Spring 4-6-2022

An Analytical Methodology To Security Constraints Management In Power System Operation

Shubo Zhang
Cleveland State University

AN ANALYTICAL METHODOLOGY TO SECURITY CONSTRAINTS MANAGEMENT IN POWER SYSTEM OPERATION

SHUBO ZHANG

Bachelor of Science in Electrical Engineering

Cleveland State University

Spring 2017

Master of Science in Electrical Engineering

Cleveland State University

Fall 2017

submitted in partial fulfillment of requirements for the degree

Doctor of Philosophy in Electrical Engineering

at the

CLEVELAND STATE UNIVERSITY

May 2022

We hereby approve this dissertation for

SHUBO ZHANG

Candidate for the Doctor of Philosophy degree for the

Department of Electrical Engineering and Computer Science

and the CLEVELAND STATE UNIVERSITY'S

College of Graduate Studies by

Dissertation Chairperson, Dr. Zhiqiang Gao

Departement & Date

Dr. Hongxing Ye

Departement & Date

Dr. Ana Stankovic

Departement & Date

Dr. Lili Dong

Departement & Date

Dr. Siu−Tung Yau

Departement & Date

Dr. Sailai Sally Shao

Departement & Date

Student's Date of Defense April 6, 2022

ACKNOWLEDGMENTS

I would like to thank my advisors, Dr. Hongxing Ye and Dr. Zhiqiang Gao for their guidance, insight and extraordinary experiences during my research, and also for providing me opportunities to grow professionally.

I would like to thank my committee members, Dr. Ana Stancvic, Dr. Lili Dong, Dr. Sally Shao, and Dr. Siu-Tung Yau for their suggestions, support and contribution to this thesis.

My sincere gratitude goes to Dr. Fengyu Wang, and Dr. Anupam Thatte for their technical insights and discussions. It helped me tremendously throughout the projects with the Midcontinent Independent System Operator.

Thanks to my labmates, Zhigang Liu and Nitin Gupta, for their continuous support and collaboration during my academic work.

My sincere appreciation goes to my dear friends, Fei Yan, Yelun Han, Alex Kazminykh, Dr. Mohammad G. Ali, Adel Alhatlani, Khalil Elassi and Jim Lahey. They are the ones who have kept me grounded and reminded me of what is truly important in life.

My deepest gratitude goes to my beloved wife, Meiqing Zhang, for her unwavering love, support and tolerance. My parents were the light that brightened the lonely path of research. I could not imagine finishing this work without their love and support that kept me motivated and confident throughout this journey.

AN ANALYTICAL METHODOLOGY

TO SECURITY CONSTRAINTS MANAGEMENT

IN POWER SYSTEM OPERATION

SHUBO ZHANG

ABSTRACT

In a deregulated electricity market, Independent System Operators (ISOs) are responsible for dispatching power to the load securely, efficiently and economically. ISO performs Security Constrained Unit Commitment (SCUC) to guarantee sufficient generation commitment, maximized social welfare and facilitating market driven economics. The large number of security constraints would render the model impossible to solve under time requirement. Developing a method to identify the minimum set of security constraints without over committing is necessary to reduce Mixed Integer Linear Programming (MILP) solution time.

To overcome this challenge, we developed a powerful tool called security constraint screening. The proposed approach effectively filters out non-dominating constraints by integrating virtual transactions and capturing changes online in real-time or look-ahead markets. Security constraint screening takes advantage of both deterministic and statistical methods, which leverages mathematical modeling and historical data. Effectiveness is verified using Midcontinent Independent System Operator (MISO) data.

The research also presented a data-driven approach to forecast congestion patterns in real-time utilizing machine learning applications. Studies have been conducted using real world data. The potential benefit is to provide the day-ahead operators with a tool for supporting decision-making regarding modeling constraints.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Background

Electricity is the lifeblood of the modern world. Rapid economic development and production activities require reliable, affordable electricity. The US power grid is one of the most complex, crucial infrastructure ever built in human history. The power system includes more than 7300 power plants, nearly 160,000 miles of high-voltage transmission lines, and millions of low-voltage transmission lines and distribution transformers, which connect 145 million customers. In 2018, the revenue of US electric power industry was 408.5 billion dollars, which is around 2% of US Gross Domestic Product (GDP) [1]. Before the 1990s, most investor-owned electric utilities were regulated and vertically integrated. Owning both the generating units and transmission lines created regional monopolies. It was not possible for customers to purchase power elsewhere. As a result, the sell side sets the price and customers take the price. In order for customers to get a fair electricity rates, state regulators oversee how these electric utilities set electricity prices. Retail electricity prices in these areas are set based on recovering the utility's operating and investment costs, including a fair rate of return on those investments [2–4].

Beginning in the 1990s, many states in the US decided to deregulate (restructuring) their electricity systems to create competition and lower costs. This transition required electric utilities to sell their generating assets and led to the creation of independent energy suppliers that owned generators. Because power lines

are a natural monopoly, electric utilities held onto these assets to become transmission and distribution utilities, and they continue to be regulated for these natural monopoly functions. The biggest impacts resulting from deregulation were changes to retail and wholesale electricity sales, with the creation of retail customer choice and wholesale markets [5].

Energy markets are auctions that are used to coordinate the production of electricity on a day-to-day basis. In an energy market, electric suppliers offer to sell the electricity that their power plants generate for a particular bid price, while load-serving entities (the demand side) bid for that electricity in order to meet their customers' energy demand. Supply side quantities and bids are ordered in ascending order of offer price. The market is cleared when the amount of electricity offered matches the amount demanded, and generators receive this market price per megawatt hour of power generated. After deregulation, regional transmission organizations (RTOs) or Independent System Operators (ISOs) replaced utilities as grid operators and became the operators of wholesale markets for electricity [6, 7].

ISOs typically run two energy markets: the day-ahead markets (DAM) and real-time markets (RTM). The day-ahead market, which represents about 95 percent of energy transactions, is based on forecasted load for the next day and typically occurs the prior morning in order to allow generators to prepare for operation. The remaining energy market transactions take place in the real-time market, which is typically run once every hour and once every five minutes to account for real-time load changes that must be balanced at all times with supply [8, 9].

Energy markets are used by ISOs to determine which units to dispatch, or run, and in what order. In the day-ahead market, ISOs compile the list of generators available for next-day dispatch and order them from least expensive to most expensive to operate. Dispatching units by lowest cost allows the market to meet energy demand at the lowest possible price. During periods of high demand, wholesale prices

rise accordingly because more high-cost units need to be dispatched in order to meet the electric load. ISO performs Security Constrained Unit Commitment (SCUC) to guarantee sufficient generation commitment, maximized social welfare and facilitating market driven economics. SCUC is formulated as a Mixed Integer Linear Programming (MILP) problem. Security constraint is mathematical formulation representing power delivered by transmission lines must be kept under thermal limit during any particular contingency. The large number of security constraints would render the model impossible to solve under time requirement. The development of a method to reduce MILP solution time by identifying the minimum set of security constraints without undercommitting is imperative [10–19].

Security Constrained Unit Commitment refers to the economic scheduling of generating units for serving the hourly load demand while satisfying temporal and operational limits of generation and transmission facilities in power systems. In a vertically integrated utility, regulatory departments apply SCUC for minimizing the operating cost. However, in deregulated market, SCUC is utilized by ISOs/RTOs to clear real-time and day-ahead markets. The objective of SCUC is to maximize the social welfare's based on energy and price bids submitted by market participants, generation supplier and load demands. SCUC is discussed extensively in the literature and well implemented in daily power system operation [20], [21]. It is a challenging task to obtain optimal solution in an acceptable time due to the enormous size of the problem. In general, SCUC must satisfy the following constraints [21–23]:

- system-wide power balance
- generation unit physical limits
- spinning and operating reserve requirements
- generation unit minimum on/off constraints
- generation unit start-up/shut-off characteristics

- bus voltage limits
- transmission flow constraints
- system operation constraints for credible contingencies

SCUC, by nature, is a non-convex and large scale mixed integer optimization problem. It belongs to the np-hard problem class. Recent improvements in modeling, optimization solver, and efficient algorithm enable system operators to obtain optimal or near-optimal solutions. The SCUC problem can be solved by Lagrangian Relaxation (LR) and Mixed Integer Linear Programming (MILP). Performance comparisons can be found in [24]. MISO has adopted MILP approach for the Co-optimized Energy and Ancillary Service Market since 2009 [25–30]. Day Ahead Market (DAM) is a financial market, and the market-clearing process requires solving SCUC problem for all planned operation periods. A near-optimal solution must be obtained within the timeframe.

In DAM, security constraints are defined as power flow transmission constraint for transmission lines considering N-1 security rule according to North American Electric Reliability Corporation (NERC). There are over, 15000 security constraints formulated for MISO's DAM. Incorporating all security constraints in the model could cause computational intractability, which means optimality and solution time requirements cannot be guaranteed. Reducing non-binding hard constraints often help reduce solution time for MILP problems. The goal of this research is to effectively reduce the size of security constraint set in the SCUC model and improve solution quality, while maintaining solution integrity. From experience, only a small fraction of security constraints are binding. MISO identified over 500 routinely binding and critical security constraints per interval, known as "watchlist". Additional constraints may be identified and added to watchlist through the interaction between Simultaneous Feasibility Test (SFT) [31–34] and DAM Security Constrained Economic Dispatch (SCED) [20, 21, 35, 36].

Figure 1.1 illustrates the typical approach to handle SCUC by decomposing the original problem into master problem and subproblems [24, 34, 37–39]. It is an iterative approach. In addition to the constraints mentioned previously, the master problem contains hundreds and thousands of security constraints as well. Solving the master problem will determine the base case unit commitment. Subproblems take the base case solution to evaluate whether any network constraints (power flow constraints in base case topology) or security constraints (power flow constraints in any creditable contingency) cause violation. Violated constraints will be added back to the master problem and be solved again. As a criterion for convergence/termination, there must be no more violations of constraints. Finally, the master problem solution is the optimal unit commitment and dispatch subject to all the physical limits and robust against all the creditable contingencies. This approach may require quite a few iterations to reach convergence, and each iteration may take longer than the previous one. Typically, ISOs require DAM SCUC to be solved under 20 minutes with less than 0.1% MILP gap. This approach usually cannot guarantee feasibility if too many contingencies are included. Therefore, empirical and heuristic approaches are applied to select the creditable contingencies. Security constraints may be overlooked in DAM SCUC, and become binding in real-time SCUC, resulting in divergence between markets and reduced social welfare. However, due to the simplistic nature, it is still being implemented as a typical industrial practice.

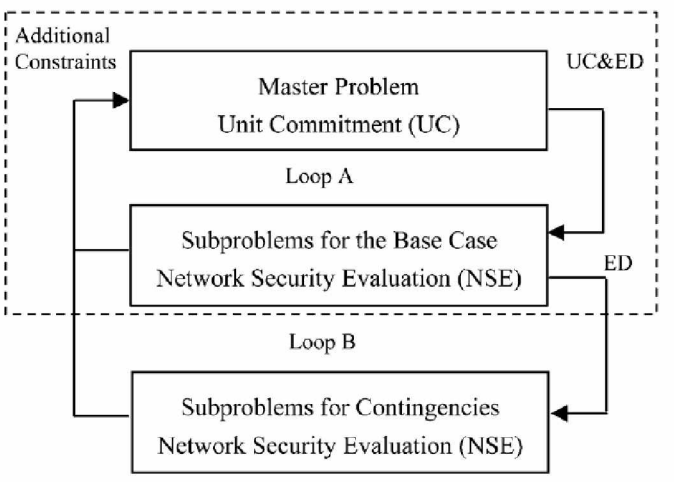

Figure 1.1: Decomposition approach for SCUC

Watchlist security constraints is an empirical approach to handle long solution time. It can be inaccurate and heavily dependent on operator's experience. Security constraint that is excluded in DAM and becomes binding in RTM will create congestion cost. It is an indicator that the SCUC model ran in DAM is not robust against uncertainty and contingency can create sub-optimal solution. Transmission congestion occurs when there is insufficient transmission capacity to deliver lower-cost generation resources to consumers, requiring the use of higher-cost generators closer to customers [40–42]. This increases the price of electricity in congested areas, as reflected in higher locational marginal prices and higher electricity prices for consumers [6, 43–49]. Table 1.1 shows the transmission congestion costs for Regional Transmission Organizations (RTOs) from 2016-2018 [50]. The goal of this research is to develop a method to find the minimal set of security constraints that can help decrease MILP solution time, while maintaining solution robustness against all N-1 contingencies.

1.2 Solution Approach and Industrial Practice

The overwhelming number of security constraints have attracted much atten-

RTO	2016	2017	2018
ERCOT	497	976	1260
ISO-NE	38.9	41.4	64.5
MISO	1400	1500	1400
NYISO	529	481	596
PJM	1023.7	697.6	1310
SPP	273.7	405.3	380.9
Total	**3762.3**	**4101.3**	**5011.3**

Table 1.1: Transmission Congestion Costs (in millions) for RTOs from 2016-2018

tion in literature. The concept of Umbrella constraint is introduced in [51] and [52]. The umbrella constraints are dominating constraints and defined as the minimum set of constraints that shapes the feasibility region of the original problem. Binding constraints is a subset of umbrella constraints, which can only be determined after solving the original MILP problem. Authors in [53] propose an efficient method to eliminate non-active constraints in standard SCUC and [54, 55] further take uncertainty into consideration. The author in [56] developed methods to eliminate large numbers of linear security constraints in the high-dimensional mixed-integer SCUC problem. Heuristic approaches are applied to remove potentially non-binding constraints in [34] and [38], which addresses the conservatism caused by large number of constraints that are maybe non-binding in nature, but dominating in the reduced constraint set. It is seen that warm start and setting proper lazy constraint could improve the SCUC solution performance [57, 58]. Our work in this proposal is an extension of [53–55].

Congestion management is an important subject for grid operators. Congestion occurs when there is a lack of transmission line capacity to deliver electricity without exceeding thermal, voltage and stability limits designed to ensure system reliability [59–68]. Congestion in ISO wholesale markets is managed through Security Constrained Economic Dispatch by providing Locational Marginal Prices (LMPs). Grid operators and market participants can see where congestion results in high

prices and act accordingly. Thus, accurate forecasting of congestion is important for decision-making. The ISOs run a two settlement market, that is, there are two sets of LMPs for each node, one in the Day-Ahead Market (DAM) and the second in the Real-Time Market (RTM). Congestion in the DAM may not match that in the RTM due to various reasons, such as outages or topology changes. Inconsistent constraints modeling between day-ahead (DA) and real-time (RT) could lead to financial divergence, operational issues, and uneconomic commitments. Constraints binding in real-time but which are not modeled in day-ahead, may lead to significant excess congestion funding (ECF). This results in inefficiency in the market. Congestion forecasting could help lower the financial divergence, as well as improve the commitment solution in the DAM [69–80].

1.3 Proposed Security Constraint Screening

We aim to reduce SCUC problem size by reducing security constraints set in the SCUC formulation. We propose a two-step screening approach to identify and eliminate non-dominating security constraints. Step 1 is a numerical method that employs necessary conditions to identify the potential dominating security constraints. Step 2 solves an LP problem to find the truly dominating constraints that further shrinks the pool of potentially dominating security constraints.

We also introduce a data-aided lazy constraint setting to further improve the performance. The contributions of this proposal are three-fold. First, we proposed a data-driven approach to determine the confidence interval of net power injection for the calculation of power flow. Second, we employed lazy constraints to resolve potential inaccuracy issue caused by data driven approach. Third, we carried out simulations and validate the performance using the real data from MISO.

A data drive approach is employed. Persistent study was performed on MISO historical data. Previous day generation were used to predict the next day unit

commitment and dispatch. The range of net power injections on individual buses were narrowed down, resulting in even fewer security constraints. The inaccuracy resulted by statistical approach was addressed by lazy constraint setting in the solver. Varies warm start strategies in combination with security constraint screening and data driven approach were tested on real data from MISO.

The contribution of this research includes:

1) A two-step approach to identify the dominating security constraints in SCUC formulation. Step 1 is deterministic method. Step 2 requires solving a group of LP problem, which is time-consuming, offline method used to pre-process the model to extract key features. Most conservative approach implemented were based on the asset's physical limitation. Online method benefited from the solution of offline method. It used numerical method to rapidly determine whether a security constraint has the potential of being dominating in the model.

2) Proposed the offline and online application of the two step security constraints method. The offline method pre-processes the model to extract key features. Most conservative approaches were implemented based on the asset's physical limitation. Online method benefited from the solution of the offline method. It used a numerical method to rapidly determine whether a security constraint has the potential of being dominating in the model.

3) Unit commitment status, dispatch quantity, and virtual power injection information were approximated by a data-driven approach to overcome the conservatives of net power injection interval, which was usually calculated according to unit minimum and maximum output. An optimistic dominating constraint set was derived. Lazy constraints were flagged in the optimization model to address the inaccuracy caused by the optimism from statistical methods.

4) Performed generator, virtual and load deviation analysis after solving base case

SCUC. The result of the deviation analysis was used to construct a pool of potentially binding security constraints in real time.

5) Evaluated the financial impact of a constraint if it became binding in real time.

6) Implemented price constraint (budget of uncertainty) after solving base case SCUC. The price constraint will further reduce the number of potentially binding constraint in real time.

7) Developed a machine learning module that was capable of determining the power flow upper/lower bounds of potentially binding constraints.

Figure 1.2 shows the implementation timeline of the proposed methods.

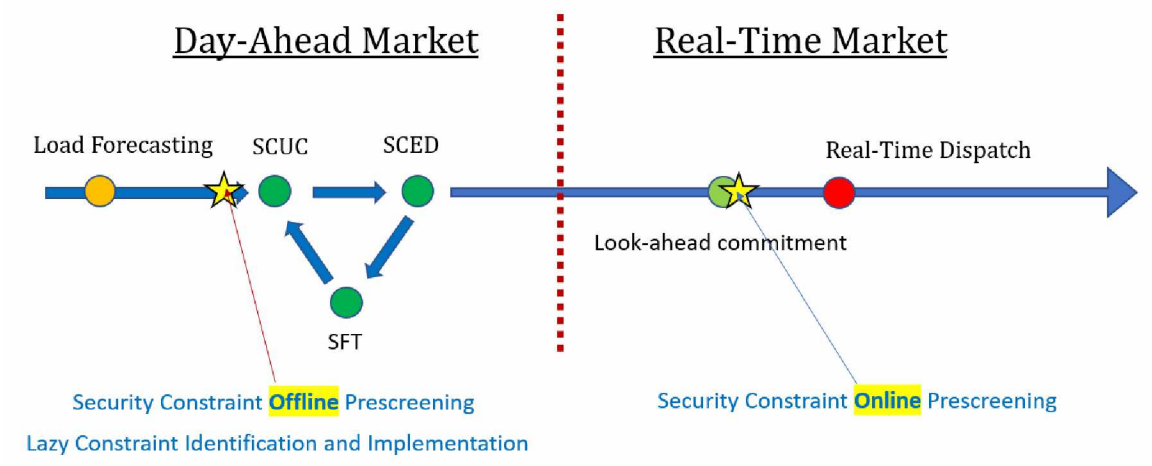

Figure 1.2: Congestion Forecasting Tool Implementation

1.4 Proposed Congestion Forecasting Tool

DAM is a forward market, which determines the optimal commitment and dispatch for the next day electricity market, through Security Constrained Unit Commitment (SCUC) and Security Constrained Economic Dispatch (SCED). Moreover, for each hour, contingency analysis is conducted by Simultaneous Feasibility Test (SFT). DA market utilizes forecasted load, forecasted renewable power, planned generator outages and credible contingencies to best estimate the state of the power system

for the next day. However, due to forecasting error and uncertainties, RT market diverges from DA planning. The gap between the two markets, or divergence, creates market inefficiency.

In recent years, machine learning techniques have been implemented in the power system operation field to tackle the problem of cyberattack detection, voltage instability, and power quality disturbance. Traditional deterministic approaches cannot deliver solutions in acceptable time given the large and complex power transmission network. The demand to summon a quick and relatively accurate solution becomes a dominating force that drives the development of heuristic methods in the modern power system [81–93]. Optimal power flow problem is a large scale root finding problem that is directly tied to the stability of the system. It is typically solved by deterministic methods such as Newton-Raphson, Gaussian-Seidel, nonlinear programming and interior point methods. Heuristic methods include swarm and bio-inspired, evolutionary-inspired, and hybrid optimization techniques [94–96]. Load forecasting and renewable forecasting have been a traditional strong field utilizing machine learning techniques [97–101]. Short term forecasting has proven to be conservative and reliable. However, long term load forecasting is more investigated in the planning period instead of day to day operation.

The goal of the research is to create a tool that estimates the gap between DA and RT markets, and offers a solution that is robust against the worst contingencies, yet the cost is within financial budget. Figure 1.3 illustrates the high level work flow of the congestion forecasting tool. It is, to the author's best knowledge, this is a novel approach to quantify the markets' divergence.

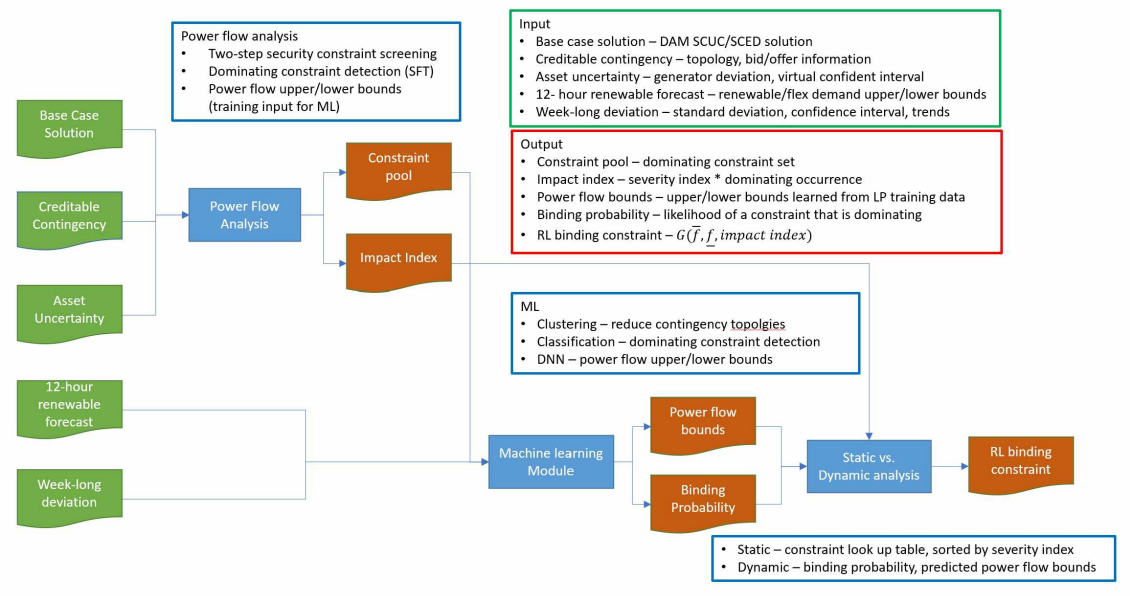

Figure 1.3: Congestion Forecasting Tool High Level Work Flow

The goal of the congestion forecasting tool is to use day-ahead market information and forecasted real-time information to predict the security constraints that can be binding in real time. The overall strategy is composed of two parts, i.e. deterministic portion (power flow analysis) and heuristic portion (machine learning).

1. Deterministic portion: Utilizing solutions from DAM SCUC, SFT and two-step security constraints screening methods to outline the potentially binding security constraint pool.

2. Machine learning portion: With forecasted information fed to the deep neural network, we calculated the power flow upper and lower bounds of all potentially congested lines, which are determined from deterministic portion. Furthermore, we calculated the probability of a line becoming binding in real time.

3. Analysis: The probability of a constraint becoming binding, as well as the impact of the constraint, can be calculated using a "severity index". In real-time operations, system operators can refer to the severity index to determine what constraints to be aware of.

The sequence of congestion forecasting tool execution is.

1. Solve DAM SCUC (base case).

2. Execute the enhanced SFT module with (pre-selected) credible contingencies.

3. Apply security constraint screening protocols to determine the potentially binding constraint pool.

4. Use forecasted data and potentially binding constraints as input to the machine learning algorithm to determine the real time binding constraints with severity analysis.

1.5 Overview

The dissertation follows the following structure. Section II presents the classical SCUC formulation. In section III, a security constraint screening method and its offline-online implementation are presented. Historical data and lazy constraints are employed to tighten the security constraint pool in the optimization model. Lazy constraint settings are incorporated in the optimization solver. On IEEE-118 and MISO test cases, the proposed screening method, data drive approach, and varying warm-start strategies are implemented to solve DAM SCUC. Section IV proposes the price constraint that addresses the economic concerns in the security constraint screening process. The virtual confidence interval considers market participant behavior and strategy, which provides economic insights and shrinks the energy traded in the day-ahead market. Section V discusses the static analysis of the consequences when a market participant does not follow the system operator's order. Derived impact index of security constraint can provide system operator a metric when considering which ones are more likely to bind in real time and exert more stress on the system. Section VI proposes a contingency screening framework (congestion forecasting) utilizing machine learning techniques. The classification algorithm can detect potentially binding constraints in real time, and a regression model was used to predict the power flow

bounds of the targeted constraints. The paper concludes with Section VII, which discusses future work that can improve and tighten the proposed model to bridge the divergence between day ahead and real time market.

