Stony Brook University

A Study on Two Optimal Planning Problems in Smart Grid

A Dissertation Presented

by

Eting Yuan

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Applied Mathematics and Statistics

Stony Brook University

August 2013

Stony Brook University
The Graduate School

Eting Yuan

We, the dissertation committee for the above candidate for the
Doctor of Philosophy degree, hereby recommend
acceptance of this dissertation.

Eugene A. Feinberg - Dissertation Advisor
Distinguished Professor, Applied Mathematics and Statistics

Joseph S.B. Mitchell - Chairperson of Defense
Professor, Applied Mathematics and Statistics

Jiaqiao Hu
Associate Professor, Applied Mathematics and Statistics

Thomas G. Robertazzi
Professor, Electrical and Computer Engineering

This dissertation is accepted by the Graduate School.

Charles Taber
Interim Dean of the Graduate School

Abstract of the Dissertation

A Study on Two Optimal Planning Problems in Smart Grid

by
Eting Yuan

Doctor of Philosophy
in
Applied Mathematics and Statistics

Stony Brook University

2013

Optimized planning of power devices and power generators is essential for saving energy and reducing costs in smart grid. Many of these optimization problems share a common characteristic and seek to optimize accumulated rewards or costs within a finite time planning horizon. For this reason, some of those problems can be formulated as finite horizon optimal planning problems. This dissertation performs detailed investigation on two representative finite horizon optimal planning problems: the Unit Commitment problem (UCP) and the Voltage and Reactive Power Control (VVC) Problem.

UCP is an important optimal power planning problems in electric grids. The purpose of UCP is to determine when to start up and shut down power generator units and how to dispatch committed units to meet electricity demands, ancillary services requirements, and security constraints in order to minimize total operational costs. This dissertation improves the traditional lagrangian relaxation (LR) approach and analyzes the effectiveness of using parallel computing for solving large-scale unit commitment problems with wind power penetration. Additionally, we investigate the potential of combining parallel computing with a rolling horizon scheme to improve the solution quality when a large amount of wind power is present.

One of the objectives of VVC is to determine the proper status of capacitor banks and transformer tap positions in a power distribution system to minimize daily power losses or the daily power consumption. In this dissertation, we propose to use an approximate stochastic annealing (ASA) algorithm for solving VVC problems. We also propose a lagrangian relaxation-dynamic programming (LR-DP) algorithm for solving VVC problems with operation limits on power devices to obtain upper and lower bounds on the performance of the optimal solution. The performance of the ASA algorithm is illustrated on a well-known PG&E 69-bus distribution network. Our testing results indicate that the ASA algorithm may yield solutions very close to the optimum within a moderate amount of computational time. This dissertation also discusses the convergence properties of the ASA algorithm for VVC problems.

to Limin Wang, Guangmei Luo and Liangjun Yuan

Contents

List of Figures

List of Tables

Acknowledgments

My sincere appreciation goes to everyone who helped me during my graduate study and along the way to complete this thesis. In particular, I would like to express my gratitude to the following people.

First of all, I would like express my earnest appreciation to my advisor, Professor Eugene A. Feinberg. He provided interesting research topics and invaluable support during the entire course of my graduate study. Without his guidance and inspiration, this thesis would not be complete.

I express my sincere gratitude to Professor Jiaqiao Hu. He gave me enormous help on the design of optimization algorithms and written skills. His technical and editorial advice was essential for the completion of this thesis.

I would like to thank Professor Joseph S.B. Mitchell for being my committee member. Moreover, he gave great lectures in the course of computational geometry, which helped me a lot in my graduate study. I also would like to thank Professor Thomas Robertazzi for being my committee member and Professor James Glimm for being my preliminary exam committee member.

My appreciation also goes to Dr. Jun Fei and all the other people in the smart gird research project. I am grateful to all my friends at Stony Brook University for their friendship and support.

Last but not least, I would like to thank my wife and my parents for their endless love and encouragement. I would not have finished this work without their support.

This material is based upon work supported by the Department of Energy under Award Number DE-OE0000220.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or

Chapter 1

Introduction

An important area of smart grid is to use new technologies and equipments to improve energy efficiency and reduce energy costs, which necessitates optimized planning of power devices and power generators. Some of those problems target at the optimization of accumulated, for example, operational costs or power consumption, over a finite time horizon, where in each time slot, optimal planning or control of power devices or generators is computed. In this dissertation, we categorize these optimization problems as finite horizon optimal planning problems. This dissertation focuses on two representative problems of this category: the Unit Commitment Problem (UCP) and the Voltage and Reactive Power Control (VVC) problem. We briefly describe these two problems in this chapter. Additionally, this chapter reviews the general mathematical models for finite horizon optimal planning problems and optimization algorithms for solving those problems.

1.1 Finite Horizon Optimal Planning Problems in Smart Grid

Many optimization problems in smart grid can be categorized as finite horizon optimal planning problems. A well-known example is the Unit Commitment Problem, which finds

the optimal schedule of power generators to minimize the total operational cost over a certain time horizon. Another example is the Voltage and Reactive Power Control problem, which finds the optimal control schedule of transformer load tap changers and shunt capacitors to minimize daily power losses or power consumption. We can also formulate the feeder reconfiguration problem as a finite horizon optimal planning problems with the objective of finding the optimal control schedule of line breakers in power distribution systems to minimize power losses. The rest of this section provides a brief description of UCP and VVC problems.

1.1.1 Unit Commitment Problem

Specifically, the purpose of UCP is to determine when to start up and shut down the power generator units and how to dispatch committed units to meet electricity demands, ancillary services requirements and security constraints. In power markets, UCPs are commonly solved by Independent System Operators (ISOs) such as NYISO [46, 48] on a daily basis. Usually, the optimal commitment schedule of power generators should satisfy a series of security constraints, and the corresponding UCP is referred to Security Constrained Unit Commitment (SCUC) Problem. The process of solving SCUC and publishing the computational results is referred to the unit commitment process. The inputs and outputs of SCUC are given in figure 1.1. ISOs receive generation and service bids from power generation companies, load bids from electricity retailers, and load and wind power forecasts from some mathematical forecasting models. Security constraints usually include: 1) total power generation should meet the total power consumption; 2) different levels of spinning reserve requirements should be satisfied to avoid power outage in cases of unexpected failure of one or more scheduled power generators and/or the sudden loss of wind power. 3) power transmission networks should be able to transmit generated electricity power to different areas of the electric grid without causing transmission congestion. With these inputs, ISOs use some optimization algorithms to compute the optimal generation schedule and ancillary

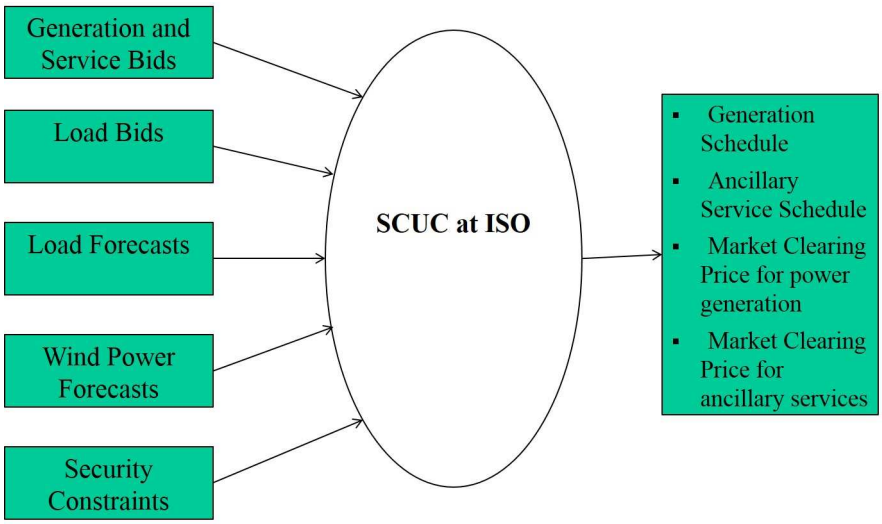

Figure 1.1: Unit Commitment Process

service schedule, which are published along with the market clearing prices of generation and ancillary services as the outputs of the unit commitment process.

Traditionally, UCPs are formulated as finite horizon deterministic optimization problems. However, as the penetration level of renewable energy resources, especially wind power, increases, stochastic UCP models are proposed for modeling the volatility of renewable energy resources. Mixed-integer linear programming [10] and lagrangian relaxation [13] are widely used for solving both deterministic and stochastic UCPs. In practice, a UCP usually has thousands of discrete and continuous variables and a large number of constraints, so it is very difficult to compute optimal solutions within an acceptable amount of time. Many heuristics are introduced to simplify the problem and reduce the computational time, but those heuristics , to a certain extent, sacrifice the optimality of the solution. In chapter 2, we propose a parallel-computing scheme for the lagrangian relaxation algorithm to boost the computational speed. Additionally, we propose a probabilistic UCP model and a rolling horizon scheme to handle the stochastic wind power generation.

1.1.2 Voltage and Reactive Power Control Problem

This dissertation studies VVC problems in power distribution systems. The VVC problem initially arose from the voltage drop along the power distribution lines, which is illustrated in figure 1.2. To comply with certain regulations, it is necessary to use some power devices and control methods to maintain voltage levels in a specific range, for example , $120V \pm 5\%$. The most widely used control devices are Transformer Load Tap Changers (LTCs) and Shunt Capacitors (SCs). LTCs function as voltage transformers and are able to increase or decrease output voltage levels by changing their tap positions. SCs inject reactive power into power distribution systems to change the electricity flow in distribution lines, and thus, to change the amount of voltage drops. Figure 1.3 illustrates the effectiveness of capacitors in reducing voltage drops. Many control methods were proposed to control voltage levels, for example in [2]. Some of these methods are based on a simple heuristic that, if the measured or estimated voltage level is below the lower limit for a certain time span, capacitors are switched on and tap positions of LTCs are increased to increase the voltage level; on the contrary, if the measured or estimated voltage level is above the upper limit, capacitors are switched off and tap positions are decreased to decrease the voltage level. The advantage of this heuristic is that it is easy to implement and is effective in maintaining voltage levels within a proper range. However, it may not be optimal.

Optimized VVC usually involves the optimal planning tap positions of LTCs and on/off states of SCs to minimize daily power losses. Thus, we consider it as a finite horizon optimal planning problem. This control scheme requires accurate day-ahead load forecast at the secondary transformer level of a power distribution feeder, which is not possible without modern smart grid technologies. In smart grid, this small-scaled load forecast is feasible by using the high resolution historical data from smart meters. Many optimization models have been proposed based on this VVC scheme [38, 40, 43, 75] during last two decades. In addition to power losses, operation costs of power devices are also considered in some of those models due to the fact that those devices such as LTCs and SCs are expensive and

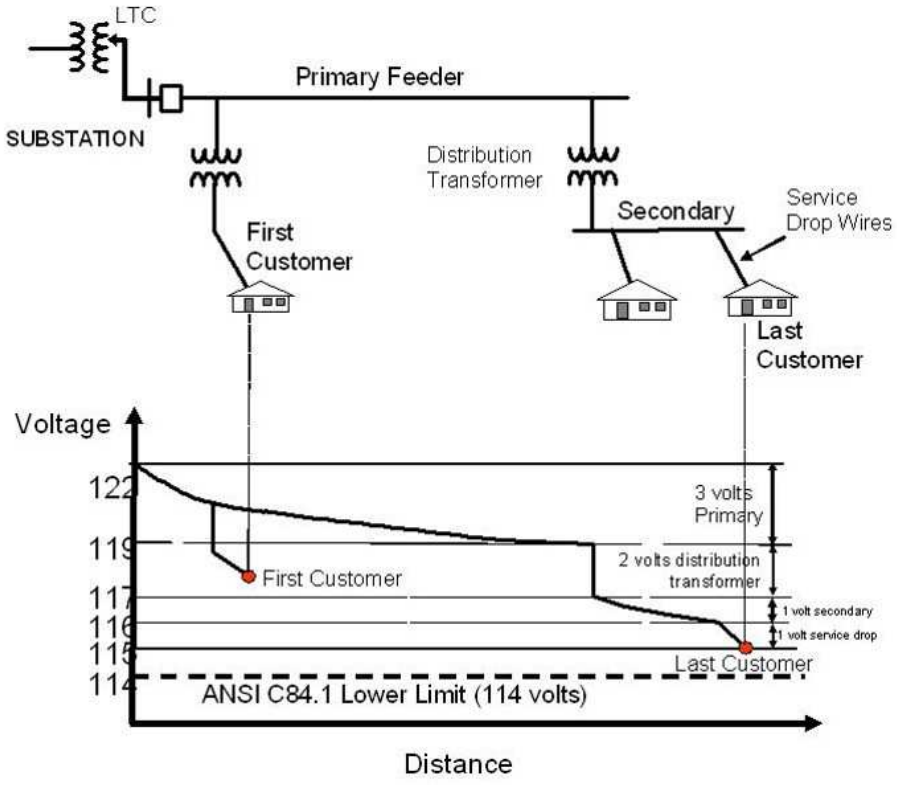

Figure 1.2: Voltage drop in a power distribution system

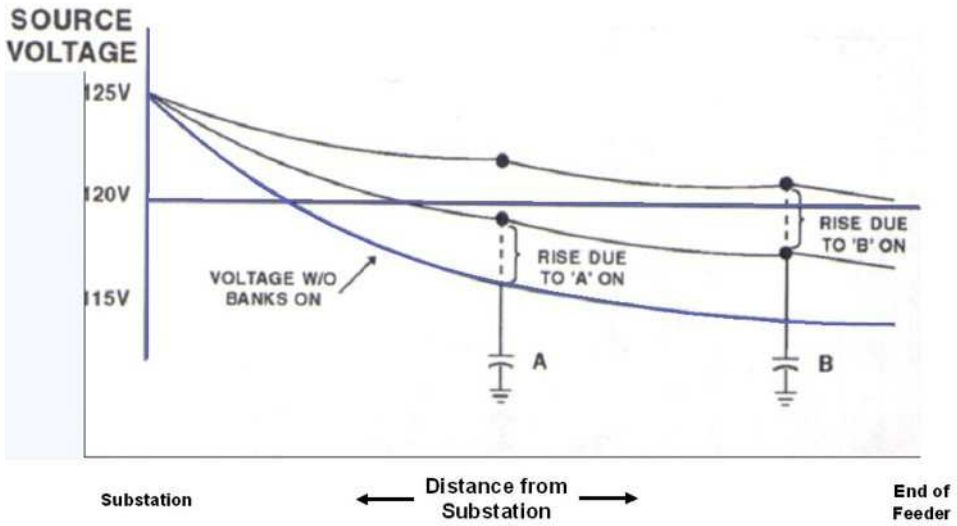

Figure 1.3: Voltage drops in a power distribution system with capacitors

have high maintenance costs, and that operations of those devices will result in disturbances of distribution systems. Two methods are used for modeling the cost of operations: 1) adding an operation cost term to the objective function; 2) imposing limits on the number of operations on those devices. This dissertation adopts both methods in chapters 3 and 4 respectively.

Another VVC scheme is Conservation Voltage Reduction (CVR) [58, 59]. This is based on load models that assume positive correlation between the power consumption and voltage levels. Thus, reducing voltage levels may reduce the power consumption. The study in [59] has shown the effectiveness of CVR in reducing the total power consumption; however, the control algorithm is largely based on heuristics and operation costs are not considered. In this dissertation, we develop mathematical models and optimization algorithms to minimize the daily power consumption in power distribution systems.

1.2 Solution Algorithms for Finite Horizon Optimal Planning Problems in Smart Grid

1.2.1 Dynamic Programming

Because we consider the accumulated cost over a finite horizon, we can formulate the optimization problem as:

$$\min_{x_t, a_t} \mathbb{E}\{\sum_{t=1}^{T} C(x_t, a_t) + C_f(x_F)\} \tag{1.1}$$

$$subject\,to \quad g(x_1, x_2, \ldots x_T, a_1, a_2, \ldots a_T) \leq 0,$$

where x_t represents the state of a power system at time t, a_t represents the action we take at time t, $C(x_t, a_t)$ represents the cost incurred at time t with state x_t and action a_t, and $C_f(x_f)$ is the final cost. The objective is to find a sequence of state-action pairs to minimize

the objective function. A brute-force method for solving this optimization problem is to enumerate all feasible sequences to find the optimal solution. The time complexity of this method is $O((|X||A|)^T)$, which makes this approach prohibitive even for some small-sized problems. Instead, we use the dynamic programming method for solving this problem.

According to the mathematical formulation above, we can model a finite horizon optimal planning problem as a finite horizon Markov Decision Process (MDP) $(\mathbb{X}, \mathbb{A}, R, P, T)$, where

- $\mathbb{X}$ is the set of all possible states, i.e., $\mathbb{X} = \{x\}$.
- $\mathbb{A}$ is the set of all actions, i,e., $\mathbb{A} = \{a\}$.
- R is the cost function, i.e, $R = C$.
- P is the transition probability from x_t to x_{t+1} given an action a_t.
- T is the planning horizon.

The objective of the finite horizon MDP is to find an optimal policy π^* in the policy space $\{\Pi : X \to A\}$, e.g. an optimal action $a^* \in A$ for each state x at time t. The standard dynamic programming algorithm can solve this problem. Let $V_t^*(x)$ be the cost accumulated from time t to T, then we have:

$$V_t^*(x) = \sum_{k=t}^{T} R(x_k, \pi^*(x_k)) + C_f(x_f).$$

The optimality Equation (1.2) and Equation (1.3) compute values of all states and the optimal policy π^* recursively.

$$V_t^*(x_t) = \min_a \mathbb{E}\{R(x_t, a) + V_{t+1}^*(x_{t+1})\}, \tag{1.2}$$

$$\pi^*(x_t) = a^* = \arg\min_a \mathbb{E}\{R(x_t, a) + V_{t+1}^*(x_{t+1})\}. \tag{1.3}$$

We apply the optimal policy at each time period, and an optimal sequence of state-action pairs is obtained. The time and space complexities are both $O(|X||A|T)$, which is a big improvement over the brute-force method with time complexity $O((|X||A|)^T)$.

The dynamic programming algorithm has been widely used for solving some optimal planning problems, including the unit commitment problem, the voltage and reactive power control problem and the feeder reconfiguration problem. In UCP, the state x can be represented by a vector $(s_1, s_2, \ldots)$, where s_i denotes the state of power generator i; the action a can be represented by another vector $(b_1, b_2, \ldots)$, where b_j denotes the operation on power generator j; the cost function can be formulated according to the operational cost function, which may include fuel cost, startup cost, etc. We can build a similar model for the VVC problem: the state x denotes the states of LTCs and SCs, action a denotes operations on LTCs and SCs, and the cost function is the power loss function or the total energy consumption function. In literature, the dynamic programming approach was proposed for solving VVC problems in [62] for solving UCPs in [24].

However, in some cases, we can only use the dynamic programming algorithm for solving small-scaled problems due to the large size of the state space X and action space A. In UCP, the state x, as mentioned previously, is represented by a vector $(s_1, s_2, \ldots)$. If we have n power generators, and each generator has m states, then $|X| = m^n$, which is exponential with respect to the number of power generators n. There are usually hundreds, or even thousands of power generators in an electric grid, so the dynamic programming approach is prohibitive in those cases. A similar problem occurs when we apply the dynamic programming algorithm for solving VVC problems. In literature, some heuristics and neural networks were used to reduce the sizes of state and action spaces. However, those methods cannot assure the optimality of solutions.

1.2.2 Stochastic Search Algorithms

We can formulate the problem in Equation (1.1) with a simpler representation in Equation (1.4) below:

$$\min_{x,y} f(x,y) \tag{1.4}$$

$$subject\,to \quad g(x,y) \leq 0,$$

where $x = (x_1, x_2, \ldots)$ is a vector of integer or discrete variables, and $y = (y_1, y_2, \ldots)$ is a vector of continuous variables. x denotes some discrete control variables in power systems, which may include tap positions of transformers, on/off states of generators, etc. y denotes continuous variables in power systems, which may include power output levels, voltage levels, etc. Because of the non-convex nature of the objective function and constraints, we can not directly apply convex programming algorithms on these problems. Instead, several well-known stochastic search algorithms may solve those problems.

Genetic Algorithm

Genetic Algorithm (GA) is an optimization algorithm that mimics the process of natural evolution, and is based on the concepts of natural selection and natural genetics. The basic structure of the algorithm is given in [22]. In GA, a number of initial candidate solutions called "population" are randomly generated. Each candidate solution is encoded into a string structure called "chromosome". Usually, each chromosome is the concatenation of binary representations of control variables, which can mutate or be altered. Chromosomes are composed of genes, which represent the values of control variables. Candidate solutions evolve towards better solutions when a sequence of genetic operators, such as parent selection, crossover and mutation, are applied, and eventually, may converge to global optimal solutions. Typically, GA follows the following steps:

1. A population of candidate solutions is initialized.

2. Fitness functions are evaluated for the population.

3. Genetic Operators are applied to generate the next generation of population.

4. If a termination condition is satisfied, then stop; otherwise, go to step 2.

In step 1, the candidate solutions are generated by random number generators. A sequence of Bernoulli random variables are generated if chromosomes consist of binary bits. In step 2, the fitness function is usually the objective function. For example, it could be the fuel cost function in UCPs or the power loss function in VVC problems and feeder reconfiguration problems. Since some candidate solutions may not satisfy all constraints, penalty terms are added to the objective function, which becomes:

$$\min_{x,y} f(x,y) + \sum_{j=1}^{Nc} \eta_j Pen_j, \tag{1.5}$$

where N_c is the number of constraints, Pen_j is the penalty for the violation of the jth constraint, and η_j is the weighting factor. In step 3, three different genetic operators are usually applied: parent selection, crossover and mutation. These operators mimic the natural evolution and selection processes. The parent selection operator stochastically selects parents for crossover from current population of candidate solutions. A candidate solution with smaller fitness function value has higher probability of being selected to pass its genes to the next generation of solutions. By applying the crossover operator, parent chromosomes are combined to produce new chromosomes that inherit their segments of genes. Different crossover schemes, such as one-point crossover, two-point crossover, uniform crossover and half uniform crossover, have been proposed for solving different kinds of problems. The mutation operator mimics the biological mutation process: one or more genes can mutate with a very small probability. Different mutation schemes are applied for different chromosome types. The genetic algorithm terminates when a termination condition is satisfied.

