers, and consumers. The Smart grid is an information-enriched energy network, and it is going to require substantial information processing, storage, and data-mining resources. An entire new software sector is rising to meet the challenges and to fill the many needs created by its arrival. Spending on the Smart grid is estimated to be $165 billion over the next 20 years, and a good portion of this cost will be on software and data services [RPT07, AW05]. The Smart grid is a complex, highly networked system that must operate in diverse and often challenging environments that combine very large complex facilities with vast numbers of edge nodes, e.g., the smart meters that are its consumer fronting boundary. The Smart meters will require sophisticated software in order to operate efficiently. Upgrading utility information and control infrastructure is critical to maintaining the reliability of the electric distribution system at a time of rising costs.

To meet the enormous challenge of creating the sustainable energy infrastructure of the future, driven by a Smart grid, researchers and practitioners need to quantitatively investigate the complex interactions between different components of the electricity grid and to evaluate the impact of new ideas and technologies, taking into consideration the interdependencies between markets, power flows, and information and communication networks [Ami05]. To assist for a quantitative understanding of the grid, there is an unprecedented need for (near) real-time visibility about the state of the grid and its loads, with volumes of data being collected from smart meters and other sensing devices added to the grid. The dissertation is a collection of design models which address these complex interactions through linear programming (LP) models. LP-based systems provide a paradigm for conceptualizing, designing, and implementing software systems with simplicity and robustness. The proposed master LP model can act autonomously and can communicate with other LP structures across open and distributed environments.

The proposed approaches in the dissertation are comprised of multiple design models as given in the Table 1. The dissertation details these models in the following chapters.

Table 1. List of linear programming models

System models	**Models**
Model 1	An LP-based resource optimization using the Dantzig-Wolfe technique
Model 2	An optimal resource assignment using branch and bound in a Smart grid
Model 3	A probabilistic energy-reallocation technique using linear programming in a Smart grid

The Model 3 contribution is a probability-based LP formulation for a directed network under uncertainty conditions with supply and demand units. Here, my contribution is on modeling and expanding basic integer linear program formulation of a bi-partite graph to network grid structure with known uncertainty. This approach is described in chapter 3. The Model 2 contribution of my work involves developing and implementing a branch-and-bound technique for allocating the distributed energy resources (DERs) to a set of demand units. Here, I discuss how

Distributed Energy Resources can maximize their preferences subjected to various equality and inequality constraints. The contribution of Model 3 is the formulation, implementation and testing of a decomposition procedure for an electric grid resource allocation problem.

In the dissertation, we treat an agent as a piece of software code that can run LP functions continuously and autonomously in an environment where other processes take place and where other agents exist. The sense of "autonomy" means that the agent activities do not require constant human guidance or intervention. I envision this architecture as a distributed system consisting of a collection of autonomous micro grids, which can make decisions themselves, connected through an electrical network and distribution middleware, which enables the Independent System Operator (ISO) to coordinate their activities and to share the resources of the smart grid system so that consumers perceive the system as a single, integrated computing facility.

Smart grid technology promises to revolutionize the way in which electricity is produced, delivered, and utilized. A fundamental problem in building open-distributed systems is to design mechanisms that compute optimal system-wide solutions effectively despite the self-interest of individual micro grids. In particular, using renewable energy sources is envisioned to result in a massively distributed power-generation and distribution system composed of a large number of generating stations operating on disparate renewable technologies. The optimal allocation of existing energy resources becomes a challenge due to the massively distributed nature of generation facilities and consumption sites, and due to the uncertainty caused by inherent random fluctuations in generation. How to allocate resources effectively and computationally to such a highly distributed system? Therefore, I target a distributed resource allocation problem to satisfy both local and global objectives to reach an optimum solution for a Smart grid application by studying the current IEEE electric-grid bus systems. I propose an iterative, distributed algorithm for its solution. The algorithm is scalable for deployment in large electricity networks because it requires fewer computations than modeling via a centralized direct LP implementation.

Objectives of the Dissertation

The main objective of this dissertation can be summarized in a single sentence: "to develop an LP model for the resource-allocation problem in a Smart grid." The dissertation focuses on a distributed linear programming technique for an electric utility's resource-allocation problem. The computational effectiveness of the Dantzig-Wolfe modeling and solution technique is developed, and associated tasks and objectives are as given below.

Objective #1. Formulate a Mathematical Model for the Smart grid Resource Allocation Problem

Task 1: To study and review prior modeling approaches in the literature to ascertain the computational benefits for resource-allocation problems. To study how these approaches can be applied in a Smart grid application by reviewing various techniques, such as LP, fuzzy logic, and heuristic methods, a literature review section is detailed in Chapter 2. Compared to the other formulation types reported in the literature, the Dantzig-Wolfe (DW) LP formulation has a much simpler structure, and I argue that it can be modeled for large-scale systems such as the IEEE 30 bus system for Smart grid. Solution algorithms for LP problems are well established and exist in commercially available software; these solvers, however, are intended for generic problems and cannot detect and take advantage of special problem structures, limiting the size of the problem that can be solved. I address how Smart grid resources can be formulated as a special case structure in order to apply the DW.

Task 2: Formulate the Dantzig-Wolfe decomposition constraints for an IEEE 14 and IEEE 30-bus system. A LP formulation of inter region constraints by decomposition process is described in chapter 5.

Task 3: Formulate bi-directional flow network constraints.

Task 4: For the 14-bus system, decompose the entire grid into multiple regions (Regions 1, 2, and 3), and formulate their constraints.

Task 5: Develop a two-region decomposition of the same problem, and compare with the three-region decomposition formulation. This task provides information about whether all decompositions yield similar performance and computational time savings.

Objective #2. Design, Develop, and Implement a Distributed Solution Procedure for the Mathematical Model

Task 1: Develop any additional constraints and objectives for the proposed problem. Here the objectives are twofold: 1) reduce the overall system failure rate and 2) reduce the repair rate of an IEEE bus system.

Task 2: Study the suitable LP solver tools to implement such a scheme.

Task 3: Design a distributed solution procedure for computing dual values for the Dantzig-Wolfe procedure, and interchange between local micro grids and global objectives.

Task 4: Implement the LP approach directly without decomposition in a large-scale algebraic LP solver such as A Mathematical Programming Language (AMPL).

Task 5: Implement Task 4 with decomposition applied.

Objective #3. Develop an Experimental Design for Testing the Procedure referred to in Objective 2

Task 1: Set up the simulation environment in AMPL for the IEEE 14 and IEEE 30-bus systems.

Task 2: Develop experimental design parameters for power flow constraints in transmission lines. Restrict flow in one direction at a time by using binary operators.

Task 3: Study the feasibility regions of the proposed mixed-integer problem.

Task 4: Develop contingency scenarios about how the method will react to and the feasibility of the solution it provides.

Task 5: Choose the Computer Processing Unit (CPU) run time as one of the main performance parameters.

Objective #4. Carry out the Experimental Testing referred in Objective 3

Task 1: Test the experimental setup for various scenarios. Conduct sensitivity analysis by simulating line failures and observing the optimum.

Task 2: Compare DW procedure with direct LP implementation, and analyze the resultant savings in computations.

Task 3: Test the scalability of the procedure as the number of resources and the system demand increase. For example, how does the procedure scale in a 30-bus system?

The results tested are discussed in detail in Chapter 6, yielding significant results on computational savings for such a decomposition procedure used by utility operators in the event of contingency scenarios. Such a decomposition formulation is neither tested nor formulated using real IEEE bus-system data. The applicability of DW procedure is the first of its kind in considering bi-directional power-flow constraints. I strongly believe that this novel technique will enable local system operators to predict, apply, maintain, and balance resource allocation effectively for their systems in a time-sensitive grid. I assert that this approach will generate broad interest in the utility market for analysis and adaptation. Moreover, the procedure is guaranteed to converge and does not require the revelation of local information from each micro grid, and all algorithm actions can be realized by programmable smart devices on the Smart grid.

Table 2. Objectives and the number of tasks

Completed Objectives	**Completed Tasks**
Objective #1	Task 1, Task 2, Task 3, Task 4, and Task 5
Objective #2	Task 1, Task 2, Task 3, Task 4, and Task 5
Objective #3	Task 1, Task 2, Task 3, Task 4, and Task 5
Objective #4	Task 1, Task 2, and Task 3

Table 2 shows the number of tasks for each objective that are completed as part of the dissertation. The dissertation has seven chapters that include this introduction chapter. Chapter 2 details the literature review on linear programming and Smart grid modeling. Two published papers are included as Chapters 3 and 4, and they describe a probabilistic resource-reallocation modeling as well as a branch and bound technique, respectively.

In summary, I have contributed to the three models using LP in the dissertation. The first model contribution is a probability-based LP formulation for a directed network under uncertainty conditions with supply and demand units. Here, my role of contribution is on modeling and expanding basic integer linear program formulation of a bi-partite graph to a network grid structure with known uncertainty. This approach is described in the published paper given in Chapter 3. The second contribution of my work involves developing and implementing a branch-and-bound technique for allocating distributed energy resources (DERs) to a set of demand units. Here, I present results that show how Distributed Energy Resources can maximize their preferences subjected to various equality and inequality constraints. This model is completely developed, implemented and tested by me using a branch and bound method. This method is outlined in the published paper of Chapter 4.

Chapter 5 detail a decomposition modeling using the Dantzig-Wolfe procedure as well as its formulation for the standard IEEE 14-bus and IEEE 30-bus systems. This procedure is my third major contribution as part of the dissertation. I have directly contributed to all phases of the formulation, implementation and testing of the method and its applicability in an electric grid structure. Chapter 6 describes the the implementation and testing phase and results of the Dantzig-Wolfe procedure. The conclusion and future work are provided in Chapter 7.

CHAPTER 2. LITERATURE REVIEW

The goal of this chapter is to provide prior work conducted with linear-programming approaches for the resource-allocation problem. Operations research (OR) modeling often concerns finding the best quantitative solution for management problems [HL01, Mom01]. The OR methods include mathematical optimization modeling as simulation, and using OR methods has grown significantly since their origination during World War II. Templeman [Tem91] describes quantitative OR methods for designing and controlling industrial and economical operation. Many private and governmental organizations have improved their operations through the successful use of mathematical programming [Wad83, Aro02, Chv83, Dan63, and SS85]. This dissertation focuses on a resource-allocation problem and applies linear programming for the solution approach.

Linear Programming in Practice

LP problems are decision problems where the purpose is to compute values for a set of decision variables in order to optimize (maximize or minimize) a linear objective function, subject to a set of linear constraints. A formal definition for the class of LP problems is given below; first, because this dissertation is primarily about solving LP problems in practice, I briefly consider the context in which such problems arise and the importance of being able to solve them.

A diverse range of real-world problems can be approximated and formulated as LP problems, and there is often great economic or other value attached to finding an optimal solution. The LP field was originally developed to plan military-logistic operations during the Second World War (The word "programming" in LP means "planning."), and since then, the range of applications has flourished. Examples include industrial diamond blending, hired-car fleet management, distribution warehousing and supply chain management, oil refining, and gas pipeline flow. (See [GPS00, BBG77, Bou01, Bou02, and Wil93] for many other applications.)

The value of being able to identify an optimum solution, as opposed to a feasible solution or sometimes no solution at all, can run into the order of many millions of dollars. For instance, a difference of 1% in the objective value in the yearlong PowerGen problem represents an annual cost difference of $520 million [Pow98].

Dr. Warren Powell of Princeton University and others developed a model for the Commercial Transport Division of North American Van Lines. Under high levels of demand uncertainty, this model dispatches thousands of trucks from customer origin to customer destination each week. Working closely with upper management, the project team developed a new type of network model for assigning drivers to loads. The model, LOADMAP, combined real-time information about drivers and loads with an elaborate forecast of future loads and truck activities to maximize profits and service. It provided management with a new understanding about the economics of truckload operations; integrated load evaluation, pricing, marketing, and load solicitation with truck and load assignment; and increased profits by an estimated $2.5 million annually, while providing a higher level of service [PSN+88].

The growth of LP as a practical technique would not have been possible without simultaneous growth in the power and availability of computing. Today, software for LP optimization is highly sophisticated, with several commercial codes being actively developed and marketed. A symbiotic relationship exists between the capacity of the codes and the growth of applications, with solutions for larger problems being demanded in less time as codes improve. This dissertation investigates a well-known, but not well-used, solution method which has the potential to solve large problems quickly by exploiting the problem structure.

LP optimization software uses two main method classes. The simplex method is a gradient descent method that moves along the edge of the feasible region [Chv83, Dan63]. Interior point methods (IPM) move through the interior of the feasible region [Wri97]. I do not dwell on these well-known methods but take the solution of an LP problem with either of these methods as granted, provided that practical considerations allow it. DW decomposition was developed in the late 1950s, a decade after the simplex method and many years before interior point methods were applied to LPs [Dan63, Dan83]. The DW procedure immediately aroused widespread interest, and many attempts were made to implement it as a computational method. Practical experiences, however, were mixed, with some claims of success but no lasting achievements, when measured by the methods used to solve practical problems. There has been no evaluation about and development of different options and strategies for computational implementations, whereas there have been continued research and development for over 50 years with the simplex method and for over 20 years with the Integer Programming Method (IPM). Perhaps the greatest challenge of Dantzig-Wolfe decomposition has been that, when viewed simply as an alternative LP optimization method, successes has been rapidly overtaken by improvements in simplex and IPM implementations. The continued improvement in LP optimization technology could be used to enhance implementation of Dantzig-Wolfe decomposition.

In the dissertation, I primarily focus on this technique referred as Dantzig-Wolfe decomposition (DW). DW is an optimization technique for solving large-scale, block-structured, linear-programming (LP) problems. Problems from many different fields, such as production planning, refinery optimization, and resource allocation, may be formulated as LP problems. Where there is some structure arising from repeated components in the problem, such as handling multiple periods, locations, or products, the problem may potentially be solved using Dantzig-Wolfe decomposition.

As preparation for our practical work, I investigate how suitable block structures can be identified in practical LP problems. I develop the decomposition algorithm from first principles, delineating the theoretical requirements and showing which aspects are open for experimentation in a practical implementation. I illustrate, geometrically, the transformation from the original problem to the Dantzig-Wolfe master problem, and I establish precisely how solutions obtained from the decomposition algorithm correspond to solutions for the original problem. I critically review previous practical work.

Smart grid control systems have used both centralized and decentralized (distributed) approaches [BCP08]. Centralized control systems have the best performance for small-scale power networks and delivering power in one direction (i.e., from substation to loads). Today, the evolution of some power-distribution routines, such as distributed power storage and distributed generators (DGs), requires deploying smart control systems [NF12]. Most traditional power control systems act preventively or reactively to events, whereas more recent control systems add active control options to their strategies [Wan01]. Control architectures for power grids have widely used central and hierarchal methods. Considering their higher efficiency and reliability, decentralized and fully distributed intelligent controllers are beginning to appear [DNS+95].

Optimization techniques have been used for power systems and studied in many resource-allocation applications [Son99, Moo01, and Sal04]. Power-distribution networks are usually designed radially with load-feed flows in one direction. This type of network experiences increased loss, decreased voltage amplitude, and voltage instability (when using a motorized maximum load) due to its radial design and, probably, its long length. One effective solution for improving the performance of distribution networks, from a technical point of view, is using distributed generation supplies.

Generally speaking, the advantages of using a distributed generation pattern can be categorized into two technical and economic aspects [KHS05].

The technical advantages of a distributed generation include decreased line loss, improved voltage profile, decreased environmental pollution, increased energy efficiency, higher-quality power, improved system reliability, and security. On the other hand, economic advantages of applying distributed generation patterns include various investments to improve facilities, decreased operational and operation costs, optimized production, decreased costs to save energy, and increased security for critical loads. A distributed allocation technique using branch and bound is studied for allocating DERS in the context of a Smart grid by Ranganathan [RN10]. A probabilistic approach using linear programming is applied for a Smart grid resource-allocation problem by Nygard [NP11]. Several other Smart grid implementations for a self-healing grid using LP are studied [PFR09]. LP-based decision support for situational awareness is outlined [RN11]. Comprehensive universal markup language (UML) representations of micro grid architecture are developed [PND11]. The preliminary results of the resource-allocation problem in a Smart grid using Dantzig-Wolfe procedure are presented [PKN12].

Development of a Distributed Linear Programming Model

The massive power blackout that caused some 5 million people in Arizona, California, and Mexico to lose electricity was apparently triggered by one person in Arizona. Figure 1 shows a map of electric outage areas. An Arizona Public Service worker "removed a piece of monitoring equipment," which set off a chain reaction across the region, according to the Associated Press [RN12]. The outage appeared to be related to a procedure an Arizona Public Service (APS) employee was conducting at the North Gila substation which is located northeast of Yuma. Operating and protection protocols typically would have isolated the resulting outage to the Yuma area. The reason the isolation did not occur in this case was mostly blamed on a lack of automated programs in place and heavy reliance on heavy manpower, although the investigation into the event is under way. Our approach addresses such events through the LP programs discussed in the next paragraph [DCN04, FES12, GPR+09].

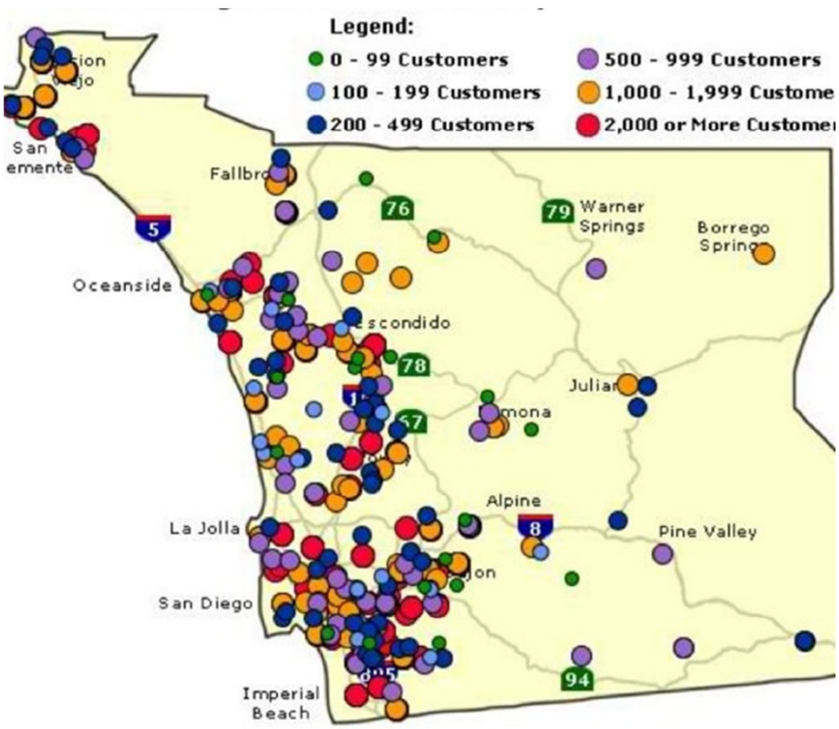


Figure 1. Electric outages in SDG&E Territory: September 8, 2011, 6:39 pm [RN12].

I restrict the attention to the general LP approach and the Dantzig-Wolfe decomposition technique in the context of a Smart grid. The following macro-grid architecture has a centralized agent called an Independent System Operator (ISO) which coordinates the micro-grid activities. An agent is a piece of software code that performs tasks autonomously in the event necessary action is needed to restore the grid process, such as self-healing the grid during an outage, running resource allocation, or scheduling tasks [Kar01, NF12]. Thus, every micro grid has an objective function and constraints that are formulated as an LP problem. I treat each individual LP program as an individual agent that monitors these micro grids and associated activities as shown in Figure 2.

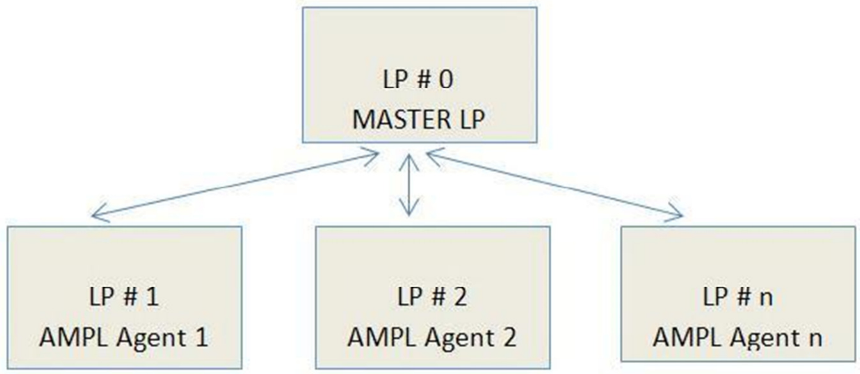


Figure 2. LP as Agents.

Similarly, any macro operations, such as coordinating all micro grids, are conducted by Independent system operator and they are treated as master LP program and constraints run by AMPL. The master LP program interacts with the sub problems via the exchange of dual variables. Figure 3 shows AOLP architecture.

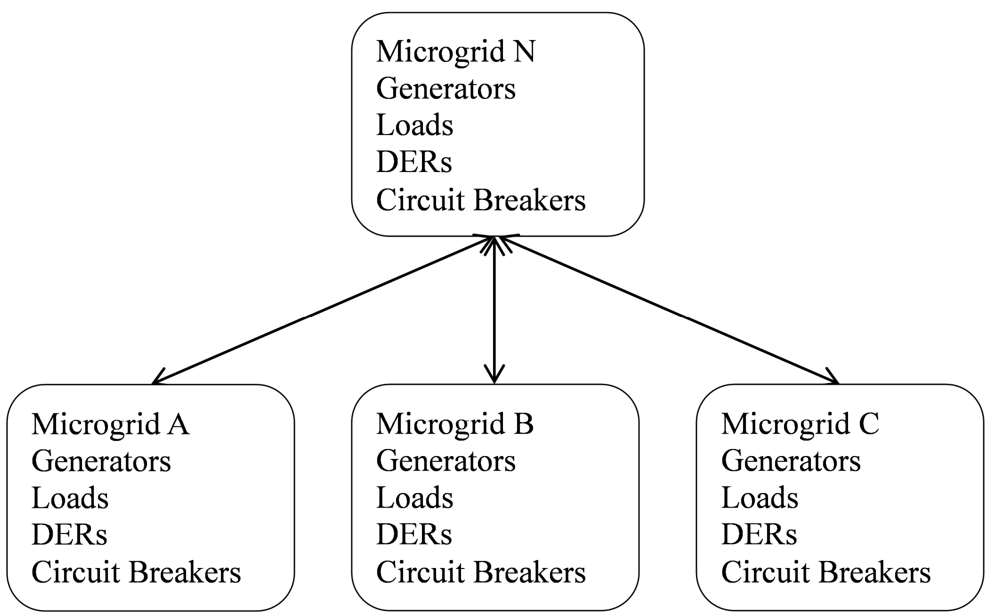


Figure 3. A distributed linear programming architecture.

As an information infrastructure with monitoring, control, and protection functions in a smart transmission grid, the wide-area measurement system (WAMS), based on a synchronized phasor measurement unit (PMU), gradually becomes an important guarantee for the security and stability of power systems. The WAMS can be used to conduct real-time monitoring and control in dynamic system states, enhancing the system's security level because it utilizes the highly precise, synchronous clock system global positioning system (GPS) to build unified time-space synchronization. The WAMS usually includes the PMUs, phasor data concentrator (PDC), control center (CC), and the high-speed data communication networks. Figure 4 shows a local micro grid with PMU-PDC integration as part of the WAMS. I assume that each block of the micro grid structure shown in Figure 4 has these units and integration in place.

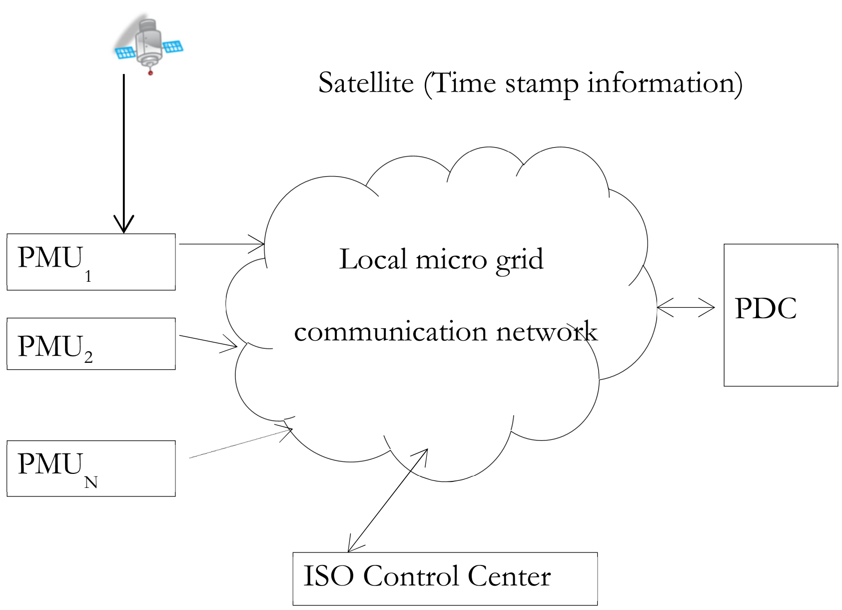


Figure 4. Local micro grid integration as part of WAMS.

The application of the Dantzig-Wolfe procedure would be significant, if it is applied to the WAMS or micro-grid architecture. In the dissertation, I will show the computational significance of a small-scale grid, such as an IEEE bus network, to demonstrate its computational efficiency. Chapters 3 and 4 discuss a resource-allocation procedure where the preliminary results motivated me to continue the computational study of the Dantzig-Wolfe procedure.

CHAPTER 3. ENERGY REALLOCATION IN A SMART GRID

Kendall E. Nygard, Prakash Ranganathan[1], Steve Bou Ghosn, Md. Minhaz Chowdhury

Davin Loegering, and Ryan McCulloch

When a malfunction occurs in a Smart grid electricity provisioning system, it is vitally important to quickly diagnose the problem and take corrective action. The self-healing problem refers to the need to take action in near real time to reallocate power in order to minimize the disruption. To address this need, I present a collection of integer linear programming (ILP) models designed to identify optimal combinations of supply sources, demand sites for them to serve, and the pathways along which the reallocated power should flow. The models explicitly support multiple time periods and the uncertainty associated with alternative sources such as wind power. Model solutions are evaluated using a simulator configured with multiple, intelligent, distributed software agents.

Introduction

A Smart grid is a digital-age electrical generation and distribution system that is fully networked, instrumented, controlled, and automated. A Smart grid is a quintessential machine-to-machine system where the major components, such as generators, relays, transformers, power lines, and electrical meters, are networked and digitally addressable with methods such as Internet Protocol (IP) addresses. Many components are also equipped with sensors and processors that are

[1] The material in this chapter was co-authored by Prakash Ranganathan and Kendall Nygard. Prakash Ranganathan had primary responsibility for developing linear programming formulation of a resource allocation problem that includes flow balance constraints and uncertainty information. Prakash Ranganathan was the primary developer of the modeling conclusions that are advanced here. Prakash Ranganathan also drafted and revised all versions of this chapter. Kendall Nygard served as proofreader and checked the LP formulation conducted by Prakash Ranganathan.

capable of carrying out intelligent actions with little or no human intervention. Available power resources in the Smart grid include conventional types of generating plants and small-scale, renewable Distributed Energy Resources (DERs).

A Smart grid provides great potential advantages for many stakeholders. At the user level, smart meters at power demand sites create possibilities for dynamically pricing electricity, making it possible for consumers to receive lower rates by shifting their usage from periods of high demand to times of low demand. Smart meters also assist utilities by reducing peak loads and allowing meters to take action to optimize resource allocation and to maximize efficiency. When disruptions occur, instrumentation in the grid immediately communicates exact information that pinpoints the location and type of problem, making maintenance and repair activities more responsive and efficient. At the transmission grid level, PMU's placed at strategic locations provide detailed information about grid health, and can trigger messages that report problems or initiate control actions.

Cascading failures that have occurred in past years highlight the need to understand the complex phenomena that can occur in power networks and to develop emergency controls and restoration procedures. In addition to mechanical failures, overloading a line can create power-supply instabilities such as phase or voltage fluctuations. A truly intelligent grid is able to predict impending fault states and failures [ADH+94, CLD+02, AS08, DCN04].

Self-healing capabilities are highly desirable in a Smart grid. I define self-healing as the ability to detect the need for corrective actions in the grid and to autonomously carry out such actions. Once a fault state is detected, the grid should perform appropriate procedures, such as dynamically controlling the power flow to restore grid components from a fault state to normal operation. Examples of common failures that occur in the power grid are power outages, low power quality, overloads that could lead to cascading failures, and service disruptions.

In our work, we model the topology of the Smart grid as an abstract network of nodes representing supply sources, demand sites, and transshipment junctions that are interconnected by links representing transmission lines. Devices such as generators, relays, and transformers are associated with specific nodes. Our models are integer linear programs that provide a self-healing capability by identifying optimal alternatives for reallocating and rerouting power when disruptions and failures occur. Failures affect the ability of certain supply sources to meet energy demands at certain sites. Our primary modeling goal is to balance the flow of power across the system to ensure that no consumer site experiences an outage, while also maximizing the overall efficiency, cost effectiveness, and reliability of the system. Our models account for multiple factors, such as availability, reliability, uncertainty, cost-effectiveness, and consumer preference. The basic modeling template is the Capacitated Transshipment Problem (CTP). An additional model structure incorporates uncertainty at supply sources and ensures that capacities (load limits) for transmission lines and through devices are not exceeded. Uncertainty of the available supply at certain sources is modeled within the integer linear programming framework using chance-constrained programming methods. The integer linear programming models provide the basis for intelligent decision making in the grid as it pertains to resource allocation. An agent-oriented simulation of Smart grid operation is available to test and evaluate alternative resource-allocation solutions.

This chapter is organized in six sections following the Introduction. Section 2 provides the problem statement and necessary background. Section 3 provides a brief review of Smart grid modeling and ILP. In Section 4, we present a collection of ILP models that capture various aspects of the self-healing problem, including an uncertainty model. Section 5 discusses the evaluation of the integer linear programming models in a Smart grid simulation environment. In Section 6, we present conclusions and describe future work.

Problem Statement

When building a Smart grid self-healing model, multiple issues need to be included. Some issues pertain to the physical infrastructure, such as the generators, busses, relays, and transmission lines. Other considerations pertain to the cyber infrastructure, such as the communication networks, storage, protocols, security, and procedures for managing the grid. Here, we focus on the issues in the physical infrastructure that involve resource allocation.