CHAPTER II

SECURITY CONSTRAINED UNIT COMMITMENT

2.1 SCUC Formulation

This section shows the classic SCUC formulation. $\mathcal{M}, \mathcal{I}, \mathcal{L}$ and $\mathcal{T}$ denotes the sets of buses, generators, transmission lines and planning period. Generators have convex production cost curve, and usually modeled as a piecewise function with a set of k segments.

$$\min_{\{p_{it}, I_{it}\}} \quad \sum_{i,t} \left(N_{i0} I_{it} + \sum_{m} IC_{ik} p_{it,k} + csu_{it} + csd_{it} \right) \tag{2.1}$$

$$\sum_{i} p_{it} = D_t, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.2}$$

$$P_i^{\min} I_{it} \leq p_{it} \leq P_i^{\max} I_{it}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.3}$$

$$0 \leq p_{it,k} \leq P_{ik}^{\max}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T}, k \in \mathcal{K} \tag{2.4}$$

$$p_{it} = \sum_{m} p_{it,k}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T}, k \in \mathcal{K} \tag{2.5}$$

$$y_{it} - z_{it} = I_{it} - I_{i,t-1}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.6}$$

$$y_{it} + z_{it} \leq 1, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.7}$$

$$csu_{it} = ST_i y_{it}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.8}$$

$$csd_{it} = SD_i z_{it}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.9}$$

$$p_{it} + sr_{it} \leq P_i^{\max}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.10}$$

$$0 \leq sr_{it} \leq MSR_i I_{it}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.11}$$

$$or_{it} = sr_{it} + (1 - I_{it})QSC_i, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.12}$$

$$\sum_i sr_{it} \geq SR_t, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.13}$$

$$\sum_i or_{it} \geq OR_t, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.14}$$

$$p_{it} - p_{i,t-1} \leq y_{it} P_i^{\min} + (1 - y_{it})RU_i, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.15}$$

$$p_{i,t-1} - p_{it} \leq z_{it} P_i^{\min} + (1 - z_{it})RD_i, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.16}$$

$$UT_i = \max\{0, \min[NT, (MU_i - TU_{i0})U_{i0}]\}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.17}$$

$$\sum_{t=1}^{UT_i}(1 - I_{it}) = 0, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.18}$$

$$\sum_{m=t}^{t+MU_i-1} I_{im} \geq MU_i y_{it}, \quad \forall t = UT_i + 1, ..., NT - MU_i + 1, i \in \mathcal{I} \tag{2.19}$$

$$\sum_{m=t}^{T}(I_{im} - y_{it}) \geq 0, \quad \forall t = NT - MU_i + 2, ..., NT, i \in \mathcal{I}, m \in \mathcal{M} \tag{2.20}$$

$$DT_i = \max\{0, \min[NT, (MD_i - TD_{i0})(1 - U_{i0})]\}, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.21}$$

$$\sum_{t=1}^{DT_i} I_{it} = 0, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.22}$$

$$\sum_{m=t}^{t+MU_i-1} (1 - I_{im}) \geq MD_i z_{it},$$

$$\forall t = DT_i + 1, ..., NT - MD_i + 1, \quad \forall i \in \mathcal{I}, t \in \mathcal{T} \tag{2.23}$$

$$\sum_{m=t}^{T}(1 - I_{im} - z_{it}) \geq 0, \quad \forall t = NT - MU_i + 2, ..., NT, i \in \mathcal{I} \tag{2.24}$$

$$\underline{F}_l \leq \sum_m \Gamma_{l,m} P_m^{\text{inj}} \leq \overline{F}_l, \quad \forall l \in \mathcal{L} \tag{2.25}$$

The objective function (2.1) is a minimization of generator production cost. I_{it} is the on/off status indicator for generator i at hour t. N_{i0} is the no-load cost. IC_{ik} is the k^{th} incremental cost from piecewise linear approximation of the generator cost curve. Equation (2.2) is the load balance constraint, where power produced by

individual generators, p_{it}, must match the overall load demand, D_t, for each planned period. (2.3), (2.4) and (2.5) are the resource limitations, generator output must be within its physical limits, $P_i^{\min}$ and $P_i^{\max}$. y_{it} and z_{it} are the start-up/shut-down indicators for generator i at hour t. (2.6) is the logic constraint that links unit status with start-up/shut-down action. (2.7) establishes the mutually exclusive relationship between start-up and shut-down action. (2.8) and (2.9) are cost constraints that are associated with start-up/shut-down action, where csu_{it} and csd_{it} are the cost variables. (2.10) to (2.14) are reserve constraints. sr_{it}/or_{it} is spinning/operating reserve variable co-optimized with power generation. SR_t/OR_t is the hourly reserve requirement. Spinning reserve is a apart of generation capacity. (2.10) establishes that such relationship, power and reserve provided by a generator cannot exceed its physical limit. Spinning reserve is constrained by deliverability, as shown in (2.11). Operating reserve is provided by spinning reserve (online) and offline reserve (2.12). Sufficient system-wide reserve, SR_t/QR_t, must be met in (2.13) and (2.14). (2.15) and (2.16) are ramping constraints, where RU_i/RD_i is the ramp-up/ramp-down limit of the generator i. If the generator starts-up or shut-down, minimum generation requirement must be satisfied, otherwise, ramping limits must be respected. (2.17) to (2.24) are the generator minimum on/off requirements, NT, UT, DT are the number of periods, uptime and downtime of the generator, MU and MD are the minimum uptime and minimum downtime constants. (2.25) is the security constraint that is discussed in details in the next section.

2.2 Security Constraint Formulation

The power flow f_l in line l can be calculated based on equation as (2.26). The literature in [102] discussed the formulation and implementation extensively.

$$f_l = \sum_m \Gamma_{l,m} P_m^{\text{inj}}, \tag{2.26}$$

where $\Gamma_{l,m}$ is the shift factor element, for line l with respect to bus m. Shift factor is the sensitivity of the line flows to the change in injections at the buses. It is calculated as follows. For generality, line and bus subscripts l, m are ignored. f is the power flow vector for $|\mathcal{L}|$ branches, P^{inj} is the power injection vector for $\mathcal{M}$ buses. $\mathbf{K_L}$ is the bus to branch incidence matrix, dimension is $|\mathcal{M}|\times|\mathcal{L}|$. $\boldsymbol{X}$ is the branch reactance matrix and the dimension is $|\mathcal{L}|\times|\mathcal{L}|$. $\boldsymbol{c}$ is the branch-phase shifter incidence matrix, while the dimension is $|\mathcal{L}|\times$ a number of phase shifters. γ is the phase angle vector, which is a number of phase shifters $\times 1$.

$$f_l = \boldsymbol{X}^{-1}\mathbf{K}_\mathbf{L}^T\theta - \boldsymbol{c}\frac{\gamma}{\boldsymbol{X}} \tag{2.27}$$

$$P^{\text{inj}} = \boldsymbol{B}\theta \tag{2.28}$$

$$f_l = \boldsymbol{X}^{-1}\mathbf{K}_\mathbf{L}^T\boldsymbol{B}^{-1}P^{\text{inj}} - \boldsymbol{c}\frac{\gamma}{\boldsymbol{X}} \tag{2.29}$$

$$f_l = \boldsymbol{\Gamma}P^{\text{inj}} \tag{2.30}$$

where

$$\boldsymbol{B} = \mathbf{K}_\mathbf{L}\boldsymbol{X}^{-1}\mathbf{K}_\mathbf{L}^T \tag{2.31}$$

The dimension of shift factor matrix $\boldsymbol{\Gamma}$, is $|\mathcal{L}|\times|\mathcal{M}|$. Shift factor shows how the flow in the branch will change if the injection at the bus changes by 1 MW. Because the reference bus always makes up for the change in the injection, shift factor values are dependent on the location of the reference bus. For detailed formulation and calculation methods, please refer to [21, 103–106].

P_m^{inj} is the net power injection at bus m, calculated as follows,

$$P_m^{\text{inj}} = \sum_{i\in m} P_i + \sum_{k\in m} P_j^v - \sum_{n\in m} D_n - \sum_{f\in m} D_f^{\text{flex}} \tag{2.32}$$

where P_i is the generator's dispatch, P_k^v is virtual bid/offer, D_n is the fixed

demand, and D_f^{flex} is the price-sensitive demand. i, j, n and f are indices for generator, virtual, fixed demand, and price-sensitive demand, respectively.

Security constraints are power flow constraints under particular contingencies. The formulation is the same as (2.25), except that shift factor matrix and power injection are updated. A line outage will change network topology, which means the shift factor matrix needs to be recalculated to produce different power flow constraints. Generators can also be disconnected from the grid due to a line outage; dispatch in base case is no longer viable. Hence, SCUC usually accounts for N-1 line outage contingencies to ensure operation security, i.e. power deliverability under contingency. Power flow under contingency must be within its contingeny limit, i.e,

$$\underline{F}_l \leq f_l \leq \overline{F}_l \tag{2.33}$$

For example, in IEEE-118 bus test system, there are 186 transmission lines. Thus, the number of N-1 security constraints that needs to be included in SCUC formulation is 69,192 = (186 + 186*185)*2, per period. For a real-world power grid with 3,000 transmission lines, 18 million security constraints needs to be modeled per period. SCUC models are impossible to solve with such constraint set. ISOs generally develop a security constraint watchlist from experience and heuristic method. In order to get a converged solution under time restriction, only hundreds of security constraints can be included for each time interval in the SCUC model. A driving force behind this research is the desire to guarantee security and improve solution quality.

CHAPTER III

TWO-STEP SECURITY CONSTRAINT SCREENING

It is a challenge to deal with the large number of security constraints in real world systems. Some ISOs/RTOs develop special procedures to identify creditable security constraints and add only those constraints into the DAM SCUC model. Iterative approaches are often employed to conduct Simultaneous Feasibility Test (SFT) so that any additional security constraint causing violation can be added to the model eventually. For instance, MISO evaluates outages and other relevant information to build the watchlist, which is a much smaller subset of the full set of security constraints. However, the system operator may still confront situations where the optimization engine cannot find an acceptable solution within the time limit. There is an emerging and practical need to find acceleration techniques to improve the optimization performance. In this section, we present a two-step approach to build the watchlist.

The two-step method introduced in this proposal is able to identify the dominating constraints in the watchlist, and the watchlist constraints are reduced to its least set. Online method takes advantages of the dual variable of the dominating security constraints. Upper limit of power flow can be checked as soon as net power injection becomes available.

Persistence studies were conducted on the generator output on MISO system and previous day commitment is used to make predictions for current day commitment. Implementation of statistical method can further reduce the conservatives of

dominating watchlist security constraint set. To resolve the constraints that may have been overlooked, we would check the power flow and set more tolerances in the commercial MILP solver to be lazy.

Step 1 is based on Theorem 3 from [54] stating the necessary condition for line congestion, to effectively identify potentially dominating security constraints set, denoted as J_1, $J_1 \subseteq U$. U is the watchlist security constraint set. In Step 1, it requires power transfer distribution factor (PTDF) and net power injection. We have to consider uncertainty into the interval of net power injection, which is calculated based on generating unit's physical limitation, virtual power bid/offer and forecasted load. Step 1 is numerical method, it is computationally effective and can take advantage of parallel processing.

Step 2 is inspired by Theorem 2 of [53] where a security constraint is inactive if and only if the feasible region of the problem is unaffected by eliminating/adding the constraint. Dominating security constraints are governed by topology; they are independent of SCUC problem and are all linear constraints. Step 2 determines the maximum power flow on a branch by solving a Linear Programming (LP) problem in respect to all other security constraints and load balance constraint. If the resulting power flow is beyond transmission capacity limit, the security constraint is a dominating constraint, otherwise redundant. Truly dominating constraints set resulted from step 2 is denoted as J_2. We have the following set relationship $J_B \subseteq J_2 \subseteq J_1 \subseteq U$, where J_B is the binding security constraint set that can only be obtained after optimization.

Step 2 can be extended to offline/online methods. Offline method uses net power injection interval in relation to resource physical limits and maximum virtual bidding quantity, determining a dominating constraint set. Feasibility set remains unchanged if real time net power injection falls within the range. Dual variable, λ, is calculated and stored while solving the offline method. Online method is implemented

when real time net power injection becomes available, the upper bound of power flow, f_l, can be calculated. To maintain conservativeness, we can consider that the potential of a security constraint is binding as a fraction between f_l and F_l, represented by an arbitrary variable, α. If $\frac{f_l}{F_l} \geq \alpha$, security constraint l is considered as potentially binding, otherwise, possibly redundant. Then it would be set as a lazy constraint in the SCUC calculation.

3.1 Step 1: Necessary Condition of Transmission Line Congestion

Given the PTDF and relative accurate load forecasting, net power injection interval solely depends on generator commitment and virtual bid/offer. In order to identify all the potentially dominating constraints, we choose the most conservative approach to determine net power injection interval by taking the physical limits of generator output and max/min of virtual bid/offer. The maximum and minimum net power injections are as the follows:

$$\overline{P}_m^{\text{inj}} = \sum_{i \in m} \overline{P}_i + \sum_{k \in m} \overline{P}_k^v - \sum_{n \in m} D_n, \tag{3.1}$$

$$\underline{P}_m^{\text{inj}} = \sum_{k \in m} \underline{P}_k^v - \sum_{n \in m} D_n - \sum_{f \in m} \overline{D}_f^{\text{flex}}. \tag{3.2}$$

Net power injection interval is $\Delta P_m^{\text{inj}} = \overline{P}_m^{\text{inj}} - \underline{P}_m^{\text{inj}}$, where overline/underline represent the maximum/minimum physical limits, respectively.

With the net power injection, we can apply Theorem 3 in [54] to determine those non-binding constraints. For self-contained, we present the result directly here. Denote $P^{\text{v}} = -\sum_m \underline{P}_m^{\text{inj}}$ and $f_l^{\text{v}} = \sum_{m=1}^{N_d} \Gamma_{l,m} \underline{P}_m^{\text{inj}}$. If there exists an integer $j \in [1, N_d]$

such that,

$$\begin{cases} \Gamma_{l,m_1} \geq \Gamma_{l,m_2} \geq \cdots \geq \Gamma_{l,m_d} & (3.3) \\ \sum_{n=1}^{j-1} \Delta P_{m_n}^{\text{inj}} \leq P^{\text{v}} \leq \sum_{n=1}^{j} \Delta P_{m_n}^{\text{inj}} & (3.4) \\ \sum_{n=1}^{j-1} \left(\Gamma_{l,m_n} - \Gamma_{l,m_j}\right) \Delta P_{m_n}^{\text{inj}} + \Gamma_{l,m_j} P^{\text{v}} + f_l^{\text{v}} \leq F_l & (3.5) \end{cases}$$

Denote the remaining security constraint set as J_1. According to our experience in real systems, J_1 are still with considerable size, and many constraints in J_1 are not binding at the optimal point. Hence, we employed a data-driven method to tighten the remaining sets by leveraging historical data. The goal was to determine the 95% confidence interval of generator output, which can be substituted in (3.1) and (3.2). The new upper and lower bound of net power injection are denoted as $\overline{P}_m^{\text{inj}*}, \underline{P}_m^{\text{inj}*}$, respectively. The new net power injection interval,

$$\Delta P_m^{\text{inj}*} = \overline{P}_m^{\text{inj}*} - \underline{P}_m^{\text{inj}*} \leq \Delta P_m^{\text{inj}}$$

denote the set of new remaining security constraints as J_1^*. Due to the smaller net power injection,

$$J_1^* \subseteq J_1$$

holds. If the true net power injection is outside $[\overline{P}_m^{\text{inj}*}, \underline{P}_m^{\text{inj}*}]$, then J_1^* might miss binding constraints. To address this issue, we employed the lazy constraint set

$$J_1 \backslash J_1^*$$

in the optimization engine. Lazy constraints will be discussed in Section 3.4.

3.2 Step 2: Linear Programming Based Offline Power Flow Maximization

J_1 detected by Step 1 are the potentially dominating security constraints, and the redundancy still exists. We applied the procedure two to further tighten security constraint set. An LP model is formulated. The objective was to maximize the f_l respecting all other security constraints in J_1. l is a dominating constraint if and only if $f_l > F_l$. It is formulated as,

$$(C) \quad f_l^{\text{off}} := \max_{\{P_m^{\text{inj}}\}} \quad \sum_m \Gamma_{l,m} P_m^{\text{inj}} \tag{3.6}$$

$$\text{s.t.} \quad \sum_m P_m^{\text{inj}} = 0, \quad m \in \mathcal{M} \qquad (\lambda_l) \tag{3.7}$$

$$\sum_m \Gamma_{l',m} P_m^{\text{inj}} \leq F_{l'}, \quad l' \in \mathcal{J}_1 \setminus l \qquad (\alpha_{l'}) \tag{3.8}$$

$$\underline{P}_m^{\text{inj}} \leq P_m^{\text{inj}} \leq \overline{P}_m^{\text{inj}}, \quad m \in \mathcal{M} \qquad (\beta_m^-, \beta_m^+) \tag{3.9}$$

where equation (3.7) is the power balance constraint. It is equivalent to $\sum_n P_n = \sum_d D_d$, but in a slightly different format. $\mathcal{M}$ is the set of bus indices. F is the power flow limit. We were able to identify the power flow violation as long as the network topology and lower/upper bound of net power injection were available. This feature enables wide range of applications. Theorem 3 in [54] is also based on the net power injection model, although identification rate might not be as high as problem (C). For simplicity, we do not present the power flow in negative direction, which is similar.

It is time-consuming to solve thousands of LPs like (C). As we are only interested in whether $f_l > F_l$ holds, it is not necessary to completely solve problem (C). Instead, we can solve problem (C*), which is a feasibility check for the optimized model. The model will be infeasible if constraint (3.14) conflicts with others, which means the maximum power flow on l cannot be greater than its power flow limit.

Hence, a security constraint l is not a dominating constraint, and vice versa.

$$(C^*) \qquad \max \quad 0 \tag{3.10}$$

$$\text{s.t.} \qquad \sum_m P_m^{\text{inj}} = 0, \quad m \in \mathcal{M} \tag{3.11}$$

$$\underline{P}_m^{\text{inj}} \leq P_m^{\text{inj}} \leq \overline{P}_m^{\text{inj}}, \quad m \in \mathcal{M} \tag{3.12}$$

$$\sum_m \Gamma_{l',m} P_m^{\text{inj}} \leq F_{l'}, \quad l' \in J_1 \setminus l \tag{3.13}$$

$$\sum_m \Gamma_{l,m} P_m^{\text{inj}} > F_l \tag{3.14}$$

3.3 Step 2: Dual Problem Based Online Screening

Offline optimization requires solving $|\mathcal{J}_1|$ LP problems to identify the non-congested lines. The computational burden could be large in multi-period scheduling problems for the real-world power systems. The online identification to tackle the computational challenge is performed based on closed form equations without solving optimization problem. At the offline optimization stage, load/net power injection information is assumed not precise. In contrast, more information will be available at the online calculation stage in real time. In the following section, we present online screening methods capturing both net power interval change and network topology update.

3.3.1 Online Method in the Abstract Form

First, we write the problem (C) in an abstract form,

$$\text{(C')} \qquad \min_{\boldsymbol{x}} \quad \boldsymbol{c}^T \boldsymbol{x} \tag{3.15}$$

$$\text{s.t.} \quad \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \qquad (\boldsymbol{\lambda}) \tag{3.16}$$

where $\boldsymbol{c}$ denotes the sensitivity factor vector for the security constraint under investi-

gation, $-\{\Gamma_{l,m}\}$, $\boldsymbol{x}$ denotes net power injection vector, P_m^{inj}, and $\boldsymbol{b}$ denotes parameters in (C) including $\overline{P}_m^{\text{inj}}$, $-\underline{P}_m^{\text{inj}}$, $F_{l'}$ etc. We change the maximization problem to a minimization one by multiplying the objective function by -1.

At online stage, more accurate net power injection will be available. Consider realized load/generation $\hat{\boldsymbol{b}}$. The question then arises as to how to find the new upper bound of power flow with $\hat{\boldsymbol{b}}$. Theoretically, one could solve (C') with updated $\hat{\boldsymbol{b}}$. However, it involves more computational burdens. Instead, we will use a closed form to find the upper bound instead of solving an LP problem. The dual form of problem (C') is

$$(D) \qquad \max_{\boldsymbol{\lambda}} \quad -\boldsymbol{b}^T\boldsymbol{\lambda} \tag{3.17}$$

$$\text{s.t.} \quad \boldsymbol{A}^T\boldsymbol{\lambda} + \boldsymbol{c} = 0, \tag{3.18}$$

$$\boldsymbol{\lambda} \geq 0. \tag{3.19}$$

Theorem III.1 *Denote the optimal point to (D) as $\boldsymbol{\lambda}^*$. With updated net power injection data $\hat{\boldsymbol{b}}$, an upper bound of power flow in branch l is*

$$-\hat{\boldsymbol{b}}^T\boldsymbol{\lambda}^*.$$

Proof. Following duality theory, we have

$$-\boldsymbol{b}^T\boldsymbol{\lambda} \leq \boldsymbol{c}^T\boldsymbol{x}$$

where $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ are feasible in (C') and (D), respectively. Denote optimal solution to (C') and (D) as $\boldsymbol{x}^*$ and $\boldsymbol{\lambda}^*$. According to strong duality,

$$\boldsymbol{c}^T\boldsymbol{x}^* = -\boldsymbol{b}^T\boldsymbol{\lambda}^*$$

holds. When $\boldsymbol{b}$ is updated to $\hat{\boldsymbol{b}}$, we denote the optimal points as $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{\lambda}}$. It is

observed that $\boldsymbol{x}^*$ might be infeasible in problem (C'), however, $\boldsymbol{\lambda}^*$ is always feasible in (D). Therefore,

$$-\hat{\boldsymbol{b}}^T\boldsymbol{\lambda}^* \leq -\hat{\boldsymbol{b}}^T\hat{\boldsymbol{\lambda}} = \boldsymbol{c}^T\hat{\boldsymbol{x}} \tag{3.20}$$

holds. According to (3.20), $\hat{\boldsymbol{b}}^T\boldsymbol{\lambda}^*$ is the **upper bound** of power flow in branch l with updated net power injection data $\hat{\boldsymbol{b}}$. ■

It is noted that optimal dual point $\boldsymbol{\lambda}^*$ to (D) is a byproduct of solving (C'). Most LP solvers provide $\boldsymbol{\lambda}^*$ when solving (C').

One may conjecture that uncertainty could be considered at offline optimization stage by setting $\boldsymbol{b}$ large enough so that

$$\boldsymbol{b} \geq \hat{\boldsymbol{b}} \tag{3.21}$$

holds. $\boldsymbol{x}^*$ is feasible for all $\hat{\boldsymbol{b}}$ in this case. However, it might have poor performance due to the large interval of net power injections. Theorem 1 holds even when $\boldsymbol{b}$ does not follow (3.21). The proposed offline-online procedure thus provides an option to find tighter upper bound of power flow.

3.3.2 Practical Formulation with Updated Net Power Injection Interval

In the real-time market, many uncertainties are materialized. The load forecast for next hours is more accurate than that in DAM. The predication of renewable generation is also with smaller errors. Furthermore, ON/OFF status for many generators are already known for next hours. This information can be utilized to narrow the net power injection. More specifically, $\underline{P}_m^{\text{inj}}$ and $\overline{P}_m^{\text{inj}}$ in equation (3.9) will be updated to capture the materialized information.

Instead of solving another LP problem with an updated constraint, we present a computationally efficient way to find the new largest power flow. Offline method requires solving problem (C) for each security constraint, which gives the maxi-

mized power flow f_l^{off}, optimal net power injection $P_m^{\text{inj}*}$ and, dual variables, i.e., $\lambda_l^*, \alpha_{l'}^*, \beta_m^{-*}, \beta_m^{+*}$. In fact, most modern optimization solver will provide dual values after solving problem (C). To show how to utilize dual values, we formulate dual problem (E) of (C) below and denoted its optimal value as $f_l^d(\underline{\mathbf{P}}^{\text{inj}}, \overline{\mathbf{P}}^{\text{inj}})$.

$$(E) \quad f_l^{\text{d}}(\underline{\mathbf{P}}^{\text{inj}}, \overline{\mathbf{P}}^{\text{inj}}) := \tag{3.22}$$

$$\max_{\substack{\lambda_l, \alpha_{l'}, \\ \{\boldsymbol{\beta}^-, \boldsymbol{\beta}^+\}}} -\sum_{l'} F_{l'}\alpha_{l'} + \sum_m (\underline{P}_m^{\text{inj}}\beta_m^- - \overline{P}_m^{\text{inj}}\beta_m^+)$$

$$\text{s.t.} \quad \lambda_l + \sum_{l' \in \mathcal{J}_1 \setminus l} \Gamma_{l',m}\alpha_{l'} - \beta_m^- + \beta_m^+ = \Gamma_{l,m}, \tag{3.23}$$

$$m \in \mathcal{M}$$

$$\beta_m^+, \beta_m^- \geq 0, m \in \mathcal{M} \tag{3.24}$$

$$\alpha_l \geq 0, \tag{3.25}$$

where $\underline{\mathbf{P}}^{\text{inj}}, \overline{\mathbf{P}}^{\text{inj}}$ are lower and upper bound vectors for all net power injections. It is noticed that $f_l^d(\underline{\mathbf{P}}^{\text{inj}}, \overline{\mathbf{P}}^{\text{inj}})$ is a function of $\underline{\mathbf{P}}^{\text{inj}}$ and $\overline{\mathbf{P}}^{\text{inj}}$.

It is observed that optimal dual values of (C), i.e. $\lambda_l^*, \alpha_{l'}^*, \beta_m^{-*}, \beta_m^{+*}$ are always feasible in (E) no matter what upper and lower bounds of net power injections are. The updated upper and lower bounds are denoted as $\hat{\underline{P}}_m^{\text{inj}}$ and $\hat{\overline{P}}_m^{\text{inj}}$, respectively. We define f_l^{on} as

$$f_l^{\text{on}} := -\sum_{l'} F_{l'}\alpha_{l'}^* + \sum_m \hat{\underline{P}}_m^{\text{inj}}\beta_m^{*-} - \sum_m \hat{\overline{P}}_m^{\text{inj}} * \beta_m^{*+}. \tag{3.26}$$

According to duality theory,

$$f_l^{\text{on}} \geq f_l^{\text{d}}(\hat{\underline{\mathbf{P}}}^{\text{inj}}, \hat{\overline{\mathbf{P}}}^{\text{inj}}) = f_l^{\text{off}}$$

holds. Therefore, f_l^{on} is an upper bound of power flow online l. In other words,

we can determine the upper bound of the maximal power flow using the closed-form equation (3.26). According to our experience, this upper bound is tight enough to remove the majority of industrious constraints in real-world systems.

3.3.3 Updated Network Topology

The method aforementioned works when there is no topology change. In this part, our goal was to quickly determine whether post line contingency power flow will cause congestion. Line outage distribution factor $\mathbf{LODF}_{ml\to ol}$ can be quickly calculated as shown in [107], where $ml \to ol$ represents monitored lines to outage lines. Pre- / post-contingency power flow, $f_{ml}^{\text{base}} \backslash f_{ml}^{\text{ctg}}$, must respect branch flow limits, F_{ml}. They are typically modeled as

$$\begin{cases} f_{ml}^{\text{ctg}} = f_{ml}^{\text{base}} + \mathbf{LODF}_{ml\to ol} f_{ol}^{\text{base}} & (3.27) \\ -F_{ml} \le f_{ml}^{\text{ctg}} \le F_{ml} & (3.28) \end{cases}$$

where $f_{ml}^{\text{base}} = \mathbf{\Gamma}_{ml} P_m^{\text{inj}*}$ is obtained from the base case.