GA has been widely used for solving optimal planning problems in smart grid. GA is used for solving UCP in [31], the optimal power flow problem in [1], and the feeder reconfiguration problem in [61]. In UCP, the on/off status of power generators are coded into binary strings (chromosomes); power output levels are computed after solving an optimal dispatching problem; penalties are added for the violation of minimum on/off time constraints and reserve constraints. In optimal power flow problems and VVC problems, tap positions and on/off states of capacitors are coded into binary strings, while voltage levels and power flows are calculated using a set of power flow equations. Penalties are added for the violation of security constraints such as voltage constraints. In feeder reconfiguration problems, the on/off states of line breakers are coded into binary strings. To improve the performance of GA, some other genetic operators such as elitism [15] are introduced for solving some optimization problems, and better performance of GA is shown after adding those operators [1].

Particle Swarm Optimization

Particle Swarm Optimization (PSO) algorithm is also popular for solving optimal planning problems in smart grid. PSO is a stochastic search algorithm introduced in [32]. PSO mimics the movements of a bird flock searching for food. Similar to GA, PSO also randomly initializes a number of candidate solutions named particles. The position of each particle is evaluated by the fitness function (objective function). In the process of iteratively searching, PSO moves particles in the multidimensional search space according to fitness function values over coordinates and velocities of those particles. For each particle, its coordinate and velocity are dynamically adjusted according to its own flying experience and the experience of its neighbors. The coordinate and velocity of a particle is updated by using its current coordinate and velocity, its best recorded position, and the best recorded position of the flock. Let x and v denote the coordinate and velocity of a particle. Let $pBest$ be the best recorded position of that particle, and $gBest$ be the best recorded position among all particles. Let k

be the iteration number. Values of x and v are updated according to the following equations:

$$v_{k+1} = k(w \cdot v_k + \psi_1 \cdot rand() \cdot (pBest - x_k) + \psi_2 \cdot rand() \cdot (gBest - x_k) \tag{1.6}$$

$$x_{k+1} = x_k + v_{k+1}$$

where ψ_1, ψ_2, k, and w are parameters of PSO. Similar to genetic algorithm, some candidate solutions generated in PSO may violate some constraints, so we should also add penalty terms to the original objective function as in Equation (1.5). PSO is initially for solving some optimal planning problems with only continuous variables; however, some control variables in smart grid are discrete. A discrete binary PSO algorithm was proposed in [33] for solving this problem, and that algorithm was applied on UCP in [20]. A similar technique was used for solving VVC problems in [40, 75].

Simulated Annealing

Simulated Annealing (SA) [34] is a stochastic search algorithm inspired by the annealing process in metallurgy. It works by emulating the physical process that a solid reaches a "frozen structure" with a minimum energy configuration if it cools down at a very slow schedule. SA has six basic elements: 1) a finite solution space, 2) an objective function, 3) neighbors of a state, 4) a transition probability measure from a state to its neighbors, 5) a cooling schedule, 6) an initial state. Let x denote a state (candidate solution) in the search process, $f(x)$ denote the objective function, S denote the solution space, $S(x)$ denote the set of neighbors of the state x, and $P(j|i)$ be the transition probability from a state i to another state j, $j \in S(i)$. SA iteratively searches for the global optimal solution according the following steps:

1. Initialize a state x_0 and a cooling schedule $\{T_k\}$.

2. Generate a state x' from $S(x_k)$ according to the transition probability $P(x'|x)$.

3. Evaluate the objective function values $f(x_k)$ and $f(x')$.

4. If $f(x_k) > f(x')$, $x_{k+1} = x'$. Else, with probability $e^{(f(x_k)-f(x'))/T_k}$, $x_{k+1} = x'$; with probability $1 - e^{(f(x_k)-f(x'))/T_k}$, $x_{k+1} = x_k$.

5. If a stopping condition is satisfied, stop; else, let $k = k + 1$, and go to step 2.

In step 3, even though a new state SA generates is worse than the old state, the new state will still be accepted with a positive probability, which prevents the search process from trapped in local optimal states.

SA is usually used for solving problems with discrete control variables. Unlike GA and PSO, which may generate candidate solutions violating one or more constraints, SA always generate feasible candidate solutions because it assigns zero transition probability to those infeasible states. However, there are two disadvantages: 1) if the feasible regions are separated from each other, SA may fail in obtaining the global optimal solution; 2) it may be challenging to find an initial feasible solution for some problems. SA was used for solving VVC problems in [39], UCPs in [74], feeder reconfiguration problems in [11], and optimal power flow problems in [55].

Approximate Stochastic Annealing

In this dissertation, we propose a novel Approximate Stochastic Annealing (ASA) algorithm for solving finite horizon optimization problems with discrete variables. The approximate stochastic annealing algorithm is inspired by the Boltzmann distribution in statistical thermodynamics. By definition, the Boltzmann distribution for the fraction of the number of particles at a state i with energy E_i is:

$$\frac{N_i}{N} = \frac{g_i e^{-E_i/(k_B T)}}{\sum_i g_i e^{-E_i/(k_B T)}}, \tag{1.7}$$

where g_i is the degeneracy of a state i, T is the temperature and k_B is the Boltzmann constant. N_i is the number of particles at state i, and N is the total number of particles. We can observe

from Equation (1.7) that when temperature is high, particles are more evenly distributed among all states with different energy levels. However, when T is low, these particles are more concentrated on states with low energy levels. We can borrow this idea for solving discrete optimization problems. Suppose a state of a particle can be represented by a vector of discrete variables x, and the energy level can be represented by the objective function value. Thus, the optimization problem becomes the problem of finding the state with the lowest energy level. Because the number of states is very large in some problems, it may be time consuming or even infeasible to calculate the energy level of each state. An alternative is that we can assume those states are distributed according to a parameterized probability distribution, and our objective is to estimate those parameters when the temperature is low. In ASA, parameters of that probability distribution are iteratively estimated using a simulation based method. At each iteration, a number of states (x's) are sampled according to the current distribution function. Then, ASA computes energy levels (objective function values) of those states, which are used to update those parameters. We expect that as the temperature slowly decreases to zero, we will have a good estimation of that probability distribution, which assigns unit probability mass to the state with the lowest energy level. This dissertation gives details of this algorithm in chapters 3 and 4. We give the convergence analysis of ASA in chapter 5.

1.2.3 Mixed-Integer Programming

As noted in Equation (1.4), optimal planning problems in smart grid have two types of variables: discrete variables (integer variables) and continuous variables. Thus, mixed-integer programming (MIP) methods may solve those problems. The most widely used algorithm is "branch and bound (B&B)", a dynamic search algorithm that searches optimal solutions through a search tree. Typically, three operations, branching, bounding and pruning, are used in this algorithm. Let S be the solution space of integer variables. The branching process divides S into smaller sets $\{S_1, S_2, \ldots\}$, whose union covers S. Then,

the optimal solution over S is the minimum of solutions over those subsets. The splitting process is performed recursively until the solution space cannot be separated. The bounding process calculates lower and upper bounds for each node in the search tree through convex programming methods. The pruning operation reduces the size of the search tree: if the lower bound of a node A is greater than the upper bound of node B, node A and its offspring can be safely discarded. In the worst case, the B&B algorithm enumerates all the candidate solutions in the solution space of discrete variables, which leads to an exponential time complexity; however, good choices of splitting points and pruning operations can significantly reduce the number of visited nodes.

One of the most important applications of mixed-integer programming is the unit commitment problem due to the fact that UCP can be formulated as a mixed integer linear programming (MILP) problem, and that linear programming problems can be solved very quickly. Even in some cases, where quadratic objective functions are used, piecewise linear functions can approximate the quadratic objective functions very well. To improve the performance of MILP, a new algorithm called "branch and cut" was proposed in [45]. The branch and cut algorithm is implemented in the commercial optimization solver IBM ILOG Cplex [29], which was used for solving UCPs with up to 100 power generators in [10]. For some other optimal planning problem such as feeder reconfiguration and VVC, MIP is rarely used because of the nonlinear or even non-convex nature of objective functions and constraints: it takes a large amount of time to calculate upper and lower bounds of a node in the search tree. MIP was used for solving VVC problem in [40], but results were not promising in terms of both optimality and computational time.

1.2.4 Interior Points Method and Lagrangian Relaxation Method

Interior points method (IPM) and Lagrangian Relaxation (LR) Methods are both convex programming techniques. We can relax integer constraints and use these two methods for solving the relaxed problem with just continuous variables. Integer variables in the solution

of the relaxed problem are rounded to integers using some heuristics. Interior point method was used in [40] for solving the VVC problem with very promising results. Note that even integer constraints are relaxed, some problems may still be non-convex, and computational results may be local optimal solutions.

Lagrangian Relaxation method is widely used for solving UCPs. A dual problem is obtained by relaxing some constraints. The dual problem is decomposed into a number of small subproblems solved by dynamic programing. We can use a simple example of UCP below to illustrate this algorithm. A simple UCP can be formulated as:

$$\min_{x} \sum_{i=1}^{N} f(x_i) \tag{1.8}$$

$$subject\,to: \qquad x_i \in X_{i,} \quad \forall i \tag{1.9}$$

$$\sum_{i=1}^{N} a_i x_i \leq b. \tag{1.10}$$

By relaxing the constraint in (1.10), a dual problem is obtained below:

$$\max_{\lambda} q(\lambda),$$

where

$$\begin{aligned} q(\lambda) &= \min_{x} \sum_{i=1}^{N} f(x_i) + \lambda \sum_{i=1}^{N} a_i x_i - \lambda b \qquad (1.11) \\ &= \sum_{i=1}^{N} \min_{x_i} \{f(x_i) + \lambda a_i x_i)\} - \lambda b \end{aligned}$$

$$subject\,to \qquad x \in X_i \quad \forall i. \tag{1.12}$$

To find the value of $q(\lambda)$, we only need to solve N small independent sub-problems: $\min_{x_i} f(x_i) + \lambda a_i x_i$. In UCP, these small sub-problems can be solved by dynamic programming with very

little computational time. Additionally, because these sub-problems are independent from each other, we can use parallel computing to solve each of them simultaneously to save computational time. This dissertation discusses details of the parallel computing scheme and the scalability in chapter 2.

1.3 Overview of This Dissertation

The rest of this dissertation is organized as follows:

In chapter 2, we propose a parallel computing scheme for solving large-scale unit commitment problem with complex security constraints and wind power penetration based on the traditional lagrangian relaxation algorithm. The UCP in this work consists of hundreds of power generators and a variety of reserve constraints and transmission constraints. We use a probabilistic UCP model and a rolling horizon scheme to handle the volatile wind power generation. We test the performance of our algorithm and the computing scheme in a simulated unit commitment process in New York Control Area. The performance and scalability of parallel computing are tested on an IBM blue gene supercomputer.

In chapter 3, we propose an Approximate Stochastic Annealing algorithm for solving voltage and reactive power control problems for minimizing power losses in power distribution systems. Voltage constraints and operations limits constraints on LTCs and SCs are considered. We also propose an algorithm that combine lagrangian relaxation and dynamic programming for solving the same problem and to test the performance of the ASA algorithm.

In chapter 3, we propose two additional VVC models. We consider operation costs of LTCs and SCs and ZIP load models in these two VVC models. One of them minimizes power losses and the other minimizes the total energy consumption. We use ASA for solving both models. We use the computational results of the dynamic programming algorithm and the simulated annealing algorithm to illustrate the performance of the ASA algorithm.

In chapter 5, we discuss the convergence properties of the ASA algorithm.

Chapter 2

Large-Scale Security Constrained Unit Commitment Problem

Nomenclature

$p_{m,i_m,t}$	electricity output level of generator i_m in zone m at time period t.
$z_{m,i_m,t}$	binary variable that is 1 if generator $i_m$in zone m is on during time period t; 0 otherwise.
$F_{m,i_m,t}$	fuel cost function of generator i_m in zone m at time period t.
$S_{m,i_m,t}$	startup cost function of generator i_m in zone m at time period t.
$R_{m,i_m,t}$	reserve service cost function of generator i_m in zone m at time period t.
$r10s_{m,i_m,t}$	10-minute spinning reserve level of generator i in zone m at time period t.
$r10ns_{m,i_m,t}$	10-minute non-synchronous reserve service level of generator i_m in zone m at time period t.

$r30s_{m,i_m,t}$	30-minute spinning reserve level of generator i in zone m at time period t.
$r30ns_{m,i_m,t}$	30-minute non-synchronous reserve service level of generator i in zone m at time period t.
$r_{m,i_m,t}$	reserve service vector defined as ($r10s_{m,i_m,t}$, $r10ns_{m,i_m,t}$, $r30s_{m,i_m,t}$, $r30ns_{m,i_m,t}$).
$CReg_{m,i_m,t}$	regulation cost function of generator i_m in zone m at time period t.
$reg_{m,i_m,t}$	regulation service level of generator i_m in zone m at time period t.
$d_{t,m}$	prediction of electricity demand of time period t in zone m.
$w_{t,m}$	prediction of wind power of time period t in zone m.
Res_{10s}	10-minute spinning reserve requirement for the whole ISO control area.
Res_{10t}	10-minute total reserve requirement for sub control area j at time t.
Λ_j	super-zone j or the jth collection of zones.
$ResLB_{j,10s,t}$	10-minute spinning reserve requirement for sub control area j at time t.
$ResLB_{j,10t}$	10-minute total reserve requirement for sub control area j at time t.
$p_{m,i_m,max}$	maximum output when generator i_m in zone m is on.
$p_{m,i_m,min}$	minimum output when generator i_m in zone m is on.
$\Gamma_{l,m}$	line flow distribution factor for the transmission line l due to the net power injection of zone m
$Tran_{i,max}$	maximum transmission capacity of transmission line l in designate direction.

$Ms_{i_m,m}$ maximum number of times that generator $i_m$in zone m is allowed to be shut down.

2.1 Introduction

The goal of unit commitment problems is to find the optimal production schedule for power generation units and the production level of each unit over a short term period in order to minimize the operational cost of the power grid [12]. To maintain the security of the electric grid, a variety of security constraints such as reserve constraints and transmission constraints, are enforced, and the resulting problem is usually called Security Constrained Unit Commitment (SCUC) problem. In New York State, UCP is solved by New York Independent System Operator (NYISO) in the day-ahead power market based on the generation and ancillary service bids, which give generation and ancillary service cost of each power generator, from Independent Power Producers (IPP), Loads bid from Load Service Entities (LSE) and Security Constraints set by NYISO and other power regulation authorities. Because of the importance of UCPs, it has been studied broadly and intensively, and many methods have been proposed in literature and used in practice [53].

Depending on the system configuration of a power grid, different optimization objectives and security constraints are considered. For the basic UCP formulation, the objective is simply to minimize the power generation cost subject to the electricity demand. However, as the liberalization of the electricity markets and advancement of optimization techniques, more and more elements are introduced. In [60], a security constrained unit commitment problem (SCUC) with transmission constraints was tackled using a lagrangian relaxation approach, where the transmission and reserve constraints were relaxed to form a dual problem and subsequently solved using subgradient methods. The test result shows that the proposed direct method can reduce the generation cost over the indirect method that does not consider transmission constraint in the dual optimization process. The algorithm was

improved in [65] to address the feasibility issue, and a unit de-commitment step was added to achieve a better solution. The AC constraints were considered in [44], and Bender's decomposition technique was used for solving the problem. Furthermore, ancillary services has been gradually introduced into the unit commitment process. In [37], Z. Li and M. Shahidehpour used Lagrangian Relaxation technique to solve the security constrained UCP with ancillary service constraints and costs; moreover, they also calculated the market clearing price of both generation and ancillary costs. Their work is important because there is a conflict between generation service and ancillary service when a generator is turned on. Additionally, some environmental elements such as carbon tax were introduced to UCP in the past two years [4]. Because of its complexity, it is unusual for ISOs to solve the problem in a single step. Instead, a multi-step approach is often adopted. For example, in the New York State, different constraints are added at different steps to decrease the complexity of the problem [46]; however, this will decrease the solution quality. Therefore, how to solve a SCUC problem in a single step with a certain time limit becomes a challenging issue.

The presence of renewable energy sources such as wind power can further increase the complexity of UCP, and a common method to handle this is to use the scenario tree technique [64] to simulate uncertainties and dynamics of wind power. However, to make the problem computationally tractable, only a very limited number of scenarios can be used. Many research projects proposed to use a rolling horizon optimization scheme rather than the traditional day-ahead scheme; see for example, the Wilmar project [73][67][68]. An alternative method is to use a probability measure to set up a probability level to limit the probability of power outage within the prescribed threshold [52]. To meet these probability requirements, one needs to set the operation reserve based on the variability of wind power. A rolling horizon approach can also be used to dynamically locate the operation reserve when new wind forecast information is available. Note that the rolling horizon approach is more computationally demanding than the traditional day-ahead scheduling process, as decisions need to be made at each time slot in an online manner and every decision requires solving a

nonlinear optimization problem involving both continuous and discrete decision variables.

In this chapter, an improved Lagrangian Relaxation (LR) method, which is adapted for parallel computing, is proposed to solve a large scale SCUC problem. Because linear generation cost functions are used, a greedy algorithm is proposed to optimize generation and ancillary services when a generator is on. By using the proposed algorithm, we expect to solve the large-scale SCUC in a single step and dramatically reduce the computational time. A system based on the power system of the New York Control Area (NYCA) is simulated to test the significance of our algorithm.

To address the variability of wind power, we follow the idea of [52] and use a probabilistic reserve constraint to describe the uncertainty of wind power. Since the wind power forecast is more accurate over shorter time periods [69][67], the probabilistic reserve constraints method is combined with a rolling horizon scheme to dynamically update the reserve constraints when more accurate wind forecast becomes available. the computational time issue is addressed by implementing our proposed solver on a parallel computing facility, and research results show that parallel computing has the potential to satisfy the computational speed requirement of the Rolling Horizon Scheme.

The organization of this chapter is as follows. A security constrained unit commit model is formulated in Section 2.2, and a solution algorithm is given in Section 2.3. Section 2.4 gives the probabilistic formulation of reserve constraints and the method used to handle these constraints. In Section 2.5, we provide a case study to illustrate the performance of our algorithms.

2.2 Formulation of the SCUC Model

This chapter formulates an SCUC model based on the realistic problem solved in New York State. Both generation service and ancillary services, including reserve services and regulations services, are optimized to minimize the total operational costs. Realistic secu-

rity constraints, including balance constraints, ancillary service requirement, load capacity constraint, transmission constraints, etc, are considered in this research. Unlike some other research, which is based on benchmark problems with up to 100 generators and limited security constraints, this work is trying to solve large-scale problem with more than 600 power generators and realistic security constraints enforced in daily power planning process in New York States. We provide the detailed formulation in the rest of this section below.

2.2.1 Objective Function

This dissertation optimizes the total operational cost, including both power supply cost and ancillary services cost. The power supply cost includes power generation cost and start-up cost. The ancillary service cost includes reserve service cost and regulation service cost. Moreover, reserve service is divided into spinning service and non-synchronous service. The formulation is given in (2.1).

$$\begin{aligned} cost \;=\; & \sum_{m=1}^{M}\sum_{i_m=1}^{I_m}\sum_{t=1}^{T}(F_{m,i_m,t}(p_{m,i_m,t}) + S_{m,i_m,t}(z_{m,i_m,t})) \\ & + \sum_{m=1}^{M}\sum_{i_m=1}^{I_m}\sum_{t=1}^{T}(R_{m,i_m,t}(r_{m,i_m,t})) \\ & + \sum_{m=1}^{M}\sum_{i_m=1}^{I_m}\sum_{t=1}^{T}(CReg_{m,i_m,t}(reg_{m,i_m,t})), \end{aligned} \tag{2.1}$$

where

$$r_{m,i_m,t} = (r10s_{m,i_m,t},\ r10ns_{m,i_m,t},\ r30s_{m,i_m,t},\ r30ns_{m,i_m,t}).$$

The first line in Equation (2.1) includes power generation function, which is a piecewise linear function, and a startup function, which is a stepwise linear function. The second line includes the reserve cost function, which is a linear function with respect to the 10-minute

and 30-minute spinning reserves and non-synchronous reserves. The third line is the linear regulation cost function. Note that a generator can provide spinning reserve or regulation services only when it is turned on, and provide non-synchronous reserve service only when it is turned off.