Physical Infrastructure Issues

Distributed Device Control Functions

Most devices associated with nodes in the system must be controllable through remote action. One example is the traditional remote relay-control circuit that is capable of tripping a circuit breaker when electrical current is higher than the threshold. A second example is adaptive control of inverters to ensure stable voltages. Fully centralized control is impossible, but local-device control with distributed intelligence is highly desirable.

Selective Load Control

The ability to selectively switch off customers under certain conditions can help avoid a wide-ranging blackout. This selection process also allows consumers to manage their energy consumption, emphasizing low-cost time periods.

Micro Grid Islanding

Distributed Energy Resources (DERs) are small-scale power generators, such as micro turbines, diesel generators, solar arrays, fuel cells, and wind farms, that are located near a customer cluster. When configured into a micro grid, these systems automatically disconnect themselves from a single point of connectivity with the primary grid when a disruption occurs. When the primary grid is returned to normal conditions, a micro grid must reconnect and resynchonize its operation.

Distributed Pathway Control

The use of alternative, redundant pathways for electricity can be utilized to maintain service under disruptive conditions. The mathematical models we develop are focused on the distributed pathway control issue, with the objective of finding an optimal set of alternative pathways for electricity to flow from supply sources to demand sites while also satisfying constraints for the transmission line's capacity [CT99, CLD+02, DCN04, DNS+95].

Smart Grid Modeling

Several models have been developed to characterize the grid functioning under various conditions. A probabilistic model of load-dependent cascading failure is presented in [KJN+04] and [Kru06]. The important area for managing consumer consumption of electricity in response to supply conditions and pricing has received attention. The role of factors such as load scheduling and market prices in driving consumer behavior and achieving energy efficiency is described in [MWJ+10] and [She95]. In [She95], user preferences are modeled using the concept of the discomfort level within an optimization-problem formulation that balances the load and minimizes user inconvenience caused by demand scheduling. In [Kad09], an energy-consumption scheduling problem is established to minimize the overall energy cost. In [JAW+10], the authors formulated a linear program for distribution management. Kadar [Kad09] developed an optimization model to design the Smart grid network infrastructure. Our work is the first development of optimization models, specifically for real-time self-healing, that directly incorporate uncertainty. Several studies were made on multi-agent based architecture in Smart grid [JW00, NGL+11M PFR09, Wan01, NS02, PKN12].

At the center of any power-system design is the control and communication architecture, comprising the hardware and protocols for exchanging critical status and control signals. In conventional electric power systems, this communication architecture can be accomplished by the Supervisory Control and Data Acquisition (SCADA) system [BW03, RI10].

Integer Linear Programming Models

Early linear programming (LP) models came into prominence and practice during World War II as a means to improve the efficiency and utilization of scarce resources. LP models have a linear objective function to minimize or maximize as well as linear constraints in the form of inequality equations. The simplex method developed by Dantzig [Dan98] has been a mainstay solution methodology, and the more recent interior point method is also prominent. Integer ILP models often arise from node-arc network formulations. Network models of this type date to the pioneering work of Ford and Fulkerson [FF10]. The work in [BBG77] on the Capacitated Transshipment Problem (CTP) gave the first full descriptions of highly efficient solution algorithms for the type of ILP that applies to the self-healing problem.

In a self-healing Smart grid, we assume that disruptions in energy availability occur due to malfunctioning or failed devices and/or inoperative transmission lines [Ami04, AS08, Ami08, BCP08, and CT99]. These disruptions affect the ability of specific supply sources to meet energy demands at specific sites. In response to the associated need to allocate electrical power in alternative ways to accomplish self-healing, we devise several optimization models for increasing complexity to assign supply sources to demand sites. More specifically, we assume that there is J distinct energy demands for which alternative supply sources must be allocated in the short term to respond to disruptions. For each of these J demands, there is a finite set of available supply sources that can be allocated to meet the demand. We index the supply sources by i = 1, 2, 3, …, I. Figure 5 shows a bipartite graph where the supply sources are nodes in the left set and where demand sites are nodes in the right set.

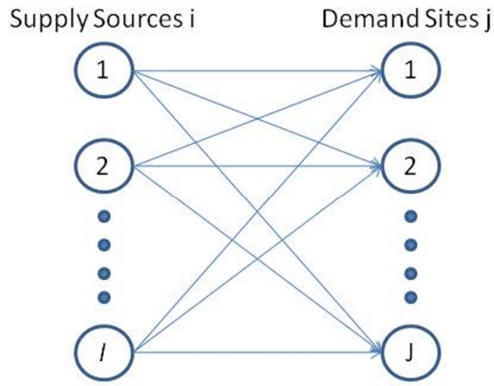


Figure 5. A bipartite graph with supply and demand sites.

The graph's arcs model intact transmission paths with multiple links that utilize sequences of transmission lines, busses, relays, transformers, capacitors, and other devices. The graph is typically not complete, with missing arcs modeling the unavailability of a viable transmission path. We use c_{ij} to denote the cost of assigning supply source i to demand site j. The objective function's parameters are evaluations of a utility function that includes multiple factors taken together, such as prices established under existing contracts, regulatory principles, prices negotiated in near real time, issues related to the transmission paths' viability, and expected reliability. Given supply source i has a specified level, s_i, of energy available; demand sites have a specified level, d_j, of energy needed; sources can supply multiple demand sites; and demand sites can be served from multiple sources. We note that available supplies and demands can be split freely in their allocations, and the variables, x_{ij}, can be viewed as power flows from supply sources to demand sites. We must also ensure that the transmission paths connecting supply sources to demand sites have sufficient capacity to bear their load levels. In a self-healing situation, we let u_{ij} denote increased load level (capacity) that can be allocated to an available pathway connecting nodes i and j. This node to pathway relationship leads to the following problem:

$$Max\ z = \sum_{i=1}^{I}\sum_{j=1}^{J} c_{ij}\, x_{ij} \tag{1}$$

Subject to:

$$\sum_{i=1}^{I} x_{ij} \geq d_j \qquad for\ all\ j \tag{2}$$

$$\sum_{j=1}^{J} x_{ij} \leq s_i \qquad for\ all\ i \tag{3}$$

$$0 \leq x_{ij} \leq u_{ij} \tag{4}$$

One limitation of this basic model is the implicit assumption that the transmission paths modeled by the arcs have no links in common, which may not be the case in practice. This leads to an expanded model formulation that breaks the bipartite graph into a more general network and includes capacities on individual links:

Notation:

The directed graph (network) has node set N and link set A = N x N. We denote typical elements:

i belongs to N, (i,j) belongs to A.

c_{ij} = utility per power flow unit on (i,j)

u_{ij} = capacity (upper bound) of (i,j)

b_i = supply of power at node i (interpret negative b_i as a demand of -b_i units)

Variable x_{ij} = power flow on link (ij).

The problem is to find the set of flows that minimizes the total cost subject to constraints which require i) "flow balance" at each node and ii) capacity restriction on each link. The formulation is as follows:

$$\text{Max } z = \sum_{(i,j)\in A} c_{ij} x_{ij}$$

Subject to:

$$\sum_{i:(i,j)\in A} x_{ij} - \sum_{j:(i,j)\in A} x_{ji} = b_i \qquad \text{for all i,j} \qquad (5)$$

$$0 \leq x_{ij} \leq u_{ij} \qquad \text{for all i,j} \qquad (6)$$

Constraint set 1 consists of flow-balance constraints. The first term in such a constraint is summed over all links with a tail at node i, referred to as the forward star of node i. Similarly, the second term is summed over all links with a head at node i, the "reverse star" of node i. This model requires that the total supply and total demand are equal. The model is known as the Capacitated Transshipment Problem (CTP) in the literature. Figure 6 illustrates the topology for this type of network.

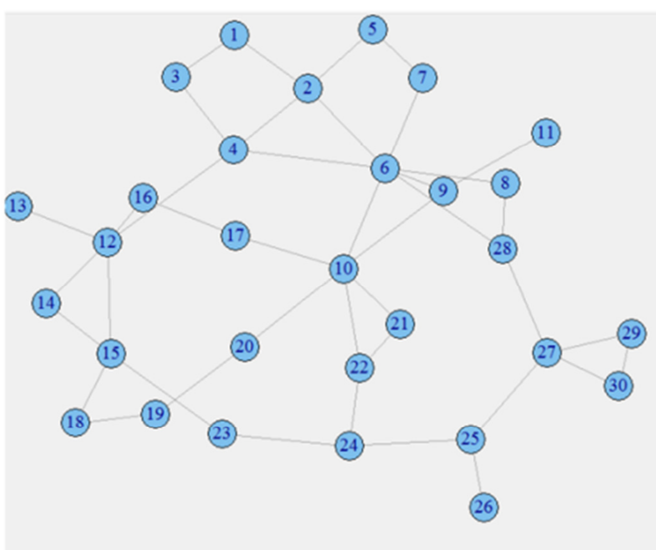


Figure 6. Smart grid topology.

More generally, it may be important to explicitly distinguish supply sources by type. For example, if a site supplies power with wind, there may be specific, important information about that source, such as uncertainty. In the following model, supply sources and demand sites are indexed and differentiated by type, p, where the index takes on values p = 1, 2, …, P. Accordingly, we now have the following notation:

Parameters:

c_{ijp} = utility per unit of flow of type p on link (i,j)

u_{ij} = capacity (upper bound) for flow on link (i,j)

b_{ip}= supply of power of type p at node i (interpret negative b_i as a demand of $-b_i$)

x_{ijp} = flow of power of type p on link (ij)

$$Max\ z = \sum_{(i,j)\in A} \sum_{p\in P} c_{ijp}\, x_{ijp}$$

Subject to:

$$\sum_{i:(i,j)\in A} x_{ijp} - \sum_{j:(i,j)\in A} x_{jip} = b_{ip} \quad \text{For all } (i,j) \tag{7}$$

$$\sum_{p=1}^{P} x_{ijp} \leq u_{ij} \qquad \text{For all (i,j)} \tag{8}$$

$$x_{ijp} \geq 0 \qquad \text{For all (i,j) and p} \tag{9}$$

In the literature, this type of modeling is known as the multi-commodity CTP. The first constraint set enforces that flow balance must occur for each type of power through every node i. The value of b_{ip} is positive at strictly supply-source nodes, negative at strictly demand-site nodes, and zero at pure transshipment nodes. The model allows for supply sources or demand sites to also serve as transshipment points, but such transshipment arrangement would be unusual in practice. The second constraint set allows for each link in the distribution system to be restricted by joint capacity over all flows that pass through it. The model is NP-complete.

Uncertainty in Resource Allocation

We now consider the possibility that supplies and demands at certain nodes are uncertain, as is often the case for supply sources such as wind or solar power. The typical power curve in Figure 7 illustrates the uncertainty of the power output obtainable from a wind machine.

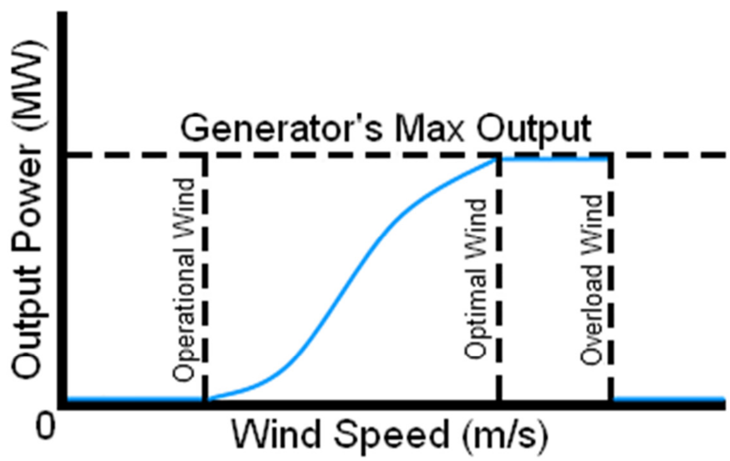


Figure 7. Wind machine power curve.

For a given source node i and power type p, we modify a constraint in set (1) to make it probabilistic as follows:

For a specific i and p

$$\Pr\left[\sum_{i:(i.j)\in A} x_{ijp} = b_{ip}\right] \geq 1 - \propto_{ip} \tag{10}$$

For an easier explanation, we assume that node i is the sole source of commodity type p and that it does not serve as a transshipment point for power originating at other sites. In this constraint, 1 - α_{ip} is the pre-assigned, smallest-allowable probability with which the power available from source i is sufficient to supply b_{ip} units for a demand site. We view α_{ip} as the acceptable risk of not receiving b_{ip} MegaWatts [MW] of electrical power from the specific DER source. For specific values of i and p, we assume that b_{ip} is a random variable that follows a statistical distribution. We note that varying the value of b_{ip} results in different flows through the network links which then, in turn, affects the links' capacity constraints. In the case where b_{ip} follows the normal distribution with mean $E\{b_{ip}\}$ and variance $var\{b_{ip}\}$, we standardize the random variable by subtracting the mean and dividing by the square root of the variance, resulting in the following equivalent, probabilistic condition:

$$Pr\left[\left[\frac{\sum_{i:(i,j)\in A} x_{ijp} - E\{b_{ip}\}}{\sqrt{Var\{b_{ip}\}}}\right] = \left[\frac{b_{ip} - E\{b_{ip}\}}{\sqrt{Var\{b_{ip}\}}}\right]\right] \geq 1 - \alpha_{ip} \tag{11}$$

The true meaning of the equation in the application should be to enforce the condition that the power distributed from supply source i to its outgoing links is at a level for which there is confidence that at least that much power will actually be delivered with a prescribed probability. Any overage would likely be dissipated. This consideration makes it legitimate to replace the equation with an inequality in the analysis:

$$Pr\left[\left[\frac{\sum_{i:(i,j)\in A} x_{ijp} - E\{b_{ip}\}}{\sqrt{Var\{b_{ip}\}}}\right] \geq \left[\frac{b_{ip} - E\{b_{ip}\}}{\sqrt{Var\{b_{ip}\}}}\right]\right] \geq 1 - \alpha_{ip} \tag{12}$$

We let Φ represent the cumulative distribution function for the standard normal distribution and let $K_{\propto_{ip}}$ be the standard normal value such that

$$\Phi\left(\mathrm{K}_{\propto_{\mathrm{ip}}}\right) = 1 - \propto_{\mathrm{ip}} \tag{13}$$

Then, probabilistic condition given above is realized if

$$\left[\frac{\sum_{i:(i,j)\in A} x_{ijp} - E\{b_{ip}\}}{\sqrt{Var\{b_{ip}\}}}\right] \geq K_{\propto_{ip}} \tag{14}$$

The above equation can be rewritten as a constraint:

$$\textstyle\sum_{i:(i,j)\in A} x_{ijp} \leq E\{b_{ip}\} + K_{\propto_{ip}}\sqrt{Var\{b_{ip}\}} \tag{15}$$

This constraint gives the condition that the power delivered will be within the upper-bound value given by the right-hand side with a probability $1 - \alpha_{ip}$. By the symmetry of the normal distribution, constraint

$$\sum_{i:(i,j)\in A} x_{ijp} \leq E\{b_{ip}\} - K_{\propto_{ip}}\sqrt{Var\{b_{ip}\}} \tag{16}$$

sets the requirement for the minimum level of power that is delivered with the prescribed probability. This is a linear constraint that is incorporated into the optimization problem as a so-called "chance constraint," effectively modeling probabilistic conditions within a linear program. As an example, suppose that the supply for node 3 is a wind source that provides power with a mean value of 7 MW and a variance of 4 MW, and that the supply has outgoing distribution links to transshipment nodes 4, 7, and 8. Node index 3 also identifies the type of power generated at node

3. If we allow a 5% risk for not meeting the supply objective, we have the following condition:

$$x_{343} + x_{373} + x_{383} \leq 7 + 1.645 * 2$$

or

$$x_{343} + x_{373} + x_{383} \leq 11.935 \tag{17}$$

The value 1.645 comes from a table of standard normal variates. The condition means that there is a 95% chance that the realizable power from the wind source is no more than 11.935 MW. Using the symmetry,

$$x_{343} + x_{373} + x_{383} \leq 7 - 1.645 * 2$$

Or

$$x_{343} + x_{373} + x_{383} \leq 3.71 \tag{18}$$

This means that at least 3.71 MW of power can be realized with 95% probability. If we increase the prescribed probability to a more stringent 99%, the standard normal variate value is 2.33, and the constraint becomes

$$x_{343} + x_{373} + x_{383} \leq 2.34 \tag{19}$$

The model can also be readily extended to multiple indexed time periods with a time-dependent, supply-demand allocation with fixed costs. This is important for consistency with the time-period planning granularity models used by most utilities.

Smart Grid Simulation

Our Smart grid simulator runs as a Multi-Agent System (MAS) using the Java Agent Development Framework (JADE). Software agents act autonomously and communicate with each other across open and distributed environments, making an agent design ideal for simulating a Smart grid. The agents can sense, act, communicate, and collaborate with each other; be empowered with degrees of autonomy; are decentralized; and have local views and knowledge. The simulation has a low-level, physical-device layer with components that can exhibit fault conditions and fail. A middle layer has agents with a knowledge base, including consumer, DER, device-managing, and monitoring agents. An upper layer consists of management agents that receive the system's state information, carry out analyses, and invoke decision-support models. The optimization models described in this chapter reside at this third level. However, the simulator is also designed to support suites for decision-support models, including fuzzy logic, statistical hypothesis testing, Bayesian belief networks, and constraint satisfaction. These agents also stream reporting information, allowing for convenient performance comparisons [GK03, FG96, KH09, Bri94].

When a three-layer optimization model generates a workable solution in a self-healing situation, it is converted into the associated corrective actions that are done at lower layers to invoke the appropriate response. Each corrective action is modeled by an agent/task pair. The task breaks into detailed roles and actions at the device and transmission-line level. A graphical user interface allows human intervention, if appropriate, or autonomous execution by simply setting initial values for parameters, conditions, and state information.

Conclusions

The optimization models developed include objective functions that maximize a utility function and constraints that ensure feasibility for the resource allocations. Stochastic information can be directly included in the constraints to model situations with known uncertainty. The agent-based simulation provides a realistic and readily validated means for evaluating the performance of the integer linear programming solutions as they would function in an operational Smart grid.

CHAPTER 4. RESOURCE ALLOCATION USING BRANCH AND BOUND

Prakash Ranganathan[2] and Kendall Nygard

The chapter describes a resource-allocation problem in a Smart grid application that is formulated and solved as a binary integer programming model. To handle power outages from the main distribution circuit, the intelligent agents in the Smart grid have to utilize and negotiate with distributed energy resource agents that act on behalf of the local generators in the grid in order to negotiate power-supply purchases to satisfy shortages. We develop a model that can optimally assign these DERs to the available multiple regional utility areas or units (RUAs) that are experiencing power shortages. This type of allocation is a resource assignment problem. The DERs in our model depict the behavior of power generated through a wind turbine, solar generation, or other renewable generation units, and the region or area refers to a centralized distribution unit. The integer programming approach is called a capacity-based Iterative Binary Integer Linear Programming (C-IBILP). All simulation results are carried out using the optimization tool box in MATLAB. Computation results exhibit very good performance for the problem instances tested and validate the assumptions made.

Distributed Energy Resources in Smart Grid

Dynamic, real-time power systems often operate in continuously changing environments, such as adverse weather conditions, sudden transformer failures, or malfunctioning of a sub-system

[2] The material in this chapter was co-authored by Prakash Ranganathan and Kendall Nygard. Prakash Ranganathan had primary responsibility for developing linear programming formulation of a resource allocation problem using branch and bound method. Prakash Ranganathan was the primary developer of the modeling, implementation and testing the conclusions that are advanced here. Prakash Ranganathan also drafted and revised all versions of this chapter. Kendall Nygard served as proofreader and checked the LP formulation conducted by Prakash Ranganathan.

in a transmission or distribution network. These disruptions, along with the complexity of our power systems, cause the energy demand and loads of a power system to fluctuate, potentially resulting in widespread outages and huge price spikes. Data from the North American Electric Reliability Council (NERC) and analyses from the Electric Power Research Institute (EPRI) indicate that the average outages from 1984 to the present time have affected nearly 700,000 customers per event annually [Ami04]. Smaller outages occur much more frequently and affect tens to hundreds of thousands of customers every few weeks or months, while larger outages occur every two to nine years and affect millions. Although preventing these outages remains a challenge, such demand changes (increases or decreases) from consumers can often be offset by distributed energy resources (DERs), which are renewable resources. Solar and wind-based power can satisfy the shortages or reduce the outage levels. In our work, we consider the use of such DER-based standby mechanisms to support any additional demand. We apply an Iterative Binary Integer Linear Programming (IBILP) technique [Web01] to optimally assign DERs for a region based on criteria such as power levels, demands, and preferences. Resource allocation for complex power system is robust with respect to demand variations and power-level fluctuations. The amount of additional power that DERs can generate and that can be effectively utilized in a power network is a measure of robustness. Hence, we argue that a capacity-based Iterative Binary Integer Linear Programming (C-IBILP) model is, inherently, a robust resource allocation.

The structure for the rest of the chapter is as follows. In Section 2, an overview and related work for the Smart grid is discussed. In Section 3, we present a general formulation of this DER assignment problem. In Section 4, we describe how to solve this problem optimally by using a branch-and-bound based (BB) algorithm with equality and inequality constraints. In Section 5, we show the experimental results.

Related Work

Mathematical programming has enjoyed a burgeoning presence in theoretical computer science, both as a framework for developing algorithms and, increasingly, as a bona fide model of computation where the limits are expressed in terms of the formulations' sizes and the formulations' integrality gaps [ABL+06, AAT05, BJN98]. Linear formulations are an appealing model of computation because both optimization and decision problems fit naturally into the framework and because both theoretically tractable and efficient practical algorithms exist to solve linear programs. For instance, state-of-the-art approaches to exactly solve large-scale instances of many NP hard problems rely on integer programming approaches that require the repeated solution of integer programs representing the problems [BJN98]. The polynomial-time algorithms of [Kru56] and other algorithms [KY97, KP94] cannot be used in this application due to high complexity and extensive run-times. Modification will be investigated in future work. We refer to a fundamental model for DER assignment as the capacity-based Iterative Binary Integer Linear Programming (C-IBILP) model. There has been little attention given to this type of approach in smart electrical-grid analyses. To our knowledge, smart grid problems of this type have not been solved for DER allocations using optimization models that perform optimal matching for supply sources' demand sites by predicting generation and market-controlled consumption. Such optimization algorithms are comparable to hard, unsolved problems in inference, optimization, and control [Web02].

Assigning DER to RUA Formulation

To illustrate our problem formulation, we assume that there are 7 areas (RUAs) and 6 DER units with the demand and preference levels shown in Figure 8 and Figure 9 respectively. We define a regional utility area (RUA) as the local-distribution power utilities within the micro grid that distribute power within their network for its loads [Ami04]. For simplicity, we name them Area 1, Area 2, …, Area 7 as illustrated in Figure 8. The power demand and the preferences in Figure 9 depict a demand-driven DER assignment problem that also accommodates preference information. The parameters in the figure are for illustration purposes.

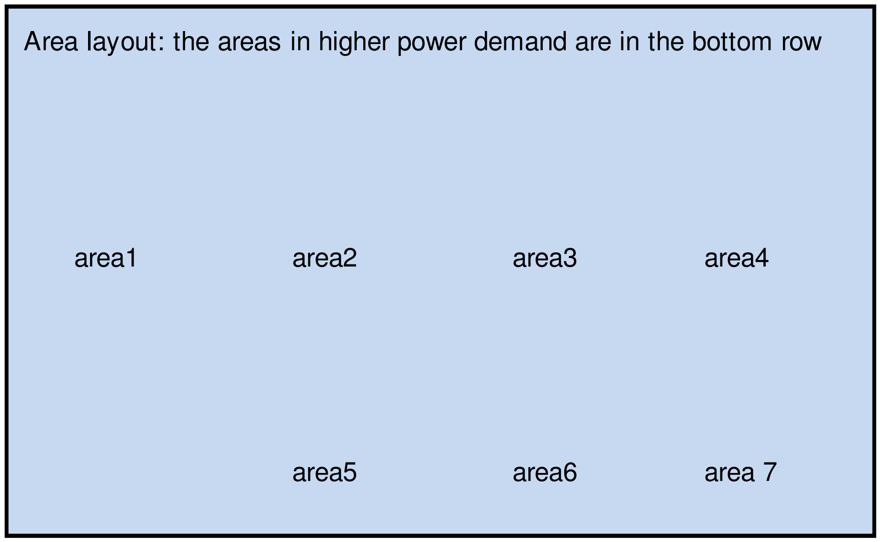


Figure 8. RUA layout.

A simple allocation "text" script in MATLAB would be as follows: text (0.1, .73, 'area1'); text (.35, .73, 'area2'); text (.60, .73, 'area3'); text (.82, .73, 'area4'); text (.35, .42, 'area5'); text (.60, .42, 'area6'); text (.82, .42, 'area 7').

For example, suppose our simulation study is charged with a need to optimally assign six DERs, DER1, DER2, DER3, DER4, DER5, and DER6, to seven regional utility areas (RUAs) based on criteria such as the power-level capacity that these DERs are able to generate and preferences in the area where these DERs wish to operate. For simplicity in our optimization procedure, we also assume that each RUA can have no more than one DER and that each DER gets exactly one RUA. The DERs can have a preference for the area that they wish to join, and their preferences are considered based on their capacity; i.e., the more power they have been able to generate, the higher the capacity is. We weigh the preferences based on the power-level capacity of

DERs through a preference weight matrix (pwm) so that the more power that the DERs can generate, the more their preferences count.

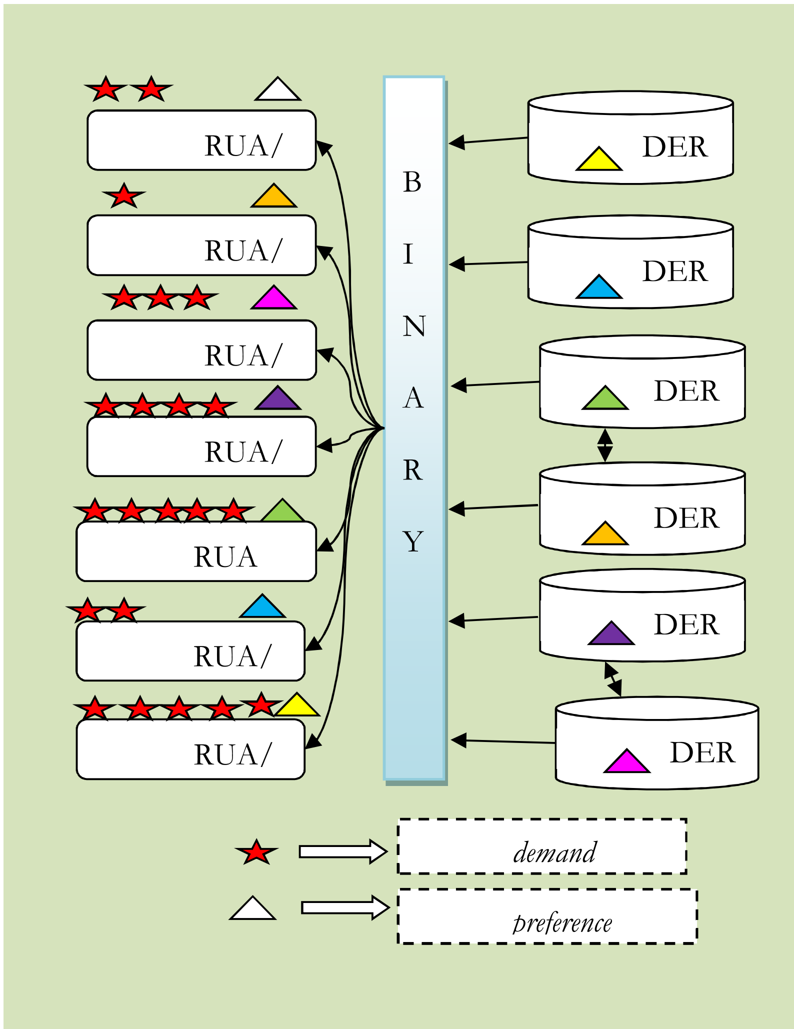


Figure 9. DER vs. RUA assignment problem.

Also, we impose multiple constraints, such as some RUAs have demand, some do not, and some demands are higher than others. DER3 and DER4 often work together, so we would like them to be no more than one RUA away from each other; DER5 and DER6 often work together, so they, too, should be no more than one RUA away from each other. Our approach to solve the assignment problem is to formulate it as a capacity-based Iterative Binary Integer Linear

Programming (C-IBILP) model and to relax the integrality constraints. Our overall objective is to maximize the satisfaction for the preferences weighted by capacity which will allocate these DERs to their areas. This is done through a binary integer programming model by defining a linear objective function. Our algorithm uses a branch-and-bound procedure with linear-programming bounds that have "minimum integer infeasibility" as the branch strategy anda "depth-first search" for the node-search strategy.

To develop our problem formulation, the first step is to choose what each element of our solution vector, $|x|$, represents. We use binary integer variables which represent the specific assignments of DERS to RUAs. If the DER is assigned to a RUA, the variable takes the value 1, and if not assigned, the variable takes the value 0. We consider the DERs in sequential order as DER1, DER2, DER3, DER4, DER5, DER6, and DER7. The nth sequence of elements in vector $|x|$ stores the assignment variables for DER n. In our example, $|x\ (1)|$ to $|x\ (7)|$ correspond to DER1 being assigned to Area 1, Area 2, etc., up to Area 7. In all, vector $|x|$ has 6 sequences of 7 elements each, or 42 elements in all. Each sequence has a single binary variable set to 1, enforcing a multiple-choice condition for each DER.

DER Capacities

We impose constraints based upon DER preference level in the area of operation driven by the capability to generate power. The concept is that, the more power a DER can generate, the higher the preference level is. For example, consider the randomly set power levels given in kilowatts (kW) below.