The traditional approach calculates post-contingency power flow from pre-contingency power flow. The proposed approach verifies congestion potential of monitored lines over an injection range, $\Delta P_m^{\text{inj}*}$. The pre-contigency net power injection is denoted as $P_m^{\text{inj}*}$. $\Delta P_m^{\text{inj}*}$ can be calculated around $P_m^{\text{inj}*}$ by factoring in topology change, load forecasting error, generator ramping rates and participation as below

$$P_m^{\text{inj}*} - \sum_{i\in\mathcal{G}_m} R_i + \underline{d} \le \Delta P_m^{\text{inj}*} \le P_m^{\text{inj}*} + \sum_{i\in\mathcal{G}_m} R_i + \overline{d}, \tag{3.29}$$

where R_i is ramping rate. Consider post-contingency power flow change due to $\Delta P_m^{\text{inj}'}$

$$\Delta f_{ml} = (\mathbf{\Gamma}_{ml} + \mathbf{LODF}_{ml\to ol}\mathbf{\Gamma}_{ol})\Delta P_m^{\text{inj}*}, \tag{3.30}$$

where $\mathbf{\Gamma}_{ol}$ is outage lines' shift factors extracted from base case. The new power

flow is

$$f_{ml}^{\text{ctg}} + \Delta f_{ml}. \tag{3.31}$$

Together with

$$-F_{ml} - f_{ml}^{\text{ctg}} \leq \mathbf{\Gamma}_{ml}^{\text{ctg}} \Delta P_{m}^{\text{inj}*} \leq F_{ml} - f_{ml}^{\text{ctg}}, \tag{3.32}$$

we can directly apply the proposed two-step screening to determine potential dominating security constraints in post contingency situation.

Topology changes discussed here can only account for line-outage contingencies and cannot account for N-1 generator contingencies. The question might arise whether this method can account for a practical SCUC loss scenario when a generator is lost. The proposed two-step method is based on the DC model considering Injection Shift Factor (ISF). Line-outage contingencies would change system topology, altering the ISF matrix, resulting in new/different security constraints. A generator outage, on the other hand, would not affect the ISF matrix, nor generating new security constraints. It will only change the net power injection on the designated bus. Since we are taking the conservative approach to estimate net power injection intervals, on a bus with generator, it would be between zero and max output power. MISO uses generator participation factor in the case of generator outage by re-allocating the missing amount of energy to nearby units. This action would not change system topology, and our conservative approach on net power injection interval would not be violated, since the generators that pick up the slack also needs to be capped by its own physical limitation.

3.4 Lazy Constraint Identification and Implementation

3.4.1 Motivation behind Lazy Constraint

The proposed two-step screening approach can significantly reduce the size of U in the SCUC model. However, only a portion of J_2 is binding at the solution point. The active constraints at the solution point were denoted as J_B. The question that naturally arises is how we can effectively shrink the size of J_2 to an approximation of J_B without jeopardizing the security. This challenge can be tackled exactly by the concept of lazy constraint. Lazy constraints are designed to incorporate user confidence in the optimization model to improve solution quality. Constraints that are believed to be less probable to bind at the optimal solution can be marked as lazy. The model is initially optimized without any lazy constraints. Once a solution is found, the lazy constraints are checked with the current solution. If any lazy constraints are violated, they will be added back to the reduced model. Optimization process continues till an optimal solution is found without any lazy constraint violations. Users should exercise the best judgement to mark lazy constraints, since the intention of lazy constraints is to shrink constraints set and optimize a smaller and easier model. On the other hand, if it is set inappropriately, it may require more iterations to restore the lazy constraints, causing solution time to increase.

It is crucial to separate screening result J_2 into the active constraint pool J^{act}, and lazy constraint pool J^{lazy}, where $J^{\text{act}} \cup J^{\text{lazy}} = J_2$ and $J^{\text{act}} \cap J^{\text{lazy}} = \emptyset$. If the active pool is over-conservative, $J^{\text{act}} \approx J_2$, performance improvement would be negligible. If the pool is over-optimistic, $J_B \backslash J^{\text{act}} \subseteq J^{\text{lazy}}$ is large and more constraints are added back in the MILP searching process, which may increase solution time, as illustrated in Figure 3.1.

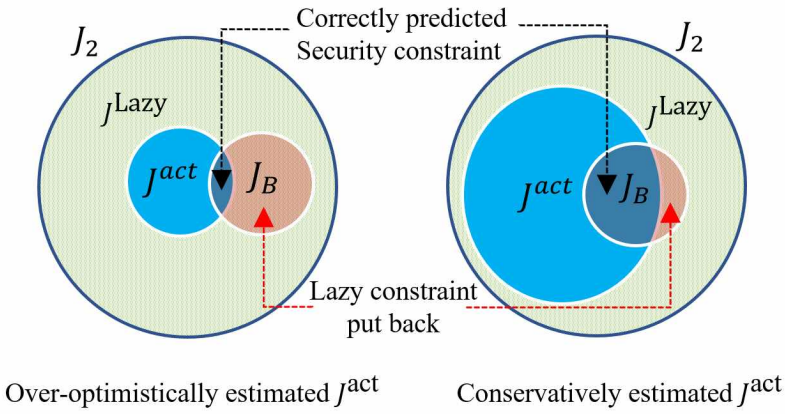

Figure 3.1: Relationship Amongst Binding Constraint, Active Constraint and Lazy Constraint

The key to establish a relatively accurate active constraint pool is to obtain an approximation of SCUC solution. We developed two approaches for achieving this goal. The first is Commitment Initialization (**CI**) inspired by [57, 58], and the second is based on Integer Relaxation (**IR**). Procedures are shown in Fig. 3.2. Both approaches solve an LP problem. The LP solution offers insight into the security constraint relationship with the SCUC model. We used this information to establish the lazy constraint pool. The basic idea was to find a much smaller active constraint pool J^{act}, and the pool of lazy constraint is

$$J^{\text{lazy}} = J_2 \setminus J^{\text{act}}$$

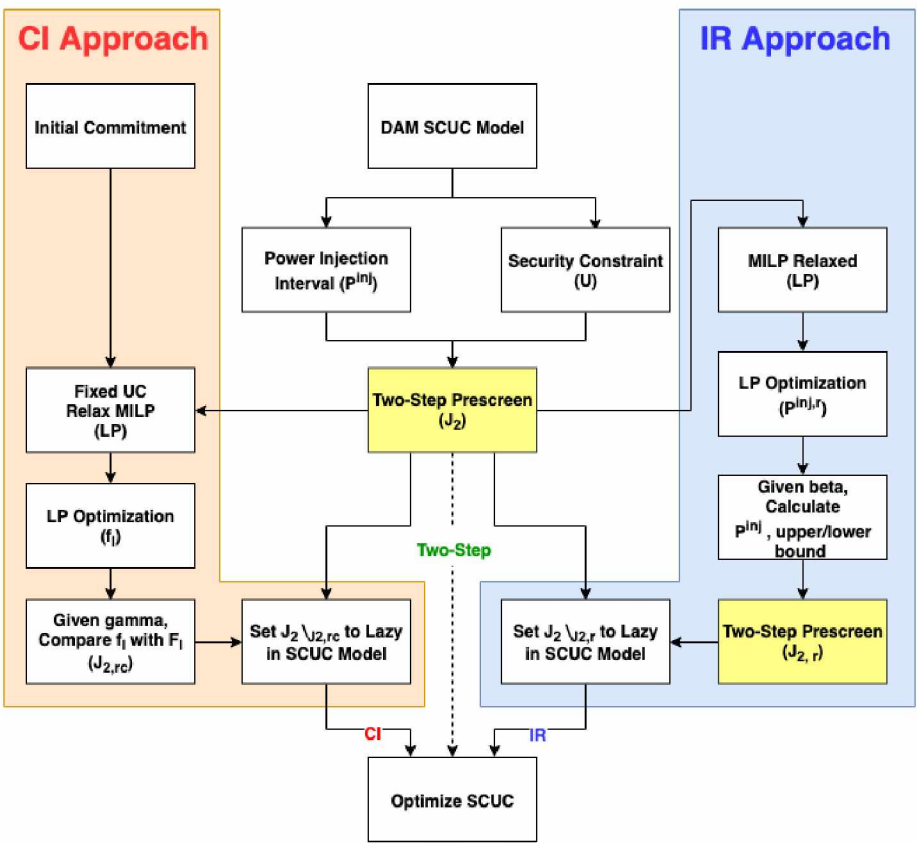

Figure 3.2: Data-Aided screening with Lazy Constraints. CI and IR provide a point statistically close to the real solution. Security constraints with values below power flow limits in certain extent are treated as lazy constraints, which reduce the size of MILP and helps solve the problem faster if set properly.

3.4.2 Integer Relaxation

Relaxing integer variables in the SCUC model will yield an LP model, which takes very little time to solve. The LP solution is by no means an accurate representation of the MILP solution. However, it provides insights to where the solution may point. We are solely interested in the net power injection from LP solution $P_m^{\mathrm{inj,r}}$. We established a boundary of net power injection, $\overline{P}_m^{\mathrm{inj,r}}$ and $\underline{P}_m^{\mathrm{inj,r}}$, to accommodate uncertainties resulting from the difference between LP and MILP solutions as shown

in (3.33) and (3.34)

$$\overline{P}_m^{\text{inj, r}} = \min\{(1+\beta) * P_m^{\text{inj,r}}, \overline{P}_m^{\text{inj}}\} \tag{3.33}$$

$$\underline{P}_m^{\text{inj, r}} = \max\{(1-\beta) * P_m^{\text{inj,r}}, \underline{P}_m^{\text{inj}}\}, \tag{3.34}$$

where β is the parameter to be tuned.

To set the lazy constraint, we take the following steps.

1. Build DAM SCUC (MILP) model
2. Apply the two-step screening method to obtain J_2
3. Relax integer variables and solve LP problem, extract $P_m^{\text{inj, r}}$
4. Select β, calculate $\overline{P}_m^{\text{inj, r}}, \underline{P}_m^{\text{inj, r}}$
5. Apply the two-step screening method on $\overline{P}_m^{\text{inj, r}}, \underline{P}_m^{\text{inj, r}}$, resulting in J^{act}
6. Obtain lazy constraint pool, $J^{\text{lazy}} = J_2 \backslash J^{\text{act}}$

3.4.3 Commitment Initialization

According to the persistence studies conducted with MISO cases, 90% of previous day commitment was identical to the current day commitment. We exploited this fact as a warm start to the SCUC model [57]. The basic idea was to get an approximated power flow by solving an LP problem with initialized commitment. The resulting power flow, f_l^c, for each of the lines was often close to the true MILP solution. Given power flow from LP solution f_l^c, power flow limit F_l, and threshold γ, if

$$\frac{f_l^c}{F_l} \geq \gamma, \qquad l \in J_2 \tag{3.35}$$

holds, the security constraint for the line l is considered as potentially binding, which is put to the active constraint set J^{act}. If equation (3.35) does not hold, this constraint

for line l is set as lazy. We have $J^{\text{lazy}} = J_2 \backslash J^{\text{act}}$.

1. Build DAM SCUC (MILP) model
2. Apply the two-step screening method to obtain J_2
3. Fix UCs with initial commitment, solve resulting LP problem
4. Calculate power flow, f_l^c, based on LP solution
5. Choose threshold γ, use (3.35) to identify J^{act}
6. Obtain lazy constraint pool $J^{\text{lazy}} = J_2 \backslash J^{\text{act}}$

Security constraints are known to be a major cause of SCUC solution deterioration. SCUC solution quality cannot be guaranteed while including all watchlist security constraints. In the combined approach of the two-step security constraint screening and lazy constraint settings, MILP model optimization performance significantly improves by having a smaller set of active constraints. Model integrity is also maintained while having the potential optimal solution check against lazy constraints.

3.5 Experimental Results

In this chapter, we applied the proposed two-step security constraint screening to a modified version of IEEE118-Bus test system. Dominating constraint detection and computational performance are reported in this section. We also used MISO provided real-world data to test the proposed offline-online security constraint screening technique. Lazy constraints are set prior to solve SCUC based on IR and CI, discussed in the previous chapter. SCUC performances amongst various techniques are compared and analyzed.

3.5.1 IEEE-118 Bus Test System Case Study

Test Setup

We applied the two-step screening method to IEEE-118 bus test system to demonstrate the effectiveness of constraint reduction and solution efficiency for both base case and line outage contingency case. To avoid confusion, scenario^0 refers to a set of pre-determined wind generation and load profiles. Scenario^n refers to the n^{th} scenario under study, where $n \in [1, 12]$. Three wind farms are added to bus 36, 77, 69 with 24 hour power output. Wind power generation hourly profile and System wide hourly load profile are shown in figure 3.3.

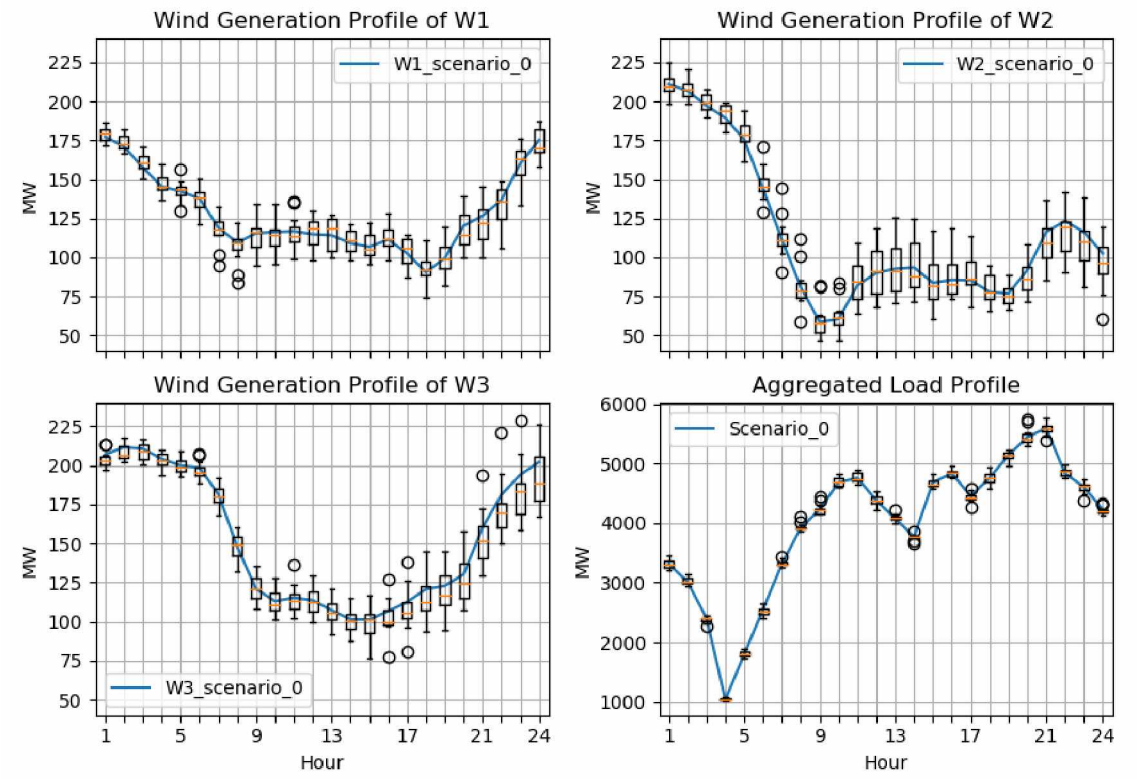

Figure 3.3: Aggregated Load Profile (scenario^{0}s are plotted with solid lines, and scenario^{n}s are in box plots for illustration)

Applying Offline and Online Method

Offline method, discussed in 3.2, is first applied to the scenario ^{0} to screen

out the dominating constraints and obtaining dual variables. Load side uncertainty is set as 10% about the load profile. Wind generator ratings were used to account for wind generation variation. Secondly, net power injection interval can be derived from (3.1) and (3.2) for scenario^n. Online method (theorem III.1) calculates power flow upper bounds by utilizing dominating constraints, dual variables from offline study and derived net power injection intervals of scenario^n. The online method would consider constraints with upper bounds exceeding their respectable limits as potentially dominating. Lastly, for verification purpose, screening offline method was applied directly to scenario^n and truly dominating constraints were determined. Results from offline method and online method were compared. There are 8,928 network constraints per scenario ($186 * 2 * 24$). Contingency case accounts for N-1 line outage security constraints with the same approach. There is a total of 1,660,608 security constraints per scenario to be screened $[(186 + 185 * 186) * 2 * 24]$.

Offline and Online Method Results

Security constraint screening results and performance summary is shown in Table 3.1 and 3.2. Each row has the following attribute,

- **Offline**: Base case (scenario^0) screening results with offline method without wind generation and load side uncertainty
- **Offline+**: Base case (scenario^0) screening results with offline method including wind generation uncertainty and 10% load side uncertainty (obtaining dual variables)
- **Online**: Scenario ^{n} screening result using online method with materialized uncertainty
- **Verify**: Scenario ^{n} screening result with offline method to verify online method result (using the same input as **Online**)

For the base case, in each time interval, there are 372 (186*2) network constraints to be screened, without adding wind generation and load side uncertainty (**Offline**). Step 1 determines that 40 constraints are potentially dominating, and step 2 determines 18 out of 40 constraints are truly dominating. Screening took 0.0332 seconds. When uncertainties are introduced into the model (**Offline+**), the screening process determines that 20 constraints are dominating. Wider net power injection intervals cause 2 additional constraints to be dominating. Around 5% of the constraints are determined to be truly dominating constraints after screening.

Similarly, the line outage contingency case contains 69,192 security constraints per time interval. As shown in Table 3.1, 0.1% of the security constraints are truly dominating (**Offline**), and once uncertainties are introduced to the model (**Offline+**), dominating security constraint rate increases to 0.15%. Offline screening process takes around 332 seconds per time interval. The significant time required for calculation also bring motivation to implement online method.

Dual variables are obtained from screening scenario ^{0} during offline stage with added uncertainty (**Offline+**). Online method is then applied to the scenario ^{n}. The results are verified by applying an offline method to a scenario ^{n} with realized uncertainty (**Verify**). Dominating constraints determined by **Online** is a superset

Table 3.1: IEEE-118 Bus System Network/Security Constraint Screening Result

	Method	Original[1]	Step1[1]	Step2[1]	Dom1[2]	Dom2[2]
Network Constraint	Offline	372	40	18	10.79%	4.70%
	Offline+	372	51	20	13.59%	5.45%
	Online	20	-	16	-	4.27%
	Verify	372	32	15	8.63%	4.10%
Security Constraint	Offline	69192	6836	66	9.88%	0.10%
	Offline+	69192	9643	101	13.94%	0.15%
	Online	101	-	69	-	0.10%
	Verify	69192	6170	53	8.92%	0.08%

[1] Constraint count at each stage

[2] Dominating constraint percentage after each screening step

of **Verify** in both network and security constraint cases. It is due to the uncertainty embedded in net power injection interval from scenario ^{0} completely covers the net power injection interval for scenario^n. Hence, power flow upper bounds determined by theorem III.1 will conservatively cover all potentially dominating security constraints in a scenario ^{n}. In the base case, on a per time interval basis, the online method determines that 16 out of 20 security constraints are dominating. Offline method verifies that 15 out of 16 constraints are truly dominating. In the contingency case, online method determines that 69 out of 101 security constraints are dominating. Offline method verifies that 53 out of 69 constraints are truly dominating. We also extended the study over a 24 hour planning period. The results are shown in 3.4.

The computational performance is shown in 3.2. Since online method is a numerical method, calculation time is remarkably short comparing with offline method. Particularly for contingency case, offline methods requires 5.43 minutes (326 seconds for checking results) per time interval and online method only took 2.4 ms. Offline screening security constraints can be defined with uncertainty bounds with proper preparation. Online method is capable of providing a truly remarkable solution approach to remove redundant security constraints in large scaled systems.

The efficiency of the online method comes at the cost of storing dual variables. The size to store dual variables is surprisingly small (manageable). Typically, $200 - 300$ MBs. For the MISO HIPPO system, a day ahead market usually have $150 - 200$ security constraints picked from watchlist. And usually around 50% of them could be potentially binding. We can determine this by applying the proposed feasibility check method after step 1. Let's say that we need to store dual variables for 100 security constraints per time interval over 36 intervals. The system contains 3000 buses. Total dual variable number is $(3000 * 2 + 99 + 1) * 100$, where $3000 * 2$ is the upper/lower bound of net power injection interval, 99 is the constraints under the current LP, and we need to calculate 100 LPs. This is a sparse matrix and depends on the precision

of data type, i.e. float32. Memory that is required to store dual variables is small.

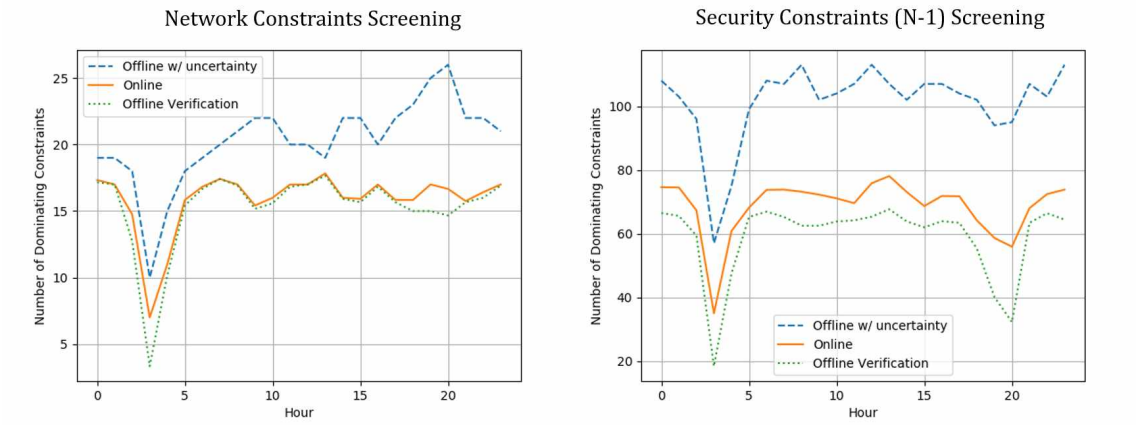

Figure 3.4: Offline-Online Screening Result (24 Hours)

Table 3.2: IEEE-118 Bus Test Case Network/Security Constraint Screening Performance Summary

Unit: Seconds	Network Constraint			Security Constraint		
	Step1	Step2	Total	Step1	Step2	Total
Offline	0.0168	0.0164	0.0332	1.0889	330.8312	331.9200
Offline +	0.0159	0.0247	0.0406	1.1218	331.9979	333.1197
Online	-	-	0.0006	-	-	0.0024
Verify	0.0072	0.0122	0.0194	1.2101	324.6765	325.8866

3.5.2 MISO Case Study

In section 3.5.1, we demonstrated the effectiveness of two-step security constraint screening on IEEE-118 bus test system. In this section, we conduct numerical simulations for MISO cases and analyze the performance impact of the proposed method in MISO market-clearing optimization engine. By using historical data and knowledge, such as Initial Commitment and uncertainty of net power injection, we integrated the security constraint screening with lazy constraint in the optimization model. The statistical approach helps tighten the net power injection range with a given confidence interval, thus building a smaller watchlist with fewer security constraints. Section 3.5.2 illustrates dominating constraints detection rate and perfor-

mance of the two-step deterministic screening method. In Section 3.5.2, dominating constraints detected by the two-step deterministic screening are enforced in the SCUC model. Lazy constraints are applied based on the procedure discussed in 3.4. Numerical simulations are carried out on MISO HIPPO project engine [108], using Gurobi 7.5 in Centos with Intel Xeon E5@3.50GHz, 64 GB RAM [109].

Dominating Constraint Detection

We apply the two-step security constraints screening to MISO DAM SCUC model prior to optimization. Dominating constraint detection rate and computation time is shown in Table 3.3. Case 17 has the fastest processing time of 13.95 seconds (step-1: 4.88s, step-2: 9.07s). 49% of the security constraints are dominating (original: 6604, step-1: 3451, step-2: 3152). In Case 1, screening time is 18.52 seconds (step-1: 6.03s, step-2: 12.49s), 43% of the security constraints are dominating (original: 9895, step-1: 4806, step-2: 4251). Step-2 needs to screen 3451 constraints in Case 17 and 4806 constraints in Case 1, which is 40% increase in constraint count. Thus, the processing time for Step-2 is increased by 38% ($\frac{12.49-9.07}{9.07}$). The two-step method is consistent across all cases, and processing time is linearly related to the number of security constraints in the model. We applied the proposed method to over 80 MISO DAM cases. Result shows that, 8075 security constraints are initially modeled on average. It has been determined that 45% of the original constraints are dominating ($3634 = 8075 * 45\%$), which will remain in the model. 4421 redundant constraints are safely removed from SCUC model prior to optimization. Average screening processing time is 16.65 seconds by solving problem (C^*) in step-2, comparing with 53.28 seconds by solving problem (C).

Application of Security Constraint Screening to MISO SCUC

In the case study, MISO DAM SCUC optimization engine terminates if MILP

Table 3.3: MISO Case Security Constraint Screening Result

	Case#	Original[1]	Constr Cnt[2]	Dom%[3]	Time[4]
Easy	3	6948	3602	51.84%	14.32
	8	9431	4168	44.19%	18.29
	12	7454	3221	43.21%	19.86
	14	6260	3130	50.00%	14.85
Hard	1	9895	4251	42.96%	18.52
	5	6950	3681	52.96%	15.70
	16	10400	4198	40.37%	18.17
	17	6604	3152	47.73%	13.95
	21	9540	5186	54.36%	23.06
	22	8300	4569	55.05%	20.72
Average[5]	-	8075	3654	45.25%	16.65

[1] Security constraint count in DAM SCUC model
[2] Dominating constraint count after screening
[3] Percentage of security constraint remained in the model
[4] Screening time, step 2 solving (C*), in seconds
[5] Average value over 80 cases

Gap reaches 0.1% or optimization time reaches 1200 seconds [34]. In this section, we considered the case that has a base solution time longer than 1000 seconds as hard case. Otherwise, it is considered as an easy case. The following solution approaches are applied to the DAM SCUC model:

- **Base solution**: Optimize DAM SCUC directly
- **Two-step solution**: Screen security constraints in DAM SCUC model and remove redundant constraints. To optimize the MILP model, no lazy constraint settings are applied.
- **IR solution**: Screening DAM SCUC model, remove redundant constraints. Apply IR approach to set lazy constraints in MILP model; optimize. β is set to 0.6.
- **CI solution**: Screen DAM SCUC model and remove redundant constraints. Apply CI Approach to set lazy constraints, then optimize. γ is set to 0.8.