2.2.2 Load Balance Constraints

In this dissertation, we assume that wind power can be generated and integrated at no additional cost, and that there is no wind curtailment. Thus, wind power will always be delivered to customers. Then, wind power can be considered as negative load in this research. Here we define the term "net load" as the difference between the electricity demand and the predicted wind power, i.e., electricity load that needs to be supplied by traditional generators, including steam generators, gas turbines and hydro power generators. In the day-ahead planning, hydro and thermal plants should meet the sum of net loads of certain control areas. The mathematical formulation is given in Equation (2.2).

$$\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} p_{i,i_m,t} = \sum_{m=1}^{M}(d_{m,t} - w_{m,t}),\ \forall t \tag{2.2}$$

2.2.3 Ancillary Service Requirements

For the entire control area, there are three kinds of reserve requirements: 10-minute spinning reserve, 10-minute total reserve, and 30-minute total reserve requirements. The mathematical formulations are given in (2.3), (2.4), and (2.5), respectively.

$$\sum_{m=1}^{M}\sum_{i_m=1}^{I_M} r10s_{m,i_m,t} \geq Res_{10spin},\ \forall t \tag{2.3}$$

$$\sum_{m=1}^{M}\sum_{i_m=1}^{I_M} (r10s_{m,i_m,t} + r10ns_{m,i_m,t}) \geq Res_{10t},\ \forall t \tag{2.4}$$

$$\sum_{m=1}^{M}\sum_{i_m=1}^{I_M}(r30s_{m,i_m,t}+r30ns_{m,i_m,t})\geq Res_{30t},\ \forall t \tag{2.5}$$

In real power market, a control area is often divided into several individual areas, which are usually called "zones". And for certain collection of zones, there are several location based reserve constraints. Letting Λ_j be the jth collection of zones, the location based reserve constraints are given by (2.6), (2.7), and (2.8).

$$\sum_{m\in\Lambda_j}\sum_{i_m=1}^{I_M} r10s_{m,i_m,t}\geq ResLB_{j,10spin},\ \forall t \tag{2.6}$$

$$\sum_{m\in\Lambda_j}^{M}\sum_{i_m=1}^{I_M}(r10s_{m,i_m,t}+r10ns_{m,i_m,t})\geq ResLB_{j,10t},\ \forall t \tag{2.7}$$

$$\sum_{m\in\Lambda_j}^{M}\sum_{i_m=1}^{I_M}(r30s_{m,i_m,t}+r30ns_{m,i_m,t})\geq ResLB_{j,30t},\ \forall t \tag{2.8}$$

In a control area operated by an ISO, those collection of zones are called "super-zones". Equations (2.6), (2.7), and (2.8) should be satisfied for all those super-zones.

Additionally, due the fluctuation of power demand and wind power, ISOs has certain requirements for regulation services, which are expressed in (2.9).

$$\sum_{m=1}^{M}\sum_{i_m=1}^{I_M} r10s_{m,i_m,t}\geq Reg_t,\ \forall t \tag{2.9}$$

2.2.4 Transmission Constraints

This dissertation also considers transmission constraints between different zones. The modeling of transmission constraints follows the method used in [65], and is given in (2.10).

$$\sum_{m-1}^{M}\Gamma_{l,m}(\sum_{i_m=1}^{I_m} p_{m,i_m,t}+w_{m,t}-d_{m,t})\leq Tram_{l,max},\ \forall l,t \tag{2.10}$$

where $\Gamma_{l,m}$ is the line flow distribution factor for the transmission line l due to the net power injection of zone m.

2.2.5 Single Generator Capacity Constraints

A generator that has been turned on may provide generation and reserve simultaneously. In practice, the sum of these services should be within the maximum power output limit. These requirements are given in (2.11), (2.12), (2.13), and (2.14). When a generator is off-line, it might provide non-synchronous reserve services, the sum of which should also be within the maximum output limit. The formulations are given in (2.16), (2.17), and (2.18). In addition, the amount of regulation service should also be within a certain limit, which is given in Equation (2.15).

$$p_{m,i_m,t} + r10s_{m,i_m,t} + r30s_{m,i_m,t} \leq p_{m,i_m,max} z_{m,i_m,t},\ \forall m, i_m, t \tag{2.11}$$

$$p_{m,i_m,min} z_{m,i_m,t} \leq p_{m,i_m,t} \leq p_{m,i_m,max} z_{m,i_m t},\ \forall m, i_m, t \tag{2.12}$$

$$r10s_{m,i_m,t} \leq r10s_{m,i_m,max} z_{m,i_m,t},\ \forall m, i_m, t \tag{2.13}$$

$$r30s_{m,i_m,t} \leq r30s_{m,i_m,max} z_{m,i_m,t},\ \forall m, i_m, t \tag{2.14}$$

$$regs_{m,i_m,t} \leq regs_{m,i_m,max} z_{m,i_m,t},\ \forall m, i_m, t \tag{2.15}$$

$$r10ns_{m,i_m,t} + r30ns_{m,i_m,t} \leq p_{m,i_m,max}(1 - z_{m,i_m,t}),\ \forall m, i_m, t \tag{2.16}$$

$$r10s_{m,i_m,t} \leq r10s_{m,i_m,max}(1 - z_{m,i_m,t}),\ \forall m, i_m, t \tag{2.17}$$

$$r30s_{m,i_m,t} \leq r30s_{m,i_m,max}(1 - z_{m,i_m,t}),\ \forall m, i_m, t \tag{2.18}$$

2.2.6 Other Constraints

Some other constraints such as minimum up time and down time constraints are also considered in our model. The detailed formulation was given in [7] and [10]. In addition, the maximum number of stops will be considered in our model, which means that a generator can only be turned off for a limited number of times on a single day. This constraint has never be considered in previous studies.

2.3 Solution Algorithm

As proposed in [65], a direct method is used for solving the SCUC problem. Compared with the work in [60], we introduce the ancillary service costs in the objective function, and add additional single generator capacity constraints to the model. Moreover, a parallel computing scheme is developed to enhance the computational speed.

2.3.1 Lagrangian Relaxation Algorithm

To solve this problem, Lagrangian Relaxation method is adopted to relax the demand, reserve, transmission, and regulations constraints. A dual problem is thus obtained, and its objective function is given in Equation (2.19).

$$
\begin{aligned}
Dual\,Cost &= Cost + \sum_{t=1}^{T}\{\lambda_{d,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} p_{m,i_m,t} - \sum_{m=1}^{M}(d_{m,t} - w_{m,t})] \\
&+ \lambda_{10s,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} r10s_{m,i_m,t} - Res_{10spin}] \\
&+ \lambda_{10t,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m}(r10s_{m,i_m,t} + r10ns_{m,i_m,t}) - Res_{10t}] \\
&+ \lambda_{30t,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m}(r30s_{m,i_m,t} + r30ns_{m,i_m,t}) - Res_{30t}] \\
&+ \sum_{j=1}^{J}\{\lambda_{j,10s,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} r10s_{m,i_m,t}\mathcal{I}(m \in \Lambda_j) - Res_{j,10spin}] \\
&+ \lambda_{j,10t,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m}(r10s_{m,i_m,t} + r10ns_{m,i_m,t})\mathcal{I}(m \in \Lambda_j) - Res_{j,10t}] \\
&+ \lambda_{j,30t,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m}(r30s_{m,i_m,t} + r30ns_{m,i_m,t})\mathcal{I}(m \in \Lambda_j) - Res_{t,10t}]\} \\
&+ \lambda_{reg,t}[\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} regs - Reg_{10spin}] \\
&+ \sum_{t=1}^{T}\sum_{l=1}^{L}\{\lambda_{tran,l,t}[Tran_{l,max} - \sum_{m=1}^{M}\Gamma_{l,m}(\sum_{i_m=1}^{I_m} p_{m,i_m,t} + w_{m,t} - d_{m,t})]\} \quad (2.19)
\end{aligned}
$$

where "$Cost$" equals the cost function in Equation (2.1). The second line of Equation (2.19) is due to the relaxation of demand constraints; lines $3-7$ are due to the relaxation of reserve constraints of the whole control area, while lines $8-13$ are due to the relaxation of location reserve constraints, line 14 is due the relaxation or regulation constraints, and lines $15-16$ are the relaxation of transmission constraints. $\lambda_{d,t}$, $\lambda_{10s,t}$, $\lambda_{10t,t}$, $\lambda_{30t,t}$,$\lambda_{j,10s,t}$, $\lambda_{j,10t,t}$, $\lambda_{j,30t,t}$, $\lambda_{reg,t}$, and $\lambda_{tran,l,t}$ are the corresponding lagrangian multipliers. For simplicity, we define $\lambda_t = \{\lambda_{d,t}, \lambda_{10s,t}, \lambda_{10t,t}, \lambda_{30t,t}, \lambda_{j,10s,t}, \lambda_{j,10t,t}, \lambda_{j,30t,t}, \lambda_{reg,t}\}$. The dual problem is to find $\max_{\lambda_t}[\min(Dual\,Cost)]$. A single generator problem is defined in Equation (2.20).

$$
\begin{aligned}
DualCost_{m,i_m} &= \sum_{t=1}^{T}\{F_{i,i_m,t}(P_{m,i_m,t}) + S_{m,i_m,t}(z_{m,i_m,t}) \\
&+ R_{m,i_m,t}(r_{m,i_m,t}) + CReg_{m,i_m,t}(reg_{m,i_m,t}) - p_{m,i_m,t}(\lambda_{d,t} + \sum_{l=1}^{L}\lambda_{tran,l,t}\Gamma_{l,m}) \\
&- r10s_{m,i_m,t}[\lambda_{10s,t} + \lambda_{10t,t} + \sum_{j=1}^{J}\mathcal{I}(m \in \Lambda_j)(\lambda_{j,10s,t} + \lambda_{j,10t,t}) \\
&- r10ns_{m,i_m,t}[\lambda_{10t,t} + \sum_{j=1}^{J}\mathcal{I}(m \in \Lambda_j)\lambda_{j,10s,t}] \\
&- r30s_{m,i_m,t}[\lambda_{30t,t} + \sum_{j=1}^{J}\mathcal{I}(m \in \Lambda_j)\lambda_{j,30t,t}] \\
&- r30ns_{m,i_m,t}[\lambda_{30t,t} + \sum_{j=1}^{J}\mathcal{I}(m \in \Lambda_j)\lambda_{j,30t,t}] \\
&- reg_{m,i_m,t}\lambda_{reg,t} \qquad (2.20)
\end{aligned}
$$

Then the dual cost can be express as follows.

$$
Dual\,Cost = \sum_{m=1}^{M}\sum_{i_m=1}^{I_m} DualCost_{m,i_m} + Extra \qquad (2.21)
$$

where $Extra$ is the difference between Equations (2.19) and (2.21). We observe that the term $Extra$ does not depend on the status of power generators, and is a constant if the values of multipliers are given.

A subgradient method is used for solving the dual problem. The value of λ_t is initialized first, and its value are used for the minimization of the dual cost function. Because the term $Extra$ is a constant, we just need to minimize the term $DualCost_{m,i_m}$ individually. Dynamic programming is used for solving the single generator problem. To calculate the one-step reward, we need to optimize the allocation of generation services and ancillary services when a generator is on or off. When a generator is on, it could provide generation services, spinning reserve services, and regulation services. The one-step cost optimization

problem in (2.22) should be solved subject to constraints (2.11), (2.13), (2.14), and (2.15). On the other hand, when a generator is off, it could provide non-synchronous reserve services. Another one-step cost optimization problem in (2.23) should be solved subject to constraints (2.17) and (2.18).

$$\begin{aligned}
&\min\{F(p_{m,i_m,t}) + R_{m,i_m,t}(r_{m,i_m,t}) \\
&+ CReg_{m,i_m,t}(reg_{m,i_m,t}) - p_{m,i_m,t}(\lambda_{d,t} + \sum_{l=1}^{L} \lambda_{tran,l,t}\Gamma_{l,m}) \\
&- r10s_{m,i_m,t}[\lambda_{10s,t} + \lambda_{10t,t} + \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)(\lambda_{j,10s,t} + \lambda_{j,10t,t})] \\
&- r30ns_{m,i_m,t}[\lambda_{30t,t} + \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)\lambda_{j,30t,t}] - reg_{m,i_m,t}\lambda_{reg,t}\}
\end{aligned} \tag{2.22}$$

$$\begin{aligned}
&\min\{R_{m,i_m,t}(r_{m,i_m,t}) - r10ns_{m,i_m,t}[\lambda_{10t,t} + \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)\lambda_{j,10s,t}] \\
&- r30ns_{m,i_m,t}[\lambda_{30t,t} + \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)\lambda_{j,30t,t}]\}
\end{aligned} \tag{2.23}$$

General linear programming techniques can be used for solving these two problems; however, it is time consuming. Because this dissertation uses piecewise generation cost functions and linear ancillary service cost functions, we propose to use a greedy algorithm for solving these problems, which can significantly reduce the computational time. Let

$$F(p) = b_0 + \begin{cases} b_1 p & a_0 \le p < a_1 \\ (b_1 - b_2)a_1 + b_2 p & a_1 \le p < a_2 \\ \ldots & \\ \sum_{k=2}^{K}(b_{k-1} - b_K)a_{k-1} + b_k p & a_{K-1} \le p \le a_K, \end{cases},$$

where we have $b_1 < b_2 < ... < b_k$ and $0 = a_0 < a_1 < ... < a_K$. We can transform it into another formula

$$F(p) = G(x_1, x_2, ..., x_K) = b_0 + \sum_{k=k}^{K} b_k x_k,$$

where $p = \sum_{k=1}^{K} x_k$, $0 \leq x_1 \leq a_1 - a_0$ and $0 \leq x_k \leq (a_k - a_{k-1}) \cdot \mathcal{I}(x_{k-1} = a_{k-1} - a_{k-2}),\ k \geq 2$. The ancillary service costs are formulated as below:

$$\begin{aligned} R(r) &= rc10s \cdot r10s + rc10ns \cdot r10ns \\ &\quad + rc30s \cdot r30s + rc30ns \cdot r30ns, \end{aligned}$$

$$CReg(reg) = creg \cdot reg,$$

where $rc10s$, $rc10ns$, $rc30s$, $rc30ns$, and $creg$ are constant cost coefficients. Then Equation (2.22) can be equivalently written as follows:

$$\min F'_{m,i_m,t}(p_{m,i_m,t}) + R'_{m,i_m,t}(r_{m,i_m,t}) + CReg'_{m,i_m,t}(reg_{m,i_m,t}), \tag{2.24}$$

where

$$\begin{aligned} F'_{m,i_m,t} \quad (\quad p_{m,i_m,t}) &= G'(x_1, x_2, ..., x_K) \\ &= b_0 + \sum_{k=k}^{K} (b_k - \lambda_{d,t} - \sum_{l=1}^{L} \lambda_{tran,l,t} \Gamma_{l,m}) x_k, \end{aligned}$$

$$\begin{aligned} R'_{m,i_m,t}(r_{m,i_m,t}) &= [rc10s_{m,i_m,t} - \lambda_{10s,t} - \lambda_{10t,t} - \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)(\lambda_{j,10s,t} + \lambda_{j,10t,t})] \cdot r10s_{m,i_m,t} \\ &+ [rc30s_{m,i_m,t} - \lambda_{30t,t} - \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)\lambda_{j,30t,t})] \cdot r30s_{m,i_m,t}, \end{aligned}$$

$$CReg'_{m,i_m,t}(reg_{m,i_m,t}) = (creg_{m,i_m,t} - \lambda_{reg,t}) \cdot reg_{m,i_m,t},$$

and Equation (2.23) is equivalent to (2.25):

$$\min R'_{m,i_m,t}(r_{m,i_m,t}), \tag{2.25}$$

where

$$\begin{aligned} R'_{m,i_m,t}(r_{m,i_m,t}) &= [rc10ns_{m,i_m,t} - \lambda_{10t,t} - \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)\lambda_{j,10t,t}] \cdot r10ns_{m,i_m,t} \\ &\quad + [rc30ns_{m,i_m,t} - \lambda_{30t,t} - \sum_{j=1}^{J} \mathcal{I}(m \in \Lambda_j)\lambda_{j,30t,t})] \cdot r30ns_{m,i_m,t} \end{aligned}$$

When a generator is "*on*", the greedy algorithm works as follows:

1. Initialize the power generation and ancillary service levels to 0.
2. Sort the linear cost coefficients, including those in generation cost function F' and ancillary cost functions R' and $CReg'$, in Equation (2.24), to form a non-decreasing list $\{c_h\}$, where $h \in \{1, 2, ..., K+3\}$.
3. Let $p_{m,i_m,t} = p_{m,i_m,min}$.
4. From $h = 1$ to $h = K + 3$, consider the following cases:
   (a) c_h is a generation cost coefficient; if $c_h > 0$, stop; else, c_h should be the generation cost coefficient of the kth segment and

   $$a_k + r10s_{m,i_m,t} + r30s_{m,i_m,t} \leq p_{m,i_m,max},$$

   then $p_{m,i_m,t} = \max(a_k, p_{m,i_m,min})$.
   (b) c_h is a 10-minute spinning reserve cost coefficient;

if $c_h > 0$, stop; else,

$$r10s_{m,i_m,t} = min(r10s_{m,i_m,max}, p_{m,i_m,max} - p_{m,i_m,t} - r30s_{m,i_m,t}).$$

(c) c_h is a 30-minute spinning reserve cost coefficient;

if $c_h > 0$, stop; else

$$r30s_{m,i_m,t} = min(r30s_{m,i_m,max}, p_{m,i_m,max} - p_{m,i_m,t} - r10s_{m,i_m,t}).$$

(d) c_h is a regulation cost coefficient;

if $c_h > 0$, stop; else, $r30s_{m,i_m,t} = reg_{m,i_m,max}$.

When a generator is "off", a similar algorithm can be applied:

1. Initialize ancillary service levels to 0.

2. Sort the linear cost coefficients of ancillary cost function R' to form a non-decreasing list $\{c_1, c_2\}$.

3. For $h = 1\, or\, 2$, consider the following cases:

   (a) c_h is a 10-minute non-synchronous reserve cost coefficient;

   if $c_h > 0$, stop;

   else,

   $$r10s_{m,i_m,t} = min(r10ns_{m,i_m,max}, p_{m,i_m,max} - r30ns_{m,i_m,t}).$$

   (b) c_h is a 30-minute non-synchronous reserve cost coefficient;

   if $c_h > 0$, stop;

   else,

   $$r30s_{m,i_m,t} = min(r30ns_{m,i_m,max}, p_{m,i_m,max} - r10ns_{m,i_m,t}).$$

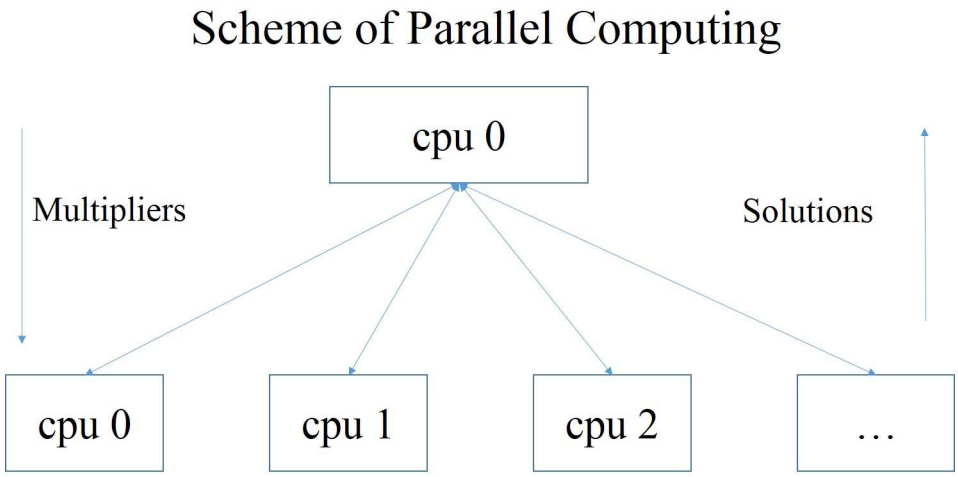

Figure 2.1: Parallel Computing Scheme to Solve UCP

2.3.2 A Parallel Computing Scheme

In the LR algorithm, the dual problem is decomposed into identical single generator problems. Therefore, it is natural to assign those problems to individual CPU's and solve them simultaneously. In every iteration, the root CPU will "broadcast" the values of lagrangian multipliers to the branch CPU's, where the single generator sub-problems are solved, and the solutions are "collected" to the root CPU to update the value of the multipliers. The computing scheme is illustrated in figure 2.3.2. Suppose there are N power generators, then the computational time required to solve the dual problem can be estimated by Equation (2.26) if the computational load is equally distributed to each branch CPU.

$$t_{dual} \approx (\frac{N}{\#\,of\,CPUs} \times t_{single} + t_{update} + t_{comm}) \times n_{iteration} \tag{2.26}$$

where t_{dual} is the computational time needed to solve the dual problem, t_{single} is the computational time for solving a single generator problem, t_{comm} is the communication time between the root CPU and branch CPU's, and $n_{iteration}$ is the number of iterations in the subgradient search process.