DER1→9 kW

DER2→10 kW

DER3→5 kW

DER4→3 kW

DER5→1.5 kW and

DER6→ 2 kW

We create a normalized weight vector based on capacity and also assume that certain DERs should be used in some preferred region or area, such as a DER with more power-generation capability being used in high-demand areas. This normalized weigh vector can be obtained in MATLAB as follows:

capacity = [9 10 5 3 1.5 2];

weight vector = capacity/sum (capacity);

```
>> capacity

capacity =

    9.0000   10.0000    5.0000    3.0000    1.5000    2.0000

>> weightvector

weightvector =

    0.2951    0.3279    0.1639    0.0984    0.0492    0.0656
```

RUA Preferences

We set up a preference weight matrix (pwm or prefmatrix) where the rows correspond to areas and the columns correspond to DERS. We assume that each DER will give values for each area so that the sum of all their choices (i.e., their columns) sums to 100. A higher number means that the DER prefers the area. We justify the use of the preference matrix by noting that limitations in algorithm scalability and data availability preclude a fully centralized solution to the problem of interest. Thus, decision making must be decentralized, and we accordingly divide the power network into many smaller RUAs, where the prefmatrix concept is applied to individual regions in the network. An example of DER preferences is shown below:

DER1 = [0; 0; 0; 0; 10; 40; 50];

DER2 = [0; 0; 0; 0; 20; 40; 40];

DER3 = [0; 0; 0; 0; 30; 40; 30];

DER4 = [1; 3; 3; 3; 10; 40; 40];

DER5 = [3; 4; 1; 2; 10; 40; 40];

DER6 = [10; 10; 10; 10; 20; 20; 20];

The i^{th} element of a DER's preference vector is the value the i^{th} RUA. Thus, the combined preference matrix is expressed as "prefmatrix":

prefmatrix = [DER1 DER2 DER3 DER4 DER5 DER6];

```
>> prefmatrix

prefmatrix =

     0     0     0     1     3    10
     0     0     0     3     4    10
     0     0     0     3     1    10
     0     0     0     3     2    10
    10    20    30    10    10    20
    40    40    40    40    40    20
    50    40    30    40    40    20
```

<u>Case 1</u>

We treat the above "prefmatrix" arrangement as case 1 for analysis. We then weigh the preference matrix by the |weightvector| to scale the columns by capacity. We also reshape this matrix as a vector in column-order so that it corresponds to our |x| vector. This task is achieved in MATLAB script as follows:

PM = prefmatrix * diag (weightvector);

```
>> diag(weightvector)

ans =

    0.2951         0         0         0         0         0
         0    0.3279         0         0         0         0
         0         0    0.1639         0         0         0
         0         0         0    0.0984         0         0
         0         0         0         0    0.0492         0
         0         0         0         0         0    0.0656
```

c = PM (:);

```
>> PM

PM =

         0         0         0    0.0984    0.1475    0.6557
         0         0         0    0.2951    0.1967    0.6557
         0         0         0    0.2951    0.0492    0.6557
         0         0         0    0.2951    0.0984    0.6557
    2.9508    6.5574    4.9180    0.9836    0.4918    1.3115
   11.8033   13.1148    6.5574    3.9344    1.9672    1.3115
   14.7541   13.1148    4.9180    3.9344    1.9672    1.3115
```

Our objective is to maximize the total preference measure weighted by capacity. This is a linear objective function, max c'*x or, equivalently, min -c'*x, with c being the DER preferences. We use the BINTPROG script of MATLAB to run our model that is defined as follows:

$$\min_{x} \; f^{T}x \; : \; \begin{cases} A.x \leq b, \\ Aeq.x = beq \\ x : \text{binary} \end{cases}$$

where

f=Vector containing the coefficients of the linear objective function.

A=Matrix containing the coefficients of the linear inequality constraints, $A \cdot x \leq b$.

b= Vector corresponding to the right-hand side of the linear inequality constraints.

Aeq= Matrix containing the coefficients of the linear equality constraints, $Aeq \cdot x = beq$.

beq= Vector containing the constants of the linear equality constraints.

x0=Initial point for the algorithm.

Options: Option structure containing the algorithm's options.

x: A binary integer solution vector—that is, its entries can only take on the values 0 or 1.

Constraints

The first set of constraints requires that each DER is assigned to exactly one area. For example, because DER2 is the second DER, we enforce the condition that |sum(x (8:14)) =1|. We represent these linear constraints in an equality matrix, Aeq, and right-hand side vector, beq, where |Aeq*x = beq|, by building the appropriate matrices. Matrix |Aeq| consists of ones and zeros. For example, the second row of |Aeq| corresponds to DER2 getting exactly one RUA, so the row pattern is as follows:

0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 ... 0 0 0

These conditions are implemented in MATLAB code as follows:

|Aeq (2, :)*x = 1| is equivalent to |sum(x (8:14)) = 1|.

```
numAREAS = 7;

numDERS = 6;

numDim = numAREAS * numDERS;

onesvector = ones (1, numAREAS);
```

Each row of Aeq corresponds to one DER.

Aeq =blkdiag (onesvector, onesvector, onesvector, onesvector, onesvector, onesvector);

beq = ones (numDERS, 1);

view the structure of Aeq, that is, where there are nonzeros (ones) Figure;

The second sets of constraints are inequalities. These constraints specify that each area has no more than one DER in it; i.e., each AREA has one DER in it or is empty. We build matrix |A| and vector |b| such that |A*x <= b| to capture these constraints. Each row of |A| and |b| corresponds to a RUA, so row 1 corresponds to the DER assigned to RUA 1. In this case, the rows have the pattern type shown below for row 1:

1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 ... 1 0 0 0 0 0 0

Each subsequent row is similar but is shifted (circularly) to the right by one spot by one. For example, row 3 corresponds to RUA 3 and enforces |A(3,:)*x <= 1| so that AREA 3 cannot have more than one DER. Figures 10 and 11 illustrate the equality and inequality constraints.

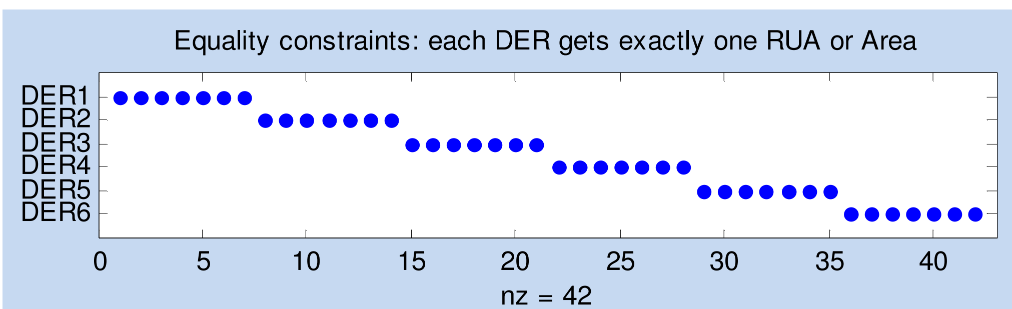


Figure 10. Equality constraints.

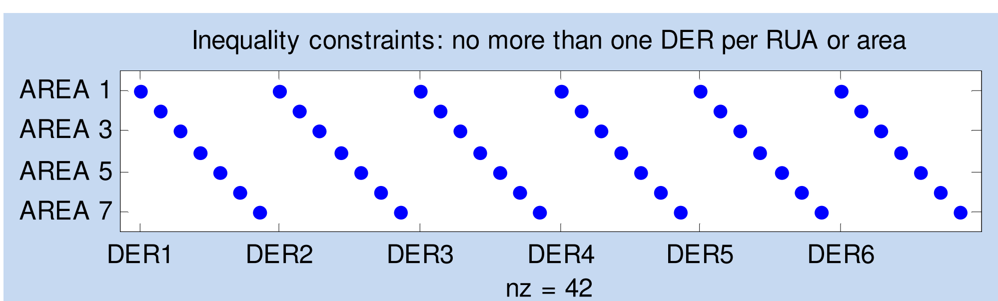


Figure 11. Inequality constraints.

A = repmat(eye(numAREAS),1,numDERS);

b = ones(numAREAS,1);

where repmat represents the replicate and tile array. Elements of the next constraint set are also inequalities, so they are added to matrix |A| vector |b| that already contain the above inequalities. We wish to enforce that DER3 and DER4 are no more than one area (RUA) from each other, and similarly for DER5 and DER6 are no more than one area away from each other. First, the symmetric distance matrix for the RUAs is built using physical locations and Manhattan (i.e., the "taxicab" metric).

D = zeros(numAREAS); // generates a 7 x 7 zero matrix

Setting up the top-right half of the matrix,

D(1,2:end) = [1 2 3 2 3 4];

D(2,3:end) = [1 2 1 2 3];

D(3,4:end) = [1 2 1 2];

D(4,5:end) = [3 2 1];

D(5,6:end) = [1 2];

D(6,end)   = 1;

The lower-left half is the same as the upper-right D = triu(D)' + D. We find the RUAs that are more than one distance unit away.

```
>> D

D =

     0     1     2     3     2     3     4
     1     0     1     2     1     2     3
     2     1     0     1     2     1     2
     3     2     1     0     3     2     1
     2     1     2     3     0     1     2
     3     2     1     2     1     0     1
     4     3     2     1     2     1     0

>> D = triu(D)' + D;
>> D

D =

     0     1     2     3     2     3     4
     2     0     1     2     1     2     3
     4     2     0     1     2     1     2
     6     4     2     0     3     2     1
     4     2     4     6     0     1     2
     6     4     2     4     2     0     1
     8     6     4     2     4     2     0
```

[AREAA,AREAB] = find(D > 1);

numPairs = length(AREAA);

This finds |numPairs| area pairs. For example, if DER3 occupies one area in the pair, then DER4 cannot occupy the other AREA in the pair; otherwise, it would be more than one unit away in terms of AREA. The same condition holds for DER5 and DER6. This situation gives |2*numPairs| additional inequality constraints which we add to |A| and |b|. By adding rows to A, we accommodate these constraints as follows:

numrows = 2*numPairs + numAREAS;

A((numAREAS+1):numrows, 1:numDim) = zeros(2*numPairs,numDim);

For each pair of AREAS in numPairs, for the |x(i)| that corresponds to DER3 in |AREAA| and for the |x(j)| that corresponds to DER 4 in |AREAB|, x(i) + x(j) <= 1; i.e., either DER3 or DER4, but not both, can occupy one of these AREAS.

Branch-and-Bound (BB) Strategy

The branch-and-bound algorithm is a well-known optimal solution method. Branch-and-bound (BB) algorithms are methods for solving non-convex global optimization problems [BB91, LW66, BJN98, Moo91]. They are exact (non-heuristic), in the sense that they calculate a provable upper and lower bound on the globally optimal objective value and that they terminate when all suboptimal feasible solutions have been eliminated. Branch-and-bound algorithms can be computationally slow. In the worst case, they require effort that grows exponentially with problem size. We achieve fast convergence in our problems. We note that, due to total unimodularity of the basic A matrix, a network-based, customized linear-programming solver could be used to provide the lower bounds very quickly in large problems. The BB algorithm is a well-known algorithm in the research community [Wol98]. An example run of the BB algorithm is shown in Figure 12, followed by a snippet of MATLAB code showing the iterative output for each node displayed in the branch-and-bound algorithm. We let the BINTPROG choose the starting point.

```
x0 = [];
f = -c;
options = optimset('Display','iter','NodeDisplayInterval',1);
[x,fval,exitflag,output] = bintprog(f,A,b,Aeq,beq,x0,options);
fval
exitflag
output
```

To reduce the number of nodes explored, the time, or the number of iterations taken, there are alternative options available. BINTPROG use the options to adjust the algorithm with differing nodes and branching-variable strategies [Moo91, MG05].

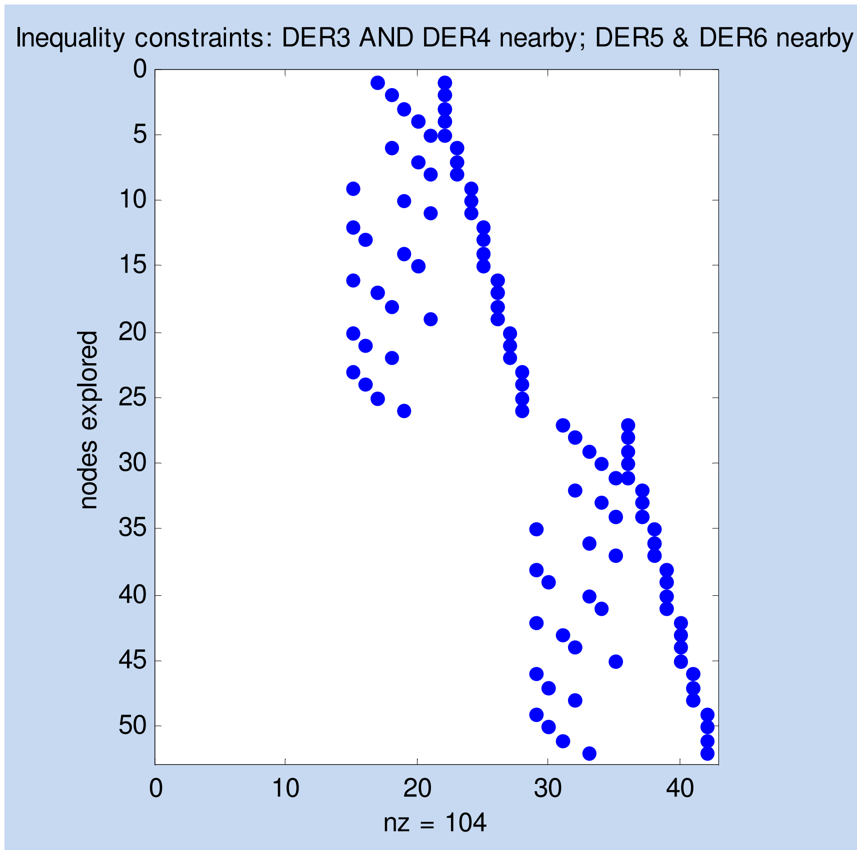


Figure 12. Branch and bound algorithm with inequality constraints.

For example, the default branching strategy is |'maxinfeas'|, which chooses the variable with the maximum integer infeasibility for the next branch, that is, the variable with the value closest to 0.5. Running the problem again with the branching strategy set to |'mininfeas'|, the option of minimum integer infeasibility is chosen (that is, the variable where the value is closest to 0 or 1, but not equal to either).

For structuring the tree, depth-first and best-node search strategy are available. For example, in "df," at each node in the search tree, if there is a child node one level down in the tree that has not already been explored, the algorithm chooses one such child to search. Otherwise, the algorithm moves to the node one level up in the tree and chooses a child node one level down from that node. In best-node (bn) strategy, the node with lowest bound on the objective function is the default. In our limited computational experience, convincing and acceptable results are quickly reached. For future work, we plan to increase the scale of our test problems and to investigate improved BB schemes.

Results

The simulation is done with a MATLAB platform. The prebuilt in command for branch and bound algorithm was used to simulate the following cases.

<u>Case 1</u>

The results show that the optimal value is reached after 163 iterations with 54 nodes participating in 1.22 seconds (case 1) using the capacity-based Iterative Binary Integer Linear Programming (C-IBILP) based branch-and-bound method which maximizes the satisfaction of the DER preferences weighted by its capacities. The final output shown in Figure 13 presents the DER allocation with RUA1, or area 1, treated as empty for optimal assignment.

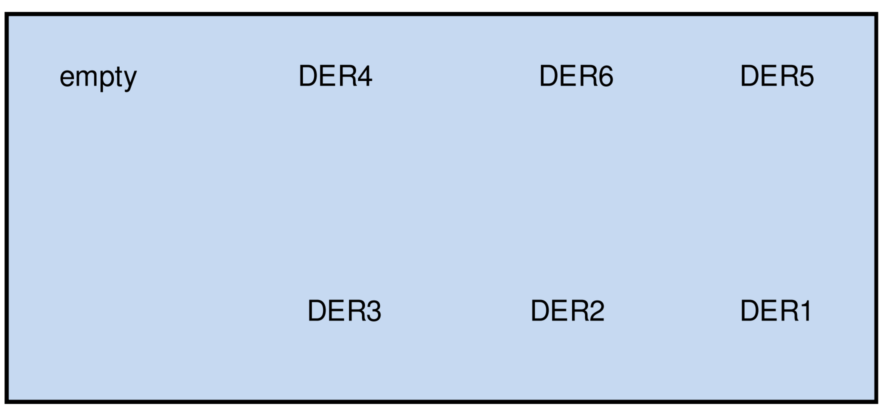


Figure 13. An optimal DER assignment solution for case 1.

Case 2

If we change the DER preferences according to the matrix shown below, then the optimal solution is reached with 13 iterations and 1 node in 0.047 seconds by using the default-node and branch strategies shown in Figure 14.

```
>> prefmatrix

prefmatrix =

     0     0     0     1     3    70
     0     0     0     3    40    10
     0     0     0     3     1    10
     0     0    70     3     2    10
    10    90    30    10    10    20
    40    40    40    60    40    20
    50    40    30    40    40    20
```

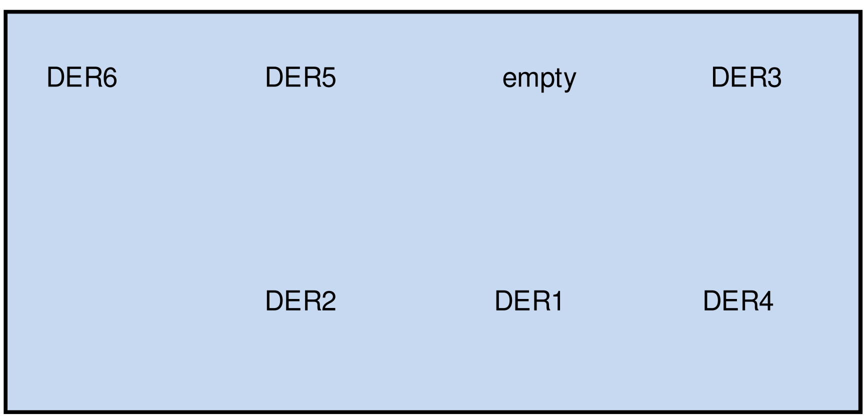


Figure 14. An optimal DER assignment solution for case 2.

Conclusions

The chapter presented a resource-assignment problem for Smart grid applications. The capacity-based Iterative Binary Integer Linear Programming (C-IBILP) model was designed to specify an optimal allocation of distributed energy resources (DERs) during power-outage periods to satisfy shortages. Computational results from two cases studied showed that our C-IBILP algorithm exhibits very good performance for the problem instances tested. A branch-and-bound algorithm for the Smart grid problem was described. It combined the extension results previously presented in the literature with new elements, such as a new lower bound that works by exploiting some properties connected with the ad-hoc branching rule we developed. Computational results established that the algorithm is very competitive. It greatly improved the results obtained by methods that have recently appeared in the literature. Our approach's limitation was that the method does not scale well for larger DERs. Our current efforts involve extending this assignment model to a more scalable assignment formulation where more DERs can serve each RUA.

CHAPTER 5. RESOURCE ALLOCATION USING DW DECOMPOSITION

Dantzig-Wolfe decomposition is a technique for dealing with linear and integer programming problems that have embedded substructures that permit efficient solution. The technique has been applied successfully in a variety of contexts [Dan63, Chv83, BJN98]. Implementing DW-decomposition-based algorithms poses various challenges. The primary challenges revolve around convergence of the dual-bound computations and, in the context of integer programming, the enforcement of integrality restrictions. The standard view of DW decomposition is that it exploits the linear-programming formulation of the Lagrangian dual. This so-called master linear program has an exponential number of variables that are handled using dynamic column generation. An alternative view is that DW is a reformulation technique that gives rise to a mixed-integer master program. Viewing DW as a reformulation technique allows for the development of a theoretical framework that facilitates the handling of branching decisions and cutting planes in the master program.

Why Decompose?

There are computational and organizational advantages when using decomposition algorithms. From a computational perspective, the advantage is that sub problems are usually easier to solve than the original problem. The sub problems are, by definition, smaller than the original problem. Moreover, in many cases, the sub problems have special properties, such as convexity, sparsity, or network constraints, that enable the use of efficient, specialized algorithms to solve them [Dan63]. By decomposing the original problem, we can take advantage of the efficient solution method available for sub problems. Decomposition algorithms also allow Smart grid problems to be solved in a distributed manner. The key point when designing a decomposition algorithm in this environment is that only limited communication between the sub problems and the master problem is required. The aim is that different engineering teams or sub groups of a Smart grid, such as the transmission side or distribution side, can solve only their own sub problems and that only a small amount of communication is required with the central coordinator.

Objective Function and Illustration of DW Algorithm

The linear-programming problem set up for the Dantzig-Wolfe solution technique can be formulated as follows:

Minimize Cx

Ax=b (Master problem constraints)

x∈ to X (where X is set of corner points

To illustrate the Dantzig-Wolfe decomposition method, we first consider a 4-bus system with 2 generators and 3 loads that plans to maximize its power based on certain constraints (Figure 15). A bus is a communication link that transports energy from one point to another point. A network model of the 4-bus system is shown in Figure 16. The buses are declared using variables x1, x2, x3, and x4. The objective is to maximize the power flow while keeping these constraints in mind. Constraints 1 and 3, namely C1 and C3, describe the line voltages that these buses should not exceed, 5 kv and 8 kv, respectively. Constraints C2 and C4 specify the average repair time needed for these buses in the event of any failure to correct themselves, and the time should not exceed 12 hours/year and 10 hours/year, respectively. Constraint C5 tells us that the total bus reactive load power should not exceed 7 kW, and constraint C6 points out that the total resistance of these buses should not exceed 17 Mega ohms.

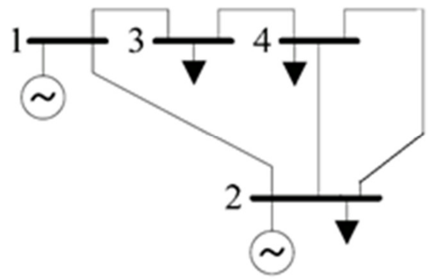


Figure 15. A 4-bus system.

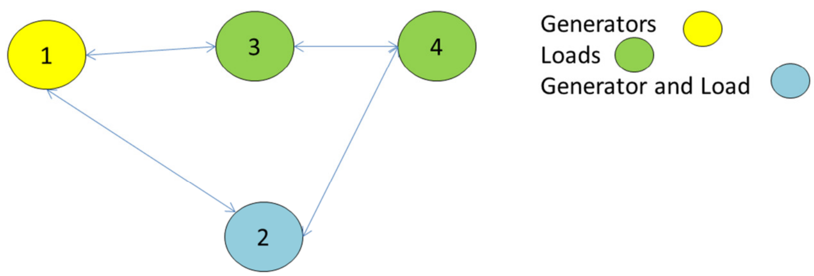


Figure 16. A network model of 4-bus system.

Let the objective function be defined as follows:

$$\text{maximize power}: \ 6x_1 + 5x_2 + 3x_3 + 4x_4;$$

Subject to

$$x_1 + x_2 \leq 5\text{kV}; \qquad \text{Constraint \# 1}$$

$$3x_1 + 2x_2 \leq 12 \text{ hours per year}; \qquad \text{Constraint \# 2}$$

$$x_3 + 2x_4 \leq 8\text{kV}; \qquad \text{Constraint \# 3}$$

$$2x_3 + x_4 \leq 10 \text{ hours per year}; \qquad \text{Constraint \# 4}$$

$$x_1 + x_2 + x_3 + x_4 \leq 7\text{kw}; \qquad \text{Constraint \# 5}$$

$$2x_1 + x_2 + x_3 + 3x_4 \leq 17\text{M}\Omega; \qquad \text{Constraint \# 6}$$

We can compute the border feasible points (corner points) using AMPL software directly or using a graphical approach. The corner points for the first two equations (i.e., C1 and C2) are (0, 0), (0,5), (2,3), and (4,0), and of the four points, point (2,3) maximizes the objective function with a value of Z=27. Similarly, the corner points for the next two equations (i.e., C3 and C4) are (0, 0) (5, 0), (0, 4), and (4, 2), and of the four points, point (4, 2) maximizes the objective function with a value of Z=20. Constraints C5 and C6 are called master constraints.

$$minimize \ \ power: -6x_1 - 5x_2 - 3x_3 - 4x_4;$$

$$\text{minimize}: \sum \mathrm{C_j \lambda_j X_j}$$

$$\text{subject to} \sum \mathrm{A\lambda_j X_j} \le \mathrm{b_i};$$

$$\text{subject to} \sum \lambda_j = 1;$$

$$\text{i.e.}, \mathrm{Y\,P_j - C_j} > 0;$$

Considering dual variables (Y) and the entering column (Pj), we can formulate the stopping criteria for the DW algorithm to terminate as follows:

$$(\mathrm{w}, \alpha)\begin{pmatrix} \mathrm{Ax_j} \\ 1 \end{pmatrix} - \mathrm{cX_j} > 0;$$

$$(\mathrm{w}, \alpha - \mathrm{C})\mathrm{X_j} + \alpha > 0;$$

where y [i.e., w and α] is the dual variable,

p_j = entering column,

c_j = objective function coefficients.

Let us introduce two slack variables, S1 and S2, to the master problem constraints because these variables do not have any impact on the objective function or on the optimal values. We create an identity matrix as shown in Table 3 for S1, S2, and λ1. We then assign the right-hand side coefficients of master constraints (C5 and C6) toward the RHS column. The coefficients of the master constraints are in the A matrix, and the initial corner values for x are taken as (2,3,4,2). The resultant product is (11, 17).

Table 3 shows the procedure for computing optimal values using the Dantzig-Wolfe technique. Table 3 has two slack variables, S1 and S2, and four convexity constraints are needed to attain an optimal solution for the above-mentioned constraints and objective function. The table provides analysis for attaining a basic feasible solution and determining which variable leaves the basis. The computation step for row and column operations is explained after the table.

Table 3. Simplex table for Dantzig-Wolfe for 4 bus system

Variables	**S1**	**S2**	**λ1**	**RHS**	$\begin{pmatrix} Ax \\ 1 \end{pmatrix}$	**θ**	**θ**
S1	1	0	0	7	11	7/11	
S2	0	1	0	17	17	1	
λ 1	0	0	1	1	1	1	
Z-Cj	0	0	0	0	47		
λ 2	1/11	0	0	7/11	1	4/11	7/4
S2	-17/11	1	0	68/11	0	20/11	17/5
λ 1	-1/11	0	1	4/11	0	7/11	4/7
Z-Cj	-47/11	0	0	-329/11	76/11		
λ 2	1/7	0	-4/7	3/7	0	4/7	3/4
S2	-9/7	1	-20/7	36/7	0	48/7	3/4
λ 3	-1/7	0	11/7	4/7	1	3/7	4/3
Z-Cj	-23/7	0	-76/7	-237/7		20/7	
λ 4	¼	0	-1	¾	1		
S2	-3	1	4	0	0		
λ 3	-1/4	0	2	¼	0		
	-4	0	8	36			

$$\begin{pmatrix} Ax_j \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 3 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \\ 4 \\ 2 \end{pmatrix} = \begin{pmatrix} 11 \\ 17 \\ 1 \end{pmatrix}$$

The minimum value of θ leaves the basis, hence row S1 with θ=7/11 is replaced with λ2 in the next set. Then, a set of row operations is performed in the following sequence.

$$\lambda_2 = \frac{s_1}{11};\ S_2 = (s_2 - 17\lambda_2);\ \lambda_1(\text{new}) = \lambda_1 - \lambda_2; Z - C = Zc - 47\lambda_2$$

Using the termination condition of the DW technique, we solve for

Max (WA-C)x + α:

$$WA = (-47/11, 0)\begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 3 \end{pmatrix}$$

$$WA = \begin{pmatrix} -47/11 & -47/11 & -47/11 & -47/11 \end{pmatrix}$$

$$WA - C = \begin{pmatrix} 19/11 & 8/11 & -14/11 & -3/11 \end{pmatrix}$$

Hence, the current objective function is modified as follows:

$$\frac{19}{11}x1 + \frac{8}{11}x2 - \frac{14}{11}x3 - \frac{3}{11}x4$$

Let us apply the same corner points into the new objective function, (0,0), (0,5), (2,3) and (4,0), to get a point that maximizes the objective function. Here it is (4,0) for the first two constraints. Similarly, (0,0) maximizes constraints c3 and c4 from border points (0,0), (5,0), (0,4), and (4,2).

Then, the current corner-point values are Xj=(4,0,0,0) and maximize at Z=76/11. Again, let us calculate the new entering column information, Pj, that will enable us to obtain $\begin{pmatrix} Ax_j \\ 1 \end{pmatrix}$.

$$P_j = \begin{pmatrix} Ax_j \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 3 \end{pmatrix}\begin{pmatrix} 4 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 4 \\ 8 \\ 1 \end{pmatrix};$$

$$B^{-1}\begin{pmatrix} Ax_j \\ 1 \end{pmatrix} = \begin{pmatrix} 1/11 & 0 & 0 \\ -17/11 & 1 & 0 \\ -1/11 & 0 & 1 \end{pmatrix}\begin{pmatrix} 4 \\ 8 \\ 1 \end{pmatrix} = \begin{pmatrix} 4/11 \\ 20/11 \\ 7/11 \end{pmatrix};$$

The value of dual variables w and α is noted (-47/11,0, 0) to improve the objective function. This process continues until an optimal solution is reached as shown in Table 4.

Table 4. Final corner points and objective

x1,x2,x3,x4	Optimal value	Iteration
(0,0,0,0)	0	Start 0
(2,3,4,2)	47	Iteration 1
(4,0,0,0)	-76/11	Iteration 2
(4,0,0,4)	20/7	Iteration 3
(4,0,0,0)	0	Terminated

As noted in Table 4, corner point (4,0,0,0) repeats, and we have obtained the preferred objective function to have a value of 0. This terminates the algorithm, enabling the calculation of the final corner point that maximizes our objective function, resulting in Z=36. An AMPL output using a direct LP implementation for our 4-bus example is shown in Figure 17.

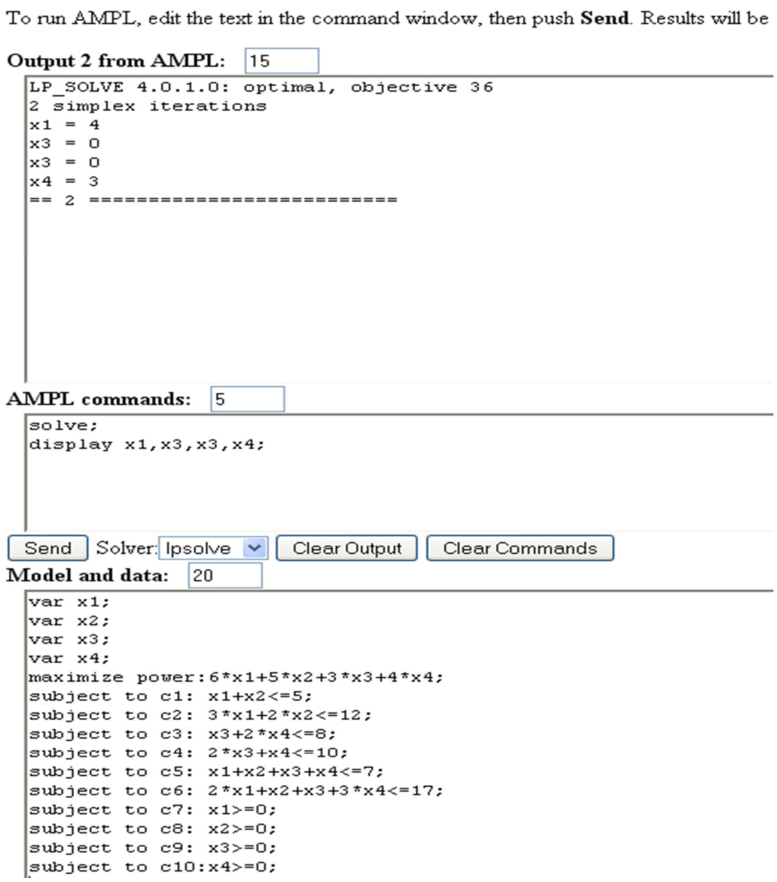


Figure 17. An AMPL run of DLP.