There are totally 46 cases where base solution time is over 300 seconds. Fig. 3.5 illustrates the SCUC optimization time comparison between base solution and best of various screening solution approaches (two-step, IR, CI). With screening techniques, 63% of the cases (26/46) can be solved with 80% or less time than base case. 22% of the cases (10/46) are solved with 50% or less time. Applications of CI and IR approach are restricted by the availability of initial commitment and generator dispatch statistics, tuning hyperparameter β and γ. Lazy constraint setting is a heuristic approach to predicting binding constraints. The results show that good lazy constraint pool could significantly improve the solution performance in terms of solution time and quality.

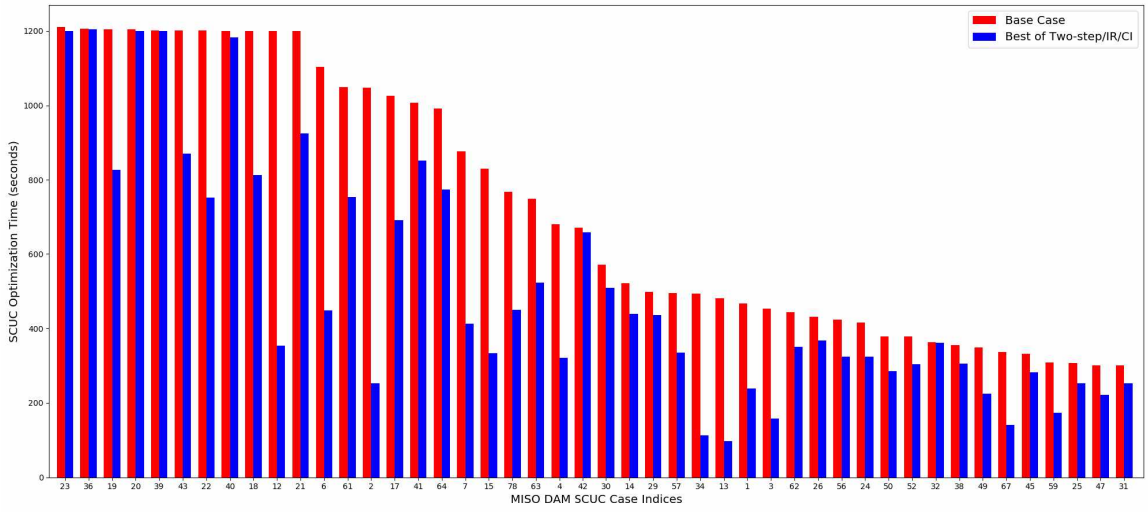

Figure 3.5: MISO Case Optimization Time Comparison

Since easy cases take fewer than 1000 seconds to reach MILP gap below 0.1%, we are mainly interested in solution time by aforementioned solution approaches. Results are shown in Table 3.4. Screening in Case 8 improves SCUC solution time by 58 (298 - 240) seconds. CI Approach determines 3138 out of 4368 constraints are lazy constraints, and at the end of optimization, 191 constraints are deemed to be non-lazy and put back to the MILP model. CI Approach further decreased solution time by 51 (240-189) seconds. With IR, more constraints are set as lazy, and results

in more putting backs, which take a toll on the solution time, 209 seconds. On the other hand, in Case 14, IR is able to achieve better solution time, 442 seconds, than CI Approach, 520 seconds. The solution time resulting from the two lazy constraint setting approaches are highly dependent on availability of initial commitment, and hyperparameters β, γ. Over-optimistic values can result in large number of constraints being set as lazy, inaccurately. From our studies on MISO SCUC easy cases, the two-step security screening method combined with lazy constraint settings shows significantly reduction in SCUC solution time. On average, the aforementioned approach saves an average of 292 seconds (30%) in SCUC optimization time compared with base solution time.

Table 3.4: Easy Case SCUC Solution Summary

Case#	Approach	Time[1]	Gap[2]	Lz_ct[3]	Lz_pb[4]
3	Base	680	0.08%	0	0
	Two-step	622	0.09%	0	0
	CI	456	0.10%	2952	191
	IR	469	0.10%	3939	860
8	Base	298	0.08%	0	0
	Two-step	240	0.08%	0	0
	CI	189	0.08%	3138	191
	IR	209	0.10%	4137	1069
14	Base	831	0.07%	0	0
	Two-step	648	0.09%	0	0
	CI	520	0.09%	2419	410
	IR	442	0.10%	3062	806

[1] SCUC optimization time, unit in seconds
[2] SCUC MILP gap (%)
[3] lazy constraint count in MILP model
[4] lazy constraint put back count after solving MILP

Hard cases take longer than 1000 seconds to reach MILP gap of 0.1%. Results are shown in Table 3.5. Incorporating screening methods helps many cases to reach 0.1% MILP gap under 1200 seconds, whereas previously they could not. In Case 17, the base solution could not reach 0.1% MILP gap within 1200 seconds, hence, the

optimization process is terminated, and the resulting MILP gap is 0.11%. The two-step approach allows MILP gap to reach 0.1% at 1188 seconds. The CI approach is able to reach MILP gap 0.09% at 1063 seconds. Screening dramatically reduce MILP solution time in some hard cases. In Case 1, the two-step approach decreases SCUC optimization time by 126 seconds from base case solution. CI and IR are able to reach 0.1% MILP GAP in 270 seconds. Total solution time is reduced by 777 seconds.

Table 3.5: Hard Case SCUC Solution Summary

Case#	Approach	Time[1]	Gap[2]	Lz_ct[3]	Lz_pb[4]
1	Base	1047	0.07%	0	0
	Two-step	921	0.06%	0	0
	CI	271	0.10%	1592	10
	IR	270	0.09%	3203	1042
5	Base	1103	0.10%	0	0
	Two-step	658	0.07%	0	0
	CI	899	0.09%	2366	166
	IR	802	0.09%	3409	1170
17	Base	1200	0.11%	0	0
	Two-step	1188	0.10%	0	0
	CI	1063	0.09%	1458	64
	IR	1200	0.08%	2087	429
21	Base	1200	0.26%	0	0
	Two-step	1200	0.19%	0	0
	CI	1200	0.10%	3549	8
	IR	1201	0.16%	5110	1752
22	Base	1200	0.23%	0	0
	Two-step	993	0.09%	0	0
	CI	872	0.09%	2742	11
	IR	669	0.09%	4557	1504

[1] SCUC optimization time, unit in seconds
[2] SCUC MILP gap (%)
[3] lazy constraint count in MILP model
[4] lazy constraint put back count after solving MILP

Case 1 and 22 simulation results in Fig. 3.6 illustrate two typical performance improvements when screening is applied. As shown in Fig. 3.6(a), the optimization engine can get the solution within 1200 seconds. When IR or CI based screening

is employed, it only takes about 25% of the original time to find the solution with similar optimal gap. In contrast, Fig. 3.6(b) shows how screening could improve the performance when the original model could not be solved within 1200 seconds. It is observed that all two-step, IR, and CI based screening help the optimization engine solve the problem within time limit.

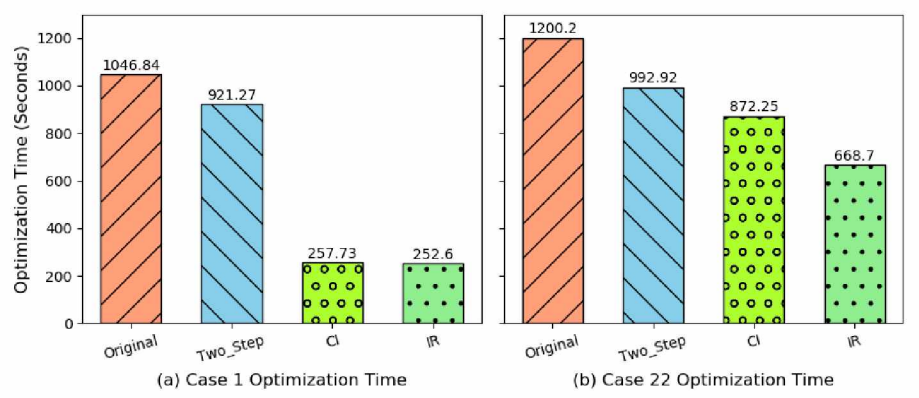

Figure 3.6: Optimization Time Comparison. Figure (a) presents the result for Case 1 where optimal gap is reached within time limit without screening. Figure (b) presents results for Case 22 where optimal gap is not reached within time limit without screening

In a few hard cases, the solver could not get a solution with required gap even if the proposed methods were applied. For instance, Case 21 in Fig. 3.7(a) shows the solver terminates after 1200 seconds. According to Fig. 3.7(b), screening helps reduce the gap significantly. IR based approach decreases the gap to 0.16% from 0.26% while CI approach almost get to 0.1% gap.

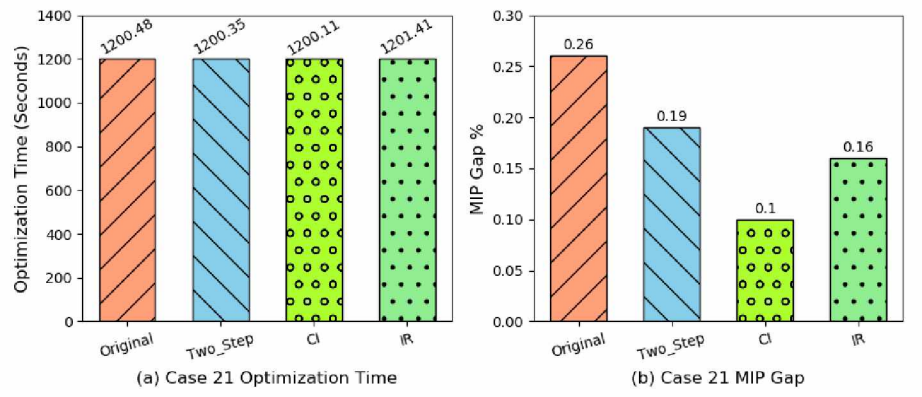

Figure 3.7: Gap Comparison. Optimal gap cannot be reached with or without screening. Figure (b) shows the gap differences when screening is applied.

CHAPTER IV

PRICE CONSTRAINTS

Some security constraints are not modelled in DAM, but they are binding in the RTM, which results in severe financial losses and inefficient dispatch. We proposed that DAM SCUC solution should account for deviations to anticipate this divergence, subject to generator dispatch deviation, load forecasting error, virtual bid and offer cleared MW. The congestion forecasting tool implements price constraints after MTLF (Midterm Load Forecasting) and before DAM SCUC to identify a potentially binding security constraint set.

As the 24 hours ahead information indicates, the DAM SCUC solution is a fixed solution. As uncertainty materialize and new and more accurate information becomes available, the real time optimal solution can possibly deviate from the DAM SCUC solution. This deviation can be possibly captured by enforcing price constraints on the participating elements, i.e. generator dispatch, virtual bid/offer fixed demand and price sensitive demand. This can be set as a percent of allowable deviation from SCUC solution. E.g. MISO allows 7% of generation dispatch deviation due to ramping limits and inaccurate voltage measurements. However, if the plant does not follow the dispatch signal and deviates more than the limit, it is going to be penalized severely.

$$\underline{P}_i \leq P_i^* \leq \overline{P}_i, \quad i \in \mathcal{I} \tag{4.1}$$

$$\underline{P}_k^v \leq P_k^{v*} \leq \overline{P}_k^v, \quad k \in \mathcal{K} \tag{4.2}$$

$$\underline{D}_n \leq D_n^* \leq \overline{D}_n, \quad n \in \mathcal{N} \tag{4.3}$$

$$\underline{D}_f^{\text{flex}} \leq D_f^{\text{flex}*} \leq \overline{D}_f^{\text{flex}}, \quad f \in \mathcal{F} \tag{4.4}$$

Generator and virtual are broken into piece-wise MW segments based on individual bidding curves.

$$P_i^* = \sum_{s \in \mathcal{S}} P_{i,s}^* \quad s \in \mathcal{S} \tag{4.5}$$

$$\underline{P}_{i,s} \leq P_{i,s}^* \leq \overline{P}_{i,s}, \quad s \in \mathcal{S}, i \in \mathcal{I} \tag{4.6}$$

$$P_k^{v*} = \sum_{s \in \mathcal{S}} P_{k,s}^{v*} \quad s \in \mathcal{S} \tag{4.7}$$

$$\underline{P}_{i,s}^v \leq P_{i,s}^{v*} \leq \overline{P}_{i,s}^v, \quad s \in \mathcal{S}, i \in \mathcal{I} \tag{4.8}$$

We would consider that the generation and demand has a deviation allowance, δ^p and δ^d. The deviation can cut off more expensive segments on generators. We can reformulate (4.1) and (4.2); the asset deviation is shown as follows:

$$(1 - \delta^p) P_i^* \leq P_i \leq (1 + \delta^p) P_i^*, \quad i \in \mathcal{I} \tag{4.9}$$

$$(1 - \delta^d) D_n^* \leq D_n \leq (1 + \delta^d) D_n^*, \quad n \in \mathcal{N} \tag{4.10}$$

Virtual deviation is closely related to the generator deviation. Generator deviation will affect the marginal unit's cleared mw and associated cleared nodal price (the most expensive segment of the marginal unit's bidding curve). If we quantify generator deviation in terms of bidding price, the marginal unit (generator)'s price

variation will also create a price range for the virtual bid/offer on the same node, which corresponds to a virtual power injection range.

$$\overline{\pi}_m, \underline{\pi}_m = \max C^p_{i,m}(\overline{P}_i, \underline{P}_i) \qquad i \in \mathcal{M} \tag{4.11}$$

$$\overline{P}^v_k, \underline{P}^v_k = (C^v_{k,m})^{-1}(\overline{\pi}_m, \underline{\pi}_m) \qquad k \in \mathcal{M} \tag{4.12}$$

Modify problem (C) formulation as problem (D). Generator and Virtual deviation penalty factor, γ^p and γ^v, are added to the new objective function. γ is a very small constant, which scales down generator/virtual cost function proportionally. The added terms in the objective of problem (D) should resemble a SCED problem, which minimize cost of generator and virtual while maintaining power balance and enforcing security constraints.

$$\begin{aligned}
(F) \qquad \max_{\{P^{\text{inj}}_m, P_i, P^v_i, D_n, D^{\text{flex}}_f\}} \quad f_l := & \sum_m \Gamma_{l,m} P^{\text{inj}}_m \\
& - \sum_m \sum_i \gamma^p_{i,m} C^p_{i,m}(P_{i,m}) \\
& - \sum_m \sum_k \gamma^v_{k,m} C^v_{k,m}(P^v_{k,m}) \\
& + \sum_m \sum_k \gamma^f_{k,m} C^{\text{flex}}_{f,m}(D^{\text{flex}}_{f,m}) \qquad (4.13)
\end{aligned}$$

s.t. (3.7), (3.8) and (3.9)

$$P^{\text{inj}}_m = \sum_{i \in m} P_i + \sum_{k \in m} P^v_k - \sum_{n \in m} D_n - \sum_{f \in m} D^{\text{flex}}_f \tag{4.14}$$

$$P_i = \sum_{s \in \mathcal{S}} P_{i,s} \quad s \in \mathcal{S} \tag{4.15}$$

$$\underline{P}_{i,s} \le P_{i,s} \le \overline{P}_{i,s}, \quad s \in \mathcal{S}, i \in \mathcal{I} \tag{4.16}$$

$$\underline{P}_i \le P_i \le \overline{P}_i, \quad i \in \mathcal{I} \tag{4.17}$$

$$P_k^v = \sum_{s \in \mathcal{S}} P_{k,s}^v \quad s \in \mathcal{S} \tag{4.18}$$

$$\underline{P}_{i,s}^v \leq P_{i,s}^v \leq \overline{P}_{i,s}^v, \quad s \in \mathcal{S}, i \in \mathcal{I} \tag{4.19}$$

$$\underline{P}_k^v \leq P_k^v \leq \overline{P}_k^v, \quad k \in \mathcal{K} \tag{4.20}$$

$$\underline{D}_n \leq D_n \leq \overline{D}_n, \quad n \in \mathcal{N} \tag{4.21}$$

$$\underline{D}_f^{\text{flex}} \leq D_f^{\text{flex}} \leq \overline{D}_f^{\text{flex}}, \quad f \in \mathcal{F} \tag{4.22}$$

The cost of asset deviation can be derived from solution of problem (D) by solving (4.23). If the cost is too much, we can form a system-wide cut and add back to problem (D), where β is between 0 and 1, representing a allowance of deviation in terms of SCUC overall cost. (4.24) is the price constraint (budget of uncertainty).

$$\begin{aligned} \text{cost}_{\text{total}}^{\text{dev}} = \sum_t (& \sum_m \sum_i C_{i,m}^p (P_{i,m} - P_{i,m}^*) \\ & + \sum_m \sum_k C_{k,m}^v (P_{k,m}^v - P_{k,m}^{v*}) \\ & - \sum_m \sum_f C_{f,m}^{\text{flex}} (D_{f,m}^{\text{flex}} - D_{f,m}^{\text{flex}*})) \end{aligned} \tag{4.23}$$

$$\text{cost}_{\text{total}}^{\text{dev}} \leq \beta \cdot \text{SCUC ObjVal} \tag{4.24}$$

SCUC objective value is not a representation of individual line congestion status. Instead, we can form cuts on each constraint by accessing SCUC solution to calculate "power flow cost". We considered that the cost are from 4 different kinds of assets, generators, virtual inc (behaves like generators), virtual dec (behaves like price sensitive load) and price sensitive load. Each of these assets can be constrained by an hourly cost, specifically, shown in the following equations. The summation of the specified hourly costs is the hourly SCUC cost. It does not account for transition cost, reserve cost etc. Basically, we only account for the active (hourly) units and its respective cost. For line l, the cost of power flow is shown as follows,

$$\omega_t^{p*} = \sum_m \sum_i C^p_{i,m,t}(P^*_{i,m,t}), \qquad t \in \mathcal{T} \tag{4.25}$$

$$\omega_t^{v*+} = \sum_m \sum_k C^v_{k,m,t}(P^{v*+}_{k,m,t}), \qquad t \in \mathcal{T} \tag{4.26}$$

$$\omega_t^{v*-} = \sum_m \sum_k C^v_{k,m,t}(P^{v*-}_{k,m,t}), \qquad t \in \mathcal{T} \tag{4.27}$$

$$\omega_t^{f*} = \sum_m \sum_f C^{\text{flex}}_{f,m,t}(D^{\text{flex}*}_{f,m,t}), \qquad t \in \mathcal{T} \tag{4.28}$$

While solving problem (D) iteratively, cost of power subject to asset deviation is,

$$\omega^p_{l,t} \leq \omega^{p*}_t \tag{4.29}$$

$$\omega^{v+}_{l,t} \leq \omega^{v*+}_t \tag{4.30}$$

$$\omega^{v-}_{l,t} \geq \omega^{v*-}_t \tag{4.31}$$

$$\omega^f_{l,t} \geq \omega^{f*}_t \qquad \forall l, \quad l \in \mathcal{L}, \quad t \in \mathcal{T} \tag{4.32}$$

As generator and virtual inc behave similarly in the optimization model, so do price sensitive load and virtual dec. What if we form the following constraints?

$$\omega^p_{l,t} + \omega^{v+}_{l,t} \leq \omega^{p*}_t + \omega^{v*+}_t \tag{4.33}$$

$$\omega^{v-}_{l,t} + \omega^f_{l,t} \geq \omega^{v*-}_t + \omega^{f*}_t \tag{4.34}$$

$$\omega^p_{l,t} + \omega^{v+}_{l,t} - \omega^{v-}_{l,t} - \omega^f_{l,t} \leq \omega^{p*}_t + \omega^{v*+}_t - \omega^{v*-}_t - \omega^{f*}_t \tag{4.35}$$

$$\omega^{v+}_{l,t} - \omega^{v-}_{l,t} - \omega^f_{l,t} \leq \omega^{v*+}_t - \omega^{v*-}_t - \omega^{f*}_t \tag{4.36}$$

$$\forall l, \forall t, \quad l \in \mathcal{L}, \quad t \in \mathcal{T}$$

We can first solve problem (C) with feasibility check method, same as two-

step offline prescreening method and can obtain the dominating security constraints (potentially binding, J_2). Then, we can formulate problem (D), and add price constraints. Identify constraint set J_3, where $J_3 \subseteq J_2 \subseteq U$

Thing needs to be addressed are:

1. Asset deviation needs a creditable range, δ^p and δ^d
2. Virtual and price sensitive demands are unrestricted from SCUC solution in problem (D), virtual is causing **MAJOR** cost deviation
3. price deviation (4.24), δ^c, could be a fixed number or a vector correlated to how far away from the power flow limit
4. Complete topology, sensitivity factor, asset location, creditable security constraint set
5. AC/DC error and loop flow

What if we are only interested in the absolute change of dispatches, or we can only account for the generator/virtual that increased in value?

$$\begin{aligned} \text{cost of deviation} = & \sum_m \sum_i C^p_{i,m}(|P_{i,m} - P^*_{i,m}|) \\ & + \sum_m \sum_k C^v_{k,m}(|P^v_{k,m} - P^{v*}_{k,m}|) \end{aligned} \tag{4.37}$$

$$P_{i,m} = \max\{P_{i,m}, P^*_{i,m}\} \tag{4.38}$$

$$P^v_{i,m} = \max\{P^v_{i,m}, P^{v*}_{i,m}\} \tag{4.39}$$

4.1 Price Constraint with LMP confidence Interval

If the set of potentially binding constraints is very large and the cost of asset deviation is too much that it is impossible to happen in real world application, we

can define a cut to further restrict the power injection interval thus, obtaining more creditable potentially binding constraints (shrunk).

The cost cut essentially modify generator's bidding curve. The use of Locational Marginal Price (LMP) confident interval may result in an infeasible model due to insufficient generation capacity. Determine the 95% confident interval of LMP at each generator/virtual node. To form a price to quantity relationship, denote as function $\Pi_m(\cdot)$. This function has to subject to hour of the day, season of the year, load level and other factors. We want this function to be dynamic.

$$\pi_m = \Pi_m(p_{i,m}) \tag{4.40}$$

Modify problem (D) constraint (4.16) and (4.19) with (4.41). We are eliminating the more expensive segments that are outside the price confident interval. If we are using 95% CI,we (4.41) will cover up to 97.5% of the price points.

$$p_{i,m}^{\text{ub}}, \quad p_{i,m}^{\text{v, ub}} = \Pi^{-1}(\overline{\pi}_m) \tag{4.41}$$

$$p_{i,m}^{\text{lb}}, \quad p_{i,m}^{\text{v, lb}} = \Pi^{-1}(\underline{\pi}_m) \tag{4.42}$$

Solve problem (D), identify potentially binding constraint, calculate deviation cost with (4.23). We have conducted tests with MISO DAM data and the result is shown in table 6.1. The price constraint method can remove up to an additional 10% of the constraints following step 2. 95% price constraint confidence interval is used in the calculation, i.e. price quantity pairs below 2.25% and above 97.5% were removed. However, the downside of this approach is the time consumption. Price constraint algorithm took around 10 minutes for a 9500 branch, 3000 bus system with 800 contingencies, where step 2 only took 3 minutes.

4.2 Virtual Confidence Interval

Virtual bid/offer accounts for about 45% of DAM capacity, and capacity and are intended to breach the gap between DAM and RTM. Virtuals are removed during real-time operation. We developed a way to use a virtual confidence interval to further shrink the net power injections that are considered in DAM. Table 4.1 shows the security constraints screening test results by removing the virtuals from the system. It mimics real time operation. "With Virtual" column is the typical DAM security constraint count, and "Without Virtual" column removes available virtual participants from the DAM clearing model. The security constraint screening technique is consistent with previous discussion. It is important to point out that "Without Virtual" column contains only 171 dominating security constraints in 36 planning periods while the entire DAM model contains 3161 dominating security constraints. It is roughly 18.5 times the computational burden during SCUC optimization.

Table 4.1: Security Constraint Screening Without Virtual

	With Virtual	Without Virtual
Offline	4231	2227
Online	3296	818
Verify	3161	171

LMP on price nodes can vary significantly due to the bidding behaviors of traditional generation units and virtual players. LMPs are not available before solving DAM SCED. However, we can determine a confidence interval of the LMP by investigating numerous historical DAM cases. Since virtual bids/offers come in finer piecewise segments, they can better represent the DAM node pricing. As illustrated in Figure 4, we first obtained the virtual LMP confidence interval by sampling from historical nodal pricing data. The upper and lower bound of pricing will correspond to an MW quantity that can be cleared. The red segment in the bid curve is the remaining MW that should be considered in security constraint filtering. The blue

segment on the virtual offer curve is the remaining MW. Overall, reducing the participating virtual quantity, will result in reduced number of potentially binding security constraints.

There are two possible ways of implementing the virtual confidence interval information. Firstly, it is for watchlist security constraint screening. It is advised to combine this approach with lazy constraint setting when going into DAM SCUC, because the potential inaccuracy caused by statistical method can be addressed by setting the less likely binding constraints to dormant during optimization. Secondly, it can be used during SFT process, where the price signal can artificially create a power injection interval.

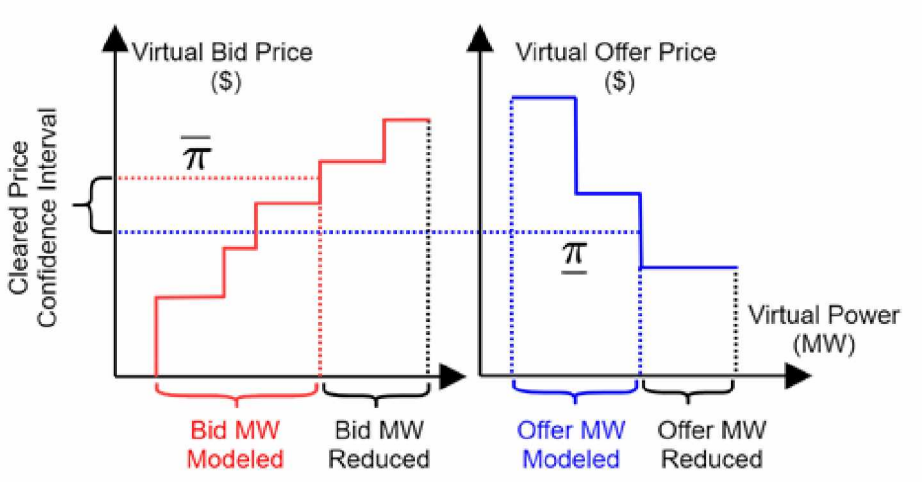

Figure 4.1: Using Virtual Confidence Interval to Reduce Net Power Injection. The red staircase bidding curve shows a typical virtual bid price quantity pair. The blue descending staircase offer curve shows a typical virtual offer price quantity pair. The cleared price confidence interval on the y-axis would cut off excessive amount of MW quantity marked on the x-axis.