2.4 Probabilistic Unit Commitment Model and Rolling Horizon (RH) Scheme

For simplicity, we only consider the probabilistic reserve constraints. Without loss of generality, we can replace the reserve constraint in Section 2.2.3 with the following constraint (2.27).

$$\mathcal{P}\{\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} p_{m,i_m,t} + \sum_{m=1}^{M}\sum_{i_m=1}^{I_m} res_{m,i_m,t} \geq \sum_{m=1}^{M} d_{m,t} - \sum_{m=1}^{M} w_m, \forall t\} \geq 1 - \alpha, \quad (2.27)$$

where $res_{m,i_m,t}$ is the general reserve service level. Using Bonferroni's inequality, we can transform Equation (2.27) to another Equation (2.28).

$$\mathcal{P}\{\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} p_{m,i_m,t} + \sum_{m=1}^{M}\sum_{i_m=1}^{I_m} res_{m,i_m,t} \geq \sum_{m=1}^{M} d_m - \sum_{m=1}^{M} w_{m,t},\} > 1 - \frac{\alpha}{T} \quad (2.28)$$

Here, we assume that $w_{m,t}$ follows a normal distribution $N(\mu^w_{m,t}, (\sigma^w_{m,t})^2)$. Thus, the above Equation can be written as (2.29).

$$\sum_{m=1}^{M}\sum_{i_m=1}^{I_m} p_{m,i_m,t} + \sum_{m=1}^{M}\sum_{i_m=1}^{I_m} res_{m,i_m,t} \geq \sum_{m=1}^{M} d_m - \sum_{m=1}^{M} \mu^w_{m,t} + z_{1-\frac{\alpha}{T}} \sum_{m=1}^{M} \sigma^w_{m,t} \quad (2.29)$$

Equation (2.29) may not accurately describe the reserve requirement, because the wind power in different zones are in general correlated. Nevertheless, for simplicity we assume that they are independent. This reserve constraints is very similar to the constraints in Section 2.2.3, and a similar LR algorithm can be applied to solve the problem.

If we assume that wind power forecasts are updated every hour, we can update Equations (2.2) and (2.29) and solve the corresponding UCP on a hourly basis. The RH scheme considers the updated wind power information in both unit commitment and economic dispatch

processes, while in traditional day-ahead scheme, the updated wind power information is only considered in economic dispatch process. Thus, by involving more accurate information in the optimization process, we expect to get better solutions with decreased operational costs.

2.5 Case Studies

2.5.1 New York Control Area

To study the effectiveness of our approach on large-scale problems, we simulated a SCUC problem based on the characteristics of New York State Control Area. NYCA is divided into 11 sub-zones with transmission interface between adjacent sub-zones. The detailed zone map is given in figure 2.5.1. One feature of power grid in New York State is that most of the electricity demand comes from the southeast area of New York state, i.e., Long Island and New York City, while a large portion of the power resources are located in the west and north parts of the state. Additionally, in the near future, most of the power farms will be located in zones $A - E$ [47], and will bring more burden to transmission lines. This uneven distribution of power generation sources and power demand makes transmission constrained unit commitment an important problem in NYCA. Moreover, locational reserve requirements are enforced to maintain the safety operation of the power grid.

We follow the practice of NYISO and divide NYCA into two super-zones, where west super-zones include load zones $A - E$ and east super-zones include zones $F - G$. Additional reserve requirements are enforced for east super-zones. In addition, similar reserve requirements are also enforced on zone K, which is Long Island. The reserve requirements can be formulated in the forms of Equations (2.6), (2.7), and (2.8). The transmission constraints are formulated in the form of Equation (2.10).

In accordance with the day-ahead power market in New York State [48], piecewise linear generation cost functions and stepwise startup cost functions are used. Each generation cost

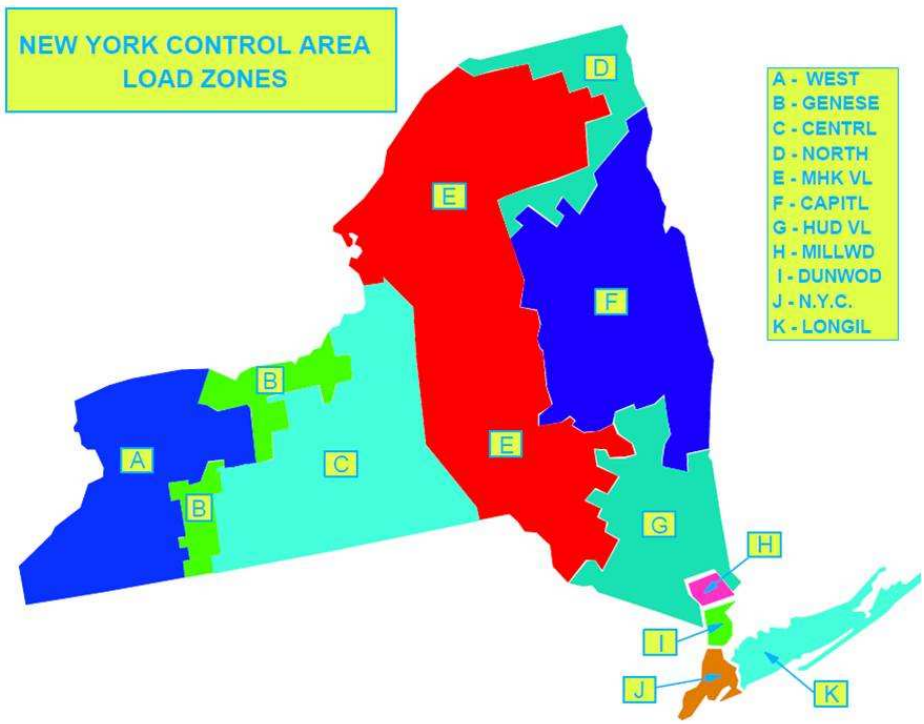

Figure 2.2: New York State Control Area

function can have up to 12 pieces. A total of 641 power generators, including nuclear plants, hydro plants, steam plants, and gas turbines, are simulated in this work. The net load for each zone is calculated by subtracting the forecasted wind power from the forecasted electricity demands. Four different wind penetration level cases are used: $1275MW$, $4250MW$, $6000MW$, and $8000MW$. According to the penetration level, the regulation requirement is adjusted as proposed in [47]. A single day (24 hour period) in August is used for the study. Because currently, the report [47] by NYISO indicates that the current reserve level is enough for the $8000MW$ penetration of wind power, so we will not consider the probabilistic reserve level management in this case.

The LR algorithm was coded in C++ and implemented on New York Blue Gene, a distributed-memory supercomputing cluster. Up to 50 nodes were used, while each node has two $700MHz$ PowerPC processors and $1G$ memory. Figure 2.3 shows the computational time required to execute the algorithm v.s. the number of CPUs used. The minimum compu-

Penetration Level (MW)	Operation Costs
1275	-1.12×10^7
4250	-1.23×10^7
6000	-1.27×10^7
8000	-1.30×10^7

Table 2.1: Total Operation Costs for different penetration level of wind power

tational time is around 180 seconds, which is much less than the 1600 seconds computational time when 1 CPU is used. The total operation costs, which is the sum of generation costs and ancillary costs, are given in table 2.1. It is interesting to note that the costs are all negative. This is reasonable because some IPPs want to assure that some plants will be selected for generation, for example, nuclear plants and some coal steam plants. Because whatever they bid for generation, they will be paid by the positive market clearing price. Thus, they have the incentive too keep those cheap power sources online. From table 2.1, it is obvious that high penetration of wind power will save money for the New York control area. The plots for the marginal regulation cost are given in figure 2.4. Because the increment in regulation requirements, the marginal cost for regulation generally increase. Because of the location-based reserve services, different sub-zones might have different marginal reserve cost services. Besides, we note that the lagrangian multiplier for transmission constraints increases as the penetration level of wind energy grows. This is reasonable for New York state because most of the wind power resources in NYCA are located in north parts while most of electricity consumption are located in southeast regions. Thus, the increased power penetration will bring more pressure on transmission lines in New York State.

2.5.2 Rolling Horizon Study

The proposed method is applied for solving large sized problems based on the ten-unit system of [31], which has been repeated 100 times so that the problem comprises 1000 units. The generator parameters are slightly perturbed because it is unrealistic to have so many identical generators. The load profile is based on the System D in [51], which has been

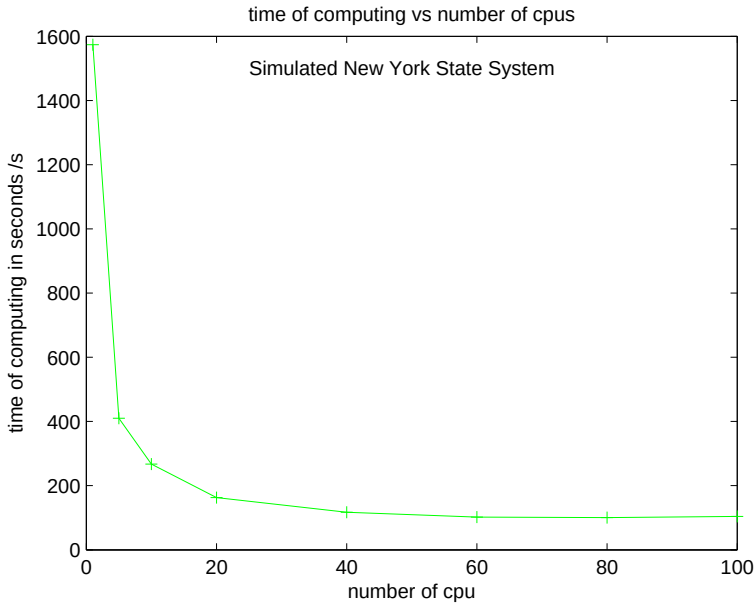

Figure 2.3: Computational time in seconds *vs* number of CPU's for NYCA case

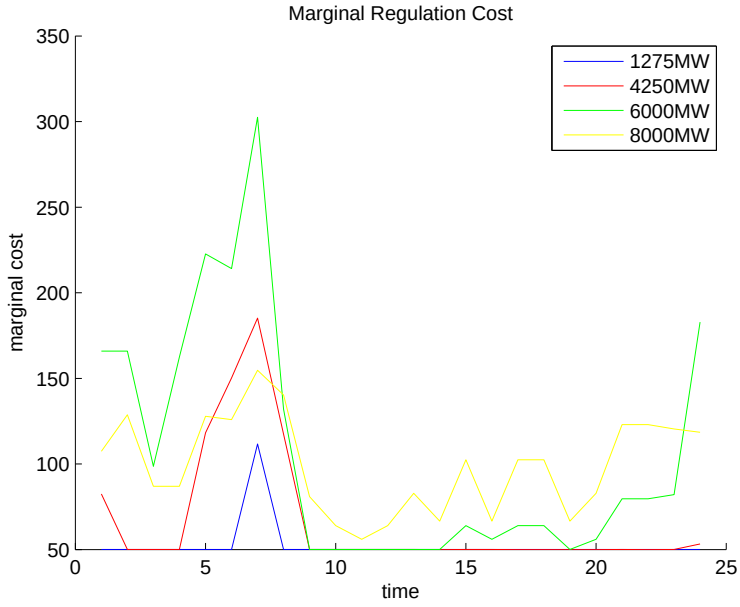

Figure 2.4: Marginal Regulation Costs for different wind penetration levels

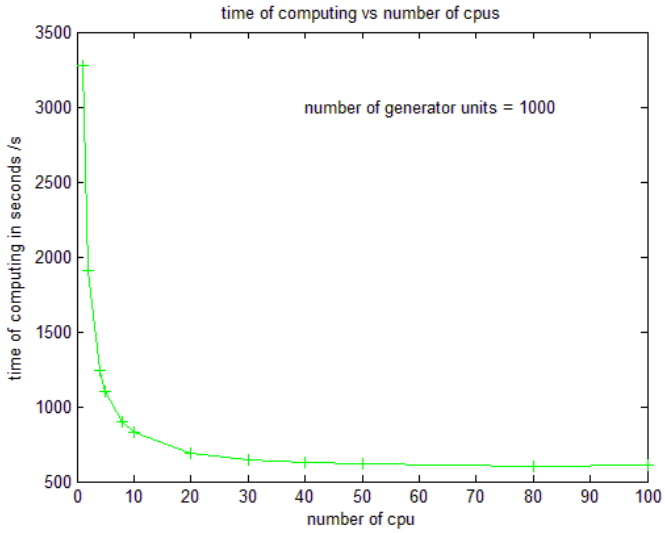

Figure 2.5: Relationship between computational time (in seconds) and number of CPU's

multiplied by 100 accordingly. We assume 25% of wind penetration level, which is close to the $8000MW$ case in NYCA. Figure 2.5 shows that the computational time decreases dramatically when the number of processors increases from 1 to 20, and the minimum computational time is about 20% of the sequential computational time. This result is not as good as that for the NYCA case, which involves much more constraints. For NYCA case, the computational time decreases from 1600 seconds (around 0.5 hours) to 3 minutes, which make it reasonable to restart the UCP solver every hour when new information is available.

We compare the result of rolling horizon approach with the traditional day-ahead planning method. In current market, the stochastic problem is solved once every day, and only the dispatching problem is solved when the real data was available. The operation costs of the next 24 hours of both approaches are compared. We use systems with different number of generators to illustrate the performance of RH algorithm. The result is shown in figure 2.6. A significant reduction of cost is observed when applying rolling horizon approach. For the stochastic problem with wind energy, the cost reduction (approximately 3%) is more significant.

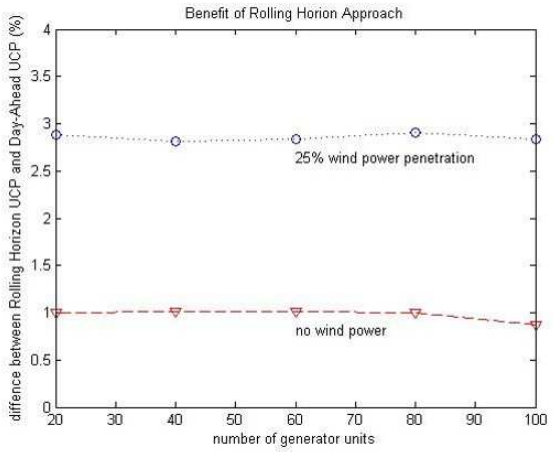

Figure 2.6: Comparison between Rolling Horizon Approach and Day-ahead Approach

2.6 Conclusion

In this dissertation, we formulate a security constrained unit commitment problem by incorporating complex ancillary services, security, and local reserve constraints, and apply this model to the New York Control Area. We investigate the impact of the increasing penetration of wind power on the New York state day-ahead power market. The test results show that parallel computing can significantly reduce the computational time, which makes it possible for rolling horizon implementation of the algorithm. Our testing results on a standard test system show that the rolling horizon approach may lead to significant cost reduction over the traditional day-ahead approach.

Chapter 3

Voltage and Reactive Power Control With Limited Operations On Power Devices

Nomenclature

N_{sc}	number of SCs in power distribution system.
N_{ltc}	number of LTCs in power distribution system.
$x^{sc}(i,t)$	binary variable that is 1 if the ith SC is connected at time t; 0 otherwise.
x^{sc}	an $N_{sc} \times T$ matrix with $x^{sc}(i,t)$ representing setting of theith SC at time t.
$x^{ltc}(i,t)$	the tap position of the ith LTC at time t.
x^{ltc}	an $N_{ltc} \times T$ matrix with $x^{ltc}(i,t)$ representing the setting of the ith SC at time t.
x	the setting of SCs and LTCs of power distribution system, i.e. (x^{sc}, x^{ltc}).
x_t	the setting of SCs and LTCs at time t
L	the number of branches in power distribution system.
N_{node}	the numbers of buses in power distribution system.
$PL(x)$	function to calculate power losses in power distribution system given a setting x

V	an $N_{node} \times T$ matrix with $V(i,t)$ representing the voltage of the ith load position at time t
I	an $L \times T$ matrix with $I(l,t)$ representing the current of the lth branch at time t
$\bar{\Delta}^{ltc}$	the operation limits of LTCs
$\bar{\Delta}^{sc}$	the operation limits and the ith SC
$\bar{V}$	the upper bound on the voltage of a bus in a distribution system
$\underline{V}$	the lower bound on the voltage of a load a distribution system

3.1 Introduction

Voltage and reactive power control (VVC) is an important practical problem in smart grid, especially in power distribution systems. The current practice of VVC is focused on maintaining voltage profiles from violating voltage constraints and pays little attention to the minimization of power losses. Because of insufficient measurements of system parameters in traditional distribution systems, it is not realistic to have accurate state estimations, making the development of an efficient algorithm for minimizing power losses difficult. However, with a large number of smart meters being installed, accurate state estimation or even state forecasting now becomes feasible. Therefore, designing efficient VVC algorithms to minimize power losses becomes an important issue. In smart grid, one of the objectives of VVC is to determine transformer tap positions and the on/off states of capacitors over a certain time period to reduce power losses in a power distribution system. Transformer load tap changers (LTCs) and circuit breakers for shunt capacitors (SCs) are very expensive devices, and cannot be operated frequently. To increase the life expectancy and save maintenance costs, the number of operations on these devices are not allowed to exceed some pre-specified operations limits within a single day. So optimal operations schedules and optimal settings of LTCs and SCs need to be computed to minimize power losses within 24 hours given the day-ahead load prediction.

The VVC problem is usually formulated as a mixed-integer nonlinear programming problem (MINLP), where tap positions of LTCs and on/off states of SCs are modeled as discrete variables, while other variables (e.g., voltages and power flows) are continuous. Many algorithms have been proposed for solving VVC problems. Dynamic programming approaches were proposed in [24–26, 38, 43]. A standard dynamic programming algorithm was used in [24] for solving a problem with 4 capacitors. However, the complexity of the algorithm increases exponentially with respect to the number of capacitors due to the well known "curse of dimensionality". To address this issue, a heuristic method was used to reduce the state space and action spaces in [43] and [38], while in [25] and [26], artificial neural network was used to play a similar role. In [42], the VVC problem was decomposed into two sub-problems to compute the optimal setting of LTCs and SCs separately. These two sub-problems were solved by dynamic programming and fuzzy control algorithms, where a coordination algorithm between these two sub-problems was proposed and a heuristic procedure was used to reduce the solution space.

Stochastic search algorithms are also popular for solving VVC problems. Genetic algorithm was used in [1] and [28] and yielded promising results; however, operations limits of transformers and capacitors were not considered. A simulated annealing algorithm was proposed in [39]. The algorithm was tested on a distribution system with 1 LTC and 11 capacitors, and test results indicated improvement of simulated annealing over conventional methods. Interior Point Method (IPM) was proposed in [40, 41]. In IPM, discrete variables were relaxed, and the resulting problem was solved using the interior point method. Newton's method was used for solving KKT conditions, and penalty terms were added to drive the discrete variables to the nearest discrete positions. The IPM method is shown to be a very efficient algorithm. However, rounding the variables to the nearest discrete positions may compromise the optimality of the solution. Additionally, several heuristic methods were introduced to reduce the complexity of the VVC problem. A time-interval based approach was proposed in [28], where the genetic algorithm was used to partition the 24-hour schedul-

ing horizon into several intervals, and switching operation was not allowed within a partition. Similarly, the work in [16] used heuristics to build priority lists of operations hours, based on which a simplified VVC problem was formulated and solved. These heuristics were effective both in reducing the computational time and in obtaining acceptable results.

In this chapter, we present a simulation-based stochastic search algorithm called Approximation Stochastic Annealing (ASA) for solving VVC problems. The ASA algorithm was initially proposed in [27] for solving finite horizon Markov Decision Processes (MDPs). The idea of using ASA in this work is motivated by the dynamic programming approach for solving VVC problems. Instead of using time-consuming backward induction algorithm, we try to find the optimal solution by searching the randomized solution space associated with the problem. At each iteration, ASA samples candidate solutions from a probability distribution over the set of all admissible solutions, and then the distribution function is modified using a Boltzmann selection scheme based on objective values of sampled solutions. The hope is that the probability distribution will gradually converge to a degenerate distribution assigning unit mass to the optimal solution. The ASA algorithm can be viewed as a stochastic search algorithm and is shown to converge to the global optimal solution in [27]. Moreover, as indicated in [27], ASA may have a faster convergence rate than the simulated annealing algorithm.

In most of the previous work on VVC, the performance of proposed algorithms was either compared with those of naive approaches (pure heuristic) or those of existing optimization algorithm. There is little investigation on how close the result of proposed algorithm is to the exact optimal solution. Because of the complexity of the VVC problem, it is very difficult to get an exact optimal solution. In this work, we propose a Lagrangian relaxation approach, which is called LR-DP algorithm, to get a lower bound on the performance of the VVC problem. By relaxing operation limit constraints, we get a dual problem which can be solved using the combination of the sub-gradient method and the dynamic programming approach. The solution of the dual problem can be viewed as the lower bound to the primal problem,

which can be compared with results from the ASA algorithm to measure the effectiveness of ASA.

The rest of the chapter is organized as follows. We provide a detailed formulation of the VVC problem in section 3.2. In section 3.3, we briefly describe the ASA algorithm and discuss the adaptation of the ASA algorithm to the VVC problem. We describe the LR-DP algorithm in section 3.4. We provide a numerical example to test the performances of ASA and LR-DP in section 3.5.

3.2 Problem formulation

3.2.1 Objective Function

The objective of the VVC problem in this chapter is to find the optimal setting of LTCs and SCs to minimize power losses in power distribution systems, and can be formulated as follows:

$$\min_{x} \sum_{t=1}^{T} \sum_{l=1}^{L} P_{t,l}(x_t), \tag{3.1}$$

where $PL_{t,l}(x_t) = |I(l,t)|^2 R_l$. $I(l,t)$ denotes the power flow on branch l at time t , and can be calculated given a specific configuration x_t using power flow equations.