The data file has the following parameters: data; param x1:=0; param x2:=0; param x3:=0; param x4:=0.

Notice that solution for both the direct LP and DW approaches is the same. The final corner values that buses can take to achieve their maximization objective can be calculated as

$= \lambda_3 * x_3 + \lambda_4 * x_4$

$= 1/4 * (4,0,0,0) + 3/4 * (4,0,0,4)$

$= (1,0,0,0) + (3,0,0,3)$

Thus, x1, x2, x3, x4 =(4,0,0,3).

For these point combinations, the final optimal value results in Z=36, hence Bus 1 and Bus 4 should be kept at 4 kv and 3 kv of generation to obtain a maximum power-flow capacity of 36 MW in the network. Thus, at each DW iteration, a relaxed version of the master problem is solved. Then, N sub problems are solved using the reduced costs of the master linear program as parameters. As a result, each sub problem generates a candidate variable that is introduced in the master problem. The current relaxed master problem is updated by including all candidate variables found by the sub problems. We code both the Dantzig-Wolfe technique and direct LP formulation for the same 4-bus problem and observe the computational savings as shown in Table 5.

Table 5. Computational savings of Dantzig Wolfe over direct approach

# of variables	**Using LP directly (in seconds)**	**D-Wolfe (in seconds)**	**Constraints**	**Difference**	**Optimum Power**	**Iterations using LP**
4	0.000999	.000183	10	8.16×10^{-4}	Z=36	2
5	0.01999	.000181	11	.019802	Z=39	3
6	0.000999	.00017	12	.01982	Z=37.66	6
7	0.000999	.00014	13	8.59×10^{-4}	Z=37	7
8	0.000999	.00014	14	8.59×10^{-4}	Z=37.1	9

Thus, Dantzig-Wolfe decomposition is an efficient optimization method when applied to large-scale problems with a special block-angular structure. Unlike the sub-gradient method, the Dantzig-Wolfe decomposition method is able to properly define new Lagrangian multipliers for subsequent sub problems. The fast and monotonic convergence is a distinct feature of the Dantzig-Wolfe decomposition [Dan63, Chv83].

LP Formulation of the IEEE 14-BUS System

The IEEE bus system is a common practice the academic and researh community uses to test any new models. The data are readily available to develop models and to perform analysis. We have used data from the system to develop our formulations. We have formulated an LP model of the standard IEEE 14-bus system using the AMPL package. The basic system is implemented and tested with bus failure and repair rates to study the impact of line voltages and the dynamic behavior of buses. The failure-rate and repair-rate data of the IEEE 14 bus is taken from [WG10, Wan01] as shown in Tables 7, 8, and 9. The objective for our formulation is to minimize the failure rate and repair rate subject to flow-balance constraints and capacity constraints. For simplicity, we have multiplied the failure rate and repair rate, defining the objective function variable as a "risk" or "loss." Hence, the goal is to minimize the risk of any energy loss for the IEEE bus system subjective to populated constraints.

A single-line diagram for the IEEE 14-bus standard system extracted from [Son99, Moo91] is shown in Figure 18. It consists of five generators with IEEE type-1 exciters, three of which are synchronous compensators that are only used for reactive power support. There are 11 loads in the system, totaling 258 MW and 81.3 Mvar. For our analysis, we have taken only the real power of the supply. The supply and demand in the IEEE 14-bus system equal 258 MW. Hence, it is a balanced system with equal values for the supply and demand units. Dynamic data for the generator exciters are selected from [WG10].

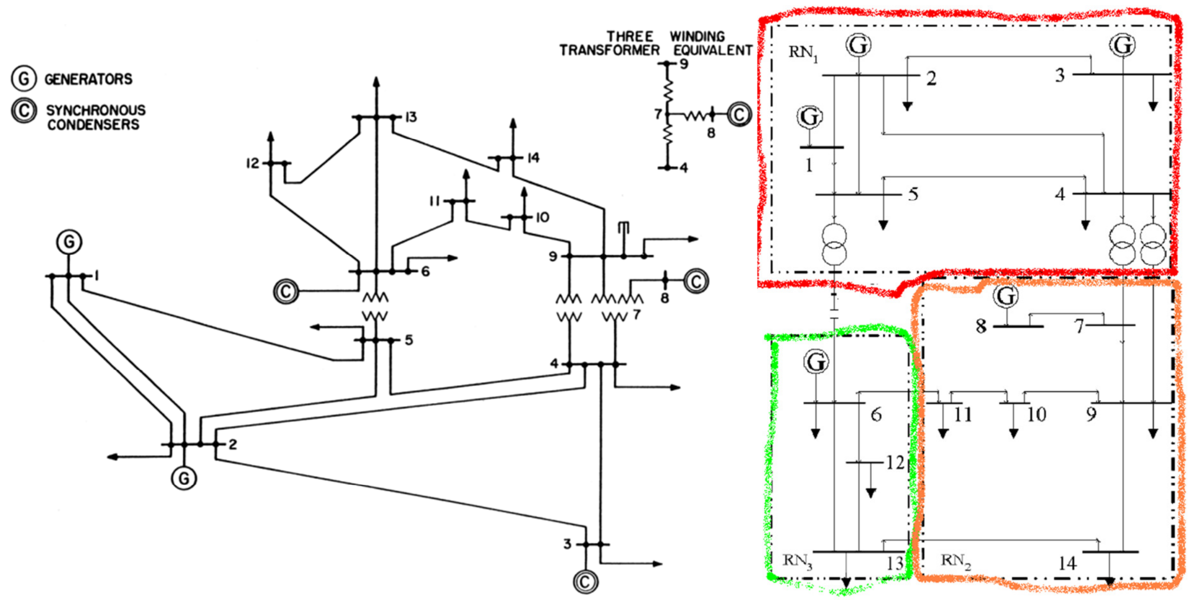


Figure 18. IEEE 14-bus system and three decomposed regions: RN1, RN2, and RN3.

The IEEE bus system shown in Figure 18 is decomposed into three regions, RN1, RN2, and RN3, as shown in Figure 19. The IEEE 14-bus network model contains 14 nodes and 18 transmission lines. A node is similar to a bus or point junction where two or multiple lines interconnect.

For example, lines 12 and 11 interconnect at node 6, and lines 14 and 12 connect at line 13. We have decomposed the system into three regions as used in a reliability study conducted in [WG10]. We may adapt random decomposition when choosing lines. In this system, we treat transmission lines 1, 2, 3, 4, and 5 as region 1; lines 7, 8, 9, 10, 11, and 14 are considered as region 2; and lines 6, 12, and 13 are region 3.

The number of generators varies in each region of the IEEE 14-bus system. For example, there are three generators in region 1 (G1, G2, G3), one generator in region 3 (G4), and one generator in region 2 (G5).

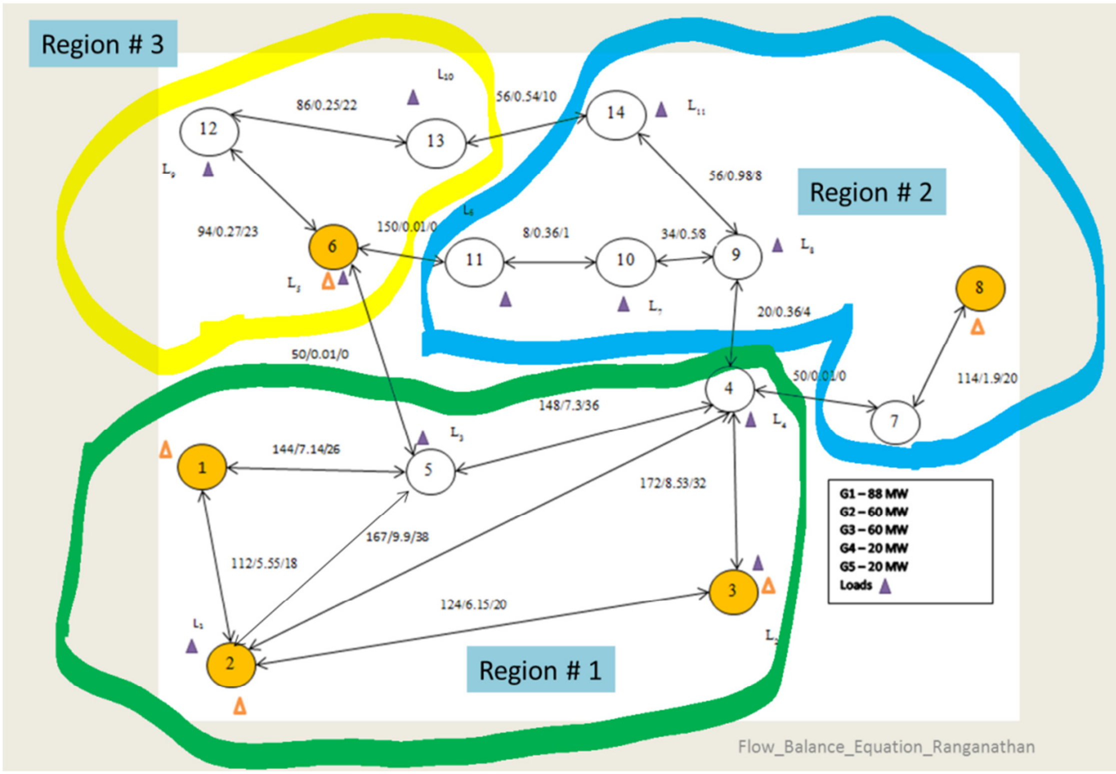


Figure 19. Network model of Regional decomposition.

In real electric networks, transformers are used to interconnect multiple regions. For our analysis, we introduce new nodes A1, A2, B1, B2, C1, and C2 that interconnect regions. This structure is shown in Figure 20. For example, nodes A2 and A1 interconnect regions 1 and 3, and nodes C1 and C2 interconnect regions 1 and 2. Similarly, nodes B1 and B2 interconnect regions 3 and 1. We include these nodes and apply flow-balance constraints in our formulation.

In Figure 20, the demands units (nodes) are represented as yellow circles, and the generators are not colored. The individual regions that interconnect to key nodes are shown separately in Figures 21, 22, and 23, respectively. For example, only nodes A1, A2, C1, and C2 from other regions are involved in our analysis for the region 1 study. These nodes represent the total supply and demand allocation from their respective regions, hence it is not necessary to include all nodes.

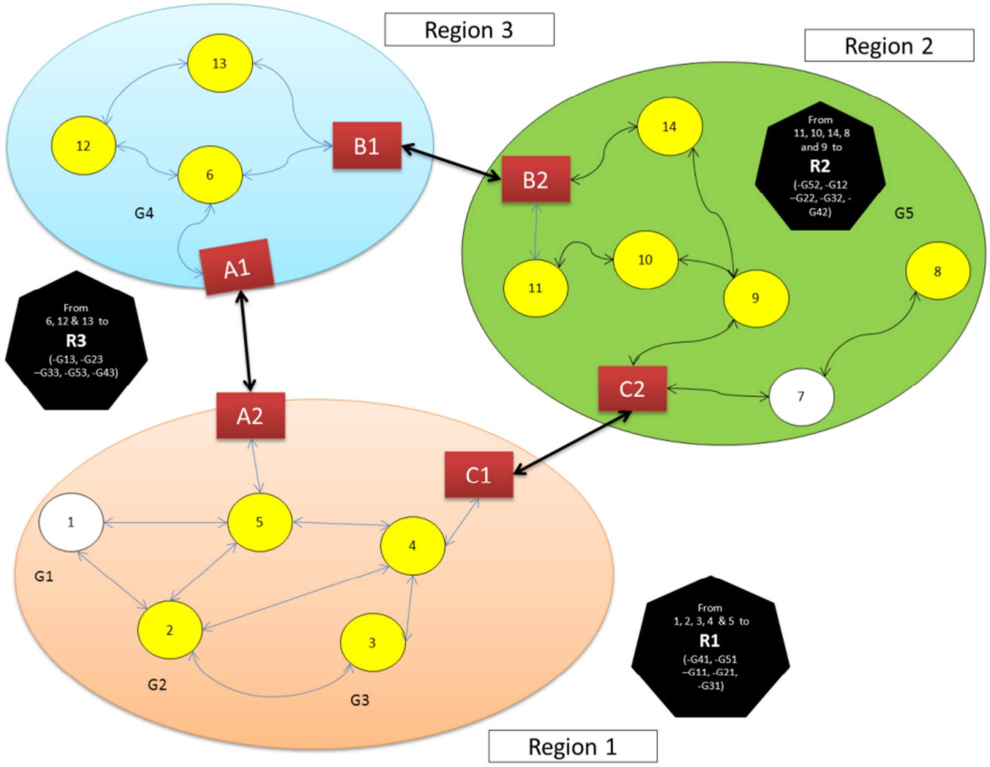


Figure 20. IEEE 14-bus network model with local R constraints.

As seen in Figure 20, we introduce new artificial nodes R1, R2, and R3 in the IEEE 14-bus system to keep excess power from reaching other regions and nodes. The presence of this node is due to the fact that we assume any loads can take power from the five generators in all three regions. By adding these new R nodes, we make sure that network follows the restriction on the available supply and demand values as per IEEE 14-bus data. For example, joint capacities at node 4 should not exceed the demand of G41. G41 has an initial allocation of 16.5 MW, thus the contribution of generator 4 to region 1 should not exceed 16.5 MW. A negative sign in the R1 node for G41 indicates that it is a demand. Figure 21 show the R1 node constraints for region 1.

We outline our basic formulation for Direct LP and Dantzig Wolfe in Table 6. The term "Direct LP" refers to allocation of the IEEE 14-bus into regions without any decomposition. The Dantzig-Wolfe formulation has variables "d" and "f" indicating the sub problems or regional constraints. In Direct LP, we combine all constraints in Ax=b notation, whereas with DW, only the master constraints are constructed. We will discuss the details in the implementation section of Chapter 6.

Table 6. LP formulation of Dantzig Wolfe and direct approach

Direct LP formulation	**Dantzig Wolfe decomposition formulation**
$min\ Z = Cx$	$min\ Z = C\lambda x$
$s.t.\ \ Ax \leq b;$ $\binom{Sub\ problems\ 1, 2 \dots n\ and}{Master\ constraints}$	$s.t.\ \ A\lambda x \leq \mathrm{b};$ $(Master\ constraint)$ $s.t.\ \ dx \leq \mathrm{f};$ $(Sub\ problem\ 1, 2, 3..n)$
$s.t.\ x \geq 0;$	$s.t. \sum_k \lambda_k = 1;$ (convexity constraint) $s.t.\ x \geq 0;$
where x is the decision variable.	where x is the decision variable.

Region 1 Constraints

The nodes that participate in region 1 are nodes 1, 2, 3, 4, and 5. The individual regional decomposition for regions 1, 2, and 3 is detailed in Figures 21, 23, and 22, respectively. The objective function for region 1 is the product of the failure rate and repair rate, which we define as the power loss or risk factor. The goal is to minimize the risk or loss for region 1 subject to the flow-balance constraints and the non-negativity additional constraints. The objective function for region 1 and the actual values for the repair and failure rates are given in Table 7.

Objective for Region 1 (Z_{LOSS})

$$Z_{LOSS-R1} = 100x_{1-2-1} + 186x_{1-5-1} + 123x_{2-3-1} + 376x_{2-4-1} + 140x_{2-5-1} + 273x_{3-4-1} + 262x_{4-5-1} + 100x_{1-2-2} + 186x_{1-5-2} + 123x_{2-3-2} + 376x_{2-4-2} + 140x_{2-5-2} + 273x_{3-4-2} + 262x_{4-5-2} + 100x_{1-2-3} + 186x_{1-5-3} + 123x_{2-3-3} + 376x_{2-4-3} + 140x_{2-5-3} + 273x_{3-4-3} + 262x_{4-5-3} + 100x_{1-2-4} + 186x_{1-5-4} + 123x_{2-3-4} + 376x_{2-4-4} + 140x_{2-5-4} + 273x_{3-4-4} + 262x_{4-5-4} + 100x_{1-2-5} + 186x_{1-5-5} + 123x_{2-3-5} + 376x_{2-4-5} + 140x_{2-5-5} + 273x_{3-4-5} + 262x_{4-5-5};$$

Table 7. Failure and Repair rates for Region 1

Lines connecting	**Failure rate (λ)**	**Repair rate (r)**
1-2	5.5552	18
1-5	7.1424	26
2-3	6.1504	20
2-4	9.9200	38
2-5	6.3488	22
3-4	8.5312	32
4-5	7.3408	36

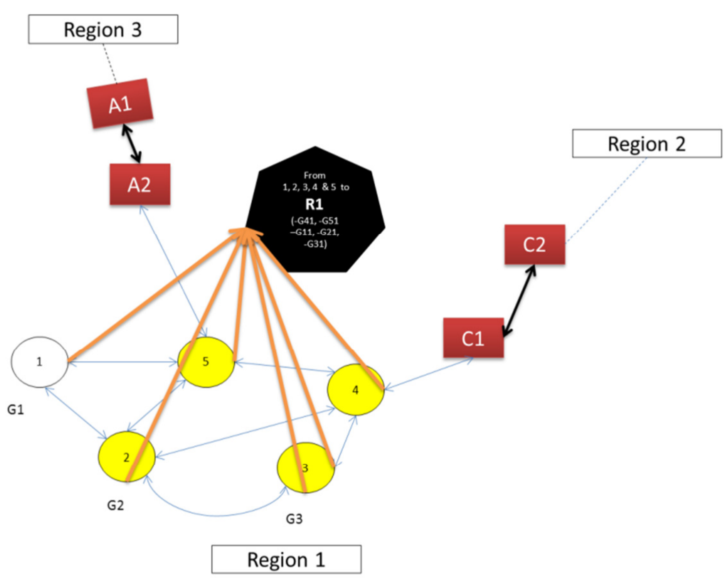


Figure 21. R1 node constraint for nodes in Region 1.

The flow balance constraints in region 1 are formulated at each node as follows:

Node 1

Here, there is no demand in node 1, so we assign zero on the right-hand side.

$$(x_{1-2-1} + x_{1-5-1}) - (x_{2-1-1} + x_{5-1-1}) + x_{1-R1-1} = 0; \tag{20}$$

$$(x_{1-2-2} + x_{1-5-2}) - (x_{2-1-2} + x_{5-1-2}) + x_{1-R1-2} = 0; \tag{21}$$

$$(x_{1-2-3} + x_{1-5-3}) - (x_{2-1-3} + x_{5-1-3}) + x_{1-R1-3} = 0; \tag{22}$$

$$(x_{1-2-4} + x_{1-5-4}) - (x_{2-1-4} + x_{5-1-4}) + x_{1-R1-4} = 0; \tag{23}$$

$$(x_{1-2-5} + x_{1-5-5}) - (x_{2-1-5} + x_{5-1-5}) + x_{1-R1-5} = 0; \tag{24}$$

Node 2

The flow balance constraints at node 2 in region 1 are formulated at each node as follows:

$$\begin{aligned}&(x_{2-1-1} + x_{2-5-1} + x_{2-4-1} + x_{2-3-1}) \\ &\qquad - (x_{1-2-1} + x_{5-2-1} + x_{4-2-1} + x_{3-2-1}) + x_{2-R1-1} \\ &\qquad \leq G_{11} - 21.7;\end{aligned} \tag{25}$$

$$\begin{aligned}&(x_{2-1-2} + x_{2-5-2} + x_{2-4-2} + x_{2-3-2}) \\ &\qquad - (x_{1-2-2} + x_{5-2-2} + x_{4-2-2} + x_{3-2-2}) + x_{2-R1-2} \\ &\qquad \leq G_{21} - 21.7;\end{aligned} \tag{26}$$

$$\begin{aligned}&(x_{2-1-3} + x_{2-5-3} + x_{2-4-3} + x_{2-3-3}) \\ &\qquad - (x_{1-2-3} + x_{5-2-3} + x_{4-2-3} + x_{3-2-3}) + x_{2-R1-3} \\ &\qquad \leq G_{31} - 21.7;\end{aligned} \tag{27}$$

$$\begin{aligned}&(x_{2-1-4} + x_{2-5-4} + x_{2-4-4} + x_{2-3-4}) \\ &\qquad - (x_{1-2-4} + x_{5-2-4} + x_{4-2-4} + x_{3-2-4}) + x_{2-R1-4} \\ &\qquad \leq G_{41} - 21.7;\end{aligned} \tag{28}$$

$$(x_{2-1-5} + x_{2-5-5} + x_{2-4-5} + x_{2-3-5}) - (x_{1-2-5} + x_{5-2-5} + x_{4-2-5} + x_{3-2-5}) + x_{2-R1-5} \leq G_{51} - 21.7; \tag{29}$$

Node 3

The flow balance constraints at node 3 in region 1 are formulated at each node as follows:

$$(x_{3-2-1} + x_{3-4-1}) - (x_{2-3-1} + x_{4-3-1}) + x_{3-R1-1} \leq G_{11} - 94.2; \tag{30}$$

$$(x_{3-2-2} + x_{3-4-2}) - (x_{2-3-2} + x_{4-3-2}) + x_{3-R1-2} \leq G_{21} - 94.2; \tag{31}$$

$$(x_{3-2-3} + x_{3-4-3}) - (x_{2-3-3} + x_{4-3-3}) + x_{3-R1-3} \leq G_{31} - 94.2; \tag{32}$$

$$(x_{3-2-4} + x_{3-4-4}) - (x_{2-3-4} + x_{4-3-4}) + x_{3-R1-4} \leq G_{41} - 94.2; \tag{33}$$

$$(x_{3-2-5} + x_{3-4-5}) - (x_{2-3-5} + x_{4-3-5}) + x_{3-R1-5} \leq G_{51} - 94.2; \tag{34}$$

Node 4

The flow balance constraints at node 4 in region 1 are formulated at each node as follows:

$$(x_{4-5-1} + x_{4-2-1} + x_{4-3-1}) - (x_{5-4-1} + x_{2-4-1} + x_{3-4-1}) + x_{4-R1-1} \leq G_{11} - 47; \tag{35}$$

$$(x_{4-5-2} + x_{4-2-2} + x_{4-3-2}) - (x_{5-4-2} + x_{2-4-2} + x_{3-4-2}) + x_{4-R1-2} \leq G_{21} - 47; \tag{36}$$

$$(x_{4-5-3} + x_{4-2-3} + x_{4-3-3}) - (x_{5-4-3} + x_{2-4-3} + x_{3-4-3}) + x_{4-R1-3} \leq G_{31} - 47; \tag{37}$$

$$(x_{4-5-4} + x_{4-2-4} + x_{4-3-4}) - (x_{5-4-4} + x_{2-4-4} + x_{3-4-4}) + x_{4-R1-4} \leq G_{41} - 47; \tag{38}$$

$$(x_{4-5-5} + x_{4-2-5} + x_{4-3-5}) - (x_{5-4-5} + x_{2-4-5} + x_{3-4-5}) + x_{4-R1-5} \leq G_{51} - 47; \tag{39}$$

Node 5

The flow balance constraints at node 5 in region 1 are formulated at each node as follows:

$$(x_{5-1-1} + x_{5-2-1} + x_{5-4-1} + x_{5,a2,1}) - \left(x_{1-5-1} + x_{2-5-1} + x_{4-5-1} + x_{a2,5,4}\right) + x_{5-R1-1} \leq G_{11} - 7.6; \tag{40}$$

$$(x_{5-1-2} + x_{5-2-2} + x_{5-4-2} + x_{5,a2,2}) - \left(x_{1-5-2} + x_{2-5-2} + x_{4-5-2} + x_{a2,5,4}\right) + x_{5-R1-2} \leq G_{21} - 7.6; \tag{41}$$

$$(x_{5-1-3} + x_{5-2-3} + x_{5-4-3} + x_{5,a2,3}) - \left(x_{1-5-3} + x_{2-5-3} + x_{4-5-3} + x_{a2,5,4}\right) + x_{5-R1-3} \leq G_{31} - 7.6; \tag{42}$$

$$(x_{5-1-4} + x_{5-2-4} + x_{5-4-4} + x_{5,a2,4}) - \left(x_{1-5-4} + x_{2-5-4} + x_{4-5-4} + x_{a2,5,4}\right) + x_{5-R1-4} \leq G_{41} - 7.6; \tag{43}$$

$$(x_{5-1-5} + x_{5-2-5} + x_{5-4-5} + x_{5,a2,5}) - \left(x_{1-5-5} + x_{2-5-5} + x_{4-5-5} + x_{a2,5,4}\right) + x_{5-R1-5} \leq G_{51} - 7.6; \tag{44}$$

Joint Capacity Constraints for Region 3

The joint capacity constraints in region 1 are formulated at each node as follows:

$$x_{1-R1-1} + x_{2-R1-1} + x_{3-R1-1} + x_{4-R1-1} + x_{5-R1-1} \leq -G_{11}; \tag{45}$$

$$x_{1-R1-2} + x_{2-R1-2} + x_{3-R1-2} + x_{4-R1-2} + x_{5-R1-2} \leq -G_{21}; \tag{46}$$

$$x_{1-R1-3} + x_{2-R1-3} + x_{3-R1-3} + x_{4-R1-3} + x_{5-R1-3} \leq -G_{31}; \tag{47}$$

$$x_{1-R1-4} + x_{2-R1-4} + x_{3-R1-4} + x_{4-R1-4} + x_{5-R1-4} \leq -G_{41}; \tag{48}$$

$$x_{1-R1-5} + x_{2-R1-5} + x_{3-R1-5} + x_{4-R1-5} + x_{5-R1-5} \leq -G_{51}; \tag{49}$$

<u>Other Constraints</u>

The other constraints in region 1 are formulated at each node as follows:

At A1,

$$\begin{aligned} &x_{6,a1,4} - (x_{a1,a2,4}) = 0; \\ &x_{a2,a1,1} - (x_{a1,6,1}) = 0; \\ &x_{a2,a1,2} - (x_{a1,6,2}) = 0; \end{aligned} \tag{50-52}$$

At A2,

$$\begin{aligned} &(x_{a2,a1,1}) = 0; \\ &x_{5,a2,2} - x_{a2,a1,2} = 0\,; \\ &x_{5,a2,3} - x_{a2,a1,3} = 0 \\ &x_{a1,a2,4} - (x_{a2,5,4}) = 0; \end{aligned} \tag{53-55}$$

$$\begin{aligned} &x_{5,a2,3} = G_{33}; \\ &x_{5,a2,2} = G_{23}; \\ &x_{5,a2,1} = G_{13}; \\ &x_{5,a2,4} = G_{43}; \\ &x_{5,a2,5} = G_{53}; \\ &x_{a2,5,4} = G_{41}; \end{aligned} \tag{56-61}$$

[END OF SUB PROBLEM 1]

Region 3 Constraints (nodes 6, 12 and 13)

The nodes that participate in region 3 are nodes 6, 12, and 13. The individual regional decomposition for region 3 is detailed in Figure 22. The objective function for region 3 is the product of the failure rate and the repair rate, which we define as the power loss or risk factor. The goal is to minimize the risk, or loss, for region 3 subject to flow-balance constraints and non-negativity additional constraints. The following equation is the objective function for region 3, and the actual values for the repair rate and failure rate are given in Table 7.

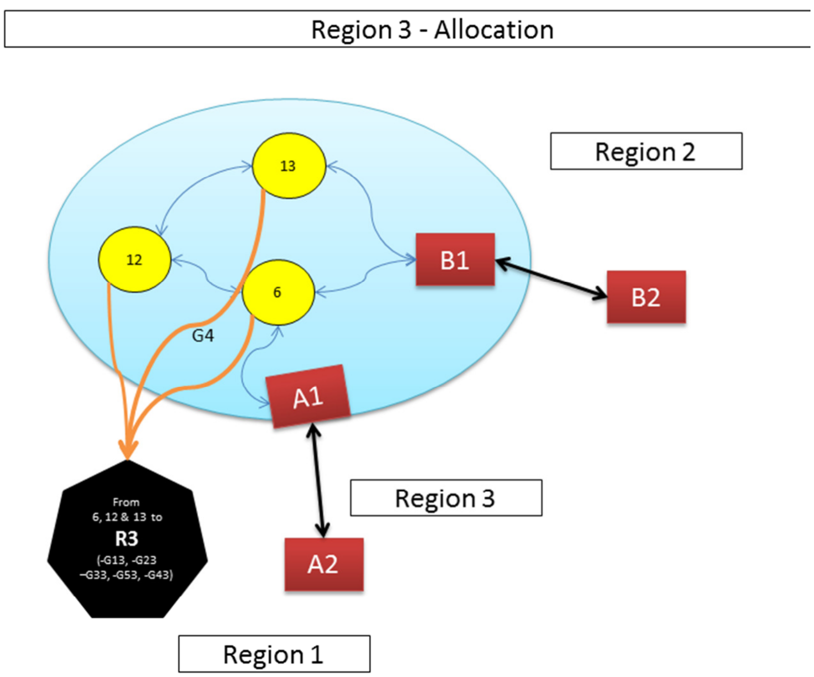


Figure 22. R3 node constraints for nodes in Region 3.

<u>Objective for Region 3 ($\boldsymbol{Z_{LOSS}}$)</u>

The objective function for Region 3 can be written as

$$Z_{LOSS-R3} = 6.3x_{6-12-1} + 5.5\,x_{12-13-1} + 6.3x_{6-12-2} + 5.5\,x_{12-13-2} + 6.3x_{6-12-3} + 5.5\,x_{12-13-3} + 6.3x_{6-12-4} + 5.5\,x_{12-13-4} + 6.3x_{6-12-5} + 5.5\,x_{12-13-5};$$

The data to calculate the coefficients for objective function are taken from Table 8.