One would ask of what the typical price ranges of the virtual transactions are. It is a very hard to answer. Mainly due to the nature of virtual transactions. Virtual participants are risk-takers. They make money by exploiting the divergence between DAM and RTM. To capture the price gap, virtual participants may bid opportunistically or conservatively depending on hour of the day, season, temperature,

and partial topology information. In other words, the virtual bids/offers differs drastically by node (bus) and time of the day. Virtual transactions accounts for over 45% of total cleared bids/offer in the DAM model. Virtual transaction's range is much wider than generation bids. For example, on a node with generating unit, the generator piece-wise bids can be ranging from 20-60, but virtual transaction may be around 0 to 200. Similar phenomena also exist for nodes with large sum of loads. Something interesting to point out is that depending on the type of physical asset present on a node, the bidding/offer behavior/strategy changes drastically. However, we are not allowed to discuss this information in the paper due to the fact it may expose certain business tactic/secrets.

As how virtuals are aggregated on individual nodes, as presented in the paper, we use the topology information, i.e. linear sensitivity factor matrix, to determine which bids are on which nodes. If node A shares the same sensitivity factor with node B with respect to all security constraints, then the bids/offers on node A and B can be aggregated. We would sort bids from node A and B in ascending order and offers in descending order. The aggregation needs to be done for each planned period individually, since topology change may occur.

CHAPTER V

STATIC ANALYSIS OF GENERATION AND LOAD DEVIATION POST DISPATCH

5.1 Generator Deviation and Load Side Uncertainty

Real-time dispatch and topologies vary from day-ahead market. This section discusses the divergence caused by generation deviation and load side uncertainty. The two-step security constraint screening method allows us to estimate the power flow bounds given to a system topology and resource information, i.e. generation limit, load forecast and contingencies. If the input is a range of generator dispatch and potential load deviation, we would be able to obtain the range of power flow in extreme conditions. The following equation serves as a power injection deviation index, where $P_m^{\text{inj}*}$ is the dispatch solution of DAM SCUC. $P_{\text{dev}}^{\text{inj}}$ in (5.1) is the power injection deviation metric we used to analyze the possible deviation resulted by utilizing offline method to calculate the upper and lower bounds of the power flow online l. Deviation Analysis contains the following studies,

- Mapping function of generator deviation and load uncertainty to power injection bounds
- Mapping function of power injection bounds to power flow bounds
- Potential binding constraint occurrence under credible contingencies and severity

$$P_{\text{dev}}^{\text{inj}} = \frac{\sum_m (\overline{P}_m^{\text{inj}} - \underline{P}_m^{\text{inj}})}{\sum_m \| P_m^{\text{inj}*} \|}, \quad m \in \mathcal{M} \tag{5.1}$$

The following figure shows power injection deviation as a function of generator deviation. Typically, generator deviation is between 3% 7%, and power injection deviation is in linear relationship with generator deviation in this region. We applied generation deviation with full range just as illustration. However, generator deviation is restricted by ramp rates and capacity. When uncertainty increase beyond 10%, there simply will not be enough resource to follow up.

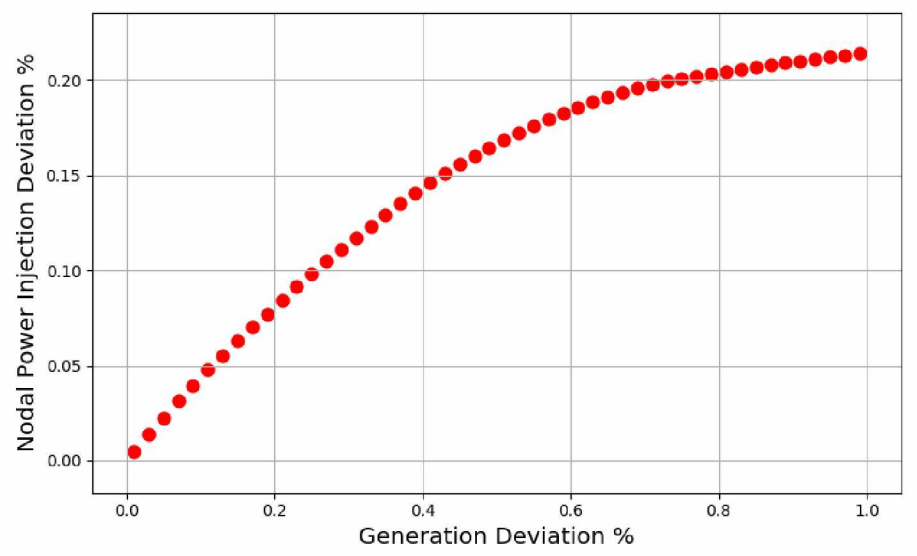

Figure 5.1: Power Injection Deviation as a Function of Generation Deviation. Note that typical generator deviation is within 7%, and power injection is linearly direct proportional to generation deviation.

Load deviation is caused by inaccuracy of load forecasting. Typically, load side uncertainty is within 10%. The following graph shows power injection deviation as a function of generator and load deviation. In the relatively small deviation region, input and output correlation is almost linear. The nonlinear relationship happens at higher input deviation percentage and is caused by lack of generation resources and load going over its limit. For example, when generator deviation is up to 17.5% and

load deviation is 10%, the nodal power injection deviation is around 16%. The linear regression tries to capture this relationship.

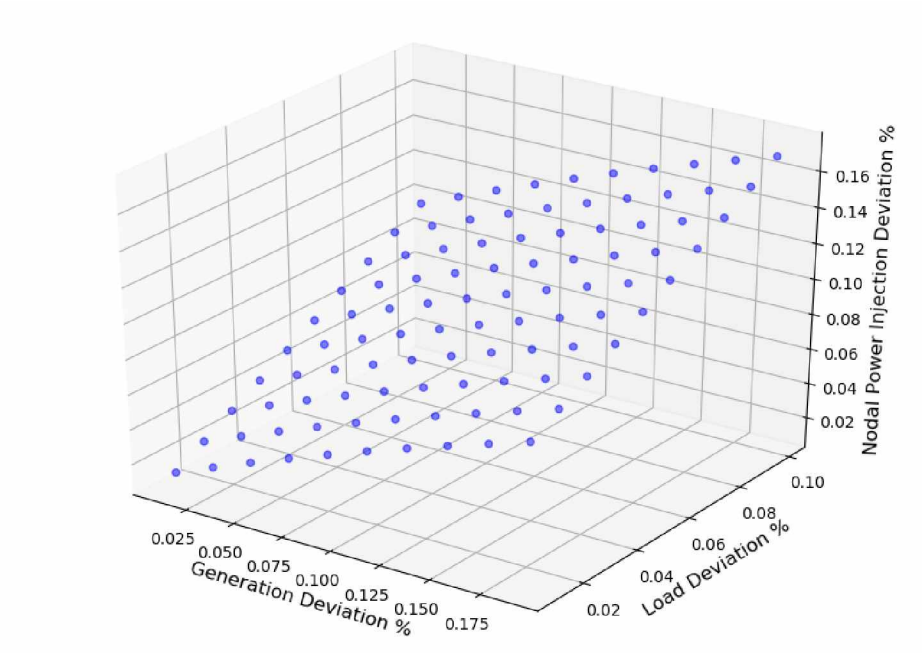

Figure 5.2: Power Flow Bounds as a Function of Generator Deviation and Load Deviation. Note that generation and load deviation have been discretized to create a finite number of pairs of uncertainties.

A simple linear regression function is used to approximate the observation that generator and load deviation post SCUC dispatch is almost linearly proportional to the nodal power injections. (5.2) shows the relationship. Note that, MSE $= 0, r^2 = 1$, it is almost a perfect linear approximation.

$$P_{\text{dev}}^{\text{inj}} = 0.40 P_{\text{dev}} + 0.92 D_{\text{dev}} \tag{5.2}$$

We explore the relationship between power injection deviation and load side uncertainty. The wind generation in MISO territory is not as potent as traditional generation sources. Table 5.1 illustrates the analysis results. "load" in the first column is the uncertainty bound that can be experienced in the system, i.e. up to 20% load

increase. However, in reality, load rarely go above 7% of the forecast value. In extreme situations, such as snow storm or heat wave, the value could deviate tremendously, but it is not our focus in this study. "Wind" column is the wind energy uncertainty bound, and we assume it can go up to 20% above the forecast value. When load and wind generation have a new bound, we can calculate the power injection deviation using (5.1), reflected in column "Power Injection". Solve problem (c) will yield a new power flow bounds based on input uncertainties.

Table 5.1: Power Injection Deviation Affected by Load and Wind Generation

Load[1]	Wind[2]	Power Injection[3]
5 %	0 %	4.97 %
10 %	0 %	9.94 %
20 %	0 %	19.88 %
0 %	10 %	0.34 %
0 %	20 %	0.68 %
5 %	10 %	5.31 %
10 %	10 %	10.28 %
10 %	20 %	10.62 %
20 %	10 %	20.22 %
20 %	20 %	20.56 %

[1] System wide load deviation
[2] System wide Wind deviation
[3] Power injection deviation

From table 5.1, it is easy to observe that power injection is greatly affected by load deviation, and wind generation uncertainty have little effect. When system-wide load is increased by 20%, power injection deviation is increased by 19.88%. If the system only experience 20% wind generation increase, power injection only varies by 0.68%. In the most extreme situation, where both load and wind generation is 20% above the forecast value, system-wide power injection increases by 20.56%.

In the next step, we attempted to create a mapping function between the power injection deviation to the power flow bounds. The following equation (5.3) serves as an indicator of power flow variation. f^* is the SCUC power flow solution,

F_l is the power flow limit and f_l^δ is the power flow deviation.

$$f_l^\delta = \frac{f_l - f_l^*}{F_l} \tag{5.3}$$

The following histogram shows the distribution of power flow deviation on all security constraints modeled in base case when fixing generator deviation to 7%, with mean of 3.65% and standard deviation of 5.44%. 95% of power flow deviation is within 12.25% of the SCUC power flow solution.

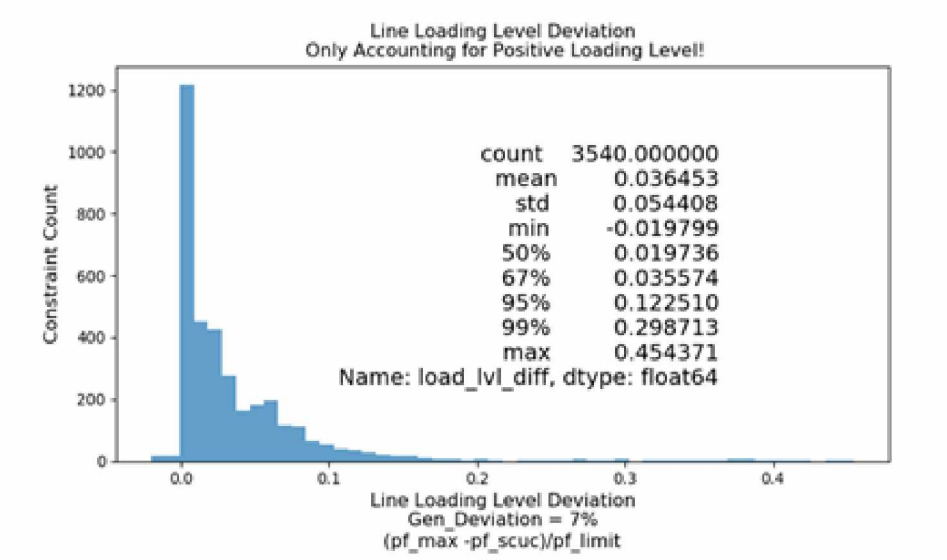

Figure 5.3: Power Flow Deviation Associated with 7% Generation Deviation. Deviations are Compiled into a Histogram for illustration purpose

When treating generator deviation as a variable, and solve power flow bounds with the two-step methods, we can generate the following graph. The relationship between generator/load deviation and power flow deviation is relatively linear at low percentage level and exponentially decaying at high percentage level. However, generator deviation above 10% is not realistic. The following figure also shows that the power flow bounds deviation is heavily skewed to the lower loading level. Standard deviation cannot accurately capture the behavior. We used Medium Absolute Devi-

ation (MAD) to capture the trend. MAD is a robust measure of the variability of a univariate sample of quantitative data.

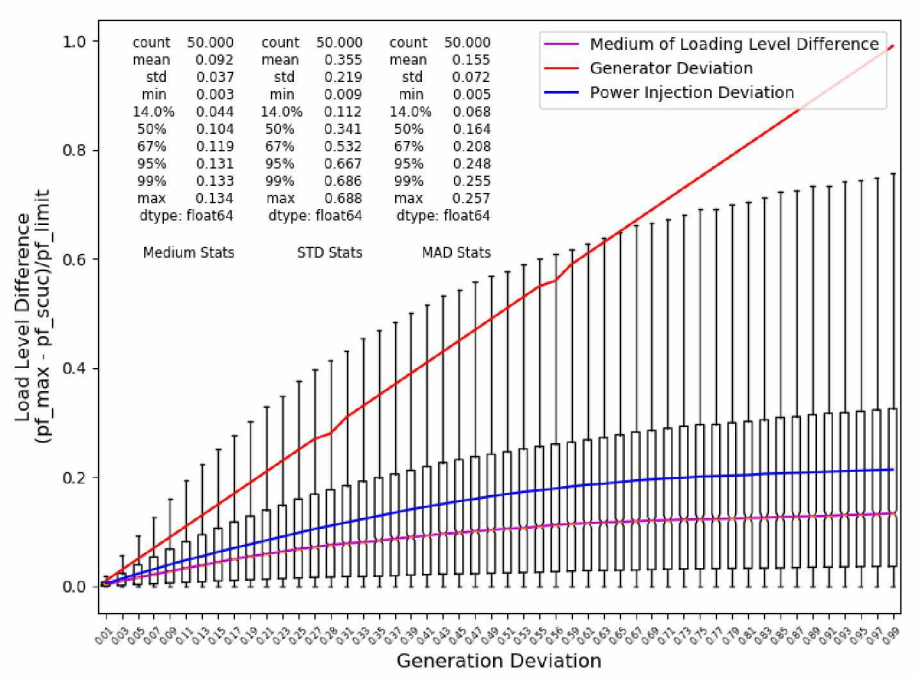

Figure 5.4: Power Flow Deviation Distribution Associated with Various Generation Deviation. Note that the assumption is generation deviation is below 7%, and the respective power flow bounds deviation is relatively linear and heavily distributed under 10%.

The following plot is security constraints sorted in descending order based on occurrence of violations. Generator deviation is set to below 20% and load deviation is below 10%. There are 36 intervals in each DAM SCUC case, we simulated power injection pretending to be the real time dispatch value. The simulated value is bounded by the uncertainty region discussed previously. Not all constraints are present in each interval, therefore, technically the maximum occurrence of a constraint is $7200 = 36 \times 100 \times 2$ (# of intervals $\times$ # of cases $\times$ # of directions). If the simulated power injection produces a power flow upper bound that is higher than its power flow limit, we will consider it as potentially binding in real time. The findings from the static analysis shows,

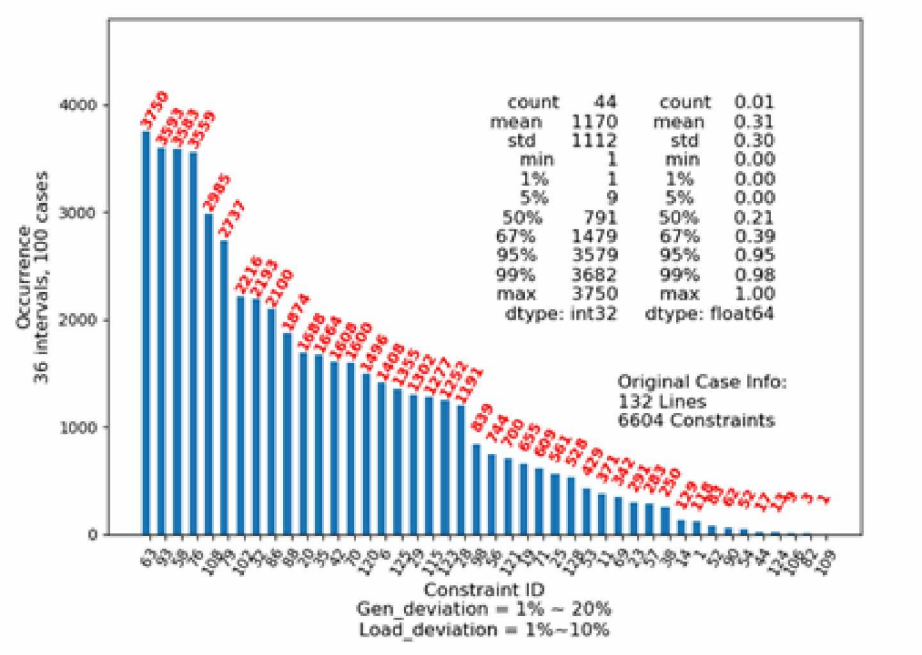

Figure 5.5: Potential Binding Security Constraints Sorted by Violation Count.

1. Generation/Load deviation is almost linearly direct proportional to power injection deviation

2. At 7% generation deviation, 5% of security constraints have the potential of becoming binding in RTM, that is approximately 9.1 constraints per interval

3. With up to 7% generation deviation, the power transfer on existing binding security constraint would increase 10%, with standard deviation of 0.026. The 95% percentile is 0.39. In other words, the power transfer would up to 39% in 95% of the situations.

4. With up to 20% generation deviation and 10% demand deviation, potentially binding security constraint would experience up to 6.5% increase with standard deviation of 2.6% and, the 95^{th} percentile is 10.9%.

5. Figure 5.5 shows the potential binding security constraints sorted by violation counts, potentially binding constraints occurrence under power injection deviation.

5.2 Deviation and Security Constraint Screening Analysis on RTS96 System

The application of security constraint offline-online screening is applied to RTS96 test case [110]. Test result is compared with IEEE-118 Bus test case. It is shown that the proposed method is consistent in terms of dominating constraint detection and computational performance. RTS96 test case consists of 120 transmission lines, 73 buses, 158 generators, and 51 loads. Day-ahead market load data is given for every hour over 366 days. And the real-time market load data is given for every 5 minutes intervals. In this experiment, we first apply the offline method to one specific period to determine the dominating security constraints (N-1 contingency) and dual variables, then use the updated real-time data (5 minutes interval, 12 data sets for the specific hour) to determine the power flow upper bound. Lastly, apply the offline method to the 12 sets of real-time data to determine the true dominating constraints and power flow upper bounds. We present dominating constraint detection and computation performance with the proposed framework.

Data analytics were performed on load variation between day-ahead market and real-time market. Since real-time data is in 5 minutes intervals and day-ahead data is in 1-hour intervals, we would take the hourly average value of real-time data to compare with day-ahead data. We would like to understand the load deviation between the two data sets. The deviation is expressed as [(Real-time MW - Day-ahead MW)/Day-ahead MW] as the statistic shown in the following figure. The 95% percentile is between -11.6% and 4%. We assumed 10% load side deviation in the experiment onward. (This also justifies the 10% load side uncertainty assumption that's been made previously). The histogram of load deviation between real-time and day ahead forecast is shown in figure 5.6.

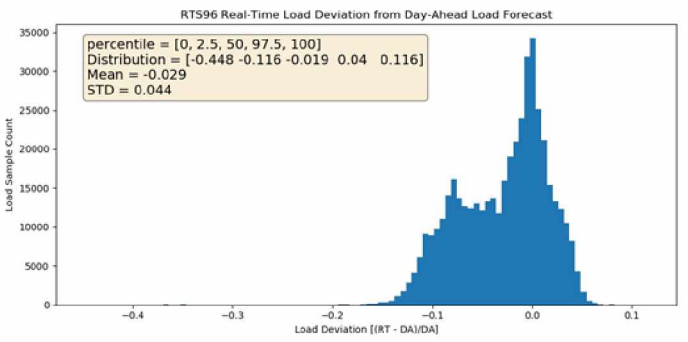

Figure 5.6: RTS96 Real-Time Load Deviation from DA Load Forecast

Table. 5.2 shows the performance of proposed offline-online security constraints method on RTS-GMLC system and IEEE-118 bus system. In Offline Screening stage, step-1 finds 20% of the security constraints potentially binding in RTS-GMLC system and 15% in IEEE-118 bus system. The computation time is 0.18 seconds and 1.12 seconds, respectively. Step-2 solves 5,685 LPs and further determines only 106 are dominating constraint, which accounts for 0.37% (=106/28,800) of N-1 security constraints in RTS-GMLC system. For IEEE-118 bus system, the dominating constraints rate is 0.15%. The LP calculations took 165 and 332 seconds, respectively. Note that we applied the modified step 2 method feasibility check, i.e. problem (C^*). On RTS-GMLC system, the feasibility check performed on 5685 constraints is 165.12 seconds; 106 constraints have the potential of becoming dominating, perform LP calculation 106 times takes 0.338 seconds. However, if we solve the LP directly 5685 times, it would take 1035.03 seconds. It is observed that the significant computation burden of step-2, even for a relatively small size system, calls for a more efficient method of real-time security constraint monitoring.

Online Screening is then applied with real-time net power injection. Online method is based on the closed-form equation (3.26). It takes 2.5 ms and 2.4 ms to finish the calculation on RTS-GMLC system and IEEE-118 bus system, respectively. Compared with 165 and 332 seconds for offline screening, the online version shows significant advantage in processing time. When the uncertainty is material-

ized, the online screening finds that 99 out of 106 security constraints are potentially dominating in RTS-GMLC, and 69 out of 101 in IEEE-118 bus system. Apply Verification to same data set used in Online Screening, 91 out of 99 constraints are truly dominating in RTS-GMLC system, and 53 out of 69 in IEEE-118 bus system. It shows that the online component has relatively accurate predication of the dominating constraints.The comparison between the two medium size test system shows that the proposed method is consistent in terms of dominating constraint detection and computational performance.

Table 5.2: RTS-GMLC/IEEE-118 bus System Security Constraint Screening Results

	Dominating Constraint Detection		Time (Seconds)	
Test Case	RTS-GMLC	IEEE118	RTS-GMLC	IEEE118
Original[1]	28,800	69,492	-	-
Step-1[2]	5,685	9,643	0.18	1.12
Step-2[3]	106	101	165	332
Online[4]	99	69	0.0025	0.0024
Verification[5]	91	53	147	326

[1] N-1 security constraints modeled
[2] Apply step-1 to the test system
[3] Apply step-2 offline screening including uncertainty
[4] Apply step-2 online screening with real-time data
[5] Verify the results by applying step-2 offline method with real-time data

5.3 Post DAM SCUC Power Injection Sampling

We sampled power injections with DAM SCUC solution as the set point, and ramp rate/standard deviation as uncertainty margin. This study aims to answer the questions that how power flow bounds would vary given the credible power injection bounds.

Figure 5.7 illustrates the idea of sampling power injection after DAM SCUC and before SFT. The red bar in the figure is the optimal solution from SCUC (power injection on bus m, for all buses). Theoretically, generation units must follow MISO

order. i.e. SCUC solution. However, there are situations where the generation unit operations are delayed. We call the difference between SCUC power injection and actual generator dispatch as "gap" or "generator deviation". And to quantify the gap. as shown below, we assumed that the gap has to be smaller than the actual ramp rate or two standard deviation around the mean. i.e. expected value (SCUC dispatch value). The distance between the two black bars is the absolute gap (generator deviation). Any point sampled between the upper bound and the expected value. we call it the sampled power flow upper bound. and any point sampled between the expected value and the absolute power injection lower bound is the sampled lower bound. The two-sampling distribution we used in the following studies are uniform distribution and normal distribution.

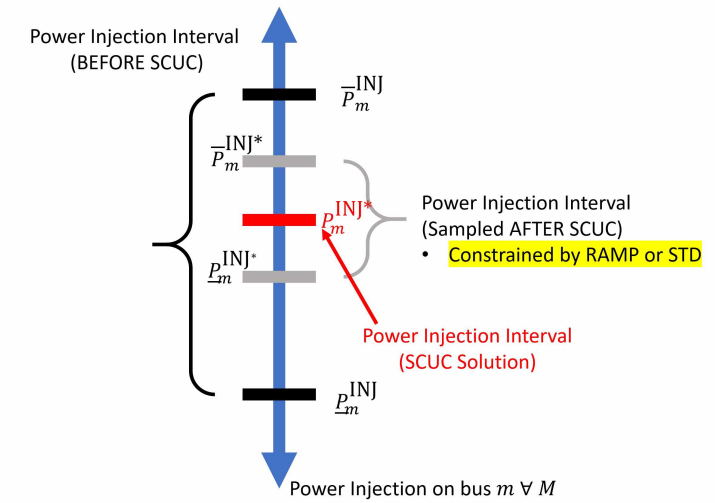

Figure 5.7: Power Injection Interval

To emphasize the goal of sampling, we outlined the uncertainty bound of power flow given as a set point and a range of power injection deviation (from the set point). Figure 5.8 illustrates the idea. The blue star is the SCUC solution, if we can clearly define the uncertainty region. i.e. ramp rate. standard deviation. We can picture that the deviated optimal solution will reside in the uncertainty region (shaded area). Red star is the new optimal solution after deviation. Power injections

sampled from the shaded region will yield individual power flow bounds. And it may resemble what happens in real time.

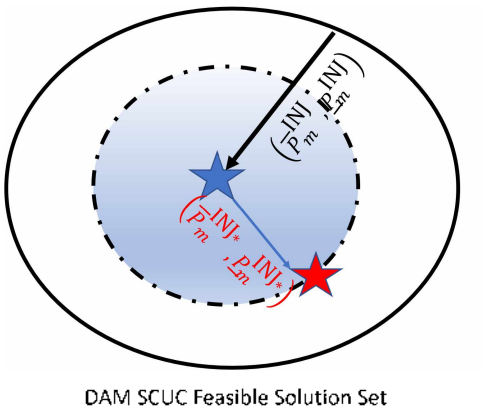

Figure 5.8: DAM SCUC Solution Set Variations

Figure 5.9 shows the sampling results. With set point being SCUC solution, power injection interval constrained by either ramp rate or two standard deviation and sampling distribution using uniform or normal, we can make the following observations.