3.2.2 Constraints

Power Flow Equations

Power flows, voltage levels, active power and reactive power should satisfy a set of power flow equations, which can be represented abstractly in the form of Equation (3.2). In this chapter, we assume that electricity loads are independent of voltage levels.

$$g(V, I, x) = 0 \tag{3.2}$$

Voltage Constraints

In a power distribution system, the voltage level of each node should be within a given range to avoid damages to electric appliances. Theses constraints are modeled in Equation (3.3) below.

$$\underline{V} \leq |V(i,t)| \leq \bar{V}, \forall i \in \{1,\ldots,N_{node}\}, t \in \{1,\ldots T\}. \tag{3.3}$$

Operation Limits on LTCs and SCs

In this chapter, maximum numbers of operations are set for LTCs and SCs. The number of switching operations of a LTC at time t is the difference between tap positions at time t and $t-1$, and the total number of switching operations is the sum of switching operations of all time periods. Theses constraints are modeled by Equations (3.4) and (3.5) below.

$$\sum_{t=1}^{T} |x^{ltc}(i,t) - x^{ltc}(i,t-1)| \leq \bar{\Delta}^{ltc} \tag{3.4}$$

$$\sum_{t=1}^{T} |x^{sc}(i,t) - x^{sc}(i,t-1)| \leq \bar{\Delta}^{sc} \tag{3.5}$$

3.3 Solution Algorithm:Approximate Stochastic Annealing

3.3.1 The Approximate Stochastic Annealing algorithm

The ASA algorithm is a simulation-based optimization algorithm developed in [27], and can be used for solving finite horizon MDPs to maximize total rewards. Given a finite state MDP (S, A, P, R, T) (S: state space; A: action space; P: transition probability function; R: reward function; T: planning horizon), candidate policies are generated from the probability

distribution over the policy space. An $|S|$-by-$|A|$-by-T stochastic matrix q_k, whose $(i,j,t)th$ entry $q_k(i,j,t)$ specifies the probability that the ith state x_i takes action a_j at time t, is constructed at each iteration k. The value $q_k(i,j,t)$ is updated based on the performance of policies generated at iteration k. The general procedure of the ASA algorithm is given below.

1. Specify a non-negative decreasing sequence $\{T_k\}$ (i.e., annealing schedule), parameter sequences $\{\alpha_k\}$ and $\{\beta_k\}$ satisfying $0 \leq \alpha_k, \beta_k \leq 1\ \forall k$. Select a sample size sequence $\{N_k\}$. Set $q_0(i,j,t) = 1/|A|\ \forall i,j,t$ and iteration counter $k = 1$.

2. Sample N_k policies $\Lambda_k := \{\pi^1, \pi^2, \ldots, \pi^{N_k}\}$ as follows:

   - with probability $1 - \beta_k$, construct a policy π^i using q_k;
   - with probability β_k, construct π^i using q_0.

   Calculate a probability mass function over the policy space: Π:

$$\hat{\phi}(q_k, \pi) = (1 - \beta_k)\phi(q_k, \pi) + \beta_k \phi(q_0, \pi),$$

$$\phi(q, \phi) = \prod_{t=1}^{T} \prod_{i=1}^{|X|} \prod_{j=1}^{|A|} [q_k(i,j,t)]^{I\{\pi \in \Pi_{i,j}(t)\}},$$

   where $I\{\cdot\}$ is the indicator function and $\Pi_{i,j}(t) := \{\pi : \pi_t(s_i) = a_j\}$.

3. Perform simulation for each $\pi \in \Lambda_k$ to calculate $V_k^\pi = \sum_{t=1}^{T} R(s_t, \pi_t(s_t))$.

4. Update matrix q_k by

$$q_{k+1}(i,j,t) = \alpha_k \frac{\sum_{\pi \in \Lambda_k} e^{V_k^\pi / T_{k+1}} / \hat{\phi}^{-1}(q_k, \pi) I\{\pi \in \Pi_{i,j}(t)\}}{\sum_{\pi \in \Lambda_k} e^{V_k^\pi / T_{k+1}} / \hat{\phi}^{-1}(q_k, \pi)} + (1 - \alpha_k) q_{k+1}(i,j,t). \tag{3.6}$$

5. If a stopping rule is satisfied, then terminate; otherwise, set $k = k + 1$ and go to step 1.

It was proven in [27] that given

1. $\alpha_k > 0 \ \forall k$, $\sum_{k=0}^{\infty} \alpha_k = \infty$, and $\sum_{k=0}^{\infty} \alpha^2 < \infty$;

2. $T_k \to 0$ as $k \to \infty$;

3. $N_k \beta_k \to \infty$ as $k \to \infty$,

V_k^{π} converges to the optimal value as $k \to \infty$ and the sequence of stochastic matrices generated at successive iterations of ASA will converge to a limiting matrix that assigns unit mass to the optimal policy π^*.

3.3.2 ASA for Voltage and Reactive Power Optimization

The original ASA algorithm is modified for solving the VVC problem. For each time period t, we let $q^{sc}(i,t)$ denote the probability of switching the ith capacitor on, and $q^{ltc}(i,j,t)$ denote the probability of moving the transformer tap of the ith LTC to the jth position. Thus, two stochastic matrices are constructed for SCs and LTCs respectively. The dimension of matrix q^{sc} for SCs is $N_{SC} \times T$, and we denote its $(i,t)th$ element by $q^{sc}(i,t)$. The dimension of matrix q_{ltc} for $LTCs$ is $N_{LTC} \times N_{tap} \times T$, and we denote its $(i,j,t)th$ element by $q^{ltc}(i,j,t)$. In the ASA algorithm, a number of settings of LTCs and SCs are sampled according to these two stochastic matrices and the objective function value for each setting is calculated. These stochastic matrices are iteratively updated based on the performance of sampled settings. Since some of these settings may not satisfy the constraints, penalty terms are added to the

objective function. The augmented objective function thus becomes:

$$\begin{aligned} h(x) &= \sum_{t=1}^{T}\sum_{l=1}^{L} PL_{t,l}(x_t) \\ &+ \eta_{ltc}\sum_{i=1}^{N_{ltc}} \max(\sum_{t=1}^{T} |x^{ltc}(i,t) - x^{ltc}(i,t-1)| - \bar{\Delta}^{ltc}, 0) \\ &+ \eta_{ltc}\sum_{i=1}^{N_{sc}} \max(\sum_{t=1}^{T} |x^{sc}(i,t) - x^{sc}(i,t-1)| - \bar{\Delta}^{sc}, 0) \\ &+ \eta_v \sum_{i=1}^{N_{node}}\sum_{t=1}^{T} (\max(V(i,t) - \bar{V}, 0) + \max(\underline{V} - V(i,t), 0)). \end{aligned} \tag{3.7}$$

On the right hand side of Equation (3.7), the second term is the penalty for the violation of operation limits on LTCs, the third term is the penalty for the violation of operation limits on SCs, and the fourth term is the penalty for the violation of voltage constraints. η_{ltc}, η_{sc} and η_v are penalty coefficients. The modified ASA algorithm in this work is given below:

1. Specify a non-negative decreasing sequence $\{T_k\}$ and a sequence $\{\beta_k\}$ satisfying $0 \leq \beta_k \leq 1\ \forall k$. Specify two non-negative decreasing sequences $\{\alpha_k^{ltc}\}$ and $\{\alpha_k^{sc}\}$ satisfying $0 \leq \alpha_k^{ltc}, \alpha_k^{sc} \leq 1\ \forall k$. Select a sample size sequence $\{N_k\}$. Initialize $q_{sc,0}(i,t) = 0.5$, $q_{ltc,0}(i,j,t) = 1/N_{tap}$.

2. Sample N_k settings of SCs and LTCs from matrices q_k^{sc} and q_k^{ltc} with probability $1-\beta_k$ and from q_0^{sc} and q_0^{ltc} with probability β_k. These settings form a set $X_k := \{x^1, x^2, \ldots, x^{N_k}\}$. Calculate the probability mass function $\hat{\phi}(q_k^{sc}, q_k^{ltc}, x)$ for each setting x as follows:

$$\phi_{sc,t}(q_{sc}, x) = \prod_{i=1}^{N_{sc}} q_{sc}(i_{sc}, t)^{x^{sc}(i,t)} (1 - q_{sc}(i,t))^{1-x^{sc}(i,t)}$$

$$\phi_{ltc,t}(q_{ltc}, x) = \prod_{i=1}^{N_{ltc}} q_{ltc}(i, x^{ltc}(i,t), t)$$

$$\phi(q_{sc}, q_{ltc}, x) = \prod_{t=1}^{T} \phi_{sc,t}(q_{sc}, x)\phi_{ltc,t}(q_{ltc}, x)$$

$$\hat{\phi}(q_{sc,k}, q_{ltc,k}, x) = (1 - \beta_k)\phi(q_{sc,k}, q_{ltc,k}, x) + \beta_k \phi(q_{sc,0}, q_{ltc,0}, x)$$

3. For each setting x, calculate the value of the augmented objective function $h(x)$ in Equation (3.7).

4. Let

$$f(x) = e^{-h(x)/T_k} \hat{\phi}^{-1}(q_{sc,k}, q_{ltc,k}, x).$$

Update the matrices q_k^{sc} and q_k^{ltc} by

$$q_{k+1}^{sc}(i,t) = \alpha_k^{sc} \frac{\sum_{x \in X_k} f(x) I\{x^{sc}(i,t) = 1\}}{\sum_{x \in x_k} f(x)} + (1 - \alpha_k^{sc}) q_k^{sc}(i,t),$$

$$q_{k+1}^{ltc}(i,j,t) = \alpha_k^{ltc} \frac{\sum_{x \in X_k} f(x) I\{x^{ltc}(i,t) = j\}}{\sum_{x \in x_k} f(x)} + (1 - \alpha_k^{ltc}) q_k^{ltc}(i,j,t).$$

5. If a stopping rule is satisfied, then terminate; otherwise, set $k = k + 1$ and go to step 1.

Note that the selection of parameters should satisfy the assumption of the original ASA algorithm to guarantee the asymptotic convergence.

3.4 Solution Algorithm: Lagrangian Relaxation-Dynamic Programming

Dynamic programming has been used for solving the VVC problem in [24] and [38]. As noted in the previous work, the complexity of dynamic programming approach grows exponentially with respect to the planning horizon and problem size. However, by relaxing the operation limits constraints, we can, to some extent, mitigate the problem. By using LR-DP, we can relax the operation limits constraints in equations (3.4) and (3.5) to get a dual problem with objective function as follows:

$$\begin{aligned} D(\mu_{ltc}, \lambda_{sc}, x) \;=\; & \sum_{t=1}^{T}\sum_{l=1}^{L} PL_{t,l}(x_t) \\ & + \sum_{i=1}^{N_{ltc}} \mu_{ltc,i}\{\sum_{t=1}^{T} |x^{ltc}(i,t) - x^{ltc}(i,t-1)| - \bar{\Delta}^{ltc}\} \\ & + \sum_{i=1}^{N_{sc}} \lambda_{sc}\{\sum_{t=1}^{T} |x^{sc}(i,t) - x^{sc}(i,t-1)| - \bar{\Delta}^{sc}\}, \end{aligned} \tag{3.8}$$

where $\mu_{ltc,i}, \lambda_{sc}$ are non-negative lagrangian multipliers. Rearranging Equation (3.8), we get:

$$\begin{aligned} D(\mu_{ltc}, \lambda_{sc}, x) = & \sum_{t=1}^{T}\{\sum_{l=1}^{L} PL_{t,l}(x_t) \\ & + \sum_{i=1}^{N_{ltc}} \mu_{ltc,i}|x^{ltc}(i,t) - x^{ltc}(i,t-1)| \\ & + \sum_{i=1}^{N_{sc}} \lambda_{sc}|x^{sc}(i,t) - x^{sc}(i,t-1)|\} \\ & - \sum_{t=1}^{N_{ltc}} \mu_{ltc,i}\bar{\Delta}^{ltc} - \sum_{t=1}^{N_{sc}} \lambda_{sc,i}\bar{\Delta}^{sc}. \end{aligned} \tag{3.9}$$

Then the dual problem becomes $\max_{\mu_{ltc},\lambda_{sc}} \min_x D(\mu_{ltc}, \lambda_{sc}, x)$ subject to constraints in equations (3.2) and (3.3). The dual problem is convex and can be solved by the sub-gradient

method. We can iteratively update μ_{ltc} and λ_{sc} by solving $\min_x D(\mu_{ltc}, \lambda_{sc}, x)$ in each iteration. The optimization problem $\min_x D(\mu_{ltc}, \lambda_{sc}, x)$ can be formulated as a finite horizon MDP (X, A, R, P, T), where

- $X = \{x_t\}$ is the set of all possible settings of LTCs and SCs at time t;
- $A = \{a := (a_{ltc}, a_{sc})\}$ is the set of all feasible operations on the LTCs and SCs;
- $R = \sum_{l=1}^{L} Pl_{t,l} + \sum_{i=1}^{N_{ltc}} \mu_{ltc,i} |x^{ltc}(i_{ltc}, t) - x^{ltc}(i_{ltc}, t-1)| + \sum_{i=1}^{N_{sc}} \lambda_{sc} |x^{sc}(i_{sc}, t) - x^{sc}(i_{sc}, t-1)|)$;
- $x_t = x_{t-1} + a$, i.e. $P(x_t + a/x_t) = 1$;
- T is a pre-defined study horizon, 24 hours in this case.
- Final reward is $-\sum_{t=1}^{N_{ltc}} \mu_{ltc,i} \Delta^{-}_{i_{ltc}} - \sum_{t=1}^{N_{sc}} \lambda_{sc,i} \Delta^{-}_{i_{sc}}$.

A standard dynamic programming method with complexity $O(|X||A|T)$ can be used for solving the MDP. Note that an action a is feasible only if the setting $x_t = x_{t-1} + a$ is feasible, i.e. the voltage constraints should be satisfied given the configuration x_t. The LR-DP algorithm for the VVC problem is given as follows:

1. Initialize μ_{ltc}, λ_{sc}, and set iteration number $k = 1$
2. Solve the optimization problem $\min_x D(\mu_{ltc}, \lambda_{sc}, x)$ and set $x^* = \text{argmin}_x D(\mu_{ltc}, \lambda_{sc}, x)$
3. Calculate the primal value $\sum_{t=1}^{T} \sum_{l=1}^{L} Ploss_{t,l}(x^*)$ if x^* satisfy the operation limits constraints. Let y_{primal} be the best primal value obtained thus far.
4. Calculate the duality gap

$$\epsilon = \frac{y_{primal} - D(\mu_{ltc}, \lambda_{sc}, x^*)}{y_{primal}}.$$

   If the duality gap is less than a given threshold, return the setting x with respect to the best primal value; otherwise, got to step 5 .

5. update μ_{ltc}, λ_{sc} as below:

$$\delta_i^{ltc} = \sum_{t=1}^{T} |x^{ltc}(i,t) - x^{ltc}(i,t-1)| - \bar{\Delta}^{ltc},$$

$$\delta_i^{sc} = \sum_{t=1}^{T} |x^{sc}(i,t) - x^{sc}(i,t-1)| - \bar{\Delta}^{sc};$$

let $\delta^{ltc} = \{\delta_i^{ltc}\}$, $\delta^{sc} = \{\delta_i^{sc}\}$ and $\delta = (\delta^{ltc}, \delta^{sc})$

$$\mu_{ltc}(i) := \max(\mu_{ltc}(i) + \alpha_k \frac{\delta_i^{ltc}}{||\delta||},\ 0),$$

$$\mu_{sc}(i) := \max(\mu_{sc}(i) + \alpha_k \frac{\delta_i^{sc}}{||\delta||},\ 0),$$

where $\{\alpha_k\}$ is a sequence satisfying $\sum_{k=1}^{\infty} \alpha_k = \infty$ and $\sum_{k=1}^{\infty} \alpha_k^2 < \infty$. Go to step 2.

We observe that the computational time in step 2 grows exponentially with respect to the number of LTCs and SCs, and may become prohibitive when the system under study is very large. However, in some realistic systems, numbers of SCs and LTCs are small, which makes the LR-DP algorithm an effective method for solving the VVC problem. Moreover, unlike some stochastic search algorithms such as genetic algorithm and particle swarm optimization, which have difficulties in evaluating the accuracy of computational results, the LR-DP algorithm provides a lower bound on the optimal value and a duality gap, which can be used to evaluate the performance of the algorithm.

3.5 Numerical Results

The well known PG&E 69-bus distribution test network with 69 buses, 68 branches and 48 loads is used to illustrate the performance of both the ASA and LR-DP algorithms. The base power and base voltage are set to 10MVA and 12.66 kV, respectively. The data of branch impedance and maximum daily active and reactive powers for all loads can be found

in [3]. The diagram of the system is given in Fig 3.1. We assume that there are 10 capacitors installed on bus 9, 19, 31, 37, 40, 47, 52, 55, 57 and 65 as in [40] with size 0.3MVar, and that a LTC with tap setting $1 \pm 0.02 \times 3$ is installed at bus 1. 7 representative load profiles in figure 3.2 are randomly assigned to nodes in figure 3.1.

3.5.1 Implementation of the ASA algorithm

The ASA algorithm is implemented using MATLAB 7.10.0 on a computer with an Intel Core2 CPU (2.40GHz), 2.0GB RAM and Ubuntu 12.04.1 OS. The step-size is set as follows: $\alpha_k^{sc} = 1/(k+100)^{0.51}$, $\alpha_k^{ltc} = 1/(k+100)^{0.6}$. The annealing schedule $T_k = 0.001 + c/(k+1)^{0.5}$, where c is the positive difference between the minimum value and median of all the objective values calculated in the step 3 of the ASA algorithm. The penalty coefficient for the violation of voltage constraints $\eta_v = 0.05$. The penalty coefficients for the violation of operation limits are $\eta_{ltc} = 1$ and $\eta_{sc} = 3$. The sample size $N_k = \max(50, \sqrt{k})$. The algorithm stops when either $k = 10000$ or the following equations are satisfied:

$$\min(1 - q_{sc,k}(i_{sc}, t), q_{sc,k}(i_{sc}, t)) < 0.001 \forall i, t$$

$$\min(1 - q_{ltc,k}(i_{ltc}, j, t), q_{sc,k}(i_{ltc}, j, t)) < 0.001 \forall i, t,$$

which indicate that all elements in those two stochastic matrices getting close to 0 or 1.

3.5.2 Implementation of the LR-DP algorithm

The LR-DP algorithm is implemented using MATLAB 7.10.0 on the same platform as that for ASA algorithm. The initial lagrangian multipliers are all set as 1. The step-size $\alpha_k = 1/(0.05 + 0.05k^{0.75})$. The algorithm stops when the iteration number k exceeds 100 or when the duality gap $\epsilon < 0.01$.

3.5.3 Result

The initial tap position is set to 0, and the initial shunt capacitor settings are 0, which means that they are switched off at the beginning. From our calculation, we observe in the results of both ASA and LR-DP algorithms that the transformer taps are at positions that lead to high voltage profiles. This is consistent with our intuition that given constant loads, higher voltage profile will bring lower electricity current, which will result in less power loss. Our ongoing study shows that if the power load is not constant and depends on the voltage profiles, the optimal tap positions in current work will not be optimal. The situation of SCs is more complicated, and no obvious pattern is observed for their settings. In many distribution systems in the United States, a portion of the capacitors are controlled by the clocks, and a switch-on time and a switch-off time are set for each of those capacitors; this means that only 2 operations are allowed for these capacitors. In this work, different operation limits between 2 and 5 are set for SCs and LTCs.

The performance of the ASA algorithm is given in table 3.1. We can see the convergence of the ASA algorithm in figure 3.3. The power losses and CPU times in table 3.1 are based on 10 independent replication runs of the ASA algorithm, and the standard deviation of power losses is also computed. The computational results of LR-DP algorithm are provided in table 3.2. Unlike ASA algorithm, the LR-DP algorithm is deterministic, so only a single run is performed.

From tables 3.1 and 3.2, we can see that, generally, power losses increase as operation limits decreases. Through computation, we observe that if there's no operation limits constraints, the maximum of operations on LTC and SC is 5. Therefore, when the operation limits is 5, the operations limits constraints can be ignored. For operation limits less than 5, lagrangian relaxation technique is used to obtain the lower bound. From table 3.2, we observe that the duality gap is less than 0.6%, which indicates that the computational result of LR-DP algorithm is very close to the optimal value. Additionally, we can see that the LR-DP algorithm is time consuming. If bad initial lagrangian values and step-sizes were

Table 3.1: Computing result of ASA algorithm

Operation Limits	Power Losses kWh	Standard Deviation kWh	CPU time (s)
2	901.09	2.61	1003
3	899.29	0.90	1093
4	898.33	1.14	1206
5	898.01	0.58	1262

chosen, the computational time could have been much longer. The computational results of ASA algorithm are compared with that of LR-DP algorithm to test the performance of the ASA algorithm. The comparison between these two algorithms are given in table 3.3. We observe that that the performance of ASA algorithm is not as good as LR-DP algorithm. However, the differences between calculated power losses are less than 0.4%, which is very small. Moreover, the ASA algorithm is significantly faster than the LR-DP algorithm: the ASA algorithm takes much less time to compute the result with only little compromise on the quality of the solution.

3.6 Conclusion

A novel Approximate Stochastic Annealing algorithm is proposed for solving the voltage and reactive power control problem with operation limits on shunt capacitors and LTCs. The proposed simulation-based algorithm samples configurations of SCs and LTCs according to a probability distribution, and a Boltzmann scheme is used to updated the probability distribution. A Lagrangian Relaxation-Dynamic Programming algorithm is also proposed for solving the problem, and a lower bound of the optimal solution is obtained. Test results on a well known PG&E 69 bus system indicate that the LR-DP algorithm can solve the problem precisely, but is time consuming, and that the ASA algorithm can solve the problem very quickly with a little compromise on the optimality of the solution. Therefore, it is concluded from numerical results that the ASA algorithm is an effective method for solving the voltage and reactive power control problem with operation limit constraints.