Table 8. Failure and Repair rates for Region 3

Lines connecting	**Failure rate (λ)**	**Repair rate (r)**
6-12	0.274	23
6-13	0.44	0.44
12-13	0.25	22

The flow-balance constraints in region 3 are formulated at each node is given as:

Node 12

$$x_{12-13-1} + x_{12-6-1} - (x_{13-12-1} + x_{6-12-1}) + x_{12-R3-1}) \leq G_{13} - 6; \quad (62)$$

$$x_{12-13-2} + x_{12-6-2} - (x_{13-12-2} + x_{6-12-2}) + x_{12-R3-2}) \leq G_{23} - 6; \quad (63)$$

$$x_{12-13-3} + x_{12-6-3} - (x_{13-12-3} + x_{6-12-3}) + x_{12-R3-3}) \leq G_{33} - 6; \quad (64)$$

$$x_{12-13-4} + x_{12-6-4} - (x_{13-12-4} + x_{6-12-4}) + x_{12-R3-4}) \leq G_{43} - 6; \quad (65)$$

$$x_{12-13-5} + x_{12-6-5} - (x_{13-12-5} + x_{6-12-5}) + x_{12-R3-5}) \leq G_{53} - 6; \quad (66)$$

Node 13

$$x_{13-12-1} + x_{13,b1,4} - (x_{12-13-1}) + x_{13-R3-1} \leq G_{13} - 13.5; \quad (67)$$

$$x_{13-12-2} + x_{13,b1,4} - (x_{12-13-2}) + x_{13-R3-2} \leq G_{23} - 13.5; \quad (68)$$

$$x_{13-12-3} + x_{13,b1,4} - (x_{12-13-3}) + x_{13-R3-3} \leq G_{33} - 13.5; \quad (69)$$

$$x_{13-12-4} + x_{13,b1,4} - (x_{12-13-4}) + x_{13-R3-4} \leq G_{43} - 13.5; \quad (70)$$

$$x_{13-12-5} + x_{13,b1,4} - (x_{12-13-5}) + x_{13-R3-5} \leq G_{53} - 13.5; \quad (71)$$

Node 6

$$(x_{6-12-1}) - (x_{12-6-1}) + x_{6-R3-1} \leq G_{13} - 11.2; \quad (72)$$

$$(x_{6-12-2}) - (x_{12-6-2}) + x_{6-R3-2} \leq G_{23} - 11.2; \quad (73)$$

$$(x_{6-12-3}) - (x_{12-6-3}) + x_{6-R3-3} \leq G_{33} - 11.2; \quad (74)$$

$$(x_{6-12-4}) - (x_{12-6-4}) + x_{6-R3-4} \leq G_{43} - 11.2; \tag{75}$$

$$(x_{6-12-5}) - (x_{12-6-5}) + x_{6-R3-5} \leq G_{53} - 11.2; \tag{76}$$

<u>Joint Capacity Constraints for Region 3</u>

$$x_{12-R3-1} + x_{13-R3-1} + x_{6-R3-1} \leq -G_{13}; \tag{77}$$

$$x_{12-R3-2} + x_{13-R3-2} + x_{6-R3-2} \leq -G_{23}; \tag{78}$$

$$x_{12-R3-3} + x_{13-R3-3} + x_{6-R3-3} \leq -G_{33}; \tag{79}$$

$$x_{12-R3-4} + x_{13-R3-4} + x_{6-R3-4} \leq -G_{43}; \tag{80}$$

$$x_{12-R3-5} + x_{13-R3-5} + x_{6-R3-5} \leq -G_{53}; \tag{81}$$

<u>Other Constraints</u>

$$x_{6,b1,4} - \left(x_{b1,6,4}\right) = 0;$$

(82-89)

$$x_{6,a1,4} - x_{a1,6,4} = 0;$$

A1:

$$x_{6,a1,4} - \left(x_{a1,a2,4}\right) = 0;$$

$$x_{a2,a1,1} - \left(x_{a1,6,1}\right) = 0;$$

$$x_{a2,a1,2} - \left(x_{a1,6,2}\right) = 0;$$

A2:

$$x_{5,a2,1} - (x_{a2,a1,1}) = 0; x_{5,a2,3} - x_{a2,a1,3} = 0;$$

$$x_{a1,a2,4} - \left(x_{a2,5,4}\right) = 0;$$

$$x_{5,a2,2} - x_{a2,a1,2} = 0$$

B1:

$$x_{13,b1,4} + x_{6,b1,4} - (x_{b1,b2,4}) = 0;$$

$$x_{b2,b1,5} - (x_{b1,6,5} + x_{b1,13,5}) = 0;$$

B2:

$$x_{14-b2-5} + x_{11-b2-5} - (x_{b2,b1,5}) = 0; \tag{90-96}$$

$$x_{b1,b2,4} - (x_{b2-14-4} + x_{b2-11-4}) = 0;$$

$$x_{6,b1,4} + x_{13,b1,4} = G_{42};$$

$$x_{b1,b2,4} = G_{42};$$

$$x_{b2,14,4} + x_{b2,11,4} = G_{42};$$

$$x_{b2,b1,5} = x_{14-b2-5} + x_{11-b2-5};$$

[END OF SUB PROBLEM 2]

Region 2 Constraints

The nodes that participate in region 2 are nodes 7, 8, 9, 10, 11, and 14. The individual regional decomposition for region 2 is detailed in Figure 23. The objective function for region 2 is the product of the failure rate and the repair rate, which we define as the power loss or risk factor. The goal is to minimize the risk, or loss, for region 2 subject to flow-balance constraints and non-negativity additional constraints. The following equation is the objective function for region 2, and the actual values for the repair rate and failure rate are given in Table 9.

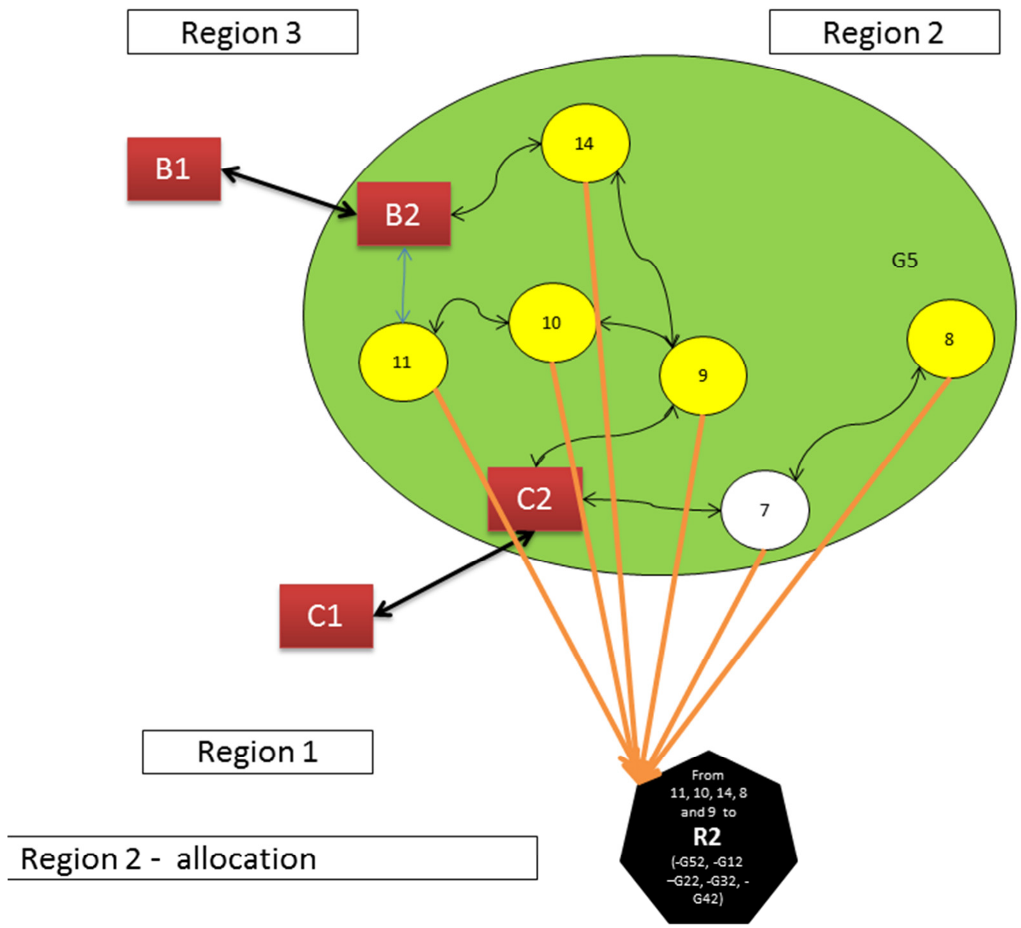


Figure 23. R2 node constraints for nodes in Region 2.

Objective for Region 2 ($\boldsymbol{Z_{LOSS}}$)

The objective function for region 2 is given as

$$Z_{LOSS-R2} = 38x_{7-8-1} + 4x_{9-10-1} + 7.84x_{9-14-1} + 2.28x_{10-11-1} + 38x_{7-8-2} + 4x_{9-10-2} + 7.84x_{9-14-2} + 2.28x_{10-11-2} + 38x_{7-8-3} + 4x_{9-10-3} + 7.84x_{9-14-3} + 2.28x_{10-11-3} + 38x_{7-8-4} + 4x_{9-10-4} + 7.84x_{9-14-4} + 2.28x_{10-11-4} + 38x_{7-8-5} + 4x_{9-10-5} + 7.84x_{9-14-5} + 2.28x_{10-11-5};$$

Table 9. Failure and Repair rates for Region 2

Lines	Failure rate (λ)	Repair rate (r)
7-8	1.9973	20
7-10	0.946	15
7-11	1.08	13
8-9	1.506	14
8-10	1.121	12
9-10	0.595	8
9-14	0.981	8
10-11	0.38	6
10-14	1.29	16
10-0	1.12	13
14-0	0.85	11

Node 7

There is no demand at node 7.

$$\left(x_{C2,7,1} + x_{C2,7,2} + x_{C2,7,3}\right) - (x_{7,C2,5}) + x_{7-R2} = 0; \tag{97}$$

$$x_{8-7-1} - (x_{7-8-1}) = 0; \tag{98}$$

$$x_{8-7-2} - (x_{7-8-2}) = 0; \tag{99}$$

$$x_{8-7-3} - (x_{7-8-3}) = 0; \tag{100}$$

$$x_{8-7-4} - (x_{7-8-4}) = 0; \tag{101}$$

$$x_{8-7-5} - (x_{7-8-5}) = 0; \tag{102}$$

Node 8

There is no demand at node 8.

$$x_{8-7-1} - x_{7-8-1} + x_{8-R2-1} = 0; \tag{103}$$

$$x_{8-7-2} - x_{7-8-2} + x_{8-R2-2} = 0; \tag{104}$$

$$x_{8-7-3} - x_{7-8-3} + x_{8-R2-3} = 0; \tag{105}$$

$$x_{8-7-4} - x_{7-8-4} + x_{8-R2-4} = 0; \tag{106}$$

$$x_{8-7-5} - x_{7-8-5} + x_{8-R2-5} = 0; \tag{107}$$

Node 9

$$(x_{10-9-1} + x_{14-9-1}) - (x_{9-10-1} + x_{9-14-1}) + x_{9,c2,1} + x_{9-R2-1} \leq G_{12} - 29.5; \tag{108}$$

$$(x_{10-9-2} + x_{14-9-2}) - (x_{9-10-2} + x_{9-14-2}) + x_{9,c2,2} + x_{9-R2-2} \leq G_{22} - 29.5; \tag{109}$$

$$(x_{10-9-3} + x_{14-9-3}) - (x_{9-10-3} + x_{9-14-3}) + x_{9,c2,3} + x_{9-R2-3} \leq G_{32} - 29.5; \tag{110}$$

$$(x_{10-9-4} + x_{14-9-4}) - (x_{9-10-4} + x_{9-14-4}) + x_{9,c2,4} + x_{9-R2-4} \leq G_{42} - 29.5; \tag{111}$$

$$(x_{10-9-5} + x_{14-9-5}) - (x_{9-10-5} + x_{9-14-5}) + x_{9,c2,5} + x_{9-R2-5} \leq G_{52} - 29.5; \tag{112}$$

Node 10

$$x_{10-11-1} + x_{10-9-1} - (x_{11-10-1} + x_{9-10-1}) + x_{10-R2-1} \leq G_{12} - 9; \tag{113}$$

$$x_{10-11-2} + x_{10-9-2} - (x_{11-10-2} + x_{9-10-2}) + x_{10-R2-2} \leq G_{22} - 9; \tag{114}$$

$$x_{10-11-3} + x_{10-9-3} - (x_{11-10-3} + x_{9-10-3}) + x_{10-R2-3} \leq G_{32} - 9; \tag{115}$$

$$x_{10-11-4} + x_{10-9-4} - (x_{11-10-4} + x_{9-10-4}) + x_{10-R2-4} \leq G_{42} - 9; \tag{116}$$

$$x_{10-11-5} + x_{10-9-5} - (x_{11-10-5} + x_{9-10-5}) + x_{10-R2-5} \leq G_{52} - 9; \tag{117}$$

Node 11

$$x_{11-10-1} + x_{11,b2,1} - \left(x_{10-11-1} + x_{b2,11,1}\right) + x_{11-R2-1} \leq G_{12} - 3.5; \tag{118}$$

$$x_{11-10-2} + x_{11,b2,2} - \left(x_{10-11-2} + x_{b2,11,2}\right) + x_{11-R2-2} \leq G_{22} - 3.5; \tag{119}$$

$$x_{11-10-3} + x_{11,b2,3} - \left(x_{10-11-3} + x_{b2,11,3}\right) + x_{11-R2-3} \leq G_{32} - 3.5; \tag{120}$$

$$x_{11-10-4} + x_{11,b2,4} - \left(x_{10-11-4} + x_{b2,11,4}\right) + x_{11-R2-4} \leq G_{42} - 3.5; \quad (121)$$

$$x_{11-10-5} + x_{11,b2,5} - \left(x_{10-11-5} + x_{b2,11,5}\right) + x_{11-R2-5} \leq G_{52} - 3.5; \quad (122)$$

Node 14

$$x_{14-9-1} + x_{14,b2,1} - \left(x_{b2,14,1} + x_{9-14-1}\right) + x_{14-R2-1} \leq G_{12} - 14.8; \quad (123)$$

$$x_{14-9-2} + x_{14,b2,2} - \left(x_{b2,14,2} + x_{9-14-2}\right) + x_{14-R2-2} \leq G_{22} - 14.8; \quad (124)$$

$$x_{14-9-3} + x_{14,b2,3} - \left(x_{b2,14,3} + x_{9-14-3}\right) + x_{14-R2-3} \leq G_{32} - 14.8; \quad (125)$$

$$x_{14-9-4} + x_{14,b2,4} - \left(x_{b2,14,4} + x_{9-14-4}\right) + x_{14-R2-4} \leq G_{42} - 14.8; \quad (126)$$

$$x_{14-9-5} + x_{14,b2,5} - \left(x_{b2,14,5} + x_{9-14-5}\right) + x_{14-R2-5} \leq G_{52} - 14.8; \quad (127)$$

Joint Capacity Constraints for Region 2

$$x_{14-R2-1} + x_{11-R2-1} + x_{10-R2-1} + x_{9-R2-1} + x_{7-R2-1} + x_{8-R2-1} \leq -G_{12}; \quad (128)$$

$$x_{14-R2-2} + x_{11-R2-2} + x_{10-R2-2} + x_{9-R2-2} + x_{7-R2-2} + x_{8-R2-2} \leq -G_{22}; \quad (129)$$

$$x_{14-R2-3} + x_{11-R2-3} + x_{10-R2-3} + x_{9-R2-3} + x_{7-R2-3} + x_{8-R2-3} \leq -G_{32}; \quad (130)$$

$$x_{14-R2-4} + x_{11-R2-4} + x_{10-R2-4} + x_{9-R2-4} + x_{7-R2-4} + x_{8-R2-4} \leq -G_{42}; \quad (131)$$

$$x_{14-R2-5} + x_{11-R2-5} + x_{10-R2-5} + x_{9-R2-5} + x_{7-R2-5} + x_{8-R2-5} \leq -G_{52}; \quad (132)$$

Other Constraints

B1:
$x_{13,b1,4} + x_{6,b1,4} - \left(x_{b1,b2,4}\right) = 0;\ x_{b2,b1,5} - \left(x_{b1,6,5} + x_{b1,13,5}\right) = 0;$

B2 (133-135)
$x_{14-b2-5} + x_{11-b2-5} - \left(x_{b2,b1,5}\right) = 0;$
$x_{b1,b2,4} - (x_{b2-14-4} + x_{b2-11-4}) = 0;$

C1: (136-139)
$x_{4,c1,1} - \left(x_{c1,c2,1}\right) = 0;$
$x_{4,c1,2} - \left(x_{c1,c2,2}\right) = 0;$
$x_{4,c1,3} - \left(x_{c1,c2,3}\right) = 0;$
$x_{c2,c1,5} - (x_{c1,4,1} + x_{c1,4,2} + x_{c1,4,3}) = 0;$

C2: (140-144)

$$x_{9,c2,4} + x_{7,c2,4} - \left(x_{c2,c1,4}\right) = 0;$$
$$x_{9,c2,5} + x_{7,c2,5} - (x_{c2,c1,5}) = 0;$$
$$x_{c1,c2,1} - (x_{c2,9,1} + x_{c2,7,1}) = 0;$$
$$x_{c1,c2,2} - (x_{c2,9,2} + x_{c2,7,2}) = 0;$$
$$x_{c1,c2,3} - (x_{c2,9,3} + x_{c2,7,3}) = 0;$$

$$x_{a2,a1,3,} = G_{33};$$
$$x_{a2,a1,2} = G_{23};$$
$$x_{a2,a1,1=} G_{13};$$
$$x_{a1,6,1} = G_{13};$$
$$x_{a1,6,2} = G_{23};$$
$$x_{a1,6,3} = G_{33};$$

(145-150)

[END OF SUB PROBLEM 3]

Master Constraints (Linking Constraints)

The master constraints are the linking constraints that connect to subproblems or regional constraints. The generator variables present in the master constraints are also included or related to the region 1, region 2, and region 3 constraints, as discussed above, during flow-balance and joint-capacity constraints. These constraints interact iteratively with the sub-constraints to reach an optimal solution via dual values and convexity constraints as proposed in the objective formulation. Constraints 153 to 158 serve as master constraints.

$$\mathrm{G}_{11} + \mathrm{G}_{12} + \mathrm{G}_{13} = 88 \quad (\mathrm{MC}\ \#\ 1)$$

$$\mathrm{G}_{21} + \mathrm{G}_{22} + \mathrm{G}_{23} = 60; \quad (\mathrm{MC}\ \#\ 2)$$

$$\mathrm{G}_{31} + \mathrm{G}_{32} + \mathrm{G}_{33} = 60 \quad (\mathrm{MC}\ \#\ 3)$$

$$\mathrm{G}_{41} + \mathrm{G}_{42} + \mathrm{G}_{43} = 25 \quad (\mathrm{MC}\ \#\ 4)$$

$$\mathrm{G}_{51} + \mathrm{G}_{52} + \mathrm{G}_{53} = 25 \quad (\mathrm{MC}\ \#\ 5)$$

(153-158)

This five set of constraints means Commodity 1 Constraints <=88; Commodity 2 Constraints <=60; Commodity 3 Constraints <=60; Commodity 4

Constraints <=25; Commodity 5 Constraints <=25;

Decomposing the IEEE 14-Bus System into Two Regions

The constraints that change when modifying three region classifications to two regions are given in the following sections.

R2 Node Constraint in Region 1

$$x_{12-R2-1} + x_{6-R2-1} + x_{13-R2-1} + x_{11-R2-1} + x_{10-R2-1} + x_{14-R2-1} + x_{9-R2-1} + x_{8-R2-1} + x_{7-R2-1} \leq -G_{12}; \tag{159}$$

$$x_{12-R2-2} + x_{6-R2-2} + x_{13-R2-2} + x_{11-R2-2} + x_{10-R2-2} + x_{14-R2-2} + x_{9-R2-2} + x_{8-R2-2} + x_{7-R2-2} \leq -G_{22}; \tag{160}$$

$$x_{12-R2-3} + x_{6-R2-3} + x_{13-R2-3} + x_{11-R2-3} + x_{10-R2-3} + x_{14-R2-3} + x_{9-R2-3} + x_{8-R2-3} + x_{7-R2-3} \leq -G_{32}; \tag{161}$$

$$x_{12-R2-4} + x_{6-R2-4} + x_{13-R2-4} + x_{11-R2-4} + x_{10-R2-4} + x_{14-R2-4} + x_{9-R2-4} + x_{8-R2-4} + x_{7-R2-4} \leq -G_{42}; \tag{162}$$

$$x_{12-R2-5} + x_{6-R2-5} + x_{13-R2-5} + x_{11-R2-5} + x_{10-R2-5} + x_{14-R2-5} + x_{9-R2-5} + x_{8-R2-5} + x_{7-R2-5} \leq -G_{52}; \tag{163}$$

R1 Node Constraint in Region 1

$$x_{1-R1-1} + x_{2-R1-1} + x_{3-R1-1} + x_{4-R1-1} + x_{5-R1-1} \leq -G_{11}; \tag{164}$$

$$x_{1-R1-2} + x_{2-R1-2} + x_{3-R1-2} + x_{4-R1-2} + x_{5-R1-2} \leq -G_{21}; \tag{165}$$

$$x_{1-R1-3} + x_{2-R1-3} + x_{3-R1-3} + x_{4-R1-3} + x_{5-R1-3} \leq -G_{31}; \tag{166}$$

$$x_{1-R1-4} + x_{2-R1-4} + x_{3-R1-4} + x_{4-R1-4} + x_{5-R1-4} \leq -G_{41}; \tag{167}$$

$$x_{1-R1-5} + x_{2-R1-5} + x_{3-R1-5} + x_{4-R1-5} + x_{5-R1-5} \leq -G_{51}; \tag{168}$$

The nodal constraints remain the same except for removal of the R3 variable. As seen in Table 10, certain nodes are not considered. The coefficients are assigned a zero if the variable is not involved in the decomposition process. Nodes B1 and B2are not considered.

Table 10. Eliminated nodes for two region decomposition

132	143	295	148	255
133	144	298	149	256
134	145	301	252	257
135	146	304	253	258
136	147	307	254	259
260	261	271	272	273
274				
G13=0	G23=0	G33=0	G43=0	G53=0

Formulating the IEEE 30-Bus System's Constraints

The Dantzig-Wolfe implementation is also tested with the next level of the IEEE bus system. The IEEE 30-bus system has 6 generators and 20 loads as shown in Figure 24. This system is much larger compared to IEEE 14-bus system that was discussed previously. The generator and load data are given in Tables 11 and Table 12, respectively.

Table 11. Generator data for the IEEE 30-bus system [WG10]

GENERATOR DATA OF THE IEEE 30-BUS TEST SYSTEM

Unit	Bus	Cost coefficients			Pmax (MW)	Pmin (MW)	Min up time (h)	Min down time (h)	Ramp up (MW)	Ramp Down (MW)	Startup Ramp (MW)	Shutdown ramp (MW)
		A ($/MWh^2)	B ($/MWh)	C ($)								
G1	1	0.0200	15.00	0	80	15	2	2	25	25	70	60
G2	2	0.0175	14.75	0	80	15	2	2	25	25	70	60
G3	13	0.0250	16.00	0	50	10	3	3	15	15	70	60
G4	22	0.0625	14.00	0	50	10	4	4	15	15	70	60
G5	23	0.0250	16.00	0	30	5	3	3	10	10	70	60
G6	27	0.0083	15.25	0	55	10	4	4	15	15	70	60

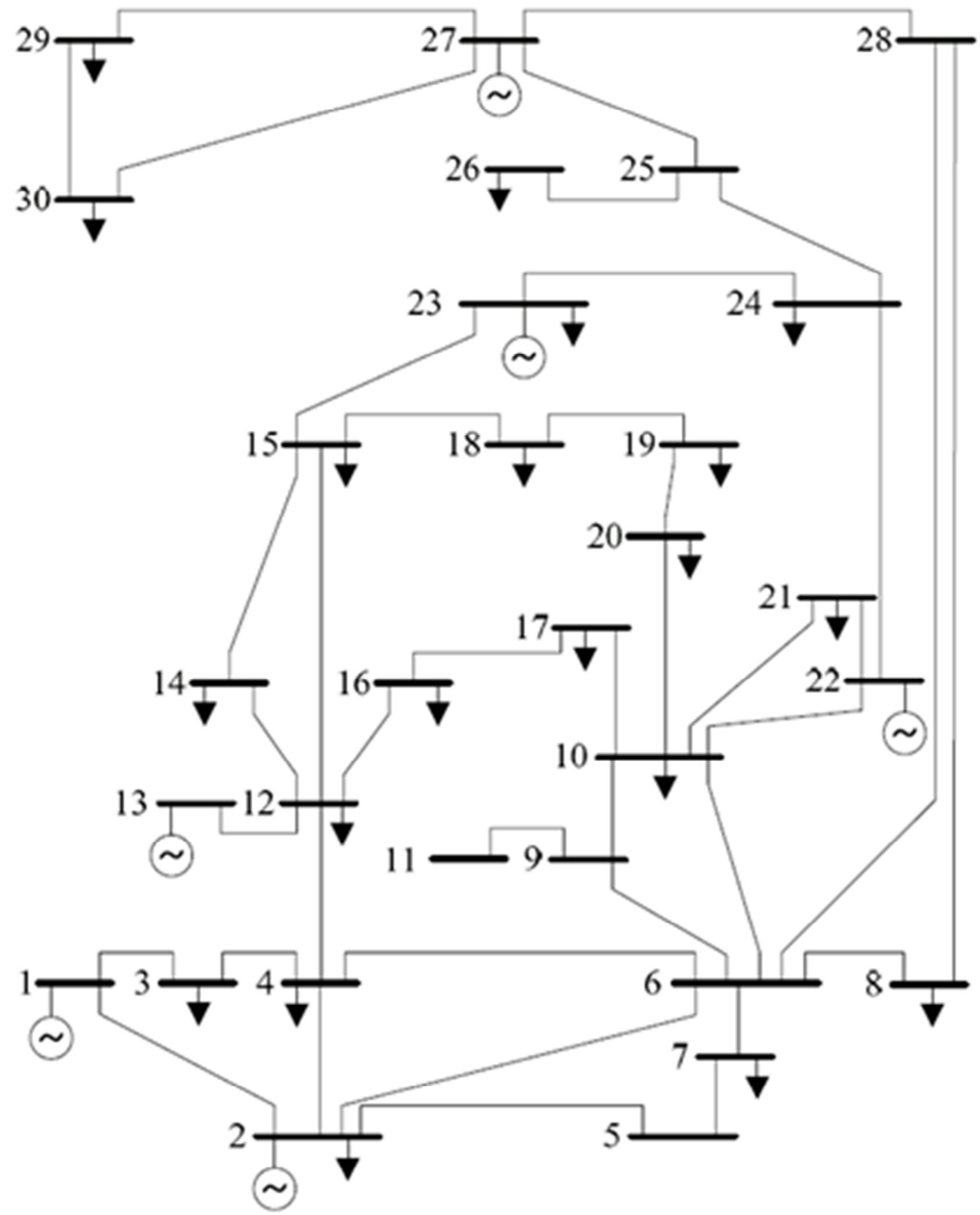


Figure 24. IEEE 30-bus system single-line diagram.

Table 12. Demand profile for the IEEE 30-bus system

Nodes	Load demand	Nodes	Load demand
1	0	16	3.5
2	21.7	17	9.0
3	2.4	18	3.2
4	67.6	19	9.5
5	34.2	20	2.2
6	0	21	17.5
7	22.8	22	0
8	30	23	3.2
9	0	24	8.7
10	5.8	25	0
11	0	26	3.5
12	11.2	27	0
13	0	28	0
14	6.2	29	2.4
15	8.2	30	10.6

The risk, or loss factor is calculated for objective-function coefficients using the failure-rate and repair-rate data shown in Table 13.

Table 13. Repair and Failure rates for the IEEE 30-bus system

NETWORK DATA OF THE IEEE 30-BUS TEST SYSTEM

Line No.	From	To	X (p.u.)	Flow limit (MW)	Failure rate	Repair rate
1	1	2	0.06	130	0.9783	0.0217
2	1	3	0.19	130	0.9841	0.0159
3	2	4	0.17	65	0.9532	0.0468
4	3	4	0.04	130	0.9172	0.0828
5	2	5	0.20	130	0.9786	0.0214
6	2	6	0.18	65	0.9497	0.0503
7	4	6	0.04	90	0.9828	0.0172
8	5	7	0.12	70	0.9760	0.0240
9	6	7	0.08	130	0.9211	0.0789
10	6	8	0.04	32	0.9494	0.0506
11	6	9	0.21	65	0.9494	0.0506
12	6	10	0.56	32	0.9211	0.0789
13	9	11	0.21	65	0.9535	0.0465
14	9	10	0.11	65	0.9509	0.0491
15	4	12	0.26	65	0.9660	0.0340
16	12	13	0.14	65	0.9838	0.0162
17	12	14	0.26	32	0.9754	0.0246
18	12	15	0.13	32	0.9598	0.0402
19	12	16	0.20	32	0.9510	0.0490
20	14	15	0.20	16	0.9494	0.0506
21	16	17	0.19	16	0.9494	0.0506
22	15	18	0.22	16	0.9236	0.0764
23	18	19	0.13	16	0.9514	0.0486
24	19	20	0.07	32	0.9509	0.0491
25	10	20	0.21	32	0.9666	0.0334
26	10	17	0.08	32	0.9824	0.0176
27	10	21	0.07	32	0.9786	0.0214
28	10	22	0.15	32	0.9612	0.0388
29	21	22	0.02	32	0.9462	0.0538
30	15	23	0.20	16	0.9498	0.0502
31	22	24	0.18	16	0.9506	0.0494
32	23	24	0.27	16	0.9181	0.0819
33	24	25	0.33	16	0.9483	0.0517
34	25	26	0.38	16	0.9537	0.0463
35	25	27	0.21	16	0.9733	0.0267
36	28	27	0.40	65	0.9818	0.0182
37	27	29	0.42	16	0.9808	0.0192
38	27	30	0.60	16	0.9564	0.0436
39	29	30	0.45	16	0.9537	0.0463
40	8	28	0.20	32	0.9537	0.0463
41	6	28	0.06	32	0.9536	0.0464

The IEEE network model for the 30-bus system is shown in Figure 25, and the decomposed regions are shown in Figure 26.