The branch under study (branch index 6296) has power flow limit of 212 MW. 1st figure is the power flow upper bound distribution using ramping constraint and uniform distribution. All power flow upper bounds are under the limit (indicated by the red vertical line). 2nd figure is the power flow upper bound distribution using generator standard deviation as constraint and uniform distribution. The majority of power flow upper bound distribution is over power flow limit. This may be an indicator that the line can be over limit. And the 3rd figure uses generator standard deviation as constraint and normal distribution. And all power flow upper bounds are under limit.

The question becomes which assumption is most realistic. Future study is required to understand the relationship between sampling distribution and constraint that is associated with the gap between DAM and RT market. The study should be done on a per branch basis.

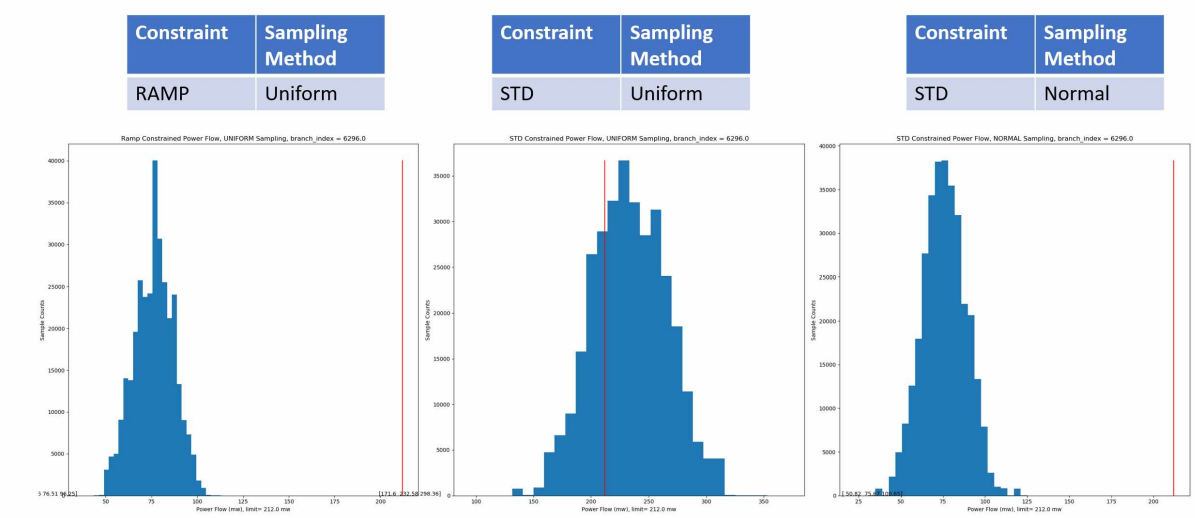

Figure 5.9: Post SCUC Power Injection Sampling and Power Flow Bounds Variation

5.4 Impact Index

In the previous sections, we discussed the power flow deviations based on power injection deviations and constraint violations given creditable contingencies. Note that the assumption is a fixed optimal solution resulted from SCUC calculation. Two important metrics derived from the analysis are the severity of violation and how frequent the violations are. We proposed to combine both metrics to form the **Impact Index**, where a single metric that can help the system operator to pick the security constraint that may exert the most stress in the system. f_l'' is the average of sampled power flow for line l for all the contingency power flows. $\overline{F_l}/\underline{F_l}$ is the upper/lower limit of line l. n is the sample size, n = # sample/ctg × # ctg. The **Severity** of the power flow violation online l can be defined as (5.4). **Violation Frequency** is defined as (5.5). The **Impact Index** is defined as (5.6).

$$\text{Severity} := \max\left\{\frac{f_l^\mu}{\overline{F}_l}, \frac{f_l^\mu}{\underline{F}_l}\right\}, \quad \text{where} \quad f_l^\mu = \frac{\frac{\sum_n f_{l,n}}{n}}{\overline{F}_l} \tag{5.4}$$

$$\text{Violation Frequency} := \frac{\text{violation count}}{n} \tag{5.5}$$

$$\text{Impact Index} := \text{Severity} * \text{Violation Frequency} \tag{5.6}$$

We use the following data from MISO DAM to demonstrate the potency and effectiveness of impact index ranking. Table 5.3 lists the max power flow calculated from (5.4), power flow limits and severity for 5 security constraints. The name of the branches are omitted due to nondisclosure agreement.

Table 5.3: Power Flow Violation Severity Calculation

Line #	Max Flow (MW)	Flow Limit (MW)	Severity (%)
1	150	115	1.30
2	225	185	1.22
3	60	50	1.20
4	175	100	1.75
5	525	500	1.05

Table 5.4 displays the frequency of violations, i.e. power flow over limit. For example, total creditable contingency count is 805, meaning there are 805 various topology changes. Line 1 is only present in 100 contingencies, and the power flow on this line violates its power flow limit by 10 times, resulting in 10% occurrence, shown in the "Frequency" column. The calculation follows (5.5).

Next, we can calculate the impact index following (5.6) and the resulting impact index is shown in 5.5. Line index is sorted based on impact index in descending order, as shown in table 5.6. Line 4 has the highest impact index of 1.58, due to having high severity (1.75) and violation frequency (90%). System operator should consider to include constraints with the highest impact index.

Table 5.4: Power Flow Violation Frequency Calculation

Line #	Violation Count	Total Scenario (MW)	Frequency (%)
1	10	100	0.10
2	75	150	0.50
3	15	250	0.06
4	18	20	0.90
5	45	300	0.15

Table 5.5: Power Flow Violation Impact Index Calculation

Line #	Severity	Frequency (%)	Impact
1	1.30	0.10	0.13
2	1.22	0.50	0.61
3	1.20	0.06	0.07
4	1.75	0.90	1.58
5	1.05	0.15	0.16

We plotted the impact index of the security constraints in descending order for the MISO DAM case in figure 5.10. The X-axis is the branch ranking, the real branch/constraint name is omitted for simplicity. The Left Y-axis shows the violation frequency and right Y-axis shows the severity scale. Impact index is marked in red. This figure provided the system operators an visual aid that offline studies can pick out the security constraints that has the highest risk of increase system stress. The right vertical line is a 20 constraint cut off. It shows that if only 20 constraints can be included for future evaluation, the indices of the constraints are presented by the impact index list. The system operator can also set an impact index threshold, the cutoff will base on the interpretation of severity and violation frequency. The latter requires correlation study between impact index and real-time constraint violation.

Table 5.6: Impact Index Sorted from Highest to Lowest

Line #	Severity	Frequency (%)	Impact
4	1.75	0.90	1.58
2	1.22	0.50	0.61
5	1.05	0.15	0.16
1	1.30	0.10	0.13
3	1.20	0.06	0.07

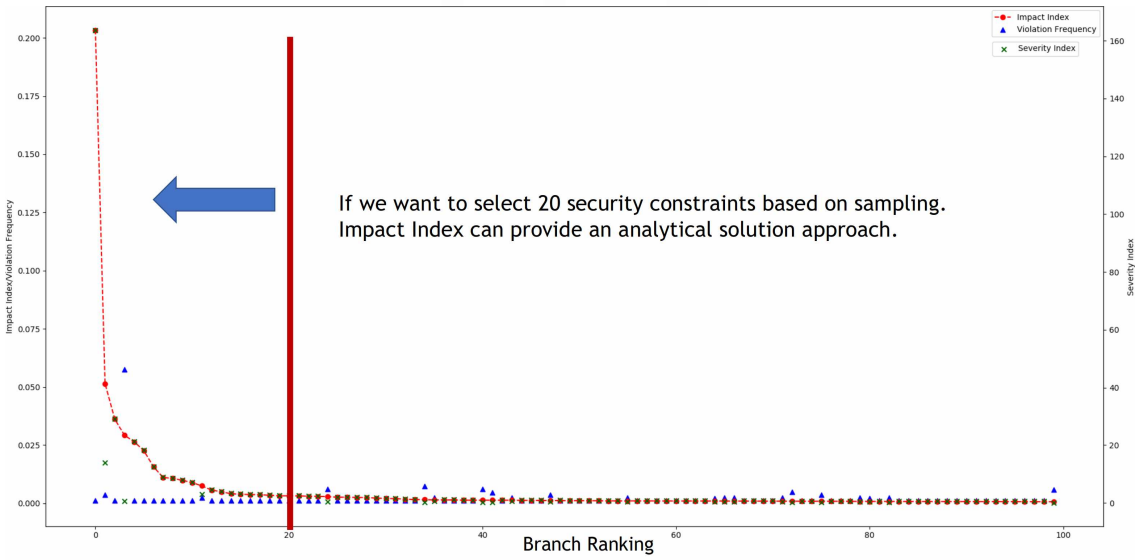

Figure 5.10: Impact Index Plotted in Descending Order

Figure 5.11 shows the distribution of the impact factors of the 9498 security constraints considered in SFT. The x-axis is the impact factor in ascending order, the higher the placement, the more stress a particular group of constraints can create stress in the system. The y-axis is the security constraint counts. It is obvious that the majority of the branches are safe from over loading. We estimated that only 8.27% (786/9498) violated the security constraint screening with uncertainties. The size of the watch list pool is 786, and it still seems large to be considered entirely in the SFT interactions with SCUC. We can follow the previous step by only picking the security constraints with the highest impact index to relax the energy dispatching protocol.

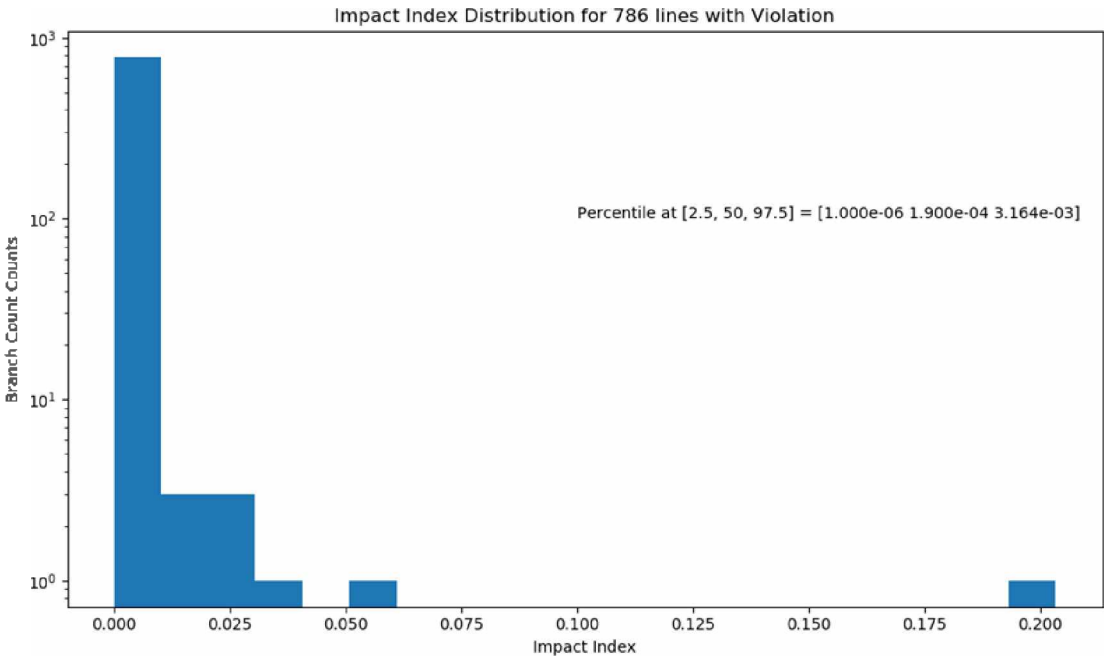

Figure 5.11: Impact Index Distribution

CHAPTER VI

CONTINGENCY SCREENING

The contingency screening procedure is an enhanced SFT module that applies the previous mentioned approach to the DAM SCUC solution (power injection) and credible contingencies to create a pool of security constraints that could be binding in real time.

Traditionally, SFT module takes the solution of DAM SCUC and apply the optimal dispatch and commitment to a set of credible contingencies. It is usually one stop solution to determine the DC power flows, and they are checked against respective power flow limits. Any violation would result in the constraint being put back into the SCUC module and solved again.

The enhanced module will take the DAM SCUC solution, add uncertainties to create power injection interval. Then use the power injection interval to determine the power flow bounds. Therefore, any possible deviation within the uncertainty assumption would be mapped into a set of power flow bounds solutions. Figure 6.1 shows the workflow. The module does the following,

- Create power injection uncertainty bounds from SCUC base case solution
- Determines potential binding security constraints with DAM SCUC as a set point
- Introduce price constraint
- Rank constraints with associated impact index

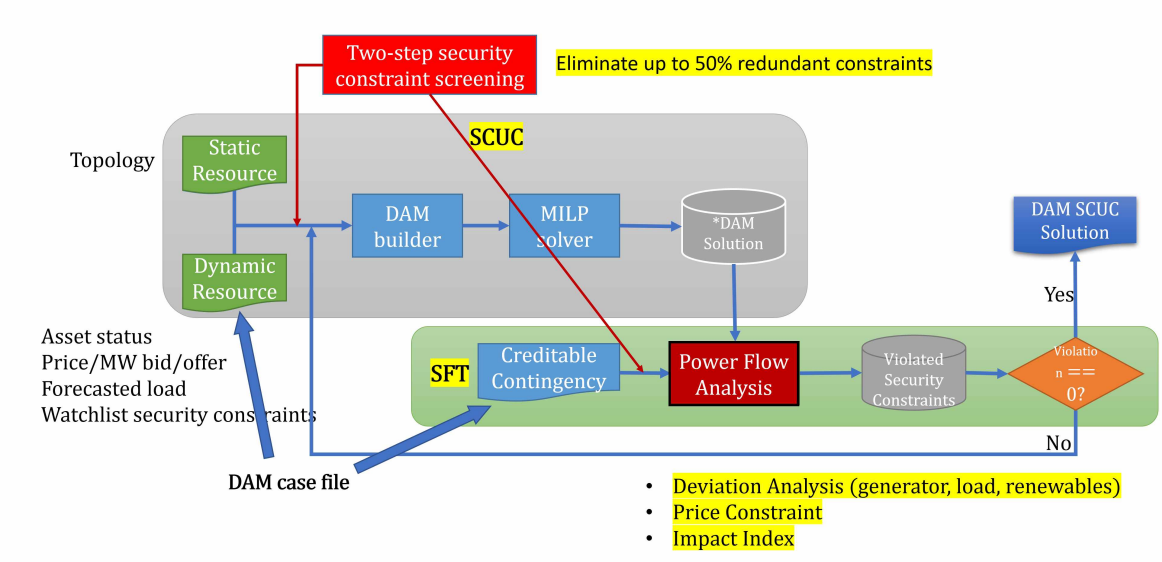

Figure 6.1: Enhanced SFT Module

The power flow upper/lower bounds are calculated using the security constraint screen in step 2. 6.2 illustrated the relationship between bounds and their respective power flow limits. the solid dots represent the power flow bounds calculated from all creditable contingencies. and the smooth lines represents the power flow limits. If the dots are bounded by the smooth lines, these constraints are redundant, and they can be safely removed from the module. When the dots are outside the bounds, we need to consider adding these constraints back to the SCUC model, since they are dominating constraints that create violations if the contingency is realized in real time.

Figure 6.2: Power Flow Bounds in SFT

We use MISO DAM data to demonstrate the effectiveness of enhanced SFT calculation. For a particular day, in a given period, the system contains 9498 security constraints, 805 creditable contingencies, and both directions of power flow have to be considered. Total constraints under evaluation is $15,291,780 = 9498 * 805 * 2$. First step should be to enumerate all the security constraints, then apply step 1, the numerical method, to determine potentially binding constraints. However, we realized it would take an extraordinary amount of time and space from our experiment. We developed a modified step 1 method to save resource, it contains three steps.

1. Step 0 Coarse: Determine the *rough estimate* (MW) of the maximum power flow for all constraints. $\Gamma^{+}_{l,m}/\Gamma^{-}_{l,m}$ denote the sensitivity factor that is greater/less than zero for a line l at bus m. P^{up+}_{m}/P^{up-}_{m} denote the power injection upper/lower bound that is greater/less than zero at bus m. If (6.1) or (6.2) is true, it means that under rough estimate conditions, i.e. not accounting for minor difference, the constraint can be violated in the forward flow direction. If (6.3) or (6.4) is true, it means that under rough estimate conditions, the constraint can be violated in the reverse flow direction. The constraint under

study will be marked for further screening and redundant constraints can be safely removed.

$$\Gamma^{+}_{l,m} P^{\text{inj}+}_{m} \geq \overline{F}_{l} \tag{6.1}$$

$$\Gamma^{-}_{l,m} P^{\text{inj}-}_{m} \geq \overline{F}_{l} \tag{6.2}$$

$$\Gamma^{+}_{l,m} P^{\text{inj}-}_{m} \leq \underline{F}_{l} \tag{6.3}$$

$$\Gamma^{-}_{l,m} P^{\text{inj}+}_{m} \leq \underline{F}_{l} \tag{6.4}$$

2. Step 0 Stacking: Stack the results of step 0 coarse into a three-dimensional matrix. When we enumerate all the security constraints from the creditable contingency, the sensitivity factor matrix was $[\mathcal{L} \times \mathcal{M}]$. After including contingency dimension, the new matrix size becomes $[\mathcal{L} \times \mathcal{M} \times \mathcal{C}]$, where $\mathcal{C}$ is the contingency set.

3. Step 0 Unique: Determine the unique security constraint based on $[\mathcal{L} \times \mathcal{M}]$ across the $[\mathcal{C}]$ dimension. In plain words, we are trying to categorize contingencies. Line outage contingencies would usually impact only a small part of the sensitivity matrix. This is mainly done by utilizing the "numpy.unique" function by Python. Another way to determine whether the difference amongst topologies is by setting a tolerance level, θ, as shown in (6.5). $\Delta\Gamma^{c}_{l,m}$ is the difference amongst sensitivity factors among various contingencies. Since, we only have either the upper or lower bound of the injection at bus m, the maximum positive value will yield the highest disturbance when multiplied by the absolute sensitivity factor difference. θ can be either a hard set value, i.e. 5 MW, or as a fraction of the power flow limit, i.e. 5 MW = 5% * 100 MW, where $\overline{F}_{l} = 100$ MW. The physical interpretation is that the variation amongst topologies can

be ignored if the power flow disturbance is under threshold, θ.

$$|\Delta\Gamma^{c}_{l,m}| \cdot \max\{\overline{P}^{\text{inj}}_{m}, |\underline{P}^{\text{inj}}_{m}|\} \leq \theta \tag{6.5}$$

Table 6.1 shows the enhanced SFT test result. Step 0 Coarse scanning filtered 243,848 out of 15,291,780 constraints can be binding. This process took 28 minutes and cost 6.3 GB of memory space. The stacking process took 116 seconds, but reduced the memory usage to 210 bytes. This is due to the use of sparse matrix. we could not do this in the previous step, because each contingency has to be generated individually. Step 0 Unique reduce the constraints by another half, 130,740 constraints remains in the pool. Lastly, we run the two-step security constraint screening method, only 177 constraints have the potential to become binding, reduced from the original 15 million constraints!

The goal of enhanced SFT is to use the power flow bounds to create an uncertainty bounds for power flows amongst various topology, i.e. creditable contingencies. Traditional SFT using a simple one-step calculation on a fixed point, the SCUC power injection, is not robust against uncertainties. Any deviation that is not "discretized" is considered and may cause unexpected operating conditions in the system where enhanced SFT creates a buffer zone to levitate stress from system operators. So long as the input uncertainties are well quantified, a potential binding constraint pool will be created based on intervals. The two challenges we face are the time and space. It took almost 40 minutes to screening through 15 million constraints, and initially took over 6 GB memory space. These problems will be addressed in future work section.

6.1 Machine Learning Applications

Congestion forecasting tool aims to use DAM participant information, contin-

Table 6.1: Enhanced SFT Test Result

	Constraint Count [1]	Constraint %[2]	Time	Memory
Step 0 Coarse[3]	243848	1.59%	28 Minutes	6.3 GB
Step 0 Stacking [4]	243848	1.59%	116 Seconds	210 Byte
Step 0 Unique [5]	130740	0.85%	81 Seconds	112 Byte
Step 1	1358	0.01%	703 Seconds	-
Step 2	177	0.00%	350 Seconds	-
Step 2 Price Constraint[6]	153	0.00%	589 Seconds	-

[1] Constraint count at each stage
[2] Dominating constraint percentage after each screening step
[3] Modified step 1
[4] Aligning constraints into a three-dimensional matrix
[5] Determine the uniqueness of constraints
[6] Include the price constraints in step 2

gency planning, assets uncertainties to create a pool of potentially congested lines. When more information becomes available, such as renewable and load forecasting, we can use machine learning technique to predict when a constraint will become binding in real time, and the power flow bounds of the constraint. The entire congestion forecasting tool workflow is composed of two major parts: offline deterministic power flow analysis and online machine learning model. The output is a list of potentially congested line in real time, sorted by severity index. Figure 6.3 shows the framework.

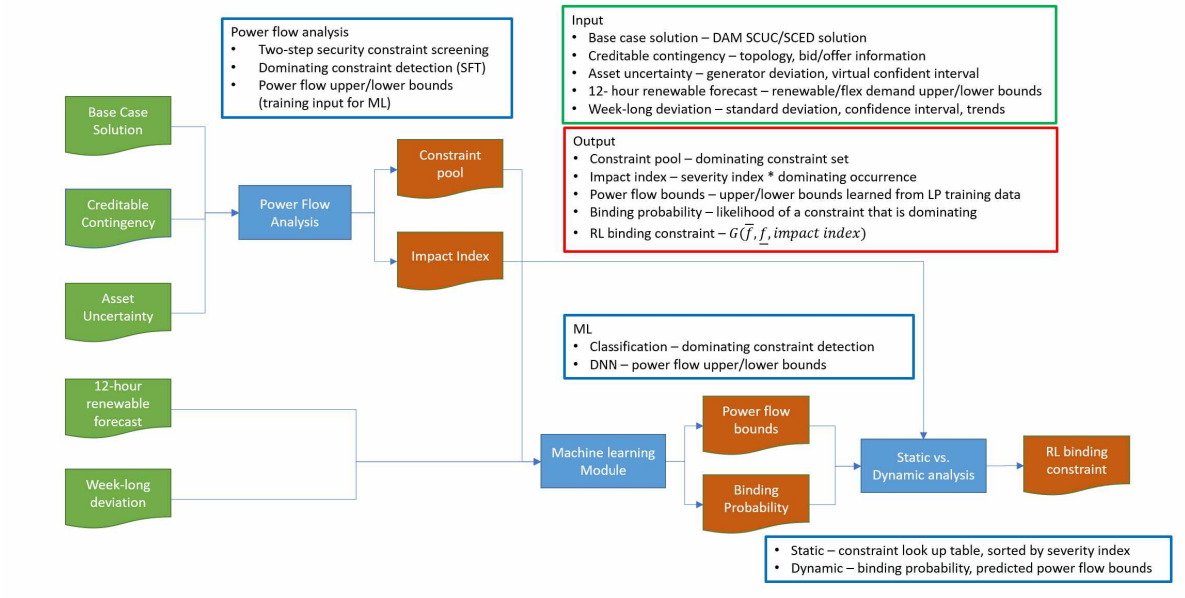

Figure 6.3: DA-RT Congestion Forecasting Workflow

Offline deterministic power flow analysis is discussed in detail in the methodology section, which contains two-step security constraint screening, SFT screening, deviation analysis, and price constraints. The input to the power flow analysis model is.

1. Base case solution — base case topology, generator parameters, load forecasted, virtual bid/offer
2. Credible contingency — pre-selected contingency topologies
3. Asset uncertainty — historical bid/offer information, generation/load deviation or confidence interval

The output to the model is.

1. Constraint pool — the complete list of potentially binding security constraints. The list is screened from base case and contingency cases
2. Impact index — discussed in Deviation Analysis section, a list of security constraints that is sorted by impact index in descending order

Machine learning module currently has two scripts, one is a classification algorithm to determine whether a security constraint is a dominating one. The inputs are system-wide load and DAM constraint sensitivity factor. The other script is a deep neural network (DNN) to predict the upper/lower bounds of targeted security constraints. The input is the same as classification algorithm. Work flow is shown in figure 6.4. Note that, in the future works, the module should include input of generator deviation, and previous day commitment. More information introduced into the module can improve the accuracy. However, the increase in complexity also bring burdens to model training. Feature selection and engineering is crucial to determine the relevant features that are most valueable to train an accurate model.

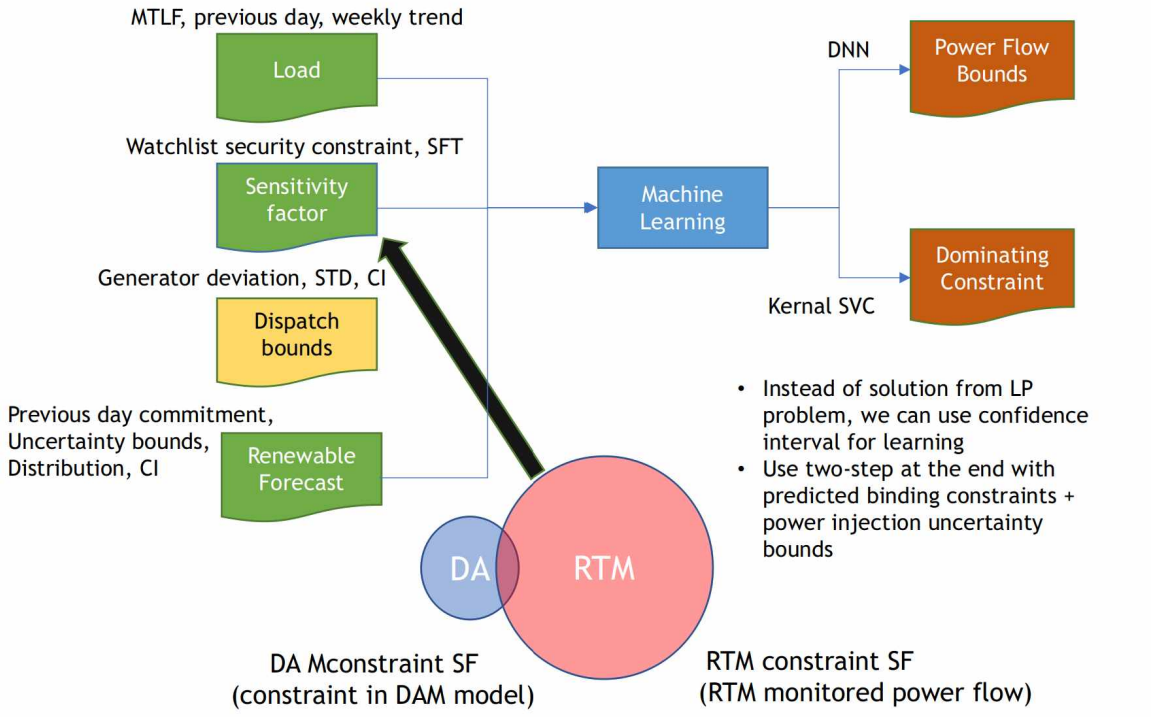

Figure 6.4: Machine Learning Module Work Flow

One of the challenges pointed out in figure 6.4 is that DAM security constraints, i.e. watchlist security constraints, are a subset of all security constraints. RTM topology is not one of the contingencies considered in DAM. This creates the situation where RTM security constraint pool is significantly larger in both spatial

and temporal perspective. In the system operator perspective, RTM data is available every 5 minutes, where DAM market clearing is in 60 minutes intervals. Using DAM market information to predict RTM status is lacking of efficient data, similar as under sampling.