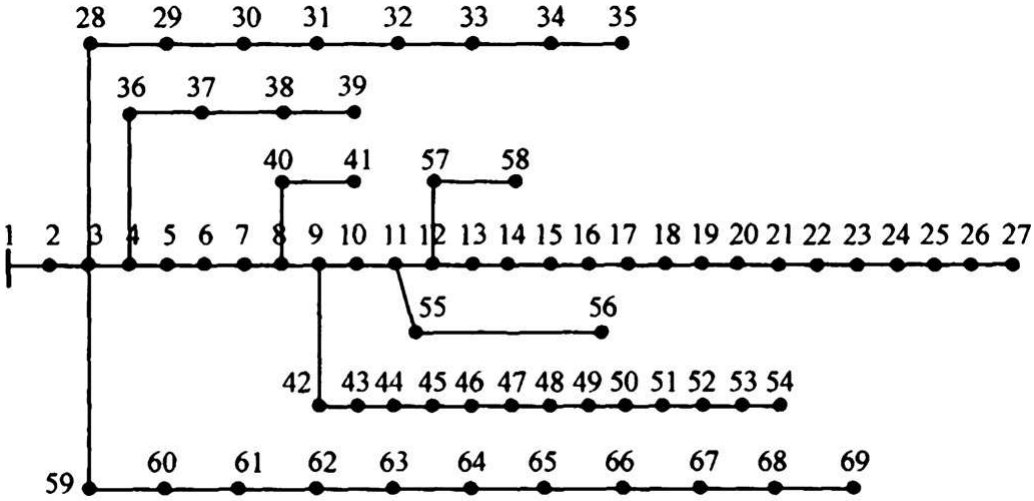

Figure 3.1: Diagram of PG&E 69-bus System

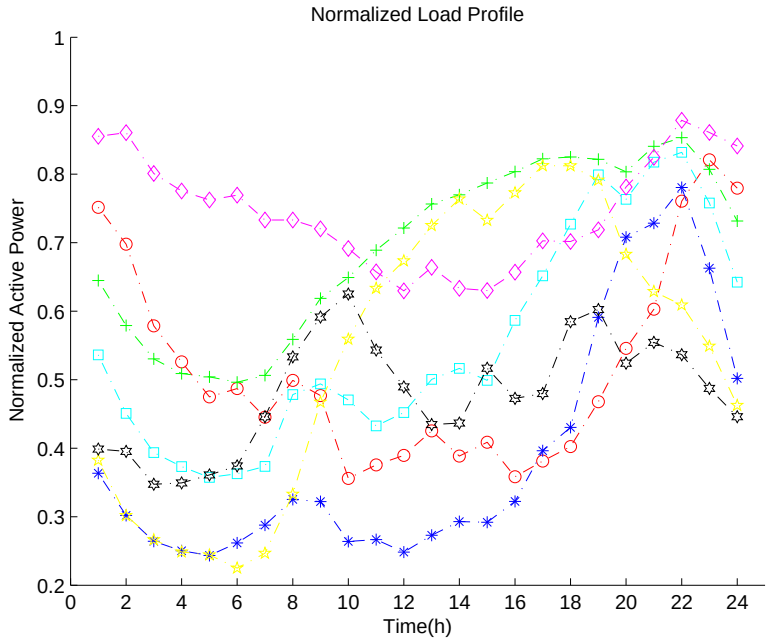

Figure 3.2: Normalized load profiles

Table 3.2: Computing result of LR-DP algorithm

Operation Limits	Power Loss (kWh)	Dual Problem (kWh)	CPU time (s)	Duality Gap
2	897.83	896.45	36004	0.15%
3	897.09	896.82	18914	0.03%
4	897.09	896.52	18935	0.06%
5	896.82	896.82	9039	0%

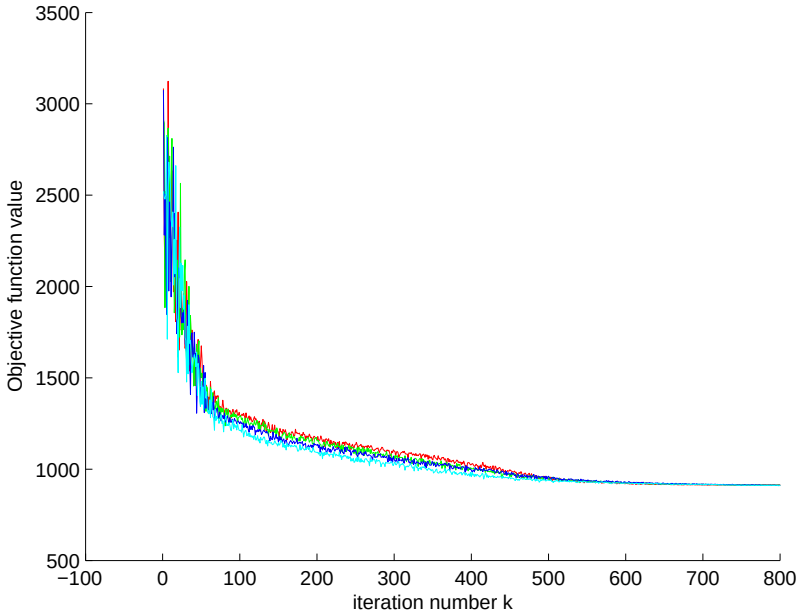

Figure 3.3: Trajectory of objective function value

Table 3.3: Comparison between ASA and LR-DP

Operation Limits	Power Loss (ASA)	CPU time (s)	Power Loss (LR-DP)	CPU time (s)
2	901.09	1003	897.83	36004
3	899.29	1093	897.09	18914
4	898.33	1206	897.09	18935
5	898.01	1262	896.82	9039

Chapter 4

Voltage and Reactive Power Control with Switching Costs and ZIP Load Model

Nomenclature

Some notations that do not appear in chapter 3 are explained below. Other notations that appear before have the same meaning as those in chapter 3.

P	an $N_{node} \times T$ matrix with $P(i,t)$ representing the active (real) power of the ith Node at time t .
Q	an $N_{node} \times T$ matrix with $Q(i,t)$ representing the reactive power of the ith node at time t .
V_0	standard voltage level. For example, $120V$.
$S_{i,0}$	load forecast of node i at V_0.
$Z^{\%}$	percentage of constant impedance load.
$I^{\%}$	percentage of constant current load.
$P^{\%}$	percentage of constant power load.

4.1 Introduction

Chapter 3 and most previous work regarding VVC assume that electricity loads are independent of voltage profiles; however, this assumption may not be valid in realistic power distribution systems. The study in [59] indicates that in some power distribution systems, the electricity loads are correlated with voltage profiles. Several load models were proposed to handle this issue. The optimal solution of VVC problems under constant load model, which assumes independence between loads and voltage profiles, may not be optimal under some other load models. Because of the correlation between electricity loads and voltage profiles, conservation voltage reduction (CVR) was proposed in [35, 36] and [59], in which the objective is to minimize the total power consumption through VVC. However, their control algorithms are based on heuristics, and switching costs of LTCs and SCs are ignored. In this chapter, we consider both the minimization of power losses and the minimization of total energy consumption. This dissertation uses the well-known ZIP load model in [59] for modeling the relationship between electricity loads and voltage profiles.

Chapter 3 uses limits on the number of operations of power devices to avoid frequent operations of those devices. However, on one hand, it is difficult to determine the optimal operations limits; on the other hand, it is relatively easy to evaluate operation costs, for example, by calculating the average maintenance cost of one operation. Therefore, it might be more useful to consider the switching costs rather than to place operation limits on those devices. This chapter adds switching costs to our VVC models.

This dissertation proposes an ASA algorithm for solving the VVC models in this chapter. We illustrate the effectiveness of ASA using the same test system as that in chapter 3. We implement the dynamic programming algorithm to get an exact optimal solution although it is extremely time-consuming. We show that the ASA algorithm generates solution very close to the optimal solution within a moderate computational time. Additionally, we illustrate the performance of ASA by comparing it with the simulated annealing (SA) algorithm.

The rest of the chapter is organized as follows. We provide detailed formulations of the

VVC problems with ZIP model in section 4.2. In section 4.3, we briefly describe the ZIP load model and a simple power flow computation algorithm when a ZIP model is involved. In section 4.4, we briefly describe the ASA algorithm for VVC problems. We describe the dynamic programming algorithm and the simulated annealing algorithm for VVC in section 4.5. We test the performance of the ASA algorithm in section 4.6.

4.2 Problem Formulation

4.2.1 Objective Functions

This chapter considers two different objective functions: the minimization of power losses and the minimization of the total energy consumption. The problem of minimizing the sum of power losses and switching costs can be formulated as:

$$\min_{x} \sum_{t=1}^{T} F_{loss}(x_t, t)$$

$$\begin{aligned} F_{loss}(x_t, t) = & \sum_{l=1}^{L} PL_{t,l}(x_t) \\ & + \eta_{ltc} \sum_{i=1}^{N_{ltc}} |x^{ltc}(i,t) - x^{ltc}(i,t-1)| \\ & + \eta_{sc} \sum_{i=1}^{N_{sc}} |x^{sc}(i,t) - x^{sc}(i,t-1)|, \end{aligned}$$

where $PL_{t,l}(x_t) = |I(t,l)|^2 R_l$. $I(t,l)$ denotes the power flow on branch l at time t , which can be calculated using power flow equations and the value of x. $\eta_{ltc} \sum_{t=1}^{T} \sum_{i=1}^{N_{ltc}} |x^{ltc}(i,t) - x^{ltc}(i,t-1)|$ is the switching cost of LTCs, and η_{ltc} is the cost of a unit step movement of the transformer tap in an LTC. $\eta_{sc} \sum_{t=1}^{T} \sum_{i=1}^{N_{sc}} |x^{sc}(i,t) - x^{sc}(i,t-1)|$ is the switching cost of SCs, and η_{sc} is the cost of a single switching operation of an SC.

The problem of minimizing the sum of the total energy consumption and switching costs can be formulated as:

$$\min_{x} \sum_{t=1}^{T} F_{total}(x_t, t)$$

$$\begin{aligned} F_{total}(x_t, t) = & \sum_{t=1}^{T} \sum_{l=1}^{L} PL_{t,l}(x_t) \\ & + \sum_{i=1}^{N} |S_i| \\ & + \eta_{ltc} \sum_{t=1}^{T} \sum_{i=1}^{N_{ltc}} |x^{ltc}(i, t-1)| \\ & + \eta_{sc} \sum_{t=1}^{T} \sum_{i=1}^{N_{sc}} |x^{sc}(i, t) - x^{sc}(i, t-1)|, \end{aligned}$$

where the second term on the right hand side is the sum of the loads at all nodes. The total energy consumption is the sum of power losses and the electricity consumption at all nodes.

4.2.2 Constraints

Power Flow Equations

The power flows, voltage levels, active power and reactive power should satisfy a set of power flow equations, which can be represented abstractly in the form of the following equation: $g(V, I, x) = 0$. This chapter uses the ZIP model to describe the relationship between voltage levels and electricity loads. The power flow computation with the ZIP load model is discussed in detail in section 4.3.

Voltage Constraints

In a power distribution system, the voltage level of each node should be within a given range to avoid damages to electric devices and appliances. Theses constraints are modeled in Equation (4.1) below.

$$\underline{V} \leq |V(i,t)| \leq \bar{V}, \forall i \in \{1,\ldots,N_{node}\}, t \in \{1,\ldots,T\} \tag{4.1}$$

4.3 Power Flow Computation

4.3.1 Power Flow Computation Using a Matrix Form

In power distribution systems, the topology of an distribution network is usually radial, and we can use the method in [66] to to compute the power flows. In [66], the voltage drop at each node can be calculated in the matrix form:

$$[\Delta V] = [DFL][I] = [BCBV][BIBC][I], \tag{4.2}$$

where $[\Delta V]$ is the voltage drop vector of length N_{node} with its ith element being the difference between the voltage levels at the substation and the ith node, $[DFL]$ is a matrix that transforms the current injection vector into the voltage drop vector, and $[I]$ is the current injection vector with its ith element being the current that the ith node injects into the system, which can be calculated as $\frac{S_i}{V_i}$. $[BIBC]$ is an $L \times L$ matrix representing the impedance of branch lines, and $[BIBC][I]$ is a vector that represents voltage drops between the two ends of all branch lines. $[BCBV][BIBC]I$ adds the voltage drop of each branch line along the paths between the substation and a node to get the voltage drop of each node. The voltage level of each node can be represented as

$$[V] = V_{sub} - [\Delta V], \tag{4.3}$$

where V_{sub} is the voltage at the substation, and $[V]$ is a vector representing the voltage levels of all nodes. Equation (4.2) can be solved recursively as follows:

$$I_i^k = \frac{S_i}{V_i}, \tag{4.4}$$

$$[\Delta V^{k+1}] = [DFL][I^k], \tag{4.5}$$

$$[V^{k+1}] = V_{sub} - [\Delta V^{k+1}], \tag{4.6}$$

where k is the iteration number.

4.3.2 ZIP Load Model

In this chapter, we use the ZIP model in [58] and [59] to model the relationship between electricity loads and voltage levels. The ZIP model can be formulated as

$$P_i = \frac{V_i^2}{V_0^2} \cdot S_{i,0} \cdot Z^{\%} \cdot cos(Z^{\theta}) + \frac{V_i}{V_0} \cdot S_{i,0} \cdot I^{\%} \cdot cos(I^{\theta}) + S_{i,0} \cdot P^{\%} \cdot cos(P^{\theta}), \tag{4.7}$$

$$Q_i = \frac{V_i^2}{V_0^2} \cdot S_{i,o} \cdot Z^{\%} \cdot sin(Z^{\theta}) + \frac{V_i}{V_0} \cdot S_{i,0} \cdot I^{\%} \cdot sin(I^{\theta}) + S_{i,0} \cdot P^{\%} \cdot sin(P^{\theta}), \tag{4.8}$$

where V_i is the voltage level at node i, V_0 is the standard voltage, for example, $120V$. $S_{i,0}$ is the electricity load at V_0 and can be estimated through load forecasting. The ZIP model assumes that the electricity load is the sum of three parts: constant impedance component, constant current component, and constant power component. $Z^{\%}$ is the percentage of the constant impedance load, $I^{\%}$ is the percentage of the constant current load, and $P^{\%}$ is the percentage of the constant power load. Z^{θ}, I^{θ} and P^{θ} are the respective phase angles of those three components. For simplicity, we assume that $Z^{\theta} = I^{\theta} = P^{\theta} = S^{\theta}$, where S^{θ} is the phase angle of $S_{i,0}$. Then, it follows that:

$$P_i = \frac{V_i^2}{V_0} \cdot P_{i,0} \cdot Z^{\%} + \frac{V_i}{V_0} \cdot P_{i,0} \cdot I^{\%} + P_{i,0} \cdot P^{\%}, \tag{4.9}$$

$$Q_i = \frac{V_i^2}{V_0} \cdot Q_{i,o} \cdot Z^{\%} + \frac{V_i}{V_0} \cdot Q_{i,o} \cdot I^{\%} + Q_{i,0} \cdot P^{\%}. \tag{4.10}$$

In this work, we assume that the load forecast are voltage dependent. In traditional electric grids, the load forecast is usually based on the historical load data at the feeder level and the weather forecast [19]. Measurements of voltage levels and loads at the customer level are usually not available. Thus, the load forecast is usually independent of voltage profiles. However, as the installation of smart meters increases, we are able to obtain the historical load and voltage data at the customer level, which makes the voltage dependent load forecast feasible in smart grid. Actually, the ZIP load model has been implemented in some widely used planning tools such as Cymdist [14].

4.3.3 Power Flow Computation with the ZIP Load Model

When the ZIP load model is used, the power flow computation method in section 4.3.1 is no longer applicable, because S_i is not a constant but a variable depending on voltage profiles. However, we can use $P_i + jQ_i$ to replace S_i in Equation (4.4). P_i and Q_i can be calculated according to Equations (4.9) and (4.10). Then, we can use the same recursive algorithm in section 4.3.1 to compute power flows.

4.4 Solution Algorithm: Approximate Stochastic Annealing

This chapter uses the ASA algorithm in section 3.3 for solving the VVC problems with switching costs and ZIP load model. Since some of the candidate solutions sampled by ASA may not satisfy voltage constraints, we add penalty terms to the objective function. The augmented objective function thus becomes:

$$h(x) = \sum_{t=1}^{T} F(x,t) + \eta_v \sum_{i=1}^{N_{node}} \sum_{t=1}^{T} (\max(V(i,t) - \bar{V}, 0) + \max(\underline{V} - V(i,t), 0)), \qquad (4.11)$$

where $F(x_t, t) = F_{loss}(x_t, t)$ if the objective is to minimize power losses, or $F(x_t, t) = F_{total}(x_t, t)$ if the objective is to minimize the total power consumption. The second term on the right hand side is the penalty for voltage violations. The implementation of the ASA algorithm is quite similar to that we use in section 3.3. Nevertheless, there are several differences:

1. This chapter adds the switching costs to our models.

2. We use the ZIP load model in this chapter, while in chapter 3, electricity loads are constants.

3. In chapter 3, we only consider the VVC problem of minimizing power losses. This chapter also solves the problem of minimizing the total power consumption.

4.5 Other Solution Algorithms

4.5.1 Dynamic Programming

Dynamic programming was used for solving VVC problems in [24] and [38]. A similar method can solve VVC problems with switching costs and the ZIP load model. We formulate the optimization problem $\min_x \sum_{t=1}^{T} F(x_t, t)$ as a finite horizon Markov Decision Process (X, A, R, P, T), where

- $X = \{x_t\}$ is the set of all possible settings of LTCs and SCs at time t;
- $A = \{a := (a_{ltc}, a_{sc})\}$ is the set of all feasible operations on LTCs and SCs;
- $x_t = x_{t-1} + a$, i.e. $P(x_t + a | x_t) = 1$;

- $R(x_t, a) = F(x_t, t)$, where $F(x_t, t) = F_{loss}(x_t, t)$ or $F(x_t, t) = F_{total}(x_t, t)$
- T is a pre-defined planning horizon, 24 hours in our case.

The standard dynamic programming method with complexity $O(|X||A|T)$ can be applied to solve the MDP. Note that an action a is feasible only if the setting $x_t = x_{t-1} + a$ is feasible, i.e. voltage constraints should be satisfied at state x_t.

The computational time in step 2 grows exponentially with respect to the number of LTCs and SCs, and may become prohibitive when the system under study is very large. However, in some realistic systems, the numbers of SCs and LTCs are small. Thus, the dynamic programming algorithm is an effective method for solving VVC problems in those cases, and the optimality of the solution is guaranteed.

4.5.2 Simulated Annealing

We implement the simulated annealing algorithm similar to that in [39], but with different objective functions and power flow equations. Let $H(x)$ be the objective function. The SA algorithm iteratively searches the optimal setting of LTCs and SCs following the steps below:

1. Initialize a feasible setting of LTCs and SCs, e.g., an initial feasible solution x_0. Specify an annealing schedule $\{T_k\}$.
2. Randomly perturb a value in x_k to get a new solution x'.
3. Solve the power flow equations. If one or more constraints are violated, go to step 2; otherwise, compute the objective function $H(x')$ and continue to step 4.
4. If $H(x') \leq H(x_k)$, let $x_{k+1} = x_k$. Else, with probability $e^{[H(x_k)-H(x')]/T_k}$, $x_{k+1} = x'$; with probability $1 - e^{[H(x_k)-H(x')]/T_k}$, $x_{k+1} = x_k$.
5. If a stop condition is satisfied, stop; else, let $k = k + 1$ and go to step 2.

Not that in step 3, we check whether the candidate solution x' generated in step 2 is feasible: if it is not feasible, we go back to step 2 and generate a new solution again. Thus, SA assures that the candidate solution in each iteration is feasible. Therefore, we use the original objective function rather than the objective function with penalty terms. $H(x') = \sum_{t=1}^{T} F_{loss}(x'_t, t)$ if the objective is to minimize power losses; $H(x') = \sum_{t=1}^{T} F_{total}(x'_t, t)$ if the objective is to minimize the total power consumption.

4.6 Experiments and Results

We still use the well known PG&E 69-bus power distribution system in chapter 3 to illustrate the performance of the ASA algorithm for solving new VVC models. The basic settings of the system is almost the same as those in chapter 3. The switching cost for a unit step movement of transformer tap is set at the expense of 0.25kWh electricity energy, and the switching cost of one operation of an SC is set at the expense of 0.5kWh electricity energy. The parameters of the ZIP load model are given as: $Z^{\%} = 0.5$, $I^{\%} = 0$, and $P^{\%} = 0.5$.

4.6.1 Implementation of the ASA algorithm

The ASA algorithm is implemented using MATLAB on a computer with an Intel Core2 CPU (2.40GHz), 2.0GB RAM and Windows 7 OS. The smoothing parameters are set as follows: $\alpha_k^{sc} = 1/(k+100)^{0.51}$, $\alpha_k^{ltc} = 1/(k+100)^{0.6}$. The annealing schedule $T_k = 0.001 + 0.5c/(k+1)^{0.5}$, where c is the difference between the minimum and the median of all objective values calculated in step 3 of the ASA algorithm. The penalty coefficient for violation of voltage constraints $\eta_v = 1$. The sample size $N_k = \max(50, \sqrt{k})$. The algorithm stops when either $k = 10000$ or the following equations are satisfied:

$$\min(1 - q_{sc,k}(i,t), q_{sc,k}(i,t)) < 0.001, \forall i, t,$$

$$\min(1 - q_{ltc,k}(i,j,t), q_{sc,k}(i,j,t)) < 0.001, \forall i,j,t,$$

which indicate that all elements in the two stochastic matrices getting close to either 0 or 1.