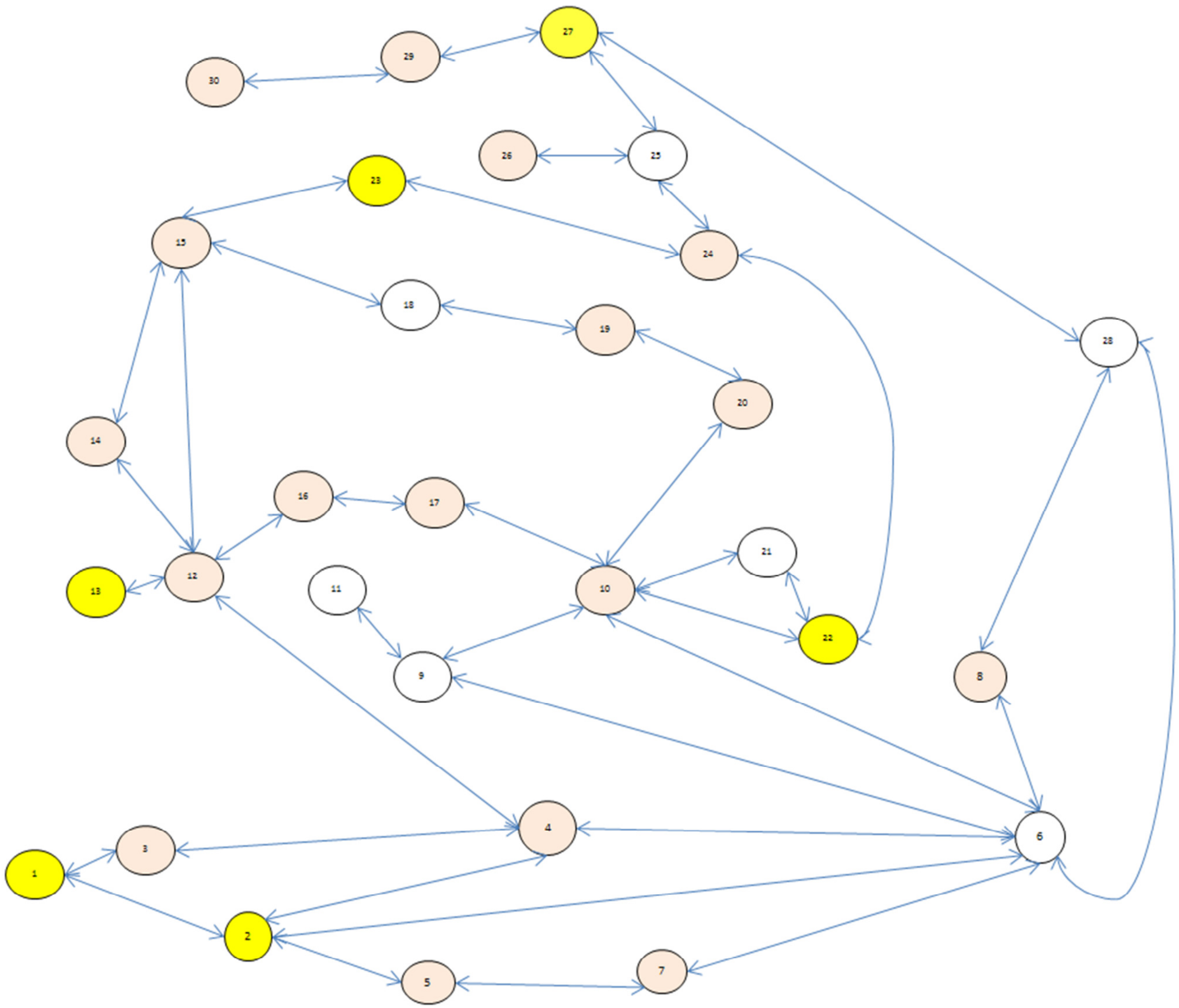


Figure 25. Network model for the IEEE 30-bus system

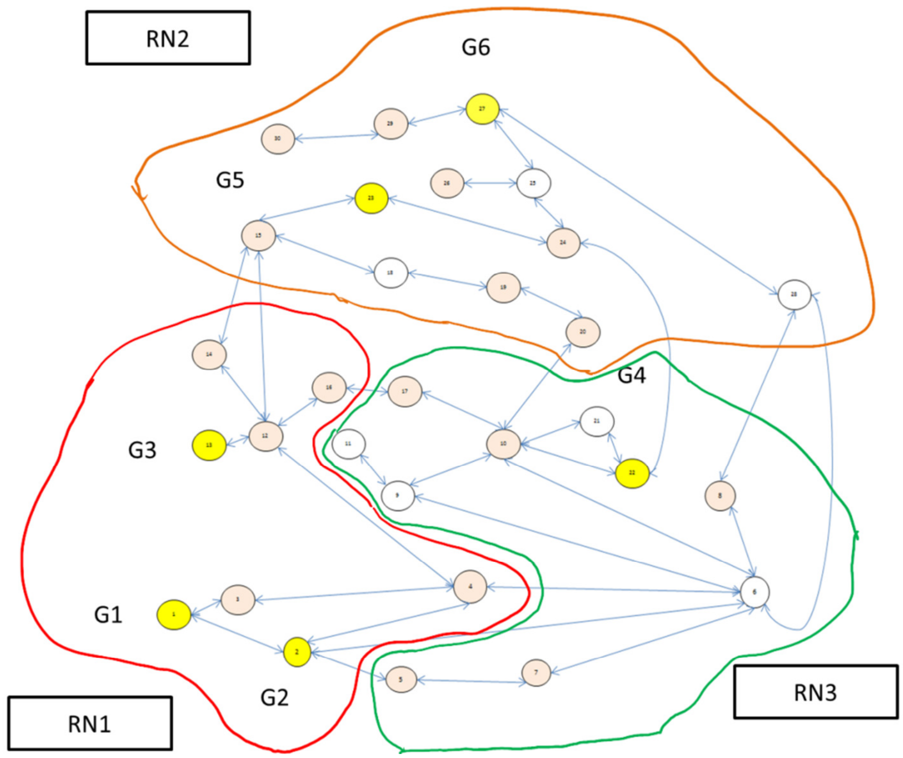


Figure 26. Network model for decomposing the IEEE 30-bus system to three regions.

Nodal Constraints for Region 1

<u>Node 1</u>

$$x_{1-2-1} + x_{1-3-1} - (x_{2-1-1} + x_{3-1-1}) + x_{1-R1-1}) \leq G_{11}; \tag{169}$$

$$x_{1-2-2} + x_{1-3-2} - (x_{2-1-2} + x_{3-1-2}) + x_{1-R1-2}) \leq G_{21}; \tag{170}$$

$$x_{1-2-3} + x_{1-3-3} - (x_{2-1-3} + x_{3-1-3}) + x_{1-R1-3}) \leq G_{31}; \tag{171}$$

$$x_{1-2-4} + x_{1-3-4} - (x_{2-1-4} + x_{3-1-4}) + x_{1-R1-4}) \leq G_{41}; \tag{172}$$

$$x_{1-2-5} + x_{1-3-5} - (x_{2-1-5} + x_{3-1-5}) + x_{1-R1-5}) \leq G_{51}; \tag{173}$$

$$x_{1-2-6} + x_{1-3-6} - (x_{2-1-6} + x_{3-1-6}) + x_{1-R1-6}) \leq G_{61}; \tag{174}$$

Node 2

$$x_{2-4-1} + x_{2-5-1} + x_{2-6-1} + x_{2-1-1} - (x_{4-2-1} + x_{5-2-1} + x_{6-2-1} + x_{1-2-1}) + (x_{2-R1-1}) \leq G_{21} - 21.7; \tag{175}$$

$$x_{2-4-2} + x_{2-5-2} + x_{2-6-2} + x_{2-1-2} - (x_{4-2-2} + x_{5-2-2} + x_{6-2-2} + x_{1-2-2}) + (x_{2-R1-2}) \leq G_{22} - 21.7; \tag{176}$$

$$x_{2-4-3} + x_{2-5-3} + x_{2-6-3} + x_{2-1-3} - (x_{4-2-3} + x_{5-2-3} + x_{6-2-3} + x_{1-2-3}) + (x_{2-R1-3}) \leq G_{23} - 21.7; \tag{177}$$

$$x_{2-4-4} + x_{2-5-4} + x_{2-6-4} + x_{2-1-4} - (x_{4-2-4} + x_{5-2-4} + x_{6-2-4} + x_{1-2-4}) + (x_{2-R1-4}) \leq G_{24} - 21.7; \tag{178}$$

$$x_{2-4-5} + x_{2-5-5} + x_{2-6-5} + x_{2-1-5} - (x_{4-2-5} + x_{5-2-5} + x_{6-2-5} + x_{1-2-5}) + (x_{2-R1-5}) \leq G_{25} - 21.7; \tag{179}$$

$$x_{2-4-6} + x_{2-5-6} + x_{2-6-6} + x_{2-1-6} - (x_{4-2-6} + x_{5-2-6} + x_{6-2-6} + x_{1-2-6}) + (x_{2-R1-6}) \leq G_{26} - 21.7; \tag{180}$$

Node 3

$$x_{1-3-1} + x_{3-4-1} - (x_{3-1-1} + x_{4-3-1}) + x_{3-R1-1}) \leq G_{11} - 2.4; \tag{181}$$

$$x_{1-3-2} + x_{3-4-2} - (x_{3-1-2} + x_{4-3-2}) + x_{3-R1-2}) \leq G_{21} - 2.4; \tag{182}$$

$$x_{1-3-3} + x_{3-4-3} - (x_{3-1-3} + x_{4-3-3}) + x_{3-R1-3}) \leq G_{31} - 2.4; \tag{183}$$

$$x_{1-3-4} + x_{3-4-4} - (x_{3-1-4} + x_{4-3-4}) + x_{3-R1-4}) \leq G_{41} - 2.4; \tag{184}$$

$$x_{1-3-5} + x_{3-4-5} - (x_{3-1-5} + x_{4-3-5}) + x_{3-R1-5}) \leq G_{51} - 2.4; \tag{185}$$

$$x_{1-3-6} + x_{3-4-6} - (x_{3-1-6} + x_{4-3-6}) + x_{3-R1-6}) \leq G_{61} - 2.4; \tag{186}$$

<u>Node 4</u>

$$\begin{aligned} &x_{4-6-1} + x_{4-3-1} + x_{4-2-1} + x_{4-12-1} \\ &\quad - (x_{6-4-1} + x_{3-4-1} + x_{2-4-1} + x_{12-4-1}) + x_{4-R1-1}) \\ &\quad \leq G_{11} - 67.6; \end{aligned} \tag{187}$$

$$\begin{aligned} &x_{4-6-2} + x_{4-3-2} + x_{4-2-2} + x_{4-12-2} \\ &\quad - (x_{6-4-2} + x_{3-4-2} + x_{2-4-2} + x_{12-4-2}) + x_{4-R1-2}) \\ &\quad \leq G_{21} - 67.6; \end{aligned} \tag{188}$$

$$\begin{aligned} &x_{4-6-3} + x_{4-3-3} + x_{4-2-3} + x_{4-12-3} \\ &\quad - (x_{6-4-3} + x_{3-4-3} + x_{2-4-3} + x_{12-4-3}) + x_{4-R1-3}) \\ &\quad \leq G_{31} - 67.6; \end{aligned} \tag{189}$$

$$\begin{aligned} &x_{4-6-4} + x_{4-3-4} + x_{4-2-4} + x_{4-42-4} \\ &\quad - (x_{6-4-4} + x_{3-4-4} + x_{2-4-4} + x_{42-4-4}) + x_{4-R1-4}) \\ &\quad \leq G_{41} - 67.6; \end{aligned} \tag{190}$$

$$\begin{aligned} &x_{4-6-5} + x_{4-3-5} + x_{4-2-5} + x_{4-12-5} \\ &\quad - (x_{6-4-5} + x_{3-4-5} + x_{2-4-5} + x_{12-4-5}) + x_{4-R1-5}) \\ &\quad \leq G_{51} - 67.6; \end{aligned} \tag{191}$$

$$x_{4-6-6} + x_{4-3-6} + x_{4-2-6} + x_{4-12-6} - (x_{6-4-6} + x_{3-4-6} + x_{2-4-6} + x_{12-4-6}) + x_{4-R1-6}) \leq G_{61} - 67.6; \tag{192}$$

Node 12

$$x_{12-13-1} + x_{12-4-1} + x_{12-16-1} + x_{12-14-1} + x_{12-15-1} - (x_{13-12-1} + x_{4-12-1} + x_{16-12-1} + x_{14-12-1} + x_{15-12-1}) + x_{12-R1-1}) \leq G_{11} - 11.2; \tag{193}$$

$$x_{12-13-2} + x_{12-4-2} + x_{12-16-2} + x_{12-14-2} + x_{12-15-2} - (x_{13-12-2} + x_{4-12-2} + x_{16-12-2} + x_{14-12-2} + x_{15-12-2}) + x_{12-R1-2}) \leq G_{21} - 11.2; \tag{194}$$

$$x_{12-13-3} + x_{12-4-3} + x_{12-16-3} + x_{12-14-3} + x_{12-15-3} - (x_{13-12-3} + x_{4-12-3} + x_{16-12-3} + x_{14-12-3} + x_{15-12-3}) + x_{12-R1-3}) \leq G_{31} - 11.2; \tag{195}$$

$$x_{12-13-4} + x_{12-4-4} + x_{12-16-4} + x_{12-14-4} + x_{12-15-4} - (x_{13-12-4} + x_{4-12-4} + x_{16-12-4} + x_{14-12-4} + x_{15-12-4}) + x_{12-R1-4}) \leq G_{41} - 11.2;; \tag{196}$$

$$x_{12-13-5} + x_{12-4-5} + x_{12-16-5} + x_{12-14-5} + x_{12-15-5} - (x_{13-12-5} + x_{4-12-5} + x_{16-12-5} + x_{14-12-5} + x_{15-12-5}) + x_{12-R1-5}) \leq G_{51} - 11.2; \tag{197}$$

$$x_{12-13-6} + x_{12-4-6} + x_{12-16-6} + x_{12-14-6} + x_{12-15-6} - (x_{13-12-6} + x_{4-12-6} + x_{16-12-6} + x_{14-12-6} + x_{15-12-6}) + x_{12-R1-6}) \leq G_{61} - 11.2; \tag{198}$$

Node 13

$$x_{13-12-1} - (x_{12-13-1}) + (x_{13-R1-1}) \leq 0; \tag{199}$$

$$x_{13-12-2} - (x_{12-13-2}) + (x_{13-R1-2}) \leq 0; \tag{200}$$

$$x_{13-12-3} - (x_{12-13-3}) + (x_{13-R1-3}) \leq 0; \tag{201}$$

$$x_{13-12-4} - (x_{12-13-4}) + (x_{13-R1-4}) \leq 0; \tag{202}$$

$$x_{13-12-5} - (x_{12-13-5}) + (x_{13-R1-5}) \leq 0; \tag{203}$$

$$x_{13-12-6} - (x_{12-13-6}) + (x_{13-R1-6}) \leq 0; \tag{204}$$

Node 14

$$x_{14-15-1} + x_{14-12-1} - (x_{12-14-1} + x_{15-14-1}) + (x_{14-R1-1}) \leq 0; \tag{205}$$

$$x_{14-15-2} + x_{14-12-2} - (x_{12-14-2} + x_{15-14-2}) + (x_{14-R1-2}) \leq 0; \tag{206}$$

$$x_{14-15-3} + x_{14-12-3} - (x_{12-14-3} + x_{15-14-3}) + (x_{14-R1-3}) \leq 0; \tag{207}$$

$$x_{14-15-4} + x_{14-12-4} - (x_{12-14-1} + x_{15-14-4}) + (x_{14-R1-4}) \leq 0; \tag{208}$$

$$x_{14-15-5} + x_{14-12-5} - (x_{12-14-5} + x_{15-14-5}) + (x_{14-R1-5}) \leq 0; \tag{209}$$

$$x_{14-15-6} + x_{14-12-6} - (x_{12-14-6} + x_{15-14-6}) + (x_{14-R1-6}) \leq 0; \tag{210}$$

Node 16

$$x_{16-12-1} + x_{16-17-1} - (x_{12-16-1} + x_{17-16-1}) + (x_{16-R1-1}) \leq G_{11} - 3.5; \tag{211}$$

$$x_{16-12-2} + x_{16-17-2} - (x_{12-16-2} + x_{17-16-2}) + (x_{16-R1-2}) \leq G_{21} - 3.5; \tag{212}$$

$$x_{16-12-3} + x_{16-17-3} - (x_{12-16-3} + x_{17-16-3}) + (x_{16-R1-3}) \leq G_{31} - 3.5; \tag{213}$$

$$x_{16-12-4} + x_{16-17-4} - (x_{12-16-4} + x_{17-16-4}) + (x_{16-R1-4}) \leq G_{41} - 3.5; \tag{214}$$

$$x_{16-12-5} + x_{16-17-5} - (x_{12-16-5} + x_{17-16-5}) + (x_{16-R1-5}) \leq G_{51} - 3.5; \tag{215}$$

$$x_{16-12-6} + x_{16-17-6} - (x_{12-16-6} + x_{17-16-6}) + (x_{16-R1-6}) \leq G_{61} - 3.5; \tag{216}$$

Artificial Nodes in Region 1

The nodes participate in region 1 are 1, 2, 3, 4, 5, 12, 13, 14, and 16. Each of these nodes is connected to R1 nodes.

$$x_{1-R1-1} + x_{2-R1-1} + x_{3-R1-1} + x_{4-R1-1} + x_{5-R1-1} + x_{12-R1-1} + x_{13-R1-1} + x_{14-R1-1} + x_{16-R1-1} \leq -G_{11}; \tag{217}$$

$$x_{1-R1-2} + x_{2-R1-2} + x_{3-R1-2} + x_{4-R1-2} + x_{5-R1-2} + x_{12-R1-2} + x_{13-R1-2} + x_{14-R1-2} + x_{16-R1-2} \leq -G_{21}; \tag{218}$$

$$x_{1-R1-3} + x_{2-R1-3} + x_{3-R1-3} + x_{4-R1-3} + x_{5-R1-3} + x_{12-R1-3} + x_{13-R1-3} + x_{14-R1-3} + x_{16-R1-3} \leq -G_{31}; \tag{219}$$

$$x_{1-R1-4} + x_{2-R1-4} + x_{3-R1-4} + x_{4-R1-4} + x_{5-R1-4} + x_{12-R1-4} + x_{13-R1-4} + x_{14-R1-4} + x_{16-R1-4} \leq -G_{41}; \tag{220}$$

$$x_{1-R1-5} + x_{2-R1-5} + x_{3-R1-5} + x_{4-R1-5} + x_{5-R1-5} + x_{12-R1-5} + x_{13-R1-5} + x_{14-R1-5} + x_{16-R1-5} \leq -G_{51}; \tag{221}$$

$$x_{1-R1-6} + x_{2-R1-6} + x_{3-R1-6} + x_{4-R1-6} + x_{5-R1-6} + x_{12-R1-6} + x_{13-R1-6} + x_{14-R1-6} + x_{16-R1-6} \leq -G_{61}; \tag{222}$$

Nodal Constraints for Region 2

Node 30

$$x_{30-29-1} - (x_{29-30-1}) + (x_{30-R2-1}) \leq G_{12} - 10.6; \tag{223}$$

$$x_{30-29-2} - (x_{29-30-2}) + (x_{30-R2-2}) \leq G_{22} - 10.6; \tag{224}$$

$$x_{30-29-3} - (x_{29-30-3}) + (x_{30-R2-3}) \leq G_{32} - 10.6; \tag{225}$$

$$x_{30-29-4} - (x_{29-30-4}) + (x_{30-R2-4}) \leq G_{42} - 10.6; \tag{226}$$

$$x_{30-29-5} - (x_{29-30-5}) + (x_{30-R2-5}) \leq G_{52} - 10.6; \tag{227}$$

$$x_{30-29-6} - (x_{29-30-6}) + (x_{30-R2-6}) \leq G_{62} - 10.6; \tag{228}$$

<u>Node 29</u>

$$x_{29-30-1} + x_{29-27-1} - (x_{30-29-1} + x_{27-29-1}) + (x_{29-R2-1}) \leq G_{12} - 2.4; \tag{229}$$

$$x_{29-30-2} + x_{29-27-2} - (x_{30-29-2} + x_{27-29-2}) + (x_{29-R2-2}) \leq G_{22} - 2.4; \tag{230}$$

$$x_{29-30-3} + x_{29-27-3} - (x_{30-29-3} + x_{27-29-3}) + (x_{29-R2-3}) \leq G_{32} - 2.4; \tag{231}$$

$$x_{29-30-4} + x_{29-27-4} - (x_{30-29-4} + x_{27-29-4}) + (x_{29-R2-4}) \leq G_{42} - 2.4; \tag{232}$$

$$x_{29-30-5} + x_{29-27-5} - (x_{30-29-5} + x_{27-29-5}) + (x_{29-R2-5}) \leq G_{52} - 2.4; \tag{233}$$

$$x_{29-30-6} + x_{29-27-6} - (x_{30-29-6} + x_{27-29-6}) + (x_{29-R2-6}) \leq G_{62} - 2.4; \tag{234}$$

<u>Node 27</u>

$$x_{27-29-1} + x_{27-25-1} - (x_{29-27-1} + x_{25-27-1}) + (x_{27-R2-1}) \leq G_{12}; \tag{235}$$

$$x_{27-29-2} + x_{27-25-2} - (x_{29-27-2} + x_{25-27-2}) + (x_{27-R2-2}) \leq G_{22}; \tag{236}$$

$$x_{27-29-3} + x_{27-25-3} - (x_{29-27-3} + x_{25-27-3}) + (x_{27-R2-3}) \leq G_{32}; \tag{237}$$

$$x_{27-29-4} + x_{27-25-4} - (x_{29-27-4} + x_{25-27-4}) + (x_{27-R2-4}) \leq G_{42}; \tag{238}$$

$$x_{27-29-5} + x_{27-25-5} - (x_{29-27-5} + x_{25-27-5}) + (x_{27-R2-5}) \leq G_{52}; \tag{239}$$

$$x_{27-29-6} + x_{27-25-6} - (x_{29-27-6} + x_{25-27-6}) + (x_{27-R2-6}) \leq G_{62}; \tag{240}$$

<u>Node 15</u>

$$\begin{aligned} &x_{15-14-1} + x_{15-12-1} + x_{15-18-1} + x_{15-23-1} \\ &\qquad - (x_{14-15-1} + x_{12-15-1} + x_{18-15-1} + x_{23-15-1}) \\ &\qquad + (x_{15-R2-1}) \leq G_{12} - 8.2; \end{aligned} \tag{241}$$

$$\begin{aligned}&x_{15-14-2}+x_{15-12-2}+x_{15-18-2}+x_{15-23-2}\\&\qquad -(x_{14-15-2}+x_{12-15-2}+x_{18-15-2}+x_{23-15-2})\\&\qquad +(x_{15-R2-2})\leq G_{22}-8.2;\end{aligned}\tag{242}$$

$$\begin{aligned}&x_{15-14-3}+x_{15-12-3}+x_{15-18-3}+x_{15-23-3}\\&\qquad -(x_{14-15-3}+x_{12-15-3}+x_{18-15-3}+x_{23-15-3})\\&\qquad +(x_{15-R2-3})\leq G_{32}-8.2;\end{aligned}\tag{243}$$

$$\begin{aligned}&x_{15-14-4}+x_{15-12-4}+x_{15-18-4}+x_{15-23-4}\\&\qquad -(x_{14-15-4}+x_{12-15-4}+x_{18-15-4}+x_{23-15-4})\\&\qquad +(x_{15-R2-4})\leq G_{42}-8.2;\end{aligned}\tag{244}$$

$$\begin{aligned}&x_{15-14-5}+x_{15-12-5}+x_{15-18-5}+x_{15-23-5}\\&\qquad -(x_{14-15-5}+x_{12-15-5}+x_{18-15-5}+x_{23-15-5})\\&\qquad +(x_{15-R2-5})\leq G_{52}-8.2;\end{aligned}\tag{245}$$

$$\begin{aligned}&x_{15-14-6}+x_{15-12-6}+x_{15-18-6}+x_{15-23-6}\\&\qquad -(x_{14-15-6}+x_{12-15-6}+x_{18-15-6}+x_{23-15-6})\\&\qquad +(x_{15-R2-6})\leq G_{62}-8.2;\end{aligned}\tag{246}$$

Similarly, flow-balance constraints for nodes 23, 26, 18, 19, 20, 24, 25, and 28 can be formulated. Due to space constraints, we have not included them here.

$$\begin{aligned}&x_{28-27-1}+x_{28-8-1}+x_{28-6-1}-(x_{27-28-1}+x_{8-28-1}+x_{6-28-1})\\&\qquad +(x_{28-R2-1})\leq G_{12};\end{aligned}\tag{247}$$

$$\begin{aligned}&x_{28-27-2}+x_{28-8-2}+x_{28-6-2}-(x_{27-28-2}+x_{8-28-2}+x_{6-28-2})\\&\qquad +(x_{28-R2-2})\leq G_{22};\end{aligned}\tag{248}$$

$$x_{28-27-3} + x_{28-8-3} + x_{28-6-3} - (x_{27-28-3} + x_{8-28-3} + x_{6-28-3}) + (x_{28-R2-3}) \leq G_{32}; \tag{249}$$

$$x_{28-27-4} + x_{28-8-4} + x_{28-6-4} - (x_{27-28-4} + x_{8-28-4} + x_{6-28-4}) + (x_{28-R2-4}) \leq G_{42}; \tag{250}$$

$$x_{28-27-5} + x_{28-8-5} + x_{28-6-5} - (x_{27-28-5} + x_{8-28-5} + x_{6-28-5}) + (x_{28-R2-5}) \leq G_{52}; \tag{251}$$

$$x_{28-27-6} + x_{28-8-6} + x_{28-6-6} - (x_{27-28-6} + x_{8-28-6} + x_{6-28-6}) + (x_{28-R2-6}) \leq G_{62}; \tag{252}$$

Joint Capacity Constraints for Nodes in region 2

$$x_{30-R2-1} + x_{29-R2-1} + x_{27-R2-1} + x_{15-R2-1} + x_{23-R2-1} + x_{26-R2-1} + x_{18-R2-1} + x_{19-R2-1} + x_{20-R2-1} + x_{24-R2-1} + x_{25-R2-1} + x_{28-R2-1} \leq -G_{12}; \tag{253}$$

$$x_{30-R2-1} + x_{29-R2-1} + x_{27-R2-1} + x_{15-R2-1} + x_{23-R2-1} + x_{26-R2-1} + x_{18-R2-1} + x_{19-R2-1} + x_{20-R2-1} + x_{24-R2-1} + x_{25-R2-1} + x_{28-R2-1} \leq -G_{22}; \tag{254}$$

$$x_{30-R2-1} + x_{29-R2-1} + x_{27-R2-1} + x_{15-R2-1} + x_{23-R2-1} + x_{26-R2-1} + x_{18-R2-1} + x_{19-R2-1} + x_{20-R2-1} + x_{24-R2-1} + x_{25-R2-1} + x_{28-R2-1} \leq -G_{32}; \tag{255}$$

$$x_{30-R2-1} + x_{29-R2-1} + x_{27-R2-1} + x_{15-R2-1} + x_{23-R2-1} + x_{26-R2-1} + x_{18-R2-1} + x_{19-R2-1} + x_{20-R2-1} + x_{24-R2-1} + x_{25-R2-1} + x_{28-R2-1} \leq -G_{42}; \tag{256}$$

$$x_{30-R2-1} + x_{29-R2-1} + x_{27-R2-1} + x_{15-R2-1} + x_{23-R2-1} + x_{26-R2-1} + x_{18-R2-1} + x_{19-R2-1} + x_{20-R2-1} + x_{24-R2-1} + x_{25-R2-1} + x_{28-R2-1} \leq -G_{52}; \quad (257)$$

$$x_{30-R2-1} + x_{29-R2-1} + x_{27-R2-1} + x_{15-R2-1} + x_{23-R2-1} + x_{26-R2-1} + x_{18-R2-1} + x_{19-R2-1} + x_{20-R2-1} + x_{24-R2-1} + x_{25-R2-1} + x_{28-R2-1} \leq -G_{62}; \quad (258)$$

Nodal Constraints for Region 3

The nodes participate in region 3 are nodes 5, 7, 6, 8, 22, 21, 10, 17, 11, and 9.

<u>Node 5</u>

$$x_{5-7-1} + x_{5-2-1} - (x_{7-5-1} + x_{2-5-1}) + (x_{5-R3-1}) \leq G_{13} - 34.2; \quad (259)$$

$$x_{5-7-2} + x_{5-2-2} - (x_{7-5-2} + x_{2-5-2}) + (x_{5-R3-2}) \leq G_{23} - 34.2; \quad (260)$$

$$x_{5-7-3} + x_{5-2-3} - (x_{7-5-3} + x_{2-5-3}) + (x_{5-R3-3}) \leq G_{33} - 34.2; \quad (261)$$

$$x_{5-7-4} + x_{5-2-4} - (x_{7-5-4} + x_{2-5-4}) + (x_{5-R3-4}) \leq G_{43} - 34.2; \quad (262)$$

$$x_{5-7-5} + x_{5-2-5} - (x_{7-5-5} + x_{2-5-5}) + (x_{5-R3-5}) \leq G_{53} - 34.2; \quad (263)$$

$$x_{5-7-6} + x_{5-2-6} - (x_{7-5-6} + x_{2-5-6}) + (x_{5-R3-6}) \leq G_{63} - 34.2; \quad (264)$$

Similarly, constraints for nodes 7, 6, 8, 22, 21, 10, 17, 11, and 9 can be formulated.