The last part of the workflow is by combining the machine learning results with the deterministic power flow analysis. We replaced the power flow bounds used to calculate the severity index with the power flow bounds predicted by DNN to create a new set of severity index. By inspecting both indexes, the operator can make an educated guess as to which constraints to look out for. The congestion forecasting tool aims to provide the DAM team with an analytic method on choosing which constraints to include in RTM.

6.2 Deep Neural Network Experiment Results

Neural Network is a machine learning model/strategy widely used in the industries for forecasting and nonlinear approximations [111]. Deep learning in our research refers to neural network with multi-layer perceptions. Figure 6.5 shows the network diagram for a two-layer neural network structure representing (6.6). The input, hidden, and output variables are represented by nodes, and the weight parameters are represented by links between the nodes, in which the bias parameters are denoted by links coming from additional input and hidden variables x_0 and z_0. Arrows denote the direction of information flow through the network during forward propagation.

$$y_k(\mathbf{x}, \mathbf{w}) = \sigma\left(\sum_{j=1}^{M} \omega_{kj}^{(2)} h\left(\sum_{i=1}^{D} \omega_{ji}^{(1)} x_i + \omega_{j0}^{(1)}\right) + \omega_{k0}^{(2)}\right) \tag{6.6}$$

$\mathbf{W}$ is the weight and bias parameter group. ω represents the weights and $h(\cdot)$ is a nonlinear activation function, we use the non-sigmoidal rectified linear unit (ReLU),

$\sigma(x) = max(x, 0)$, for this experiment [112]. The neural network model is a nonlinear function from a set of input variable x_i to a set of output variable y_k controlled by a vector $\mathbf{w}$ of adjustable parameters. Forward propagation and backward propagation are utilized to calculate the weights. It is thoroughly discussed in [113]. We used ADAM as our optimizer [114]; it is a built-in to tensor-flow [115]. (6.7) is the typical loss evaluation, where $\mathbf{t}_n$ is the prediction vector. By minimizing the loss, our goal is to determine the best weights, $\mathbf{w}$, that yields the most accurate prediction for the neural network model.

$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}||\mathbf{y}(x_n, \mathbf{w}) - \mathbf{t}_n||^2 \tag{6.7}$$

Security constraint screen step 2 solves problem (C), the objective is the power flow upper bound of line l. It would be extremely time-consuming to solve N-1 contingency for all 186 line outage contingencies of the IEEE-118 bus system. Our goal of the DNN is to replace the extensive LP programming with a trained neural network. The training input of DNN is [hour, branch index, flow direction, contingency index, individual loads and 118bus system N-1 contingency sensitivity factor], input vector dimension is $[123 \times 1]$. The output is [power flow upper bound $\overline{F}_l$, power flow lower bound $\underline{F}_l$], the output vector dimension is $[2 \times 1]$. The training set size is 20 million $= (185 + 1) * 186 * 2 * 24 * 12$, (185 contingencies + base) $\times$# branches$\times$# direction $\times$# periods $\times$# scenarios. During training, we used an 80/20 train/test data split, input is scaled by normalization, the dropout rate is 0.2 and 10 folds of cross validation.

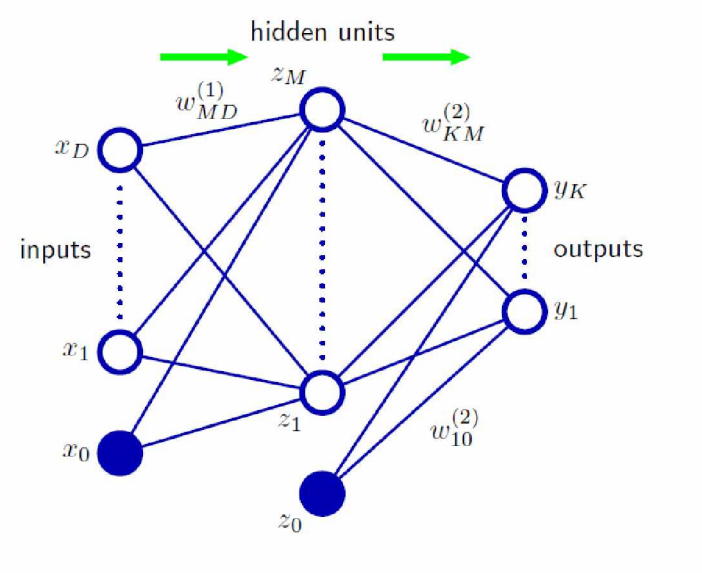

Figure 6.5: Two-layer Neural Network Diagram

Figure 6.6 shows the DNN results on IEEE-118 bus with N-1 contingency. Note that not necessarily more nodes and layers will make the prediction better. In our case, we attempted to predict the upper and lower bounds of power flows post DAM SCUC with added uncertainties. Experimental result shows that two hidden layers with 80 and 55 nodes will have the least loss value.

In figure 6.6, the x-axis is the actual upper/lower bounds which has been calculated by the linear programming method (step 2), and the y-axis is the predicted upper/lower bounds by the DNN. It is observed that for constraints with lower power flow bounds, the variance of prediction could be larger than the ones with higher bound values. In other words, the prediction is more accurate for constraints with larger power flow bounds. We can also observe that non-zero predictions while the ground truths are zero (dots reside on the y-axis while x is equal to zero). These predictions are due to the bound calculation are zeros in the training set. It is unavoidable for the IEEE-118 bus system. We imagine that real world system will have less zero bounds.

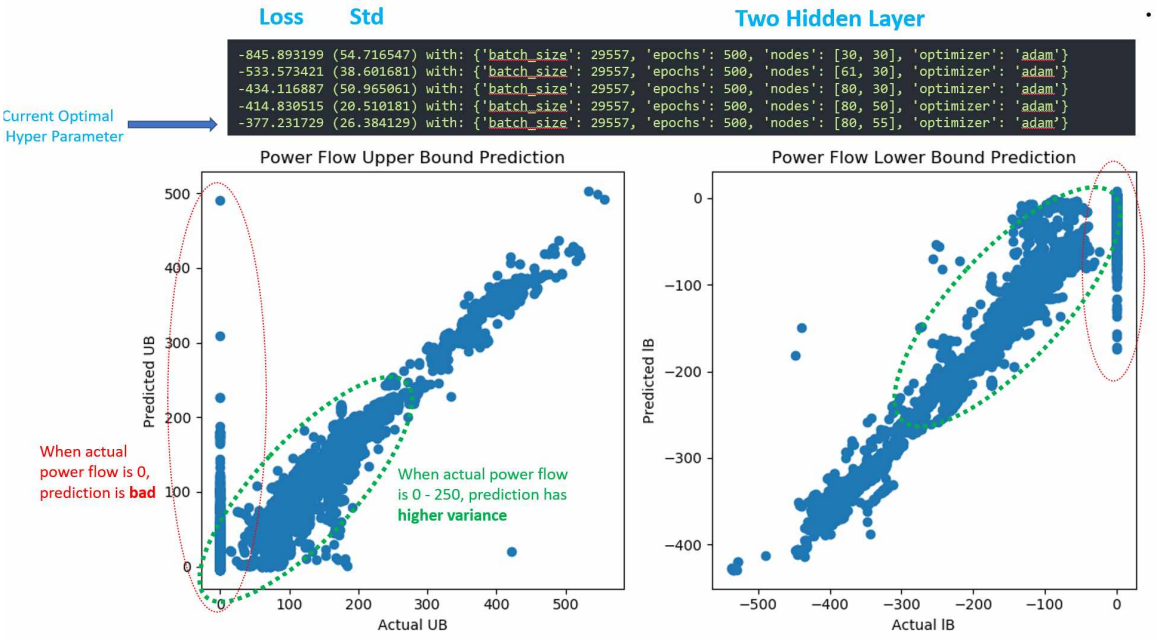

Figure 6.6: DNN on IEEE-118 Bus N-1 Contingency

Table 6.2 shows the DNN dominating constraint prediction in comparison with the deterministic methods. During training stage, we evaluated N-1 contingency power flow upper/lower bounds with 24 hours of data. The base case is the dominating constraints screening by the proposed two-step methods. "Base +" is the base case data with 10% load side uncertainty and generation deviation post SCUC calculation. The dominating constraint pool increased from 66 to 101 due to increasing uncertainty. Next, we trained the model from the previous discussion and used an isolated data set for prediction. Results in "DNN" shows that 69 out of the 101 security constraints are dominating. Lastly, we verified that 53 out of the 59 security constraints are truly dominating. With our training setup, completely calculating step 2 LPs would take over 60 minutes. However, with DNN we can reach almost identical accuracy without a fraction of a second. This is being interpreted as a scenario where we trade space for time. Offline training can be done where time is abundant, and hard drive space is unlimited. Once we need to determine the dominating constraints in real time, the trained model can quickly calculate the power

flow bounds with good accuracy.

Table 6.2: DNN Dominating Constraint Prediction

	Constraints	Step 1	Step 2
Base	69192	6896	66 (0.10%)
Base +	69192	9649	101 (0.15%)
DNN	101	NA	**69 (0.10%)**
Verify	69192	8075	53 (0.08%)

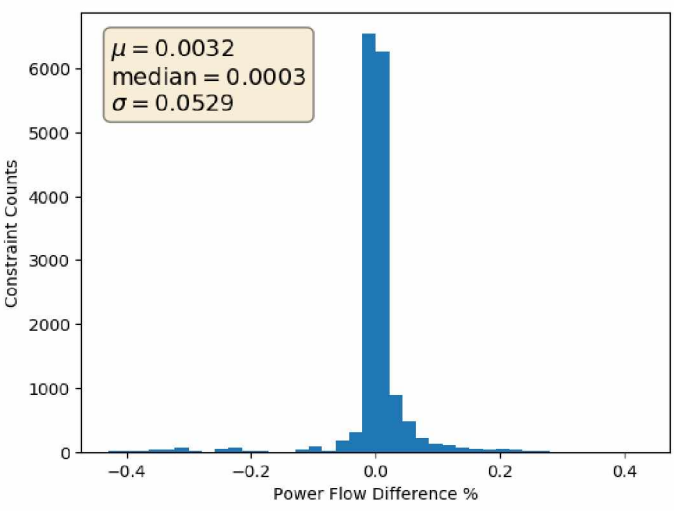

Figure 6.7: Power Flow Difference Comparison Actual vs. Predicted

Figure 6.7 shows the power flow difference between the real power flow, $(\overline{f}_l, \underline{f}_l^*)$, and DNN predicted power flow bounds, $(\overline{f}_l^{\text{dnn}}, \underline{f}_l^{\text{dnn}})$. The metric is calculated as in (6.8) and (6.9). The mean is 0.32% with a standard deviation of 5.29%. DNN predicted power flow bounds are very accurate in this case study.

$$\%\Delta f = \frac{\overline{f}_l^* - \overline{f}_l^{\text{dnn}}}{\overline{f}_l^*} * 100\% \tag{6.8}$$

$$\%\Delta f = \frac{\underline{f}_l^* - \underline{f}_l^{\text{dnn}}}{\underline{f}_l^*} * 100\% \tag{6.9}$$

6.3 Classification Dominating Constraints

Classification algorithm is well known as supervised learning technical in machine learning. In this section, we apply classical classification algorithms [111] to predict whether a constraint is dominating and training input is the same as DNN. Algorithms utilized are logistic regression, KNN, SVM, kernel SVC, naive Bayes, decision tree and random forest. Following is the standard definition of measurement matrix, we also use the confusion matrix to quantify our results, table 6.3.

- True Positive (TP): Observation is positive, and is predicted to be positive.
- False Negative (FN): Observation is positive, but is predicted negative.
- True Negative (TN): Observation is negative, and is predicted to be negative.
- False Positive (FP): Observation is negative, but is predicted positive.

Table 6.3: Confusion Metric

	Predicted 0	Predicted 1
Actual 0	True Negative	False Positive
Actual 1	False Negative	True Positive

Accuracy, recall, precision and F-measure are calculated as follows. The metric that caused most concern is the False Negative results. False negative measurements are the actual dominating constraints being indicated as redundant. Typically, redundant constraints would only cause the SCUC to take longer to solve, but missing dominating constraints will alter the optimal solution, insufficient dispatch may occur.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{6.10}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6.11}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6.12}$$

$$\text{F - measure} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} \tag{6.13}$$

Table 6.4 shows the machine learning results with various methods. From the data, our biggest challenge is the imbalanced data set, 5% of the constraints are dominating and 95% are redundant. Training with imbalanced data set may cause the algorithm to be over optimistic, and even with high accuracy, the application is insignificant due to the majority of the result to be indicated as redundant. Typically, undersampling, oversampling and SMOTE are implemented to tackle the problem. We used the standard Python package = Easyensemble to under sample the data set, i.e. only taking 10% of the sample (5% true and 5% false) in each training data set, keep on shuffling in the false data. Kernel SVM showed promise.

Table 6.4: Classification Result

Logistic Regression		KNN		SVM		Kernel SVC	
36602	13	36490	125	36600	15	36585	30
291	41	296	36	296	36	263	69

Naive Bayes		Decision Tree		Random Forest	
33075	3529	36491	124	36467	148
214	118	172	160	155	177

Table 6.5 shows the results from undersampled kernel SVM. During training with 10% of data set, only 1 constraint was false negative. When using the entire data set for prediction, we were not only able to capture 2 dominating constraints. Kernel SVM is the best classification algorithm we found to produce the lowest false negative, while having an acceptable accuracy rate. The downside is the recall value is high, almost 100%, and precision is low, around 25%. This is due to the 4868 redundant constraints we were not able to detect. Overall, classifying dominating constraint

is a trade-off between whether we want few missing dominating constraints or less redundant constraints.

Table 6.5: Undersampled Kernel SVM Result

Training Set		Entire Set	
329	9	178216	4868
1	321	2	1646

6.4 Clustering Experiment Results

Different line outage contingencies may have similar impact on the sensitivity factors on the same branch. We ran the unsupervised learning, i.e. clustering, on IEEE-118 bus system with 186 branches. The goal is to reduce the number of contingencies needs to be considered in congestion forecasting tool. In this experiment, for each branch, input X = [# contingencies, # bus] = [1 + 185, 118]. Note that each branch has 186 data entries, and each has 118 features. Perform clustering on each branch (186 times), clustering algorithm is kMeans. Measurement of distance is Within Cluster Sum of Squares (wcss) or Sum of Squared Error (distortion). Use elbow method to find an optimal k (number of clusters).

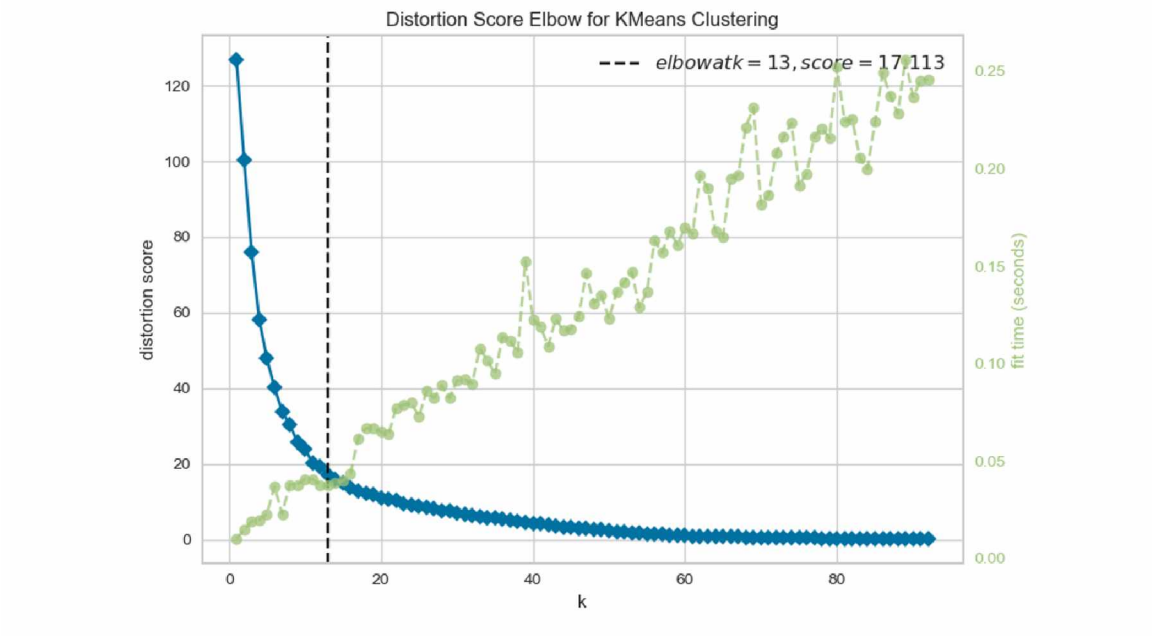

Figure 6.8: IEEE-118 Bus N-1 Contingency Clustering

From figure 6.8, we can observe that 9 branches have a sensitivity factor of 0 with over all contingencies, maybe due to islanding, which cannot be clustered. Out of 177 branches that can be clustered, the majority of the branch SF can be clustered using 10 to 15 clusters. In other words, 186 contingencies can be reduced into 10 to 15 contingencies. The elbow method shows that 13 cluster is the optimal number of contingencies that can be condensed into.

CHAPTER VII

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In this research, we propose to build a watchlist of potentially binding security constraints to tighten the SCUC formulation, which helps to improve solution quality and shorten solution time. We propose a two-step approach as a screening process to identify and eliminate the non-binding security constraints in the SCUC formulation.

In order to give MIP model a warm start, power injection uncertainty and initial commitment are also incorporated. Thus, potentially binding constraints are marked. Potentially non-binding constraints are set as lazy in the MIP solver, the size of the problem is further reduced. We also introduced software acceleration techniques to improve performance of the screening process. Data driven security constraint screening process is tested on IEEE-118 bus system and MISO test cases, which proves to be effective. Our work has been published in [116, 117].

This research also presents a data driven approach to forecast congestion patterns in the Real Time Market. MISO is interested in congestion forecasting because constraints that are missed in the Day Ahead but bind in Real Time could increase congestion funding concerns and result in a suboptimal unit commitment solution. The ability to forecast congestion is an important need for forward processes, to ensure market alignment and reliability.

The approach comprises two portions: a deterministic portion that leverages previous research done on the security constraint screening; followed by a machine

learning portion that uses well known algorithms to solve two problems – a binary classification problem to predict which constraints will bind, and a regression problem to predict the bounds of the constraints.

The deterministic portion leverages the High-Performance Power Grid Optimization (HIPPO) tool, which was developed by MISO research and development team (along with partners). The objective is to provide the operators with a list of potentially binding constraints. Along with this list, the upper and lower bounds of potential flow on the constraints is provided, which could be used as the training input to the machine learning portion. The uncertainty due to generators not following dispatch, and renewable & load forecast errors, is considered.

The machine learning approach was tested on an IEEE-118 bus system. These preliminary results show that the basic concept of utilizing machine learning models to predict binding constraints has some promise. Further studies should be conducted using real MISO system data, in order to fully evaluate the approach. The potential benefit is to provide the Day Ahead operators with a tool for supporting decision-making regarding modeling of constraints.

7.2 Future Work

The proposed two-step security constraint screening method can significantly shorten optimization time of DAM SCUC. In order words, it helps to increase social welfare. A variety of applications can be derived from it, and the challenges need to be addressed.

1. The proposed offline-online method can only be applied when there is no network topology change during online stage. The dual variables produced in offline stage are not valid for online calculation, if the LP model changes, i.e. different network topology. Traditionally, network operator uses Line Outage Distribution Factors (LODF) to estimate the power flow impact on monitored

lines due to line outage. It is worthwhile to investigate the impact.

2. Virtual participants are financial players in DAM, total bid and offer quantity accounts for about 45% of total market capacity. The virtual power cleared in DAM, is a financial tool that helps breach the gap between DAM and RTM. Even though the virtual power would create physical congestions, the price signal, when created, can help us estimate RTM congestion status. Heuristic methods can be applied to estimate the virtual power signal, and it can lead to a much narrower net power injection range, which would further eliminate redundant security constraints.

3. DAM is a financial market, and RTM is a physical market. Real-time grid security is a crucial component in daily power system operation. We have been working on the DAM so far. We can extend the research to estimate real-time congestion forecasting/management by estimating the difference between the two markets from historical data.Creating a watchlist of security constraints for a system operator to monitor in real-time would yield a short but creditable potential congestion.

7.2.1 Verify Assumptions

We made few assumptions that needs to be verified with regard to the divergence between DAM and RTM.

1. Generator deviation - We assume that generator deviation can be quantified or constrained by ramp rate, or standard deviation. And the deviation can capture the shift in power injection between the two markets. The assumption needs to be verified in perspective of nodal injection error and optimal solution shift.

2. Deviation distribution – Assuming that the distributions are uniform or normal, we used the sampling method to simulate real-time injection. Does the real-time

deviation follow this behavior?

3. Real-time topology - we assume that real time topology is one of the credible contingencies in SFT. The topology shift needs to be checked amongst, DAM, RTM and credible contingencies included in SFT.

7.2.2 Challenges in Deterministic Approach (Power Flow Analysis)

Majority of the work done in deterministic approach is purely based in the perspective of DAM. As shown in Figure 5.8, the question becomes whether the newly drawn circle (deviated feasible solution set) covers the real time solution.

1. Quantify the gap between DAM and RTM in terms of asset deviation (instead of financial means) – in terms of power injection, sensitivity factor, power flows (direction) and objective value.

2. Identify congestion driver – deviation of generator, load, virtual or renewable?

3. Reduced calculation time – the power flow analysis takes a minute to complete, and computation efficiency needs to be addressed.

7.2.3 Challenges in Heuristic Approach (Machine Learning)

1. Sensitivity factor variation - Line outages create topology change; sensitivity factor is the direct measurement of topology change in a DC network. Credible contingencies that contain all the line outage definitions are a part of input to the SFT module. We have observed that majority of line outages only affect a small portion of the entire network, and some contingencies have similar effect on the topology change. It is possible to group these contingencies together to reduce the amount of "duplicated" security constraints. If given the entire set

of contingencies $(n-k)$, we can perform statistical analysis on the sensitivity factors, and understand how much they vary and the impact on the power flow bounds. The current study created the mapping/transfer function between power flow bounds to power injection bounds, i.e. $(\overline{f}, \underline{f}) = \mathcal{G}(\overline{P}^{\text{inj}}, \underline{P}^{\text{inj}})$. If we also include sensitivity factor, the function becomes $(\overline{f}, \underline{f}) = \mathcal{G}(\overline{P}^{\text{inj}}, \underline{P}^{\text{inj}}, \overline{\Gamma}, \underline{\Gamma})$.

2. Metric that evaluate the gap between DAM and RTM - Future study is required to quantify the gap between DAM and RTM. Quantifying the gap is one crucial question. What metric should we use? Traditionally, we use money and asset deviation as measurement. However, this method is flawed in terms of capturing the congestion drivers, i.e. it may show correlation but very hard to deduce the congestion cause.

3. Feature selection - Currently, we are using sensitivity factor, system load and topology information to predict the power flow bounds. When more features are introduced, key features need to be identified through feature engineering or PCA.

4. Model precision - Power system (power flow) behaviors vary hourly and seasonally. Specific models should be trained to suit the situation.

7.2.4 Alternate Approaches for the Deterministic Method

1. MISO enforces a shift factor cutoff (1.5%). However, the flow in such instances still contributes to the SE flow. Thus, we could use the Hippo SFT without the cutoff, then we can evaluate the impact on constraints. We could consider the DA MTLF case (i.e. without virtuals, use the forecasted load for next day).

2. Another approach could be to use the DA PROBE simulation tool. This can evaluate many more contingencies than our market engine. We could run DA

PROBE against the DA MTLF case. The advantage is that the inputs are in CSV format, and thus are relatively easy to change.

BIBLIOGRAPHY

[1] U.S. Energy Information Administration, "Electric Power Annual 2018," p. 239. [Online]. Available: https://www.eia.gov/electricity/annual/pdf/epa.pdf

[2] H. L. Willis and L. Philipson, *Understanding Electric Utilities and De-Regulation*, 2nd ed. Boca Raton: CRC Press, Jan. 2017.

[3] M. Amin and J. Stringer, "The Electric Power Grid: Today and Tomorrow," *MRS Bulletin*, vol. 33, no. 4, pp. 399–407, Apr. 2008, publisher: Cambridge University Press. [Online]. Available: https://www.cambridge.org/core/journals/mrs-bulletin/article/electric-power-grid-today-and-tomorrow/D75D0366741B4F24D3DD96F3C7122A6D

[4] M. Delmas and Y. Tokat, "Deregulation, governance structures, and efficiency: the U.S. electric utility sector," *Strategic Management Journal*, vol. 26, no. 5, pp. 441–460, 2005, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/smj.456. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/smj.456

[5] A. Rahimi and A. Sheffrin, "Effective market monitoring in deregulated electricity markets," *IEEE Transactions on Power Systems*, vol. 18, no. 2, pp. 486–493, May 2003, conference Name: IEEE Transactions on Power Systems.

[6] H. Singh and A. Papalexopoulos, "Competitive procurement of ancillary services by an independent system operator," *IEEE Transactions on Power Sys-*

tems, vol. 14, no. 2, pp. 498–504, May 1999, conference Name: IEEE Transactions on Power Systems.