4.6.2 Implementation of the Dynamic Programming Algorithm

The dynamic programming algorithm is implemented using MATLAB on the same platform as the ASA algorithm. Standard backward induction algorithm is implemented.

4.6.3 Implementation of the Simulated Annealing Algorithm

We compare the results generated by the ASA algorithm with those generated by the simulated annealing algorithm in [39]. The annealing schedule $T_{k+1} = T_k \cdot 0.9995$, and $T_1 = 1000$. The search process ends when $T_k < 0.001$. At step 2 of the SA algorithm in section 4.5.2, a control device, e.g. LTC or SC, is randomly selected, and its status is perturbed to generate a new solution.

4.6.4 Results

To illustrate the significance of ZIP model in VVC, we perform experiments in two cases. In case 1, the constant load model is used, and the correlation between electricity loads and voltage profiles is ignored. In case 2, the ZIP load model is used.

Case 1: Constant Load Model

In this case, we assume that electricity loads are independent of voltage profiles. In this case, the minimization of power losses is equivalent to the minimization of total power consumption, because the electricity loads at all nodes are constants. The computational results are given in table (4.1) and table (4.2). We observe that the results from the ASA algorithm are almost the same as those from the dynamic programming algorithm, which generates

the exact global optimal solution. But the computational time of the ASA algorithm is 80% less.

Because the constant load model is used in this case, the optimal voltage at the feeder is at the highest possible value ($1.04p.u.$) without violating voltage constraints: the transformer tap moves from neutral position to position +2, and stays at this position in the next 24 hours. This is consistent with our intuition that, given constant electricity loads, higher voltage level results in lower electricity current, and thus, yields less power loss. Additionally, we observe fewer switching operations in table (4.2) than in table (4.1) , which is consistent with our intuition that switching costs of LTCs and SCs will result in fewer switching operations in optimal solutions.

Case 2: ZIP Load Model

First, we perform an experiment with the objective of minimizing power losses. The computational results are given in tables 4.3 and 4.4. We observe that the computing result in this case has more frequent LTC tap movements than case 1, which may be due to the fact that an increase in voltage level under the ZIP model may not result in lower energy losses: an increase in voltage level will result in more energy consumption, which may bring more energy losses. Therefore, the optimal tap position may not be the highest feasible position, and may change as hourly electricity loads vary in a planning period. Additionally, optimal operation schedules of SCs in this case are also different from those in case 1. Thus, the computational results support our conjecture that the optimal controlling schedules of SCs and LTCs are different under different load models.

Similar to our observations in case 1, computational results from the ASA algorithm are very close to those from the dynamic programming algorithm. Additionally, the ASA algorithm also uses much less computational time than the dynamic programming approach does in this case.

Second, we perform an experiment with the objective of minimizing the total energy con-

sumption. The computational results are given in tables (4.5) and (4.6). In this experiment, we observe different operations of LTCs and SCs from the previous problem of minimizing power losses. This shows that the optimal solution for the minimization of power losses may not be optimal for the minimization of the total power consumption. Additionally, we observe lower tap positions during the day time, when the power consumption is high, than during the midnight, when the power consumption is low, which indicates that the conservation voltage reduction is more effective for peak loads than for off-peak loads. Computational results also indicate that although the objective function has changed, the ASA algorithm still performs very well, and is able to generate solutions very close to the exact optimal solution using much less computational time than dynamic programming.

Comparison Between ASA and SA

An advantage of ASA over SA is that ASA samples a number of candidate solutions over the entire solutions space in each iteration, which may prevent the algorithm from getting trapped in local optimal points, while SA samples a single candidate solution in each iteration and is more likely to be trapped in local optimal points. Thus, ASA may generate better solutions than SA does. The computational results in this chapter support our conjecture to some extent. In all experiments in this work, we observe smaller objective values using the ASA algorithm than using the SA algorithm. Although ASA evaluates much more solutions in each iteration than SA does, it requires a much smaller number of iterations to yield a good solution. Overall, ASA uses less computational time than SA does, which is consistent with the computational results in [27].

4.7 Conclusion

An Approximate Stochastic Annealing algorithm is proposed for solving the voltage and reactive power control problem with switching costs and the ZIP load model. The objectives

	ASA	Dynamic Programming	Simulated Annealing
Objective Values (kWh)	896.82	896.82	899.56
Computational Time (s)	1132	6015	1327
Tap Position Changes (LTCs)	2	2	2
Switching Operations (SCs)	14	14	49

Table 4.1: Computational results of power loss minimization: constant load model and zero switching costs

	ASA	Dynamic Programming	Simulated Annealing
Objective Values (kWh)	902.33	902.33	904.43
Computational Time (s)	831	6455	1245
Tap Position Changes (LTCs)	2	2	2
Switching Operations (SCs)	8	8	10

Table 4.2: Computational results of power loss minimization: constant load model and none-zero switching costs

	ASA	Dynamic Programming	Simulated Annealing
Objective Values (kWh)	910.10	909.83	912.55
Computational Time (s)	994	6195	1282
Tap Position Changes (LTCs)	8	8	8
Switching Operations (SCs)	19	15	43

Table 4.3: Computational results of power losses minimization: ZIP load model and zero switching costs

	ASA	Dynamic Programming	Simulated Annealing
Objective Values (kWh)	915.36	915.26	915.41
Computational Time (s)	976	6112	1212
Tap Position Changes (LTCs)	3	3	3
Switching Operations (SCs)	7	7	7

Table 4.4: Computational results of power losses minimization: ZIP model and none-zero switching costs

	ASA	Dynamic Programming	Simulated Annealing
Objective Values (kWh)	56685.5	56685.5	56690.1
Computational Time (s)	1029	6180	1322
Tap Position Changes (LTCs)	12	12	10
Switching Operations (SCs)	9	9	47

Table 4.5: Computational results of total power consumption minimization: ZIP load model and zero switching costs

	ASA	Dynamic Programming	Simulated Annealing
Objective Values (kWh)	56691.5	56691.4	56698.4
Computational Time (s)	746	6093	1261
Tap Position Changes (LTCs)	10	10	10
Switching Operations (SCs)	5	5	13

Table 4.6: Computational results of total power consumption minimization: ZIP model and none-zero switching costs

of both power loss minimization and total power consumption minimization are considered. The proposed ASA algorithm is used for solving VVC problems with different objective functions. This chapter also implements the dynamic programming algorithm and the simulated annealing algorithm to test the performance of the ASA algorithm. The computational results on the PG&E 69-bus system indicate that the ASA algorithm can solve problems with different objective functions and load models very quickly with very little compromise on the optimality of the solutions. The computational results also illustrate the superiority of the ASA algorithm over the SA algorithm for solving VVC problems in terms of both optimality and computational time.

Chapter 5

Convergence Analysis of the ASA Algorithm

5.1 Introduction

The study in [27] shows that ASA generates policies that converge to the global optimal policy for solving finite horizon MDPs. In this chapter, we show that under some assumptions, ASA generates solutions that converge to optimal solutions of VVC problems. We make some minor modifications of the proof in [27] to prove the convergence of the ASA algorithm for VVC problems.

This chapter is organized as follows. First, we give a generalized formulation of VVC problems. Second, we give the ASA algorithm for that formulation. Finally, we provide the proof of convergence of the ASA algorithm.

5.2 A Generalized Formulation of VVC Problems

Without loss of generality, we can formulate the VVC problem as a mixed-integer nonlinear programming problem below:

$$\min_{x,y} f(x,y) \tag{5.1}$$

$$subject\,to: \quad g(x,y) = 0$$

$$h(x,y) \leq 0,$$

where x represents integer variables, e.g, tap positions and states of capacitors, and y represents continuous variables, which include power flows, voltages, etc. $g(x,y) = 0$ represents power flow equations. $h(x,y) \leq 0$ represents other security constraints, for example, voltage constraints. Because we can compute the value of y from x using power flow equations $g(x,y) = 0$, we can rewrite the problem above as:

$$\min_{x} f_1(x) \tag{5.2}$$

$$subject\,to: \quad h_1(x) \leq 0,$$

where functions f_1 and h_1 do not have closed forms, and are neither convex nor differentiable. If penalty terms are added for the violation of constraints, we can rewrite the problem as an unconstrained discrete optimization problem below:

$$\min_{x} H(x) \tag{5.3}$$

$$H(x) = f_1(x) + \eta \max(h_1(x), 0),$$

Note that if η is large enough, the optimal solution for the unconstrained optimization problem is also optimal for the constrained optimization problem above.

5.3 Approximate Stochastic Annealing for the Generalized Formulation of VVC Problems

Assume that x is a vector of size n, and $x(i) \in \{0, 1, 2, \ldots, m-1\}$. Initialize a matrix q with dimension $n \times m$, and $q(i,j)$ denotes the probability that $x(i) = j$. Set $q(i,j) = 1/m, \forall i, j$. Specify a non-negative decreasing sequence $\{T_k\}$ (i.e., annealing schedule), parameter sequences $\{\alpha_k\}$ and $\{\beta_k\}$ satisfying $0 \leq \alpha_k, \beta_k \leq 1, \forall k$. Select a sample size sequence $\{N_k\}$. Set iteration number $k = 0$. The procedure of the ASA algorithm is given below:

1. Sample N_k values of x, which form a set X_k, from q_k with probability $1-\beta_k$, and from q_0 with probability β_k.

2. For each x, calculate objective $H(x)$.

3. Update q by:

$$q_{k+1}(i,j) = \alpha_k \frac{\sum_{x \in X_k} e^{-H(x)/T_{k+1}} \hat{\phi}^{-1}(q_k, x) I\{x(i) = j\}}{\sum_{x \in X_k} e^{-H(x)/T_{k+1}} \hat{\phi}^{-1}(q_x, x)} + (1 - \alpha_k) q_k(i,j), \qquad (5.4)$$

   where

$$\hat{\phi}(q_k, x) = (1 - \beta_k)\phi_k + \beta_k \phi_0.$$

$$\phi_k = \prod_{i=1}^{n} q_k(i, x(i))$$

4. If a stopping condition is satisfied, then stop; otherwise, set $k = k + 1$ and go to step 1.

5.4 Analysis of Convergence

This chapter proves that under certain conditions, the probability matrix q_k converges to a matrix that assigns probability 1 to the optimal solution x^* with the minimum objective

value $H(x^*)$.

Let $\mathcal{F}_k$ be the σ-field generated by the sets of samples up to iteration $k-1$, i.e., $\mathcal{F}_k = \sigma\{X_1, X_2, \ldots, X_{k-1}\}$. The conditional probability and expectation are with respect to $\hat{\phi}(q_k, x)$ at the kth iteration. Let S denote the solution space of x. Define:

$$Y_k = \sum_{x \in S} e^{-H(x)/T_{k+1}} I\{x(i) = j\},$$

$$Z_k = \sum_{x \in S} e^{-H(x)/T_{k+1}},$$

$$\hat{Y}_k = N_k^{-1} \sum_{x \in X_k} e^{-H(x)/T_{k+1}} \hat{\phi}^{-1}(q_k,x) I\{x(i) = j\},$$

$$\hat{Z}_k = N_k^{-1} \sum_{x \in X_k} e^{-H(x)/T_{k+1}} \hat{\phi}^{-1}(q_k,x).$$

Let

$$g_{k+1}(x) = \frac{e^{-H(x)/T_{k+1}}}{\sum_{x \in S} e^{-H(x)/T_{k+1}}},$$

where $g_{k+1}(x)$ can be considered as a Boltzmann probability mass function of x. Let

$$\bar{g}_{k+1}(x) = \frac{e^{-H(x)/T_{k+1}} \hat{\phi}^{-1}(q_k,x)}{\sum_{x \in X_k} e^{-H(x)/T_{k+1}} \hat{\phi}^{-1}(q_k,x)},$$

where X_k is the set of samples generated at iteration k. Thus, we can consider $\bar{g}_{k+1}(x)$ as an empirical Boltzmann probability mass function that approximates $g_{k+1}(x)$. The proof of convergence follows the proof in [27]. However, we make some modifications to adapt it for the optimization problem in this chapter.

Firstly, we show that the sequence of idealized Boltzmann distribution $\{g_k\}$ converges to a limiting distribution that concentrates only on the optimal solution x^*.

Lemma 5.1. *If $T_k \to 0$ as $k \to \infty$, then $E_{g_k}[I\{x(i) = j\}] \to I\{x^*(i) = j\}, \forall i, j$ as $k \to \infty$.*

Proof. :

$$
\begin{aligned}
&|E_{g_k}[I\{x(i)=j\}] - I\{x^*(i)=j\}| \\
&\leq E_{g_k}[|I\{x(i)=j\} - I\{x^*(i)=j\}|] \quad since\, |E[\cdot]| < E[|\cdot|] \\
&= \sum_{x\neq x^*} |I\{x(i)=j\} - I\{x^*(i)=j\}| g_k(x) + |I\{x^*(i)=j\} - I\{x^*(i)=j\}| g_k(x^*) \\
&\leq \sum_{x\neq x^*} |I\{x(i)=j\} - I\{x^*(i)=j\}| \frac{e^{-H(x)/T_k}}{\sum_{x\in S} e^{-H(x)/T_k}} \\
&= \sum_{x\neq x^*} |I\{x(i)=j\} - I\{x^*(i)=j\}| \frac{e^{-H(x)/T_k}}{\sum_{x\in S} e^{-H(x)/T_k}} \\
&\leq \frac{\sum_{x\neq x^*} e^{-H(x)/T_k}}{e^{-H(x^*)/T_k} + \sum_{x\neq x^*} e^{-H(x)/T_k}} \\
&= \frac{\sum_{x\neq x^*} e^{(H(x^*)-H(x))/T_k}}{1 + \sum_{x\neq x^*} e^{(H(x^*)-H(x))/T_k}} \\
&\leq \sum_{x\neq x^*} e^{(H(x^*)-H(x))/T_k}
\end{aligned}
$$

Since for $x \neq x^*$, $H(x^*) - H(x) < 0$, then, as $T_k \to 0$, $H(x^*) - H(x))/T_k \to -\infty$. Therefore, $\sum_{x\neq x^*} e^{(H(x^*)-H(x))/T_k} \to 0$, since $|S|$ is finite. □

Secondly, we show that the estimated Boltzmann mass function $\frac{\hat{Y}_k}{\hat{Z}_k}$ is an asymptotically unbiased estimator of $\frac{Y_k}{Z_k}$.

Lemma 5.2. *If $N_k\beta_k \to \infty$ as $k \to \infty$, then*

$$E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k] \to \frac{Y_k}{Z_k}$$

as $k \to \infty$ w.p. 1.

Proof. :

$$\begin{aligned}\frac{\hat{Y}_k}{\hat{Z}_k} - \frac{Y_k}{Z_k} &= \frac{\hat{Y}_k}{\hat{Z}_k} - \frac{\hat{Y}_k}{Z_k} + \frac{\hat{Y}_k}{Z_k} - \frac{Y_k}{Z_k} \\ &= \frac{\hat{Y}_k(Z_k - \hat{Z}_k)}{\hat{Z}_k Z_k} + \frac{\hat{Y}_k - Y_k}{Z_k}.\end{aligned}$$

Taking the conditional probability with respect to $\mathcal{F}_k$ at both sides yield:

$$\begin{aligned}E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k] - \frac{Y_k}{Z_k} &= E[\frac{\hat{Y}_k(Z_k - \hat{Z}_k)}{\hat{Z}_k Z_k}|\mathcal{F}_k] + E[\frac{\hat{Y}_k - Y_k}{Z_k}|\mathcal{F}_k] \\ &= E[\frac{\hat{Y}_k(Z_k - \hat{Z}_k)}{\hat{Z}_k Z_k}|\mathcal{F}_k] \\ &\leq \frac{E[(Z_k - \hat{Z}_k)|\mathcal{F}_k]}{|Z_k|} \quad since\, \frac{\hat{Y}_k}{\hat{Z}_k} \leq 1 \\ &\leq \frac{1}{|Z_k|} E[(Z_k - \hat{Z}_k)^2|\mathcal{F}_k]^{1/2} \quad H\ddot{o}lder's\, inequality \\ &\leq \frac{1}{|Z_k|\sqrt{N_k}} E[e^{-2H(x)/T_{k+1}} \hat{\phi}^{-2}(q_k, x)|\mathcal{F}_k]^{1/2} \\ &= \frac{1}{|Z_k|\sqrt{N_k}} (\sum_{x \in S} e^{-2H(x)/T_{k+1}} \hat{\phi}^{-1}(q_k, x))^{1/2} \\ &= \frac{1}{\sqrt{N_k}} (\sum_{x \in S} ((1-\beta_0)\phi(q_k, x) + \beta_0 \phi(q_0, x))^{-1} g_{k+1}(x) \frac{e^{-H(x)/T_{k+1}}}{Z_k})^{1/2} \\ &\leq \frac{1}{\sqrt{N_k}} (\sum_{x \in S} (\beta_0 \phi(q_0, x))^{-1} g_{k+1}(x))^{1/2} \\ &\leq \frac{1}{\sqrt{N_k}} (\sum_{x \in S} (\beta_0 \phi(q_0, x))^{-1} g_{k+1}(x))^{1/2} \\ &= \frac{(\sum_{x \in S} \phi(q_0, x)^{-1} g_{k+1}(x))^{1/2}}{\sqrt{N_k \beta_0}}.\end{aligned}$$

Because $\phi(q_0, x)^{-1} g_{k+1}(x)$ is bounded and $|S|$ is bounded, $(\sum_{x \in S} \phi(q_0, x)^{-1} g_{k+1}(x))^{1/2}$ is bounded. Therefore $E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k] - \frac{Y_k}{Z_k} \to 0$ as $N_k \beta_0 \to \infty$. Thus, the desired result holds. □

Finally, we show that under some assumptions, the stochastic matrix q converges to the matrix that assigns unit mass to the optimal solution x^*.

Theorem 5.1. *Assume the following conditions hold:*

1. $\alpha_k > 0, \forall k$; $\sum_{k=0}^{\infty} \alpha_k = \infty$ *and* $\sum_{k=0}^{\infty} \alpha_k^2 < \infty$;

2. $T_k \to 0$ *as* $k \to \infty$;

3. $N_k \beta_k \to \infty$ *as* $k \to \infty$.

Then

$$q_k(i,j) \to I\{x^*(i) = j\}, \forall i, j \quad as\, k \to \infty\, w.p.\, 1.$$

Proof. :

Let $\eta_k := q_k(i,j) - I\{x^*(i) = j\}$. We can rewrite the recursive Equation (5.4) in the form below:

$$\eta_{k+1} = \eta_k - \xi_k, \tag{5.5}$$

where

$$\xi_k = \alpha_k(\eta_k + I\{x^*(i) = j\} - \frac{\hat{Y}_k}{\hat{Z}_k}) \tag{5.6}$$

Let $U_k = E[\xi_k|\mathcal{F}_k]$ and $\Delta_k = \xi_k - U_k$.

(i) Firstly, we show that $P(|\eta_k| > \epsilon, \eta U_k < 0\, i.o.) = 0, \forall \epsilon > 0$.

$$\begin{aligned} U_k &= E[\xi_k|\mathcal{F}_k] \\ &= \alpha_k(\eta_k + I\{x^*(i) = j\} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]) \qquad (5.7) \\ &= \alpha_k(\eta_k + I\{x^*(i) = j\} - \frac{Y_k}{Z_k} + \frac{Y_k}{Z_k} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]) \qquad (5.8) \end{aligned}$$

Then,

$$\eta_k U_k = \alpha_k(\eta_k^2 + \eta_k(I\{x^*(i) = j\} - \frac{Y_k}{Z_k}) + \eta_k(\frac{Y_k}{Z_k} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]) \tag{5.9}$$

Because η_k is bounded, $\eta_k(I\{x^*(i) = j\} - \frac{Y_k}{Z_k}) \to 0$ and $\eta_k(\frac{Y_k}{Z_k} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]) \to 0$ as $k \to \infty$ (Lemma 5.1 and Lemma 5.2). Thus, If $|\eta_k| > \epsilon$, we have $\eta_k U_k > 0$ when k is sufficiently large. Therefor, we have $P(|\eta_k| > \epsilon, \eta U_k = 0\, i.o.) = 0$.

(ii) Secondly, because $\eta_k + I\{x^*(i) = j\} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]$ is bounded and $\alpha_k \to 0$ as $k \to \infty$,

then $|U_k|(1+|\eta_k|)^{-1} \to 0$ as $k \to \infty$.

(iii) Thirdly,we show that $\sum_{k=1}^{\infty} E[|\Delta_k|^2] < \infty$.

By definition, we have

$$\Delta_k = \alpha_k(E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k] - \frac{\hat{Y}_k}{\hat{Z}_k})$$

$$\sum_{k=1}^{\infty} E[|\Delta_k|^2] = \sum_{k=1}^{\infty} \alpha_k^2 E[(\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k] - \frac{\hat{Y}_k}{\hat{Z}_k})^2]$$

Because $E[(\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k] - \frac{\hat{Y}_k}{\hat{Z}_k})^2]$ is bounded and $\sum_{k=1}^{\infty} \alpha_k^2 < \infty$, then, $\sum_{k=1}^{\infty} E[|\Delta_k|^2] < \infty$.