<u>Node 9</u>

$$x_{9-11-1} + x_{9-10-1} + x_{9-6-1} - (x_{11-9-1} + x_{10-9-1} + x_{6-9-1}) + (x_{9-R1-1}) \leq G_{13}; \quad (265)$$

$$x_{9-11-2} + x_{9-10-2} + x_{9-6-2} - (x_{11-9-2} + x_{10-9-2} + x_{6-9-2}) + (x_{9-R1-2}) \leq G_{23}; \quad (266)$$

$$x_{9-11-3} + x_{9-10-3} + x_{9-6-3} - (x_{11-9-3} + x_{10-9-3} + x_{6-9-3}) + (x_{9-R1-3}) \leq G_{33}; \tag{267}$$

$$x_{9-11-4} + x_{9-10-4} + x_{9-6-4} - (x_{11-9-4} + x_{10-9-4} + x_{6-9-4}) + (x_{9-R1-4}) \leq G_{43}; \tag{268}$$

$$x_{9-11-5} + x_{9-10-5} + x_{9-6-5} - (x_{11-9-5} + x_{10-9-5} + x_{6-9-5}) + (x_{9-R1-5}) \leq G_{53}; \tag{269}$$

$$x_{9-11-6} + x_{9-10-6} + x_{9-6-6} - (x_{11-9-6} + x_{10-9-6} + x_{6-9-6}) + (x_{9-R1-6}) \leq G_{63}; \tag{270}$$

Joint Capacity Constraints for Region 3

$$x_{5-R3-1} + x_{7-R3-1} + x_{6-R3-1} + x_{8-R3-1} + x_{22-R3-1} + x_{21-R3-1} + x_{10-R3-1} + x_{17-R3-1} + x_{11-R3-1} + x_{9-R3-1} \leq -G_{13}; \tag{271}$$

$$x_{5-R3-2} + x_{7-R3-2} + x_{6-R3-2} + x_{8-R3-2} + x_{22-R3-2} + x_{21-R3-2} + x_{10-R3-2} + x_{17-R3-2} + x_{11-R3-2} + x_{9-R3-2} \leq -G_{23}; \tag{272}$$

$$x_{5-R3-3} + x_{7-R3-3} + x_{6-R3-3} + x_{8-R3-3} + x_{22-R3-3} + x_{21-R3-3} + x_{10-R3-3} + x_{17-R3-3} + x_{11-R3-3} + x_{9-R3-3} \leq -G_{33}; \tag{273}$$

$$x_{5-R3-4} + x_{7-R3-4} + x_{6-R3-4} + x_{8-R3-4} + x_{22-R3-4} + x_{21-R3-4} + x_{10-R3-4} + x_{17-R3-4} + x_{11-R3-4} + x_{9-R3-4} \leq -G_{43}; \tag{274}$$

$$x_{5-R3-5} + x_{7-R3-5} + x_{6-R3-5} + x_{8-R3-5} + x_{22-R3-5} + x_{21-R3-5} + x_{10-R3-5} + x_{17-R3-5} + x_{11-R3-5} + x_{9-R3-5} \leq -G_{53}; \tag{275}$$

$$x_{5-R3-6} + x_{7-R3-6} + x_{6-R3-6} + x_{8-R3-6} + x_{22-R3-6} + x_{21-R3-6} + x_{10-R3-6} + x_{17-R3-6} + x_{11-R3-6} + x_{9-R3-6} \leq -G_{63}; \tag{276}$$

The additional constraints that interconnect regions

$x_{a1,14,1} - (x_{14,c1,1}) = G_{12};$

$x_{a1,14,2} - (x_{14,c1,2}) = G_{22};$

$x_{a1,14,3} - (x_{14,c1,3}) = G_{32};$

$x_{a1,14,4} - (x_{14,c1,1}) = G_{42};$

$x_{a1,14,5} - (x_{14,c1,5}) = G_{52};$

$x_{a1,14,6} - (x_{14,c1,6}) = G_{62};$

$x_{a2,5,1} - (x_{5,b1,1}) = G_{13};$

$x_{a2,5,2} - (x_{5,b1,2}) = G_{23};$

$x_{a2,5,3} - (x_{5,b1,3}) = G_{33};$ (277-296)

$x_{a2,5,4} - (x_{5,b1,4}) = G_{43};$

$x_{a2,5,5} - (x_{5,b1,5}) = G_{53};$

$x_{a2,5,6} - (x_{5,b1,6}) = G_{63};$

$x_{a2,a1,1} = G_{11};$

$x_{a2,a1,2} = G_{21};$

$x_{a2,a1,3} = G_{31};$

$x_{a2,a1,4} = G_{41};$

$x_{a2,a1,5} = G_{51};$

$x_{a2,a1,6} = G_{61}$; and additional constraints.

Thus, the LP formulation of the IEEE 14-bus and 30-bus systems is developed. Now, we investigate the AMPL implementation of Dantzig Wolfe procedure and results in Chapter 6.

CHAPTER 6. IMPLEMENTATION AND TESTING OF DANTZIG WOLFE PROCEDURE

This chapter discusses the AMPL implementation and results run for the IEEE 14-bus and IEEE 30-bus systems. The environment for the AMPL modeling software is discussed regarding how to specify the model, data, and run file information.

Overview of Modeling in AMPL and Results

Practical, large-scale mathematical programming involves more than just the minimization or maximization of an objective function subject to constraint equations and inequalities. Before any optimizing algorithm can be applied, some effort must be expended to formulate the underlying model and to generate the requisite computational data structures. If algorithms could deal with optimization problems as people do, then the formulation and generation phases of modeling might be relatively easy. In reality, however, there are many differences between the form in which human modelers understand a problem and the form in which algorithms solve it. Reliable translation from the "modeler's form to the algorithm's form" is often a considerable expense.

In the traditional approach for translation, the work is divided between a human and a computer. First, a person who understands the modeler's form writes a computer program where the output represents the required data structures. Then, a computer compiles and executes the program to create the algorithm's form. This arrangement is often costly and error-prone; most seriously, the program must be debugged by a human modeler even though the algorithm's output form is not meant for people to read.

In the important special case of linear programming, the largest part of the algorithm's form is the representing the constraint coefficient matrix. Typically, this matrix is a very sparse matrix where rows and columns number in the hundreds or thousands, and where nonzero elements appear in intricate patterns. A computer program that produces a compact representation of the coefficients is called a matrix generator.

Compared to previous languages, AMPL is notable for the generality of its syntax and for the similarity of its expressions to the algebraic notation customarily used in the modeler's form. AMPL offers a variety of types and operations to define indexing sets as well as a range of logical expressions. AMPL draws considerable inspiration from the XML prototype language [Fou83], incorporating many changes and extensions.

AMPL is a new language designed to make these steps easier and less error-prone. AMPL closely resembles the symbolic algebraic notation that many modelers use to describe mathematical programs, yet it is regular and formal enough to be processed by a computer system; it is particularly notable for the generality of its syntax and for the variety of its indexing operations. We have implemented a translator that takes a linear AMPL model and the associated data as input and produces output suitable for standard Dantzig Wolfe linear-programming optimizers.

Lagrangian Relaxation Procedure

Dual decomposition, and more generally Lagrangian relaxation, is a classical method for combinatorial optimization [Sal04]. Dual decomposition leverages the observation that many decoding problems can be decomposed into two or more subproblems, together with linear constraints that enforce some notion of agreement among solutions for the different problems. The subproblems are chosen such that they can be solved efficiently using exact combinatorial algorithms. The agreement constraints are incorporated using Lagrange multipliers, and an iterative algorithm—for example, a sub-gradient algorithm—is used to minimize the resulting dual variables. Dual decomposition algorithms have the following properties. They are typically simple and efficient. For example, sub-gradient algorithms involve two steps for each iteration: first, each subproblem is solved using a combinatorial algorithm; second, simple additive updates are made to the Lagrange multipliers. They have well-understood formal properties, particularly through connections to linear-programming (LP) relaxations. In cases where the underlying LP relaxation is tight, they produce an exact solution for the original decoding problem, with a certificate of optimality. In cases where the underlying LP is not tight, heuristic methods can be used to derive a good solution; alternatively, constraints can be added incrementally until the relaxation is tight, at which point an exact solution is recovered.

Dual decomposition, where two or more combinatorial algorithms are used, is a special case of Lagrangian relaxation (LR). It is useful to consider LR methods that utilize a single combinatorial algorithm, together with a set of linear constraints that are, again, incorporated using Lagrange multipliers. Utilizing a single combinatorial algorithm is qualitatively different from dual-decomposition approaches, although the techniques are very closely related. Lagrangian relaxation has a long history in the combinatorial optimization literature, going back to the seminal work of Held and Karp in 1971 [HK71], who derive a relaxation algorithm for the traveling salesman problem.

The Lagrangian relaxation of general LP is given as

$$\mathrm{Z} = \min \mathrm{Cx} \quad \text{subject to } \mathrm{Ax} \leq \mathrm{b}, \mathrm{Bx} \leq \mathrm{d} \text{ and } \mathrm{x} \geq 0$$

The DW version of the LP in equation is given by

$$\text{Minimize } Z = cx - \lambda^k (Ax - b_0) \tag{297}$$

subject to

$$Bx \leq b; x_i \geq 0; b_i \geq 0 \tag{298}$$

Figure 27 illustrates the interaction between the sub problems and master problems via dual variables and theta. The Lagrangian multiplier is chosen iteratively by the AMPL code to solve the problem.

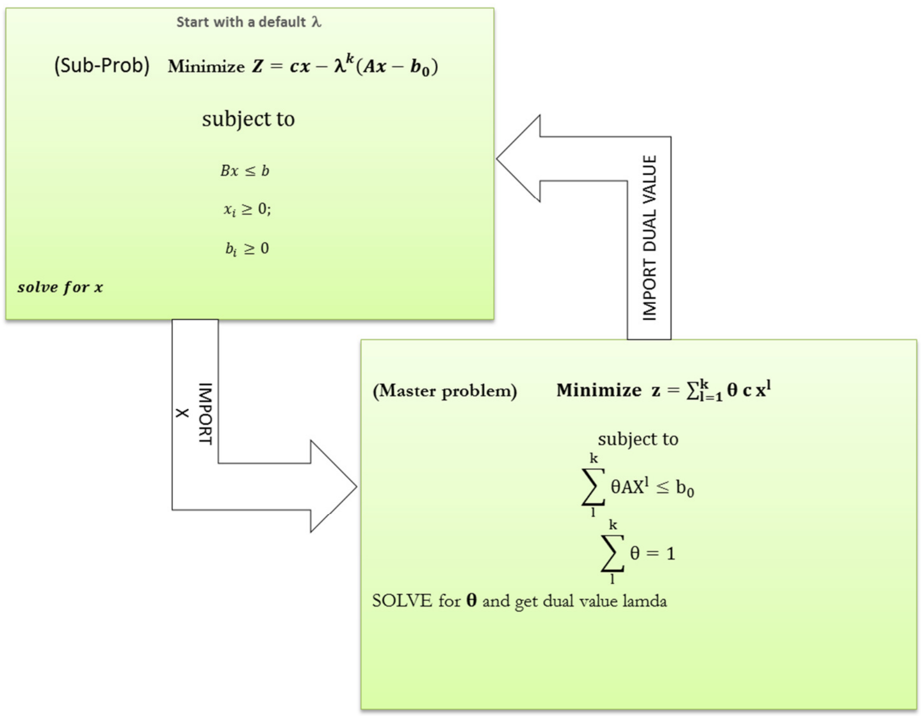


Figure 27. Lagrangian relaxation of the Dantzig-Wolfe decomposition.

Computational Results of IEEE Bus System

In testing this approach, we observe that the Lagrangian relaxation version of DW performs poorly when compared to other models such as Direct LP and Direct DW implementation. Table 14 indicates the real power values of the loads and generators for an IEEE 14-bus system. There is a balance in the total supply and demand for this system which equals 258 MW. This system has 5 generators and 11 loads as seen in Table 14. The initial assumption on generator values to individual regions is shown in Table 15.

Table 14. Supplies and demand profile for the IEEE 14-bus system

	L1(2)	**L2(3)**	**L3(5)**	**L4(4)**	**L5(6)**	**L6(11)**	**L7(10)**	**L8(9)**	**L9(12)**	**L10(13)**	**L11(14)**	**Supply**
G1 (1)	0.55	1.7	.14	4.47	.15	.16	.52	.11	.424	7.67	8.21	88
G2 (2)	1	0.15	6.34	9.92	6.35	6.36	6.72	7.31	6.624	6.87	7.41	60
G3 (3)	6.15	1	12.79	8.53	15.88	15.89	16.25	16.84	16.15	16.4	16.9	60
G4 (8)	11.92	10.53	9.34	2.0	3.31	3.3	2.94	2.35	3.58	3.83	3.33	25
G5 (6)	6.35	15.88	0.01	7.35	1	0.01	0.37	0.96	0.274	0.44	0.98	25
	21.7	94.2	7.6	47	11.2	3.5	9	29.5	6	13.5	14.8	258

Table 15. Intial allocation of generator values for three regions

Generators	**Region 1**	**Region 2**	**Region 3**
G1	58.10	19.36	9.68
G2	39.65	13.20	6.60
G3	39.65	13.20	6.60
G4	16.50	5.50	2.75
G5	16.52	5.50	2.75

To run the DW procedure, we make initial assumptions about the generator values as shown in Table 15. This assumption is reasonable when considering the number of loads and generators in the given system.

A print screen view of model, data, and run file is shown in Figure 28.

```
# ----------------------------------------
# DANTZIG-WOLFE DECOMPOSITION
# (model file)
# ----------------------------------------
### SUBPROBLEM ###
param cr; #no. of complicating rows
param or; # no. of other rows

param nsub; # no of subproblems
param nv; #no. of variables
param n_start {1..nsub}; #no. of variables
param n_end {1..nsub}; #no. of variables

param K >= 1 default 1;
param a {1..cr, 1..nv};
param b{1..cr};
param c{1..nv};

param d {1..or, 1..nv};
param f {1..or};

param lambda {1..cr} >= 0 default 100;
param xl {1..K, 1..nv} ;#>= 0 default 0;
var x {1..nv} >= 0;

### MASTER PROBLEM ###

minimize Master_ov : sum {s in 1..nsub, l in 1..K} sum {i in n_start[s]..n_end[s]} c[i]*lambda[l]* xl[l,i];

subject to Master_row {j in 1..cr}: sum{ s in 1..nsub,l in 1..K, i in n_start[s]..n_end[s]} a[j,i]*xl[l,i] <= b[j];

subject to Convexity :sum {l in 1..K} lambda[l] = 1;


### SUB PROBLEM ###

minimize Subproblem_ov {s in 1..nsub}:

sum {i in n_start[s]..n_end[s]} c[i]*x[i]-sum {l in 1..cr}lambda[l]*(sum {i in n_start[s]..n_end[s]} a[l,i]*x[i] - b[l]);

subject to Subconstraints {s in 1..nsub,j in n_start[s]..n_end[s]}: sum {i in n_start[s]..n_end[s]} d[j,i]*x[i] <= f[j];
```

Figure 28. Snapshot of the AMPL model file showing DW Implementation.

The number of master constraints or complicating constraints is indicated using the “param” command in AMPL. For example, the “cr” variable refers to master constraints, and the “or” variable refers to region constraints or other constraints. The subproblem classification is determined using the “nsub” parameter. The convexity constraint is denoted using the lamda variable. The subproblem matrices are denoted using variables d and f in the model file. Separate data and run files are used to run the program. The subproblems setting as seen in AMPL is shown in Figure 29 and the commodity constraints are shown in Figure 30.

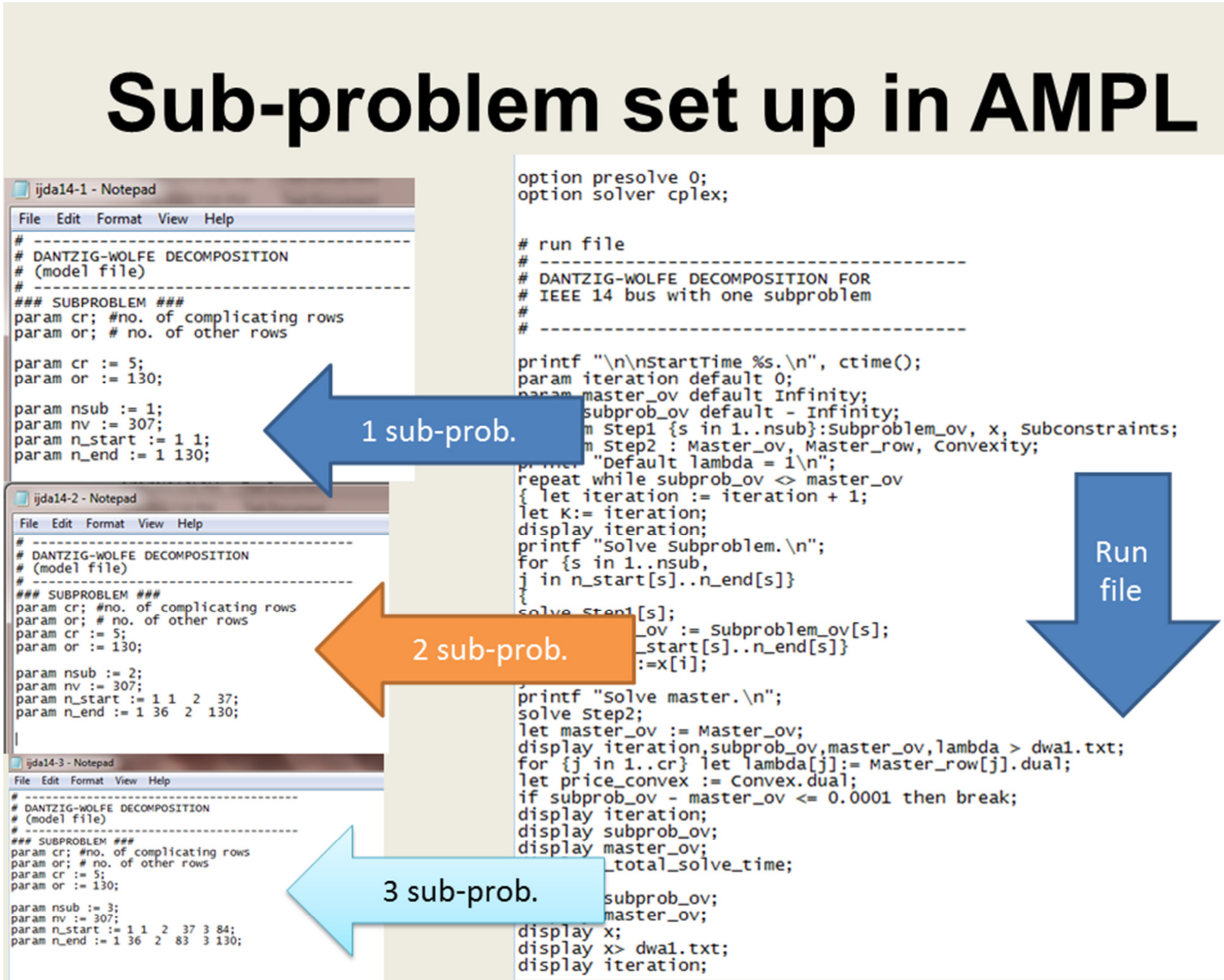
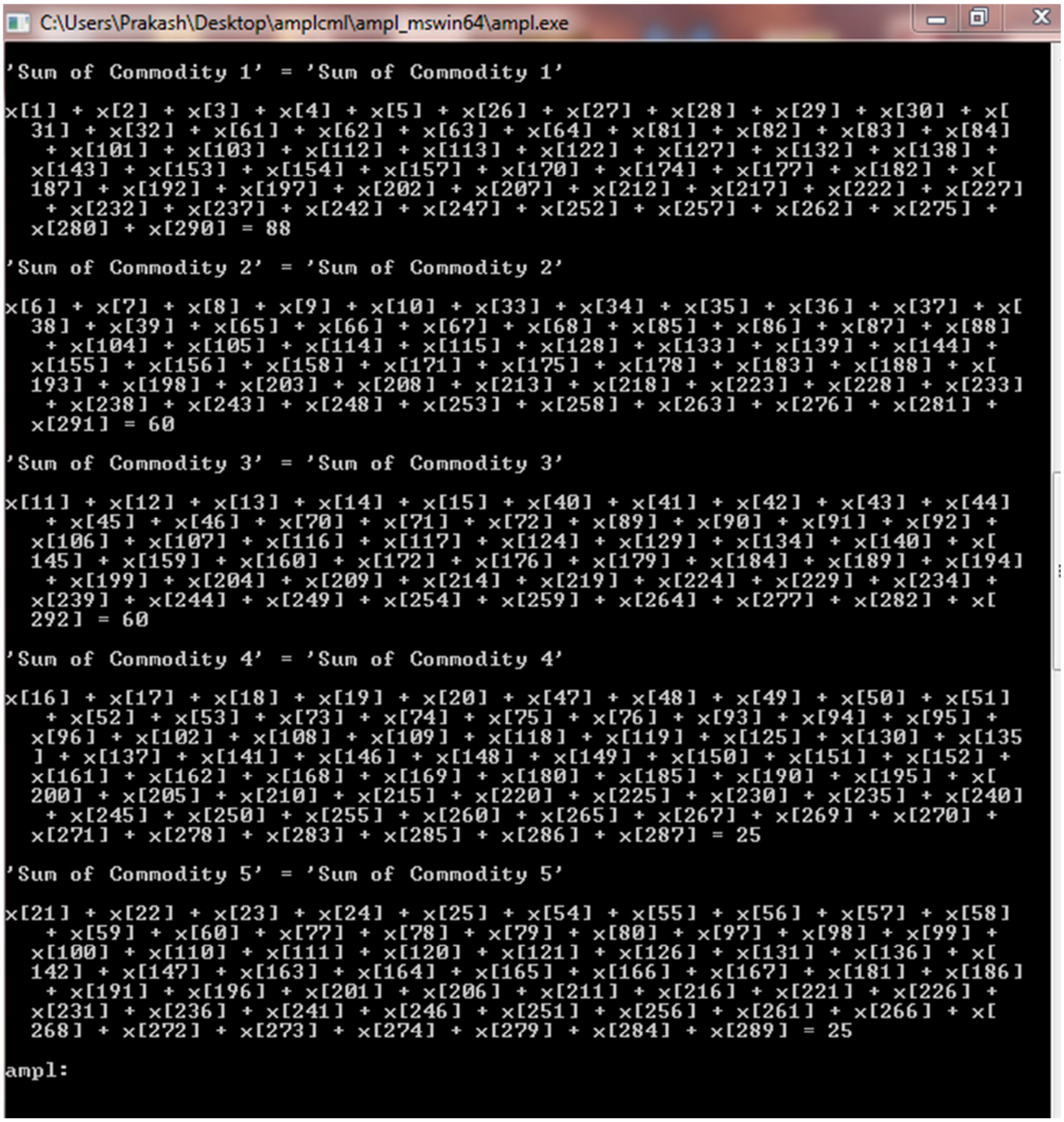


Figure 29. AMPL model file of DW.

The model file in Figure 29 show how the interactions between the master and sub problems occur iteratively and the process in which dual values are calculated. The run file is embedded in the data file itself for convenience. As indicated in Figure 30, the commodity constraints for each generators in the IEEE 14 bus system is satisfied. This figure indicate that our algorithm did not exceed and within limits of the capacities of these generators. For example, the generator 1 capacity has not exceeded more than 88 MW and similarly generator 2 is within its limit of 60 MW. The nodes that participate is included when performing aggregate "sum" operation.



```
C:\Users\Prakash\Desktop\amplcml\ampl_mswin64\ampl.exe
'Sum of Commodity 1' = 'Sum of Commodity 1'

x[1] + x[2] + x[3] + x[4] + x[5] + x[26] + x[27] + x[28] + x[29] + x[30] + x[
  31] + x[32] + x[61] + x[62] + x[63] + x[64] + x[81] + x[82] + x[83] + x[84]
   + x[101] + x[103] + x[112] + x[113] + x[122] + x[127] + x[132] + x[138] +
  x[143] + x[153] + x[154] + x[157] + x[170] + x[174] + x[177] + x[182] + x[
  187] + x[192] + x[197] + x[202] + x[207] + x[212] + x[217] + x[222] + x[227]
   + x[232] + x[237] + x[242] + x[247] + x[252] + x[257] + x[262] + x[275] +
  x[280] + x[290] = 88

'Sum of Commodity 2' = 'Sum of Commodity 2'

x[6] + x[7] + x[8] + x[9] + x[10] + x[33] + x[34] + x[35] + x[36] + x[37] + x[
  38] + x[39] + x[65] + x[66] + x[67] + x[68] + x[85] + x[86] + x[87] + x[88]
   + x[104] + x[105] + x[114] + x[115] + x[128] + x[133] + x[139] + x[144] +
  x[155] + x[156] + x[158] + x[171] + x[175] + x[178] + x[183] + x[188] + x[
  193] + x[198] + x[203] + x[208] + x[213] + x[218] + x[223] + x[228] + x[233]
   + x[238] + x[243] + x[248] + x[253] + x[258] + x[263] + x[276] + x[281] +
  x[291] = 60

'Sum of Commodity 3' = 'Sum of Commodity 3'

x[11] + x[12] + x[13] + x[14] + x[15] + x[40] + x[41] + x[42] + x[43] + x[44]
   + x[45] + x[46] + x[70] + x[71] + x[72] + x[89] + x[90] + x[91] + x[92] +
  x[106] + x[107] + x[116] + x[117] + x[124] + x[129] + x[134] + x[140] + x[
  145] + x[159] + x[160] + x[172] + x[176] + x[179] + x[184] + x[189] + x[194]
   + x[199] + x[204] + x[209] + x[214] + x[219] + x[224] + x[229] + x[234] +
  x[239] + x[244] + x[249] + x[254] + x[259] + x[264] + x[277] + x[282] + x[
  292] = 60

'Sum of Commodity 4' = 'Sum of Commodity 4'

x[16] + x[17] + x[18] + x[19] + x[20] + x[47] + x[48] + x[49] + x[50] + x[51]
   + x[52] + x[53] + x[73] + x[74] + x[75] + x[76] + x[93] + x[94] + x[95] +
  x[96] + x[102] + x[108] + x[109] + x[118] + x[119] + x[125] + x[130] + x[135
  ] + x[137] + x[141] + x[146] + x[148] + x[149] + x[150] + x[151] + x[152] +
  x[161] + x[162] + x[168] + x[169] + x[180] + x[185] + x[190] + x[195] + x[
  200] + x[205] + x[210] + x[215] + x[220] + x[225] + x[230] + x[235] + x[240]
   + x[245] + x[250] + x[255] + x[260] + x[265] + x[267] + x[269] + x[270] +
  x[271] + x[278] + x[283] + x[285] + x[286] + x[287] = 25

'Sum of Commodity 5' = 'Sum of Commodity 5'

x[21] + x[22] + x[23] + x[24] + x[25] + x[54] + x[55] + x[56] + x[57] + x[58]
   + x[59] + x[60] + x[77] + x[78] + x[79] + x[80] + x[97] + x[98] + x[99] +
  x[100] + x[110] + x[111] + x[120] + x[121] + x[126] + x[131] + x[136] + x[
  142] + x[147] + x[163] + x[164] + x[165] + x[166] + x[167] + x[181] + x[186]
   + x[191] + x[196] + x[201] + x[206] + x[211] + x[216] + x[221] + x[226] +
  x[231] + x[236] + x[241] + x[246] + x[251] + x[256] + x[261] + x[266] + x[
  268] + x[272] + x[273] + x[274] + x[279] + x[284] + x[289] = 25

ampl:
```

Figure 30. Snapshot of commodity constraints.

Figure 31 shows the AMPL allocation output for all variables involved in the IEEE 14-bus simulation. The total number of variables involved in the allocation process is 307 varaibles. Each variable is represented as a "node" in AMPL modeling.

x [*] :=

1	79.7893	78	0	155	0	232	0.729333
2	-52.0857	79	0	156	0	233	0
3	0	80	0	157	0	234	0
4	0	81	0	158	0	235	0.744513
5	-27.7036	82	0	159	0	236	0.733333
6	101.849	83	0	160	0	237	0
7	-52.0857	84	0	161	-21.1407	238	0
8	22.06	85	0	162	4.75549	239	0
9	0	86	0	163	0	240	0
10	-27.7036	87	0	164	0	241	0
11	79.7893	88	0	165	0	242	0
12	-52.0857	89	0	166	0	243	0
13	0	90	0	167	0	244	0
14	0	91	0	168	4.75549	245	0
15	-27.7036	92	0	169	0	246	0
16	79.7893	93	0	170	0	247	0
17	-52.0857	94	0	171	0	248	0.733566
18	0	95	0	172	0	249	0.733566
19	0	96	0	173	0	250	0
20	-27.7036	97	0	174	0	251	0
21	79.7893	98	0	175	0	252	0
22	-52.0857	99	0	176	0	253	0
23	0	100	0	177	36.4289	254	0
24	0	101	-4.245	178	0	255	-0.0335394
25	-27.7036	102	27.7371	179	0	256	0
26	0	103	0	180	0	257	0
27	19.2964	104	-4.245	181	0	258	0
28	66.4964	105	0	182	36.4289	259	0
29	0	106	-4.245	183	0	260	0
30	0	107	0	184	0	261	0
31	0	108	-4.245	185	0	262	0
32	0	109	0	186	0	263	0
33	0	110	-4.245	187	0	264	0
34	19.2964	111	0	188	0	265	0
35	66.4964	112	-15.445	189	0	266	0
36	0	113	-29.935	190	0	267	0
37	0	114	-15.445	191	0	268	0
38	0	115	-29.935	192	-14.712	269	0
39	0	116	-15.445	193	-14.6993	270	0
40	0	117	-29.935	194	-14.6993	271	0
41	41.3564	118	-15.445	195	-14.7335	272	0
42	66.4964	119	-29.935	196	-14.7	273	0
43	0	120	-15.445	197	-15.5413	274	0
44	22.06	121	-29.935	198	-15.5329	275	0
45	0	122	0	199	-15.5329	276	0
46	0	123	0	200	-15.511	277	0
47	0	124	0	201	-15.5333	278	0
48	19.2964	125	0	202	0	279	0
49	66.4964	126	0	203	0	280	0
50	0	127	0	204	0	281	0
51	0	128	0	205	0	282	0
52	0	129	0	206	0	283	0
53	0	130	0	207	0	284	0
54	0	131	0	208	0	285	0
55	19.2964	132	4.245	209	0	286	0
56	71.4233	133	4.245	210	0	287	0
57	0	134	4.245	211	0	288	0
58	0	135	4.245	212	0	289	0
59	4.92691	136	4.245	213	0	290	0
60	0	137	-47.68	214	0	291	0
61	0	138	0	215	0	292	0
62	0	139	0	216	0	293	0.476826
63	0	140	0	217	0.0120015	294	-0.0382921
64	0	141	0	218	-0.000697857	295	-0.438534
65	0	142	0	219	-0.000697857	296	0.698703
66	0	143	0	220	0	297	-0.0555203
67	0	144	0	221	0	298	-0.643183
68	0	145	0	222	4.24133	299	0.698703
69	0	146	0	223	4.96643	300	-0.0555203
70	0	147	0	224	4.96643	301	-0.643183
71	0	148	52.4355	225	4.24451	302	1.679
72	0	149	52.4355	226	4.23333	303	-0.135366
73	0	150	-21.1407	227	0	304	-1.54364
74	0	151	-21.1407	228	0	305	1.67697
75	0	152	-21.1407	229	0	306	-0.133333
76	0	153	0	230	0	307	-1.54364
77	0	154	0	231	0		

Figure 31. Allocation results of the IEEE 14 bus system.

The individual nodes responsible for each commodity are listed in Figure 32. The computational results of DW using 14 bus is shown in Table 16.