(iv) Finally, we show that

$$P(\liminf |\eta_k| > 0, \sum_{k=1}^{\infty} |U_k| < \infty) = 0.$$

From (i), we have:

$$\begin{aligned} |U_k| &\geq |\alpha_k(\eta_k + I\{x^*(i) = j\} - \frac{Y_k}{Z_k} + \frac{Y_k}{Z_k} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k])| \\ &\geq \alpha_k(|\eta_k| - |(I\{x^*(i) = j\} - \frac{Y_k}{Z_k})| - |\frac{Y_k}{Z_k} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]|) \end{aligned}$$

Let Ω be the set of paths generated by the ASA algorithm such that for each $\omega \in \Omega$, $\liminf |\eta_n| > 0$. Then for each $w \in \Omega$, there exists a positive integer N_1 such that $|\eta_k| > \delta, \forall k > N_1$. Because of Lemma 5.1 and Lemma 5.2, there exists a positive integer N_2 such that $|(I\{x^*(i) = j\} - \frac{Y_k}{Z_k})| + |\frac{Y_k}{Z_k} - E[\frac{\hat{Y}_k}{\hat{Z}_k}|\mathcal{F}_k]| < \alpha_k\delta/2, \forall k > N_2$. Let $N = \max(N_1, N_2)$, we have

$$\sum_{k=1}^{\infty} |U_k| = \sum_{k=1}^{N} |U_k| + \sum_{k=N+1}^{\infty} |U_k| \geq \sum_{k=1}^{N} |U_k| + \frac{\delta}{2} \sum_{k=N+1}^{\infty} \alpha_k = \infty. \tag{5.10}$$

Therefore, we have $P(\liminf |\eta_n| > 0, \sum_{k=1}^{\infty} |U_k| < \infty) = 0$.

Since we have the results in (i)-(iv), the main theorem in [17] shows that $\eta_k \to 0$ as $k \to 1$ $w.p.1$. □

Chapter 6

Summary and Conclusions

6.1 Summary

Finite horizon optimal planning problem is an important category of optimization problems in smart grid. Many well-known optimization problems in power systems fall into this category. This dissertation investigates two of them: the unit commitment problem and the voltage and reactive power control problem.

Unit commitment problem is a power generation planning problem for finding the optimal generation and service schedule of power generators to minimize total operational costs. This dissertation adapts the traditional lagrangian relaxation algorithm for parallel computing, which significantly reduces the computational time of solving UCPs, and make it possible for solving large scale security constrained unit commitment problems on-line within an acceptable time limit. We use the rolling horizon scheme to handle the volatile wind power generation. In the rolling horizon scheme, we solve UCP each hour using the updated wind power forecasts and load forecasts in the next 24 hours. Our test results show that the rolling horizon scheme yields solutions with smaller total operational costs than the traditional day-ahead scheme does.

In this dissertation, we model the voltage and reactive power control problem as a finite

horizon planning problem that finds the optimal planning of LTCs and SCs over a 24-hour horizon. Firstly, we use a VVC model to minimize energy losses with constant load model and operation limits constraints on LTCs and SCs. A novel stochastic search algorithm called Approximate Stochastic Annealing is proposed for solving the optimization model. ASA samples candidate solutions from a probability distribution over the solution space. The distribution function is then modified using a Boltzmann selection scheme. Asymptotically, the probability distribution converges to a degenerated distribution that assigns unit mass to the optimal solution. This dissertation also proposes a lagrangian relaxation based algorithm: operations limit constraints are relaxed to form a dual problem, which is solved by a combination of the subgradient method and the dynamic programming algorithm. We use the PG&E 69 power distribution system to illustrate the performance of both algorithms.

Secondly, this dissertation solves VVC problems with ZIP load model and switching costs. We modify the existing matrix-form power flow computation method to accommodate ZIP load model. We consider two objective functions: the minimization of power losses and the minimization of total power consumption. We also use the ASA algorithm for solving the VVC optimization problems. We compare the results from the ASA algorithm with those from the dynamic programming algorithm and the simulated annealing algorithm to illustrate its performance.

Thirdly, we discuss the convergence properties of the ASA algorithm for solving VVC problems.

6.2 Conclusions

The following conclusions are obtained from this dissertation:

- The traditional Lagrangian-Relaxation method for solving UCPs can be adapted for a parallel computing scheme, which greatly reduces the computational time for solving unit commitment problems, especially for solving large-scale security constrained unit

commitment problems.

- The rolling horizon scheme significantly reduces total operational costs when compared with the traditional day-ahead computing scheme. The parallel computing method is a good fit for the rolling horizon scheme, which requires solving UCPs within a short time window.

- The ASA algorithm can solve VVC problems with a moderate computational time but little compromise on the optimality of solutions.

- The Lagrangian Relaxation-Dynamic Programming is also effective for solving VVC problems with operation limit constraints. It yields solutions with very small duality gaps. However, the long computational time is a major disadvantage of this algorithm.

- We propose new formulations of VVC problems with switching costs, ZIP load model and different objective functions. Our computing results indicate that the ASA algorithm can effectively solve these new formulations in terms of both optimality and computational time.

6.3 Limitations and Future Work

Despite of the contribution of this dissertation for solving two finite horizon optimal planning problems, our work has some limitations, which are listed below. Meanwhile, we suggest potential methods to overcome those limitations.

- Some finite horizon optimal planning problems including UCP and VVC problems are mixed-integer programming problems, and they are NP-hard problems. Those algorithms in this dissertation do not solve the problem of the "curse of dimensionality", because their worst case time complexities are still non-deterministic polynomial. A potential method for overcoming the "curse of dimensionality" is the approximate dynamic programming algorithm (ADP) [6, 54]. As we mentioned earlier, many of

those optimal planning problems can be formulated as MDPs. Due to large state and action spaces, traditional dynamic programming methods may not be applicable. The approximate dynamic programming algorithm may be a good alternative for solving those MDPs.

- The wind power model we used in chapter 2 is a simple Gaussian model, which is a naive model for the wind power forecast. A more sophisticated wind power forecast model is necessary to test the real performance of the rolling horizon scheme.

- We can only scale up to 30 CPUs in the parallel computing method for UCP. The computational time does not decrease significantly when more CPUs are used, and the communication time among CPUs becomes the dominant factor in the total computational time. A detailed study on the job assignment and the communication schedule is needed to improve the scalability.

- The VVC problems in this dissertation are deterministic and assume accurate forecasts of electricity loads. However, in load forecast models, they are usually stochastic. Moreover, distributed renewable energy resources are increasingly involved in power distribution systems, and they are volatile. Therefore, stochastic VVC models may be better choices than deterministic models. And thus, stochastic control and planning algorithms are needed for solving those stochastic VVC models.

- As indicated in [27], ASA can solve stochastic optimization problems. Similarly, we may use ASA for solving stochastic VVC problems.

Bibliography

[1] A. G. Bakirtzis, P. N. Biskas, C. E. Zoumas, and V. Petridis, "Optimal power flow by enhanced genetic algorithm", *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 229-236, May 2002.

[2] M. E. Baran and M.-Y. Hsu, "Volt/Var control at distribution Substations", *IEEE Transactions on Power Systems*, vol. 14, no. 1, pp. 312-318, Feb. 1999

[3] M. E. Baran and F. F. Wu, "Optimal capacitor placement on radial distribution systems", *IEEE Transactions on Power Delivery*, vol. 4, no. 1, pp. 725-734, Jan. 1989.

[4] R. Barth, H. Brand, P. Meibom and C. Weber, "A stochastic unit-commitment model for the evaluation of the impacts of integration of large amounts of intermittent wind power", *International Conference on Probabilistic Methods Applied to Power Systems*, pp. 1-8, June 2006.

[5] D. Bertsekas, G. Lauer, N., Jr. Sandell and T. Posbergh, "Optimal short-term scheduling of large-scale power systems", *IEEE Transactions on Automatic Control*, vol. 28, no. 1, pp. 1-11, Jan. 1983.

[6] D. P. Bertskas, J. N. Tsitsiklis and J. Tsitsiklis, "Neuro-Dynamic Programmingm, 1st Edition", Athena Scientific, May 1996.

[7] J. Birge and F. Louveaux, "Introduction to stochastic programming", New York, Berlin, Heidelberg: Springer, vol. 2, 2000.

[8] C. C. Caroe and R. Schultz, "A two-stage stochastic program for unit commitment under uncertainty in a hydro-thermal power system", *Konrad-Zuse-Zentrum fur Informationstechnik*, 1998.

[9] P. Carpentier, G. Gohen, J.-C. Culioli and A. Renaud, "Stochastic optimization of unit commitment: a new decomposition framework", *IEEE Transactions on Power Systems*, vol. 11, no. 2, pp. 1067-1073, May. 1996.

[10] M. Carrión and J. M. Arroyo, "A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem", *IEEE Transactions on Power Systems*, vol.21, no.3, pp.1371,1378, Aug. 2006.

[11] H.-C. Chang, C.-C. Kuo, "Network reconfiguration in distribution system using simulated annealing", *Electric Power System Research*, vol. 29, pp. 227-238, May 1994

[12] A. I. Cohen,and V. R. Sherkat, "Optimization-based methods for operations scheduling", *Proceedings of the IEEE*, vol. 75, no. 12, pp. 1574-1591, Dec. 1987.

[13] A. I. Cohen and S. H. Wan, "A Method for Solving the Fuel Constrained Unit Commitment Problem", *IEEE Transactions on Power Systems*, vol. 2, no. 3, pp. 608-614, Aug. 1987.

[14] Cooper Power Systems, Cymdist, *http://www.cyme.com/software/cymdist/*

[15] L. Davis, "Handbook of Genetic Algorithms", New York: Van Nostrand 1991.

[16] Y. Deng, X. Ren, C. Zhao and D. Zhao, "A heuristic and algorithmic combined approach for reactive power optimization with time-varying load demand in distribution systems", *IEEE Transactions on Power Systems*, vol. 17, no. 4, pp. 1068-1072, Nov. 2002.

[17] S. N. Evans and N. C. Weber, "On the almost sure convergence of a general stochastic approximation procedure", *Bulletin of the Australian Mathematical Society*, vol. 34, pp. 335-342, 1986.

[18] A. Fabbri, T. GomezSanRoman, J. RivierAbbad,and V. H.MendezQuezada, "Assessment of the Cost Associated With Wind Generation Prediction Errors in a Liberalized Electricity Market", *IEEE Transactions on Power Systems*, vol. 20, no. 3, pp. 1440-1446, Aug. 2005.

[19] E. A. Feinberg and D. Genethliou, "Load Forecasting", *Applied Mathematics for Restructured Electric Power Systems: Optimization, Control, and Computational Intelligence* (J. H. Chow, F.F. Wu, and J.J. Momoh, eds.), Spinger, pp. 269-285, 2005.

[20] Z.-L. Gaing, "Discrete particle swarm optimization algorithm for unit commitment", *IEEE Power Engineering Society General Meeting*, vol. 1, pp. 418-424, 2003.

[21] J. Goez, J. Luedtke, D. Rajan and J. Kalagnanam, "Stochastic Unit Commitment Problem", *Tech Report, IBM Research Division*, Dec. 2008.

[22] D. E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", *Addison-Wesley, Reading, MA*, 1989.

[23] W. J. Hobbs, G. Hermon, S. Warner and G. B.Shelbe, "An enhanced dynamic programming approach for unit commitment", *IEEE Transactions on Power Systems*, vol. 3, no. 3, pp. 1201-1205, Aug. 1988.

[24] Y.-Y. Hsu and H.-C. Kuo, "Dispatch of capacitors on distribution system using dynamic programming", *IEE Proceedings, Transmission and Distribution*, vol. 140, no. 6, pp. 433-438, Nov. 1993.

[25] Y.-Y. Hsu and F.-C. Lu, "A combined artificial neural network-fuzzy dynamic programming approach to reactive power/voltage control in a distribution substation", *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1265-1271, Nov. 1998.

[26] Y.-Y. Hsu and C.-C Yang, "A hybrid artificial neural network-dynamic programming

approach for feeder capacitor scheduling", *IEEE Transactions on Power Systems*, vol. 9, no. 2, pp. 1069-1075, May 1994.

[27] J. Hu and H. S. Chang, "An Approximate Stochastic Annealing algorithm for finite horizon Markov decision processes", *2010 49th IEEE Conference on Decision and Control (CDC)*, pp. 5338-5343, Dec. 2010.

[28] Z. Hu, X. Wang, H. Chen and G. A. Tayloy, "Generation, Volt/Var control in distribution systems using a time-interval based approach", *IEE Proceedings, Transmission and Distribution*, vol 150, no. 5, pp. 548-554, Sep. 2003.

[29] IBM ILOG Optimizer, *http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer*.

[30] R. A. Johnson and D. W. Wichern, "Applied Multivariate Statistical Analysis, 6th edition", Prentice Hall, 2007

[31] S. A. Kazarlis, A. G. Bakirtzis and V. Petridis, "A genetic algorithm solution to the unit commitment problem", *IEEE Transactions on Power Systems*, Vol. 11, no. 1, pp. 83-92, Feb. 1996.

[32] J. Kennedy and R. Eberhart, "Particle swarm optimization", *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, pp. 1942-1948, Nov 1995.

[33] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm optimization", *Proceedings of IEEE International Conference on Neural Networks*, Vol. 4, pp. 4104-4108, Perth, Australia, 1997.

[34] S. Kirkpatric, C. D. Gelatt, and M. P. Vecci, "Optimization by simulated annealing", *Science*, vol 220, pp. 671-680, 1983.

[35] D. Kirshner, "Implementation of conservation voltage reduction at Commonwealth Edison", *IEEE Transactions on Power Systems*, vol 4, no. 4, pp. 1178-1182, Nov. 1990

[36] D. M. Lauria, "Conservation Voltage Reduction (CVR) at Northeast Utilities", *IEEE Transactions on Power Delivery*, vol 2, no. 4, pp. 1186-1191, Oct. 1987

[37] Z. Li and M. Shahidehpour, "Security-constrained unit commitment for simultaneous clearing of energy and ancillary services markets", *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 1079-1088, May. 2005.

[38] R.-H. Liang and C.-K. Cheng, "Dispatch of main transformer ULTC and capacitors in a distribution system", *IEEE Transactions on Power Delivery*, vol. 16, no. 4 pp. 625-630, Oct. 2001.

[39] R.-H. Liang and Y.-S. Wang, "Fuzzy-based reactive power and voltage control in a distribution system", *IEEE Transactions on Power Systems*, vol 18, no. 2, pp. 610-618, Apr. 2003.

[40] M. B. Liu, C. A. Canizares and W. Huang, "Reactive Power and Voltage Control in Distribution Systems With Limited Switching Operations", *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 889-899, May 2009.

[41] M. Liu, S. K. Tso and Y. Cheng, "An extended nonlinear primal-dual interior-point algorithm for reactive-power optimization of large-scale power systems with discrete control variables", *IEEE Transactions on Power Systems*, vol. 17, n0. 4, pp. 982-991, Nov. 2002.

[42] Y. Liu, P. Zhang and X. Qiu, "Optimal vol/var control in distribution systems", *International Journal of Electrical Power and Energy Systems*, vol. 24, no. 4, pp. 271-276, 2002.

[43] F.-C. Lu and Y.-Y. Hsu, "Fuzzy dynamic programming approach to reactive power/voltage control in a distribution substation", *IEEE Transactions on Power Systems*, vol. 12, no. 2, pp. 681-688, May 1997.

[44] H. Ma and S. M. Shahidehpour, "Unit commitment with transmission security and voltage constraints", *IEEE Transactions on Power Systems*, vol. 14, no. 2, pp. 757-764, May. 1999.

[45] J. E. Mitchell, "Branch and cut algorithms for combinatorial optimization problems", *Handbook of Applied Optimization*, pp. 65-77, 2002.

[46] NYISO, "Day-Ahead Scheduling Manual", June 2001.

[47] NYISO, "Final Report of the NYISO 2010 Wind Generation Study", 2010.

[48] NYISO, "Market Participants User's Guide", May 2011.

[49] W. Ongsakul and N. Petcharaks, "Unit commitment by enhanced adaptive Lagrangian relaxation", *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 620-628, Aug. 2004.

[50] The Open MPI Project, Open MPI, *www.open-mpi.org*.

[51] U.A. Öztürk, "The Stochastic Unit Commitment Problem: A Chance Constrained Programming Approach Considering Extreme Multivariate Tail Probabilities", *Ph.D. Thesis*, University of Pittsburgh, 2003.

[52] U.A. Öztürk, M. Mazumdar and B.A. Norman, "A solution to the stochastic unit commitment problem using chance constrained programming", *IEEE Transactions on Power Systems*, vol. 15, no. 3, pp. 1589-1598, Aug. 2004.

[53] N. P. Padhy, "Unit commitment-a bibliographical survey", *IEEE Transactions on Power Systems*, vol. 19, no. 2, pp. 1196-1205, May 2004.

[54] W. B. Powell, "Approximate Dynamic Programming: Solving the Curses of Dimensionality, 2nd Edition", Wiley, 2010

[55] C. A. Roa-Sepulveda, B. J. Pavez-Lazo, "A solution to the optimal power flow using simulated annealing", *International Journal of Electrical Power & Energy Systems*, vol.26, pp 47-57, Jan. 2003.

[56] P. A. Ruiz, C.R. Philbrick, K. W. Cheung and P. W. Sauer, "Wind power day-ahead uncertainty management through stochastic unit commitment policies", *Power Systems Conference and Exposition, 2009. PSCE '09. IEEE/PES*, pp. 1-9, Mar. 2009.

[57] P. A. Ruiz, C.R. Philbrick, E. Zak, K. W. Cheung and P. W. Sauer, "Uncertainty Management in the Unit Commitment Problem", *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 642-651, May. 2009.

[58] M. Sadeghi and G. A. Sarvi, "Determination of ZIP Parameters with Least Squares Optimization Method", *IEEE Electrical Power Energy Conference (EPEC)*, pp. 1-6, 2009

[59] K. P. Schneider, F. K. Tuffner, J. C. Fuller and R. Singh, "Evaluation of Conservation Voltage Reduction (CVR) on a national level", *Pacific Northwest National Lab Technical Report*, July 2010

[60] J. J. Shaw, "A direct method for security-constrained unit commitment", *IEEE Transactions on Power Systems*, vol. 10, no. 3, pp. 1329-1342, Aug. 1995.

[61] S. Sivanagaraju, Y. Srikanth and E. Jagadish Babu, "An efficient genetic algorithm for loss minimum distribution system reconfiguration", *Electric Power Components and Systems*, vol. 34, pp. 249-258, 2006

[62] W. L. Snyder, H. D. Powell and J. C.Rayburn, "Dynamic Programming Approach to Unit Commitment", *IEEE Transactions on Power Systems*, vol. 2, no. 2, pp. 339-348, May 1987.

[63] S. Takriti and J. R. Birge, "Using integer programming to refine Lagrangian-based unit commitment solutions", *IEEE Transactions on Power Systems*, vol. 15, no. 1, pp. 151-156, Aug. 2000.

[64] S. Takriti, John R. Birge and E. Long, "A Stochastic Model for the Unit Commitment Problem", *IEEE Transactions on Power Systems*, vol. 11, no. 3, pp. 1497-1508, Aug. 1996.

[65] C.-L. Tseng, S. S. Oren, C. S. Cheng, C. Li, A. J. Svoboda and R. B. Johnson, "A transmission-constrained unit commitment method in power system scheduling", *Decision Support Systems*, vol. 24, no. 3-4, pp. 297-310, 1999.

[66] J.-H. Teng, "A Direct Approach for Distribution System Load Flow Solutions", *IEEE Transactions on Power Delivery*, Vol. 18, no. 3, pp. 882-887, July 2003

[67] A. Tuohy, E. Denny, and M. O'Malley, "Rolling Unit Commitment for Systems with Significant Installed Wind Capacity", *Power Tech, 2007 IEEE Lausanne*, pp. 1380-1385.

[68] A. Tuohy, P. Meibom, E. Denny and M. O'Malley, "Unit Commitment for Systems With Significant Wind Penetration", *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 592-201, May. 2009.

[69] B. C. Ummels, M. Gibescu, E. Pelgrum, W. L. Kling and A. J. Brand, "Impacts of Wind Power On Thermal Generation Unit Commitment and Dipatch", *IEEE Transactions on Energy Conversion*, vol. 27, no. 1, pp. 44-51, Mar. 2007.

[70] J. Valenzuela and M. Mazumdar, "Statistical Analysis of Electric Power Production Cost", *IIE Transactions*, vol. 32, pp. 1139-1148, 2000.

[71] J. Wang, A. Botterud, V. Miranda, C. Monteiro and G. Sheble, "Impact of Wind Power Forecasting On Unit Commitment and Dispatch", *8th Int. Workshop on Large-Scale Integration of Wind Power into Power System*, Oct. 2009.

[72] J. Wang, M. Shahidehpour and Z. Li, "Security-Constrained Unit Commitment With Volatile Wind Power Generation", *IEEE Transactions on Power Systems*, vol. 23, no. 3, pp. 1319-1327, Aug. 2008.

[73] Wind Power Integration in Liberalised Electricity Markets Wilmar project, Available at *http://www.wilmar.risoe.dk*

[74] F. Zhang and F. D. Galiana, "Unit commitment by simulated annealing", *IEEE Transactions on Power Systems*, vol. 8, pp. 988-996, Aug. 1993.

[75] W. Zhang and Y. Liu, "Multi-objective reactive power and voltage control based on fuzzy optimization strategy and fuzzy adaptive particle swarm", *Journal of Electrical Power & Energy Systems*, vol. 30, no. 9, pp. 525-532, 2008.

[76] F. Zhuang and F. D. Galiana, "Towards a more rigorous and practical unit commitment by Lagrangian relaxation", *IEEE Transactions on Power Systems*, vol. 3, no. 2, pp. 763-773, May. 1988.