Table 16. Computational results on the IEEE 14-bus system

Parameters	**Direct LP**	**Lag. with 1 sub-prob. (Relax.)**	**Lag with 2 sub-prob. (Relax.)**	**Lag with 3 sub-prob. (Relax.)**	**Lag with 10 sub-prob. (Relax.)**	**DW with 1 sub-prob.**	**DW with 2 sub-prob.**	**DW with 3 sub-prob.**	**DW with 10 sub-prob.**
Computation time (sec)	1.5	4.781	4.64	4.6875	4.796	2.437	2.421	2.328	2.375
Variables	307	307	307	307	307	307	307	307	307
Total # of Constraints SP: Sub-problem MC – Master Constraints	135	SP 1: 130 MP: 5	SP 1: 1-36 SP 2: 37-130 MP: 5	SP 1: 1-36 SP 2: 37-83 SP 3: 84-130 MP: 5	SP 1: 1-20 SP 2: 21-40 SP 3: 41-50 SP 4: 51-60 SP 5: 61-80 SP 6: 81-90 SP 7: 91-100 SP 8: 101-110 SP 9: 111-115 SP10:116-130 MP: 5	SP 1: 125 MP: 5	SP 1: 1-36 SP 2: 37-130 MP: 5	SP 1: 1-36 SP 2: 37-83 SP 3: 84-130 MP: 5	SP 1: 1-20 SP 2: 21-40 SP 3: 41-50 SP 4: 51-60 SP 5: 61-80 SP 6: 81-90 SP 7: 91-100 SP 8: 101-110 SP 9: 111-115 SP10:116-130 MP: 5
Cost	14015	14015	14015	14015	14015	14015	14015	14015	14015
Allocation result	Same	Same	Same	Same	Same	Same	Same	Same	Same

1	X-1-2-1	51	X4-2-4	101	x5,a2,1	151	xa1-6-4	201	x14-9-5	251	x11-R2-5	301	G33
2	X1-5-1	52	X3-2-4	102	xa2,5,4	152	xa1-a2-4	202	x9-10-1	252	x14-b2-1	302	G41
3	X2-1-1	53	X2-R1-4	103	x5-R1-1	153	xa2-a1-1	203	x9-10-2	253	x14-b2-2	303	G42
4	X5-1-1	54	X2-5-5	104	x5,a2,2	154	xa1-6-1	204	x9-10-3	254	x14-b2-3	304	G43
5	X1-R1-1	55	X2-4-5	105	x5-R1-2	155	xa2-a1-2	205	x9-10-4	255	x14-b2-4	305	G51
6	X1-2-2	56	X2-3-5	106	x5,a2,3	156	xa1-6-2	206	x9-10-5	256	x14-b2-5	306	G52
7	X1-5-2	57	X5-2-5	107	x5-R1-3	157	x5-a2-1	207	x9-14-1	257	xb2-14-1	307	G53
8	X2-1-2	58	X4-2-5	108	x5,a2,4	158	x5-a2-2	208	x9-14-2	258	xb2-14-2		
9	X5-1-2	59	X3-2-5	109	x5-R1-4	159	x5-a2-3	209	x9-14-3	259	xb2-14-3		
10	X1-R1-2	60	X2-R1-5	110	x5,a2,5	160	xa2-a1-3	210	x9-14-4	260	xb2-14-4		
11	X1-2-3	61	X3-4-1	111	x5-R1-5	161	xa2-5-4	211	x9-14-5	261	xb2-14-5		
12	X1-5-3	62	X2-3-1	112	x6-12-1	162	xb1-b2-4	212	x9-c2-1	262	x14-R2-1		
13	X2-1-3	63	X4-3-1	113	x12-13-1	163	xb2-b1-5	213	x9-c2-2	263	x14-R2-2		
14	X5-1-3	64	X3-R1-1	114	x6-12-2	164	xb1-6-5	214	x9-c2-3	264	x14-R2-3		
15	X1-R1-3	65	X3-4-2	115	x12-13-2	165	xb1-13-5	215	x9-c2-4	265	x14-R2-4		
16	X1-2-4	66	X2-3-2	116	x6-12-3	166	x14-b2-5	216	x9-c2-5	266	x14-R2-5		
17	X1-5-4	67	X4-3-2	117	x12-13-3	167	x11-b2-5	217	x9-R2-1	267	x7-R2-4		
18	X2-1-4	68	X3-R1-2	118	x6-12-4	168	xb2-14-4	218	x9-R2-2	268	x7-R2-5		
19	X5-1-4	69	X3-4-3	119	x12-13-4	169	xb2-11-4	219	x9-R2-3	269	x13-b1-4		
20	X1-R1-4	70	X2-3-3	120	x6-12-5	170	xc2-7-1	220	x9-R2-4	270	x6-b1-4		
21	X1-2-5	71	X4-3-3	121	x12-13-5	171	xc2-7-2	221	x9-R2-5	271	xb1-b2-4		
22	X1-5-5	72	X3-R1-3	122	x12-6-1	172	xc2-7-3	222	x10-11-1	272	xb2-b1-5		
23	X2-1-5	73	X3-4-4	123	x12-6-2	173	x7-c2-5	223	x10-11-2	273	xb1-6-5		
24	X5-1-5	74	X2-3-4	124	x12-6-3	174	x7-R2-1	224	x10-11-3	274	xb1-13-5		
25	X1-R1-5	75	X4-3-4	125	x12-6-4	175	x7-R2-2	225	x10-11-4	275	x4-c1-1		
26	x2-5-1	76	X3-R1-4	126	x12-6-5	176	x7-R2-3	226	x10-11-5	276	x4-c1-2		
27	x2-4-1	77	X3-4-5	127	x13-12-1	177	x8-7-1	227	x11-10-1	277	x4-c1-3		
28	x2-3-1	78	X2-3-5	128	x13-12-2	178	x8-7-2	228	x11-10-2	278	x4-c1-4		
29	x5-2-1	79	X4-3-5	129	x13-12-3	179	x8-7-3	229	x11-10-3	279	x4-c1-5		
30	x4-2-1	80	X3-R1-5	130	x13-12-4	180	x8-7-4	230	x11-10-4	280	xc1-c2-1		
31	x3-2-1	81	X4-5-1	131	x13-12-5	181	x8-7-5	231	x11-10-5	281	xc1-c2-2		
32	x2-R1-1	82	X4-3-1	132	x12-R3-1	182	x7-8-1	232	x10-R2-1	282	xc1-c2-3		
33	X2-5-2	83	X5-4-1	133	x12-R3-2	183	x7-8-2	233	x10-R2-2	283	xc1-c2-4		
34	X2-4-2	84	X4-R1-1	134	x12-R3-3	184	x7-8-3	234	x10-R2-3	284	xc1-c2-5		
35	X2-3-2	85	X4-5-2	135	x12-R3-4	185	x7-8-4	235	x10-R2-4	285	x9-c2-4		
36	X5-2-2	86	X4-3-2	136	x12-R3-5	186	x7-8-5	236	x10-R2-5	286	xc2-c1-4		
37	X4-2-2	87	X5-4-2	137	X13-B1-4	187	x8-R2-1	237	x11-b2-1	287	x7-c2-4		
38	X3-2-2	88	X4-R1-2	138	x13-R3-1	188	x8-R2-2	238	x11-b2-2	288	x9-c2-5		
39	X2-R1-2	89	X4-5-3	139	x13-R3-2	189	x8-R2-3	239	x11-b2-3	289	xc2-c1-5		
40	X2-5-3	90	X4-3-3	140	x13-R3-3	190	x8-R2-4	240	x11-b2-4	290	xc2-9-1		
41	X2-4-3	91	X5-4-3	141	x13-R3-4	191	x8-R2-5	241	x11-b2-5	291	xc2-9-2		
42	X2-3-3	92	X4-R1-3	142	x13-R3-5	192	X10-9-1	242	xb2-11-1	292	xc2-9-3		
43	X5-2-3	93	X4-5-4	143	x6-R3-1	193	X10-9-2	243	xb2-11-2	293	G11		
44	X4-2-3	94	X4-3-4	144	x6-R3-2	194	x10-9-3	244	xb2-11-3	294	G12		
45	X3-2-3	95	X5-4-4	145	x6-R3-3	195	x10-9-4	245	xb2-11-4	295	G13		
46	X2-R1-3	96	X4-R1-4	146	x6-R3-4	196	x10-9-5	246	xb2-11-5	296	G21		
47	X2-5-4	97	X4-5-5	147	x6-R3-5	197	x14-9-1	247	x11-R2-1	297	G22		
48	X-2-4-4	98	X4-3-5	148	x6-B1-4	198	x14-9-2	248	x11-R2-2	298	G23		
49	X2-3-4	99	X5-4-5	149	xb1-6-4	199	x14-9-3	249	x11-R2-3	299	G31		
50	X5-2-4	100	X4-R1-5	150	x6-a1-4	200	x14-9-4	250	x11-R2-4	300	G32		

Figure 32. Snapshot of nodes and variables in the IEEE 14 bus simulation.

The complete list of all variables involved in the IEEE 14-bus system is indicated through an Excel snapshot in Figure 33.

The model is extended and tested with the IEEE 30-bus system. The results show that directly implementing LP takes a little longer than the decomposition scheme. Moreover, the Dantzig-Wolfe relaxation procedure takes much longer and performs the worse compared to

Dantzig Wolfe procedure. The Dantzig-Wolfe implementation runs faster and provides reasonable computational time savings. Table 16 reflects the 14-bus system performances with multi-region decomposition. The total number of variables in the 14-bus and 30-bus systems is 307 and 650 variables with 130 and 225 sub-constraints. The computational results of the IEEE 30 bus with various decomposition structures are given in Table 17. The allocation and objective value of cost parameter yield in same solution. The interactions between the master and sub problems take 212 iterations to attain an optimal cost for three region decomposition. The same formulation can be decomposed into two regions with some modifications. Certain variables that link to nodes are not considered. The coefficients are assigned as zero if the variable is not involved in the decomposition process. For example, joint-capacity constraint R3 is not involved in the two-region problem. Similarly, nodes B1 and B2 are not considered or removed.

Table 17. Computational results on the IEEE 30-bus system

Parameters	**Direct LP**	**Lag with 1 sub-prob. (Relax.)**	**Lag with 2 sub-prob. (Relax.)**	**Lag with 3 sub-prob. (Relax.)**	**Lag with 10 sub-prob. (Relax.)**	**DW with 1 sub-prob.**	**DW with 2 sub-prob.**	**DW with 3 sub-prob.**	**DW with 10 sub-prob.**
Computation time: (sec)	4.5	14.781	14.765	14.6875	14.96	3.4375	3.421	3.38	3.95
Variables	650	650	650	650	650	650	650	650	650
Total # of Constraints SP: Sub-problem MC – Master Constraints	275	SP 1: 270 MP: 5	SP 1: 1-36 SP 2: 37-275 MP: 5	SP 1: 1-36 SP 2: 37-150 SP 3: 150-275 MP: 5	SP 1: 1-20 SP 2: 21-40 SP 3: 41-50 SP 4: 51-60 SP 5: 61-80 SP 6: 81-90 SP 7: 91-100 SP 8: 101-110 SP 9: 111-115 SP10:116-275 MP: 5	SP 1: 270 MP: 5	SP 1: 1-36 SP 2: 37-275 MP: 5	SP 1: 1-36 SP 2: 37-83 SP 3: 84-275 MP: 5	SP 1: 1-20 SP 2: 21-40 SP 3: 41-50 SP 4: 51-60 SP 5: 61-80 SP 6: 81-90 SP 7: 91-100 SP 8: 101-110 SP 9: 111-115 SP10:116-275 MP: 5
Cost	67485.4	67485.4	67485.4	67485.4	67485.4	67485.4	67485.4	67485.4	67485.4
Allocation result	Same	Same	Same	Same	Same	Same	Same	Same	Same

Our results show huge savings for computational cost and response time with the entire IEEE 30-bus system compared to Direct LP and Lagrangian relaxation formulations. The key contribution is that we have developed, implemented, and tested the Dantzig-Wolfe procedure in the IEEE bus system with various decomposition structures. It is important to note that all decomposition results and methods result in same cost and allocation values, the main contribution of this dissertation.

Sensitivity analysis of the IEEE 14-bus system and the IEEE 30-bus system for loss (failures and repair rates) is investigated. Finding the optimal solution for a linear programming model is important, but it is not the only information available. There is a tremendous amount of sensitivity information, or information about what happens when data values are changed. When formulating a problem as a linear program, we have to invoke a certainty assumption: we have to know what values the data took; finally, decisions are made based on that data from the LP run. Often, this assumption is somewhat dubious: the data might be unknown, guessed, or otherwise inaccurate. How can we determine the effect on the optimal decisions if values such as the failure or repair rates change? Clearly, some numbers in the data are more important than others. Can we find the "important" numbers? Can we determine the effect of misestimating? Linear programming offers extensive capabilities to address these questions. In the model, I test the sensitivity of our LP formulation with respect to the effect on line failures. I have simulated certain line failures by treating a certain variables to zero and notice the change in the allocation procedure.

In an IEEE 14 bus system, I simulated lins failures I Region 1 by treating $\mathrm{x}_{12}, \mathrm{x}_{23}$ and x_{24} to 0. I was able to observe the re-allocation of power to loads via other lines such as x_{15} and x_{54} to other regions and eventually reaching an optimum value. Similarly, in IEEE 30 bus system, the lines $\mathrm{x}_{25}, \mathrm{x}_{57}$ and x_{76} was set to zero and the re-routing of allocation can be seen via x_{13}, x_{34} lines to other regions. The failure scenario thus can easily be modeled by treating those lines as zero. Thus it is evident from these AMPL runs, the sensitivity of DW allocation process and transmission lines in the IEEE 14 bus and 30 bus system are bound to flow limits set on the transmission lines.

The inferences about the results and possible future tasks related to this procedure are discussed in the final chapter with conclusions.

CHAPTER 7. CONCLUSION

A distributed linear-programming model has been created, developed, implemented, and tested. Two standard IEEE bus systems were modeled and successfully decomposed in multiple ways into sub problems. The problem was solved iteratively in each case, and directly supports resource allocation in a Smart grid environment. I have shown that thean LP-based design using Dantzig-Wolfe decomposition can execute and quickly determine the primary resource scheduling and allocation issues in the event that a failure occurs in the grid. The decomposition procedure can be easily managed by system operators. In the study using the 4-bus, 14-bus, and 30-bus systems, the results indicate that the computational benefits of the Dantzig-Wolfe approach enable fast responses on the order of a millisecond to a few seconds as network size increases. Although the 30-bus system is not a large bus network, the results clearly indicate a faster computation time if an appropriate Dantzig-Wolfe structure is formulated. This approach can enable system operators in the electric grid to respond to any allocation request for resources in the event of any outages or line failures. The key contribution of the dissertation is the design, development, and testing of a procedure that successfully decomposes an optimization problem that is defined over a large grid, but can be solved in regional pieces.

The following inferences are made for my defined problem:

Inference 1: The larger size of decomposing into regions does not guarantee computational savings for the overall problem. For example, the computational time savings is greater with the 3-region decomposition of the IEEE network rather than in the 10-region decomposition network, an important and interesting contribution of this dissertation because it conveys that not all decompositions can yield savings for the computation time. However, the solution procedure is decomposed by regions, which significantly spreads out the computational load. In addition, the procedure demonstrates that feasible resource allocation solutions can be obtained on an intermediate basis, allowing solutions to be terminated early with a still valuable heuristic problem solution.

Inference 2: The Dantzig-Wolfe decomposition performs better in the IEEE 30-bus system than the IEEE 14-bus system. This finding is probably due to increased variables and large number of constraints and many interrelated complicated constraints with the subproblems in the network's structure tested. Also, a large number of iterations between the subproblems and the master model were required.

Inference 3: The Lagrangian Relaxation procedure performs poorly compared to actual Dantzig-Wolfe version. This suggests that a more sophisticated procedure for setting Lagrangian multipliers is need.

Inference 4: Direct LP formulation performs better in the IEEE 14-bus system compared to the decomposition scheme This is likely due to the relative small size of the test problems.

Inference 5: All models result in the identical resource allocation and identical cost as measured by the objective function value. This demonstrates that the computational procedures, although solved through decomposition, still provide the best possible solution.

Inference 6: The number of iterations for interactions between master and sub problems is different, ranging from 100-300 iterations. The approach takes multiple iterations to reach an optimal cost as solution of our objective, which demonstrates the appropriate and accurate interactions between the master and sub problem constraints.

Inference 7: The proposed DW method is tested for scalability up to IEEE 30 bus system. Due to the large number of constraints for a given formulation, scalability issues remain.

Inference 8: The benefits and significance of decomposition by regions yields reasonable savings on time computations compared to running all constraint sets directly.

In summary, the key contribution is that I have developed, implemented, and tested the Dantzig-Wolfe procedure in the IEEE bus system with various decomposition structures. It is important to note that all decomposition structure results in the same cost and allocation values. This is a significant and main contribution for this dissertation. In addition, I have modeled the LP formulation for resource allocation with known uncertainty information included in chapter 3. A branch and bound based algorithm for resource allocation is also presented in chapter 4 as part of the contribution.

In this work, I define scalability is the ability of a DW process to handle a growing amount of sub-problem or constraints in a capable manner, or its ability to be enlarged to accommodate that growth. For example, it can refer to the capability of a system to increase total throughput (such as computational time) under an increased load when resources constraints are added.

The DW algorithm said to scale if it is suitably efficient and practical when applied to large problems, such as when there are a large number of participating nodes, as is the case of a typical distributed Smart grid system. The dissertation work shows that feasible scalability up to the level of IEEE 30 bus system with 650 variables and 325 constraints.

The application of the evaluated decompositions can be implemented on a hierarchical US electric grid. For example, the practicality for true large scale grid can be envisioned as distributing resources among the 4 Independent System Operators (ISOs), which are the NYISO, MISO, Western Interconnection and Southern Interconnection systems.

Regarding future plans for individuals who wish to study this problem, I recommend that they test with large-scale systems such as the standard 118-bus, 300-bus, and 1000-bus models. The efficiency of the Dantzig-Wolfe procedure relies on how complicated constraints are greatly involved in sub-problem variables and solutions.

REFERENCES

[AAT05] Alekhnovich. M, Arora. S, and Tourlakis. I, Towards strong non approximability results in the lovasz-schrijver hierarchy. Theory of Computing, ACM, 2005, pp. 294-303.

[ABL+06] Arora. S, Bollobas. B, Lorasz. L, and Tourlakis. I, Proving integrality gaps without knowing the linear program, Theory of Computing, February 2006, pp. 19-51.

[ADH+94] Adler. R, Daniel. S, Heising. C, Lauby. M, Ludorf. R, and White, T, An IEEE survey of US and Canadian overhead transmission outages at 230 kV and above, IEEE Transaction on Power Delivery, January 1994, pp. 21-39.

[Ami04] Amin. M, Balancing market priorities with security issues: Interconnected system operations and control under the restructured electricity enterprise, IEEE Power and Energy Magazine, July 2004, pp. 30-38.

[Ami05] Amin. M, Powering the 21st Century: We can and must modernize the grid, IEEE Power and Energy Magazine, March 2005, pp. 93-95.

[Aro02] Arora. S, Proving integrality gaps without knowing the linear program. Proceedings of the 43rd Symposium on Foundations of Computer Science, 2006, pp. 313-322.

[AS08] Amin. M, and Schewe. P, Preventing blackouts, Scientific American., Vol. 296, August 2008, pp. 60-67.

[AW05] Amin. M, and Wollenberg. B, Toward a Smart grid, IEEE Power and Energy Magazine, September 2005, pp. 34-38.

[BB91] Balakrishnan. B, and Balemi. S, Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems, International Journal of Robust and Nonlinear Control, December 1991, pp. 295-317.

[BBG77] Bradley. G, Brown. G, and Graves. G, Design and Implementation of Large Scale Primal Transshipment Algorithms, Management Science, 1977, pp. 1-34.

[BCP08] Bignucoloa. F, Caldona. R, and Prandonib, V, Radial MV networks voltage regulation with distribution management system coordinated controller, Electric Power Systems Research, April 2008, pp. 634-664.

[BJN98] Barnhart. C, Johnson. E, and Nemhauser. G, Branch-and-Price: Column generation for solving huge integer programs, Operations Research, 1998, pp. 316–329.

[Bou01] Boutilier. C, Planning and programming with first-order markov decision processes: Insights and challenges. Morgan Kaufmann, MA, 2001.

[Bou02] Boutilier. C, A POMDP formulation of preference elicitation problems. Proceedings of the Eighteenth National Conference on Artificial Intelligence, CA, USA, 2002, pp. 239-246.

[Bri94] Marick. B, The craft of software testing: Subsystem testing including object-based and object-oriented testing, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1994.

[BW03] Bailey. D, and Wright. E, Practical SCADA for industry, Newnes, Oxford, UK, 2003.

[Chv83] Chvatal. V, Linear programming, Freeman, New York, New York, USA, 1983.

[CLD+02] Carreras. B. Lynch. V, Dobson. I, Chaoes, N, Critical points and transitions in an electric power transmission model for cascading failure blackouts, Chaos, December 2002, pp. 985-994.

[CT99] Camponogara. C, and Talukdar. S, Agent Cooperation: Distributed Control Applications, Proceedings of the International Conference on Intelligent System Application to Power Systems, VA, April 1999, pp.1-6.

[Dan63] Dantzig. G.B, Linear Programming and Extensions, Princeton University Press, Princeton, New Jersey, USA, 1963.

[Dan83] Dantzig. G.B, Reminiscences about the origins of linear programming , Mathematical Programming: The State of the Art Springer Lecture Notes, Springer Verlag, Berlin, 1983, pp. 78-86.

[Dan98] Dantzig. G.B, Linear Programming and Extensions, Princeton University Press, 1998.

[DCN04] Dobson. I., Carreras. B. and Newman. D.E, Probabilistic load-dependent cascading failure with limited component interactions, IEEE International Symposium on Circuits and Systems, Canada, 2004, pp.15-32.

[DLW+89] Dewitt. C, Lasdon. S, Waren D, Brenner. A, and Melhem. A, OMEGA: An Improved Gasoline Blending System for Texaco. Interfaces, 1989, pp. 85-101.

[DNS+95] Dwyer. A, Nielsen. R, Stangl. J, and Markushevich. N, Load to Voltage Dependency Tests at B.C. Hydro, IEEE Transactions on Power Systems, May 1995, pp. 709-715.

[FES12] Farag. E, El-Saadany. F, and Seethapathy. R, A Two Ways Communication-Based Distributed Control for Voltage Regulation in Smart Distribution Feeders, IEEE Transaction on Smart grid, March. 2012, pp 766-772.

[FF10] Ford. R, and Fulkerson. R, Flows in Networks, Princeton University Press, NJ, 2010.

[FG96] Franklin. S, and Graesser. A, Is it an agent, or just a program?: A taxonomy for autonomous agents, Proceedings of 3rd International Workshop on Agent Theories, Architectures and Languages, Springer-Verlag, NY, 1996, pp. 21-35.

[Fou83] Fourer. R, Modeling Languages versus Matrix Generators for Linear Programming. ACM Transactions on Mathematical Software, 1983, pp. 143-183.

[GK03] Gilmore. S, and Kloul. L, A unified approach to performance modeling and verification. Paper presented at Dagstuhl seminar on Probabilistic Methods in Verification and Planning, Germany, May 2003.

[GPR+09] Greitzer. F, Podmor. R, Robinson. M, and Ey. Pamela, Naturalistic decision making for Power System Operators, International Conference on Naturalistic Decision Making (NDM), London, 2009.

[GPS00] Gueret. C, Prins. C, and Sevaux. M, Programmation lineaire, Editions Eyrolles, Paris, 2000.

[HL01] Hillier. F, and Lieberman. G, Introduction to Operations Research, McGraw-Hill, UK, 2001.

[JAW+10] Javed. F, Arshad. N, Wallin. F, Vassileva. I, and Dahlquist. E, Engineering Optimization Models at Runtime for Dynamically Adaptive Systems, IEEE International Conference on Engineering of Complex Computer Systems, Singapore, 2010, pp. 253-254.

[JW00] Jennings. N, and Wooldridge. M, Agent-Oriented Software Engineering, Proceedings of the 9th European Workshop on Modeling Autonomous Agents in a Multi-Agent World, Spain 2000.

[Kad09] Kadar. P., Multi Objective Optimization of Smart grid Structure, Proceedings of 15th International Conference on Intelligent System Applications to Power Systems, Greece, September 2009

[Kar01] Wiegers. K., Software Requirements, Second edition, Microsoft Press, ISBN-10-0-7356-1879-8.

[KH09] Karnouskos. S, and Holanda. N, Simulation of a Smart grid City with Software

Agents, Proceedings of European Modeling Symposium (EMS 2009), Athens, Greece, November 2009.

[KHS05] Khattam. W, Hegazy. Y, and Salama. Y. An integrated distributed generation optimization model for distribution system planning, IEEE Transactions on Power Systems, May 2005, pp. 1158-1165.

[KJN+04] Kirschen. D, Jawayeera. D, Nedic. D, and Allan, N, A probabilistic indicator of system stress, IEEE Transactions on Power Systems, 2004, pp. 1650-1657.

[KP94] Kozina. G, and Perepelista. V. Interval spanning trees problem: Solvability and computational complexity, Interval Computations, 1994, pp. 42-50.

[Kru56] Kruskal. J, On the shortest spanning subtree of a graph and the travelling salesman problem, Proceedings of the American Mathematical Society, 1956, pp. 48-50.

[Kru06] Krutz. R, Securing SCADA systems, Wiley Publishing Inc, 2006.

[KY97] Kouvelis. P, and Yu. P, Robust Discrete Optimization and its Applications. Kluwer Academic Publishers, MA, 1997.

[LW66] Lawler. L, and Wood. E, Branch-and-bound methods: A survey, Operations Research, 1966, pp. 699-719.

[MG05] Montemanni. R, and Gambardella. L, A branch and bound algorithm for the robust spanning tree problem with interval data, European Journal of Operational Research, March 2005, pp. 771-779.

[Mom01] Momoh. J, Electric Power System Applications of Optimization, Marcel Dekker, NY, 2001.

[Moo91] Moore. R, Global optimization to prescribed accuracy. Computers and Mathematics with Applications, 1991, pp. 25-39.

[MWJ+10] Mohsenian-Rad. M, Wong. V, Jatskevich. J, and Schober. R, Optimal and autonomous incentive-based energy consumption scheduling algorithm for Smart grid, Proceedings in IEEE Power Engineering Society Conference on Innovative Smart grid Technologies, January 2010, pp.1-8.

[NF12] Nguyen. C, Flueck. A, Agent Based Restoration with Distributed Energy Storage Support in Smart grids, IEEE Transactions on Smart grid, pp. 1029-1038, June 2012.

[NGL+11] Nygard. K, Ghosn. S, Loegering. D, McCulloch. R, and Ranganathan. P, Implementing a Flexible Simulation of a Self-Healing Smart grid, 2011 International Conference on Modeling, Simulation and Visualization Methods, Nevada, July 2011.

[NPG+11] Nygard. K, Ranganathan. P, Ghosn. S, Loegering. D, McCulloh. R, and Chowdhury.M, Optimization Models for Energy Reallocation in a Smart grid, IEEE Machine to Machine Communications and Networking Workshop, Shanghai, China, April 2011, pp.186-191.

[NS02] Nagata. T, and Sasaki. H, A Multi-agent Approach to Power System Restoration, IEEE Transactions on Power Systems, May 2002, pp. 457-462.

[NW88] Nemhauser. G, and Wolsey. L, Integer and Combinatorial Optimization, John Wiley & Sons, NY, 1988.

[PFR09] Pipattanasomporn. M, Feroze. H, and Rahman. S, Multi-Agent Systems in a Distributed Smart grid: Design and Implementation, Proceedings of IEEE PES 2009 Power Systems Conference and Exposition, Seattle, Washington, USA, March 2009, pp.1-6.

[PKN12] Ranganathan. P and Nygard. K, A Smart Agent Oriented Linear Programming Control in electric grid, Annual Electric Power and Energy Conference, Canada, October 2012, pp. 102-106.

[PSN+88] Powell. W, Sheffi. Y, Nickerson. S, Butterbaugh. K, and Atherton. S, Maximizing Profits for North American Van Lines Truckload Division: A new Framework for pricing and Operations. Interfaces, 1988, pp. 21-41.

[Pow98] PowerGen plc, Private communication, 1998.

[RI10] Rahimi. R, and Ipakchi. A, Demand response as a market resource under the Smart grid paradigm, IEEE Transactions on Smart grid, December 2010, pp. 52-66.

[RN10] Ranganathan. P, and Nygard. K, An optimal resource assignment problem in Smart grid, The Second International Conference on Future Computational Technologies and Applications, Portugal, November 26, 2010, pp.75-82.

[RN12] Ranganathan. P, and Nygard. K, A Smart Agent Oriented Linear Programming (AOLP) architecture in Electric grid, 2012 Annual Electric Power and Energy Conference, London, Canada, October 10-12, 2012.

[RPT07] Rahman. S, Pipattanasomporn. M, and Teklu. Y, Intelligent Distributed Autonomous Power Systems (IDAPS), Proceedings of IEEE PES Annual General Meeting, Tampa, Florida, USA, 2007.

[Sal04] Salam. S, Comparison of Lagrangian relaxation and truncated dynamic programming methods for solving hydrothermal coordination problems, Proceedings of International Conference on Intelligent Sensing and Information Processing, 2004, pp. 265-270.

[She95] Shenker. S, Fundamental Design Issues for the Future Internet, IEEE Journal of Select Areas Communication, 1995, pp. 1176-1188.

[Son99] Song. Y, Modern Optimization Techniques in Power Systems, Kluwer Academic

Publishers, 1999.

[SS85] Sullivan. R, and Secrest. C, A simple Optimization DSS for Production Planning at Dairymans Cooperative Creamery Association, Interfaces, 1985, pp. 46-53.

[Tem91] Templeman. B, Optimization and decision support systems in civil engineering. Routledge, Gordon and Breach Science Publishers Ltd, 1991.

[Wad83] Waddell. R, A Model for Equipment Replacement Decision and Policies, Interfaces, 1983, pp. 1-7.

[Wan01] Wang. H, Multi-agent co-ordination for the secondary voltage control in power system contingencies, Proceedings of IEEE Generation, Transmission and Distribution, Jan 2001, pp. 61-66.

[Web01] Mathworks, Inc., Binary Integer Programming (BINTPROG), http://www.mathworks.com, accessed 08.31.2010.

[Web02] http://www.oe.energy.gov/smartgrid.htm, accessed 08.31.2010.

[WG10] Wang. C, and Groot. M, Managing end-user preferences in the Smart grid, ACM International Conference on Energy efficient computing and networking, Germany, April 2010, pp. 357-363.

[Wil93] Williams. P, Model building in mathematical programming, 3rd revised edition, John Wiley & Sons, Chichester, 1993.

[Wol98] Wolsey. L, Integer Programming, John Wiley & Sons, NY,1998.

[Wri97] Wright. S, Primal-dual interior point methods, SIAM, 1997, pp.145-157.