[7] W. Hogan, "INDEPENDENT SYSTEM OPERATOR : PRICING AND FLEXIBILITY IN A COMPETITIVE ELECTRICITY MARKET," 1998. [Online]. Available: https://www.semanticscholar.org/paper/INDEPENDENT-SYSTEM-OPERATOR-%3A-PRICING-AND-IN-A-Hogan/a332ebda2c4e2dfa49ec278424da99ec019e2001

[8] M. C. Caramanis and J. M. Foster, "Coupling of day ahead and real-time power markets for energy and reserves incorporating local distribution network costs and congestion," in *2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sep. 2010, pp. 42–49.

[9] F. Wang, X. Ge, P. Yang, K. Li, Z. Mi, P. Siano, and N. Duić, "Day-ahead optimal bidding and scheduling strategies for DER aggregator considering responsive uncertainty under real-time pricing," *Energy*, vol. 213, p. 118765, Dec. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220318727

[10] L. Wu, M. Shahidehpour, and T. Li, "Stochastic Security-Constrained Unit Commitment," *IEEE Transactions on Power Systems*, vol. 22, no. 2, pp. 800–811, May 2007, conference Name: IEEE Transactions on Power Systems.

[11] Y. Fu, M. Shahidehpour, and Z. Li, "Security-constrained unit commitment with AC constraints," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 1001–1013, May 2005, conference Name: IEEE Transactions on Power Systems.

[12] J. D. Guy, "Security Constrained Unit Commitment," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-90, no. 3, pp. 1385–1390, May 1971, conference Name: IEEE Transactions on Power Apparatus and Systems.

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

[14] 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, conference Name: IEEE Transactions on Power Systems.

[15] C. Liu, M. Shahidehpour, Y. Fu, and Z. Li, "Security-Constrained Unit Commitment With Natural Gas Transmission Constraints," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1523–1536, Aug. 2009, conference Name: IEEE Transactions on Power Systems.

[16] A. Khodaei and M. Shahidehpour, "Transmission Switching in Security-Constrained Unit Commitment," *IEEE Transactions on Power Systems*, vol. 25, no. 4, pp. 1937–1945, Nov. 2010, conference Name: IEEE Transactions on Power Systems.

[17] Y. Fu, M. Shahidehpour, and Z. Li, "AC contingency dispatch based on security-constrained unit commitment," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 897–908, May 2006, conference Name: IEEE Transactions on Power Systems.

[18] X. Sun, P. B. Luh, M. A. Bragin, Y. Chen, J. Wan, and F. Wang, "A Novel Decomposition and Coordination Approach for Large Day-Ahead Unit Commitment With Combined Cycle Units," *IEEE Transactions on Power Systems*, vol. 33, no. 5, pp. 5297–5308, Sep. 2018, conference Name: IEEE Transactions on Power Systems.

[19] X. Guan, Q. Zhai, and A. Papalexopoulos, "Optimization based methods for unit commitment: Lagrangian relaxation versus general mixed integer programming," in *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, vol. 2, Jul. 2003, pp. 1095–1100 Vol. 2.

[20] M. Shahidehpour, H. Yamin, and Z. Li, *Market Operations in Electric Power Systems: Forecasting, Scheduling, and Risk Management.* John Wiley & Sons, May 2003.

[21] A. J. Wood, B. F. Wollenberg, and G. B. Sheble, *Power Generation, Operation and Control*, 3rd ed. Wiley, 2013.

[22] D. Streiffert, R. Philbrick, and A. Ott, "A mixed integer programming solution for market clearing and reliability analysis," in *IEEE Power Engineering Society General Meeting, 2005*, Jun. 2005, pp. 2724–2731 Vol. 3, iSSN: 1932-5517.

[23] Y. Fu, Z. Li, and L. Wu, "Modeling and Solution of the Large-Scale Security-Constrained Unit Commitment," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 3524–3533, Nov. 2013, conference Name: IEEE Transactions on Power Systems.

[24] ——, "Modeling and Solution of the Large-Scale Security-Constrained Unit Commitment," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 3524–3533, Nov. 2013.

[25] M. A. Mirzaei, A. S. Yazdankhah, B. Mohammadi-Ivatloo, M. Marzband, M. Shafie-khah, and J. P. S. Catalão, "Stochastic network-constrained co-optimization of energy and reserve products in renewable energy integrated power and gas networks with energy storage system," *Journal of Cleaner Production*, vol. 223, pp. 747–758, Jun. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0959652619307048

[26] A. Khodaei and M. Shahidehpour, "Microgrid-Based Co-Optimization of Generation and Transmission Planning in Power Systems," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1582–1590, May 2013, conference Name: IEEE Transactions on Power Systems.

[27] C. He, L. Wu, T. Liu, and M. Shahidehpour, "Robust Co-Optimization Scheduling of Electricity and Natural Gas Systems via ADMM," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 2, pp. 658–670, Apr. 2017, conference Name: IEEE Transactions on Sustainable Energy.

[28] X. Zhang, M. Shahidehpour, A. S. Alabdulwahab, and A. Abusorrah, "Security-Constrained Co-Optimization Planning of Electricity and Natural Gas Transportation Infrastructures," *IEEE Transactions on Power Systems*, vol. 30, no. 6, pp. 2984–2993, Nov. 2015, conference Name: IEEE Transactions on Power Systems.

[29] K. W. Hedman, M. C. Ferris, R. P. O'Neill, E. B. Fisher, and S. S. Oren, "Co-Optimization of Generation Unit Commitment and Transmission Switching With N-1 Reliability," *IEEE Transactions on Power Systems*, vol. 25, no. 2, pp. 1052–1063, May 2010, conference Name: IEEE Transactions on Power Systems.

[30] W. Wei, F. Liu, and S. Mei, "Distributionally Robust Co-Optimization of Energy and Reserve Dispatch," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 1, pp. 289–300, Jan. 2016, conference Name: IEEE Transactions on Sustainable Energy.

[31] J. Zhu and K. Cheung, "Flexible simultaneous feasibility test in energy market," in *IEEE PES General Meeting*, Jul. 2010, pp. 1–7, iSSN: 1944-9925.

[32] R. Rangarajan and Z. Wang, "An improved simultaneous feasibility test to alleviate revenue inadequacy in FTR markets," in *2014 North American Power Symposium (NAPS)*, Sep. 2014, pp. 1–6.

[33] V. Sarkar and S. Khaparde, "A robust mathematical framework for managing simultaneous feasibility condition in financial transmission rights auction," in *2006 IEEE Power Engineering Society General Meeting.* Montreal, Que., Canada: IEEE, 2006, p. 6 pp. [Online]. Available: http://ieeexplore.ieee.org/document/1709067/

[34] Y. Chen, A. Casto, F. Wang, Q. Wang, X. Wang, and J. Wan, "Improving Large Scale Day-Ahead Security Constrained Unit Commitment Performance," *IEEE Transactions on Power Systems*, vol. 31, no. 6, pp. 4732–4743, Nov. 2016.

[35] J. Zhu, *Optimization of power system operation*, second edition ed., ser. IEEE Press series on power engineering. Hoboken, New Jersey : IEEE Press: Wiley ; Piscataway, NJ, 2015.

[36] H. Chen, Ed., *Power grid operation in a market environment: economic efficiency and risk mitigation*, ser. IEEE Press series on power engineering. Piscataway, NJ: IEEE Press/Wiley, 2017, oCLC: ocn933721938.

[37] D. A. Tejada-Arango, P. Sanchez-Martin, and A. Ramos, "Security Constrained Unit Commitment Using Line Outage Distribution Factors," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 329–337, Jan. 2018. [Online]. Available: http://ieeexplore.ieee.org/document/7886335/

[38] A. S. Xavier, F. Qiu, F. Wang, and P. R. Thimmapuram, "Transmission Constraint Filtering in Large-Scale Security-Constrained Unit Commitment," *IEEE Transactions on Power Systems*, vol. 34, no. 3, pp. 2457–2460, 2019.

[39] S. Cvijic and J. Xiong, "Security constrained unit commitment and economic dispatch through benders decomposition: A comparative study," in *2011 IEEE Power and Energy Society General Meeting*, Jul. 2011, pp. 1–8, iSSN: 1944-9925.

[40] A. Salloum, Y. M. Al-Abdullah, V. Vittal, and K. W. Hedman, "Impacts of Constraint Relaxations on Power System Operational Security," *IEEE Power and Energy Technology Systems Journal*, vol. 3, no. 3, pp. 99–108, Sep. 2016, conference Name: IEEE Power and Energy Technology Systems Journal.

[41] V. Van Acker, P. Shamsollahi, C. Cathey, R. Dillon, and D. Gray, "Impact of MW dependent constraint violation penalties on market results at southwest power pool," in *PES T D 2012*, May 2012, pp. 1–8, iSSN: 2160-8563.

[42] Y. M. Al-Abdullah, M. Abdi-Khorsand, and K. W. Hedman, "The Role of Out-of-Market Corrections in Day-Ahead Scheduling," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 1937–1946, Jul. 2015, conference Name: IEEE Transactions on Power Systems.

[43] J. B. Cardell, "Marginal Loss Pricing for Hours With Transmission Congestion," *IEEE Transactions on Power Systems*, vol. 22, no. 4, pp. 1466–1474, Nov. 2007, conference Name: IEEE Transactions on Power Systems.

[44] X. Feng, J. Pan, LeTang, H. Chao, and J. Yang, "Economic evaluation of transmission congestion relief based on power market simulations," in *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, vol. 2, Jul. 2003, pp. 1018–1024 Vol. 2.

[45] F. Li and R. Bo, "Congestion and Price Prediction Under Load Variation," *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 911–922, May 2009, conference Name: IEEE Transactions on Power Systems.

[46] Q. Wang, G. Zhang, J. D. McCalley, T. Zheng, and E. Litvinov, "Risk-Based Locational Marginal Pricing and Congestion Management," *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2518–2528, Sep. 2014, conference Name: IEEE Transactions on Power Systems.

[47] H. Chao, F. Li, L. Trinh, J. Pan, M. Gopinathan, and D. Pillo, "Market based transmission planning considering reliability and economic performances," in *2004 International Conference on Probabilistic Methods Applied to Power Systems*, Sep. 2004, pp. 557–562.

[48] A. Kumar, S. C. Srivastava, and S. N. Singh, "Congestion management in competitive power market: A bibliographical survey," *Electric Power Systems Research*, vol. 76, no. 1, pp. 153–164, Sep. 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779605001604

[49] A. Kumar, S. Srivastava, and S. Singh, "A zonal congestion management approach using real and reactive power rescheduling," *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 554–562, Feb. 2004, conference Name: IEEE Transactions on Power Systems.

[50] Jesse Schneider, Grid Strategies LLC, "Transmission congestion costs in the U.S. RTOs," 2019 (accessed August 10, 2020). [Online]. Available: https://watt-transmission.org/2019/09/17/transmission-congestion-costs-in-the-u-s-rtos/

[51] A. J. Ardakani and F. Bouffard, "Identification of Umbrella Constraints in DC-Based Security-Constrained Optimal Power Flow," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 3924–3934, Nov. 2013.

[52] ——, "Acceleration of Umbrella Constraint Discovery in Generation Scheduling Problems," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 2100–2109, Jul. 2015.

[53] Q. Zhai, X. Guan, J. Cheng, and H. Wu, "Fast Identification of Inactive Security Constraints in SCUC Problems," *IEEE Transactions on Power Systems*, vol. 25, no. 4, pp. 1946–1954, Nov. 2010.

[54] H. Ye, J. Wang, and Z. Li, "MIP Reformulation for Max-Min Problems in Two-Stage Robust SCUC," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1237–1247, Mar. 2017.

[55] H. Ye and Z. Li, "Necessary Conditions of Line Congestions in Uncertainty Accommodation," *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 4165–4166, Sep. 2016.

[56] R. Madani, J. Lavaei, and R. Baldick, "Constraint Screening for Security Analysis of Power Networks," *IEEE Transactions on Power Systems*, vol. 32, no. 3, pp. 1828–1838, May 2017.

[57] Y. Chen, "MISO R&D on improving the efficiency of market clearing software." [Online]. Available: https://www.ferc.gov/CalendarFiles/20190625100440-3%20-%20T1_Chen_wide.pdf

[58] Y. Chen, F. Wang, Y. Ma, and Y. Yao, "A Distributed Framework for Solving and Benchmarking Security Constrained Unit Commitment with Warm Start," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 711–720, 2020.

[59] R. Fang and A. David, "Transmission congestion management in an electricity market," *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 877–883, Aug. 1999, conference Name: IEEE Transactions on Power Systems.

[60] A. Pillay, S. Prabhakar Karthikeyan, and D. P. Kothari, "Congestion management in power systems – A review," *International Journal of Electrical Power & Energy Systems*, vol. 70, pp. 83–90, Sep. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061515000411

[61] V. K. Prajapati and V. Mahajan, "Reliability assessment and congestion management of power system with energy storage system and uncertain renewable resources," *Energy*, vol. 215, p. 119134, Jan. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220322416

[62] M. Reza Salehizadeh, A. Rahimi-Kian, and M. Oloomi-Buygi, "Security-based multi-objective congestion management for emission reduction in power system," *International Journal of Electrical Power & Energy Systems*, vol. 65, pp. 124–135, Feb. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061514005821

[63] J. Hazra and A. K. Sinha, "Congestion Management Using Multiobjective Particle Swarm Optimization," *IEEE Transactions on Power Systems*, vol. 22, no. 4, pp. 1726–1734, Nov. 2007, conference Name: IEEE Transactions on Power Systems.

[64] M. Esmaili, N. Amjady, and H. A. Shayanfar, "Stochastic congestion management in power markets using efficient scenario approaches," *Energy Conversion and Management*, vol. 51, no. 11, pp. 2285–2293, Nov. 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S019689041000138X

[65] D. Gan and D. Bourcier, "Locational market power screening and congestion management: experience and suggestions," *IEEE Transactions on Power Systems*, vol. 17, no. 1, pp. 180–185, Feb. 2002, conference Name: IEEE Transactions on Power Systems.

[66] E. Bompard, P. Correia, G. Gross, and M. Amelin, "Congestion-management schemes: a comparative analysis under a unified framework," *IEEE Transactions on Power Systems*, vol. 18, no. 1, pp. 346–352, Feb. 2003, conference Name: IEEE Transactions on Power Systems.

[67] E. Shayesteh, M. Parsa Moghaddam, S. Taherynejhad, and M. K. Sheikh-EL-Eslami, "Congestion Management using Demand Response programs in power market," in *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, Jul. 2008, pp. 1–8, iSSN: 1932-5517.

[68] L. Yao, P. Cartwright, L. Schmitt, and X.-P. Zhang, "Congestion Management of Transmission Systems Using FACTS," in *2005 IEEE/PES Transmission Distribution Conference Exposition: Asia and Pacific*, Aug. 2005, pp. 1–5, iSSN: 2160-8644.

[69] Q. Zhou, L. Tesfatsion, and C.-C. Liu, "Short-Term Congestion Forecasting in Wholesale Power Markets," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 2185–2196, Nov. 2011, conference Name: IEEE Transactions on Power Systems.

[70] L. Min, S. T. Lee, P. Zhang, V. Rose, and J. Cole, "Short-term probabilistic transmission congestion forecasting," in *2008 Third International Conference on Electric Utility Deregulation and Restructuring and Power Technologies*, Apr. 2008, pp. 764–770.

[71] Y. Li and N. Yu, "Learning to Arbitrage Congestion in Electricity Market with Virtual Bids," in *2021 IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*, Oct. 2021, pp. 01–06.

[72] G. Li, C.-c. Liu, and H. Salazar, "Forecasting Transmission Congestion Using Day- Ahead Shadow Prices," in *2006 IEEE PES Power Systems Conference and Exposition*, Oct. 2006, pp. 1705–1709.

[73] A. Loland, E. Ferkingstad, and M. Wilhelmsen, "Forecasting transmission congestion," *The Journal of Energy Markets*, vol. 5, no. 3, pp. 65–83, Sep. 2012. [Online]. Available: http://www.risk.net/journal-of-energy-markets/technical-paper/2197368/forecasting-transmission-congestion

[74] Y. Liu and F. F. Wu, "Impacts of Network Constraints on Electricity Market Equilibrium," *IEEE Transactions on Power Systems*, vol. 22, no. 1, pp. 126–135, Feb. 2007, conference Name: IEEE Transactions on Power Systems.

[75] K. Zheng, Q. Chen, Y. Wang, C. Kang, and L. Xie, "Unsupervised Congestion Status Identification Using LMP Data," *IEEE Transactions on Smart Grid*, vol. 12, no. 1, pp. 726–736, Jan. 2021, conference Name: IEEE Transactions on Smart Grid.

[76] M. B. Nappu, A. Arief, T. K. Saha, and R. C. Bansal, "Investigation of LMP forecasting for congested power systems," in *2012 22nd Australasian Universities Power Engineering Conference (AUPEC)*, Sep. 2012, pp. 1–6.

[77] S. Raikar and M. Ilic, "Assessment of transmission congestion for major electricity markets in the US," in *2001 Power Engineering Society Summer Meeting. Conference Proceedings (Cat. No.01CH37262)*, vol. 2, Jul. 2001, pp. 1152–1156 vol.2.

[78] K. Singh, N. P. Padhy, and J. Sharma, "Influence of Price Responsive Demand Shifting Bidding on Congestion and LMP in Pool-Based Day-Ahead Electricity Markets," *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 886–896, May 2011, conference Name: IEEE Transactions on Power Systems.

[79] X. Yan and N. A. Chowdhury, "Mid-term electricity market clearing price forecasting: A multiple SVM approach," *International Journal of Electrical Power & Energy Systems*, vol. 58, pp. 206–214, Jun. 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061514000362

[80] J. Zarnikau, C. H. Tsai, and C. K. Woo, "Determinants of the wholesale prices of energy and ancillary services in the U.S. Midcontinent electricity market," *Energy*, vol. 195, p. 117051, Mar. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544220301584

[81] L. Yin, Q. Gao, L. Zhao, B. Zhang, T. Wang, S. Li, and H. Liu, "A review of machine learning for new generation smart dispatch in power systems," *Engineering Applications of Artificial Intelligence*, vol. 88, p. 103372, Feb. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0952197619302982

[82] C. Rudin, D. Waltz, R. N. Anderson, A. Boulanger, A. Salleb-Aouissi, M. Chow, H. Dutta, P. N. Gross, B. Huang, S. Ierome, D. F. Isaac, A. Kressner, R. J. Passonneau, A. Radeva, and L. Wu, "Machine Learning for the New York City Power Grid," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 2, pp. 328–345, Feb. 2012. [Online]. Available: http://ieeexplore.ieee.org/document/5770269/

[83] J.-P. Lai, Y.-M. Chang, C.-H. Chen, and P.-F. Pai, "A Survey of Machine Learning Models in Renewable Energy Predictions," *Applied Sciences*, vol. 10, no. 17, p. 5975, Jan. 2020, number: 17 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2076-3417/10/17/5975

[84] O. A. Alimi, K. Ouahada, and A. M. Abu-Mahfouz, "A Review of Machine Learning Approaches to Power System Security and Stability," *IEEE Access*, vol. 8, pp. 113 512–113 531, 2020, conference Name: IEEE Access.

[85] M. Sarhani and A. E. Afia, "Electric load forecasting using hybrid machine learning approach incorporating feature selection," p. 7, May 2015.

[86] X. Chen, Z. Y. Dong, K. Meng, Y. Xu, K. P. Wong, and H. W. Ngan, "Electricity Price Forecasting With Extreme Learning Machine and Bootstrapping," *IEEE Transactions on Power Systems*, vol. 27, no. 4, pp. 2055–2062, Nov. 2012. [Online]. Available: http://ieeexplore.ieee.org/document/6184354/

[87] A. Mosavi, M. Salimi, S. Faizollahzadeh Ardabili, T. Rabczuk, S. Shamshirband, and A. R. Varkonyi-Koczy, "State of the Art of Machine Learning Models in Energy Systems, a Systematic Review," *Energies*, vol. 12, no. 7, p. 1301, Jan. 2019, number: 7 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1996-1073/12/7/1301

[88] G. Dalal, E. Gilboa, S. Mannor, and L. Wehenkel, "Chance-Constrained Outage Scheduling Using a Machine Learning Proxy," *IEEE Transactions on Power Systems*, vol. 34, no. 4, pp. 2528–2540, Jul. 2019, conference Name: IEEE Transactions on Power Systems.

[89] A. S. Khwaja, A. Anpalagan, M. Naeem, and B. Venkatesh, "Joint bagged-boosted artificial neural networks: Using ensemble machine learning to improve short-term electricity load forecasting," *Electric Power Systems Research*, vol. 179, p. 106080, Feb. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779619303992

[90] X. Lei, Z. Yang, J. Yu, J. Zhao, Q. Gao, and H. Yu, "Data-Driven Optimal Power Flow: A Physics-Informed Machine Learning Approach,"

IEEE Transactions on Power Systems, vol. 36, no. 1, pp. 346–354, Jan. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9115822/

[91] R. Eskandarpour and A. Khodaei, "Machine Learning Based Power Grid Outage Prediction in Response to Extreme Events," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3315–3316, Jul. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7752978/

[92] L. Duchesne, E. Karangelos, and L. Wehenkel, "Machine learning of real-time power systems reliability management response," in *2017 IEEE Manchester PowerTech*. Manchester, United Kingdom: IEEE, Jun. 2017, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/7980927/

[93] S. M. Miraftabzadeh, F. Foiadelli, M. Longo, and M. Pasetti, "A Survey of Machine Learning Applications for Power System Analytics," in *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*. Genova, Italy: IEEE, Jun. 2019, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/document/8783340/

[94] M. Ebeed, S. Kamel, and F. Jurado, "Optimal Power Flow Using Recent Optimization Techniques," in *Classical and Recent Aspects of Power System Optimization*. Elsevier, 2018, pp. 157–183. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/B9780128124413000070

[95] F. Hasan, A. Kargarian, and A. Mohammadi, "A Survey on Applications of Machine Learning for Optimal Power Flow," in *2020 IEEE Texas Power and Energy Conference (TPEC)*. College Station, TX, USA: IEEE, Feb. 2020, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/9042547/

[96] N. Guha, Z. Wang, M. Wytock, and A. Majumdar, "Machine Learning for AC Optimal Power Flow," *arXiv:1910.08842 [cs, eess, stat]*, Oct. 2019, arXiv: 1910.08842. [Online]. Available: http://arxiv.org/abs/1910.08842

[97] N. G. Paterakis, E. Mocanu, M. Gibescu, B. Stappers, and W. van Alst, "Deep learning versus traditional machine learning methods for aggregated energy demand prediction," in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*. Torino: IEEE, Sep. 2017, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/8260289/

[98] R. Ahmadiahangar, T. Haring, A. Rosin, T. Korotko, and J. Martins, "Residential Load Forecasting for Flexibility Prediction Using Machine Learning-Based Regression Model," in *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*. Genova, Italy: IEEE, Jun. 2019, pp. 1–4. [Online]. Available: https://ieeexplore.ieee.org/document/8783634/

[99] A. Almalaq and G. Edwards, "A Review of Deep Learning Methods Applied on Load Forecasting," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. Cancun, Mexico: IEEE, Dec. 2017, pp. 511–516. [Online]. Available: http://ieeexplore.ieee.org/document/8260682/

[100] H. Sangrody, N. Zhou, S. Tutun, B. Khorramdel, M. Motalleb, and M. Sarailoo, "Long term forecasting using machine learning methods," in *2018 IEEE Power and Energy Conference at Illinois (PECI)*. Champaign, IL, USA: IEEE, Feb. 2018, pp. 1–5. [Online]. Available: http://ieeexplore.ieee.org/document/8334980/

[101] M. Negnevitsky, P. Mandal, and A. Srivastava, "Machine Learning Applications for Load, Price and Wind Power Prediction in Power Systems,"

in *2009 15th International Conference on Intelligent System Applications to Power Systems*. Curitiba: IEEE, Nov. 2009, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/document/5352820/

[102] A. Conejo and J. Aguado, "Multi-area coordinated decentralized DC optimal power flow," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1272–1278, Nov. 1998, conference Name: IEEE Transactions on Power Systems.

[103] S. Greene, I. Dobson, and F. L. Alvarado, "Sensitivity of transfer capability margins with a fast formula," *IEEE Transactions on Power Systems*, vol. 17, no. 1, pp. 34–40, 2002.

[104] T. Guler, G. Gross, and Minghai Liu, "Generalized Line Outage Distribution Factors," *IEEE Transactions on Power Systems*, vol. 22, no. 2, pp. 879–881, May 2007. [Online]. Available: http://ieeexplore.ieee.org/document/4162596/

[105] Jiachun Guo, Yong Fu, Zuyi Li, and M. Shahidehpour, "Direct Calculation of Line Outage Distribution Factors," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1633–1634, Aug. 2009. [Online]. Available: http://ieeexplore.ieee.org/document/5130240/

[106] A. D. Domínguez-García and P. W. Sauer, "Measurement-Based Estimation of Linear Sensitivity Distribution Factors and its Application in Online Analysis Tools," p. 21.

[107] J. Guo, Y. Fu, Z. Li, and M. Shahidehpour, "Direct Calculation of Line Outage Distribution Factors," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1633–1634, Aug. 2009.

[108] Y. Chen, F. Wang, J. Wan, and F. Pan, "Developing Next Generation Electricity Market Clearing Optimization Software," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*. Portland, OR: IEEE, Aug. 2018, pp. 1–5.

[109] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2020.

[110] C. Grigg, P. Wong, P. Albrecht, R. Allan, M. Bhavaraju, R. Billinton, Q. Chen, C. Fong, S. Haddad, S. Kuruganty, W. Li, R. Mukerji, D. Patton, N. Rau, D. Reppen, A. Schneider, M. Shahidehpour, and C. Singh, "The IEEE Reliability Test System-1996. A report prepared by the Reliability Test System Task Force of the Application of Probability Methods Subcommittee," *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 1010–1020, Aug. 1999, conference Name: IEEE Transactions on Power Systems.

[111] C. M. Bishop, *Pattern recognition and machine learning*, ser. Information science and statistics. New York: Springer, 2006.

[112] J. Schmidt-Hieber, "Nonparametric regression using deep neural networks with ReLU activation function," *The Annals of Statistics*, vol. 48, no. 4, pp. 1875–1897, Aug. 2020, publisher: Institute of Mathematical Statistics. [Online]. Available: https://projecteuclid.org/journals/annals-of-statistics/volume-48/issue-4/Nonparametric-regression-using-deep-neural-networks-with-ReLU-activation-function/10.1214/19-AOS1875.full

[113] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[114] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Dec. 2014. [Online]. Available: https://arxiv.org/abs/1412.6980v9

[115] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan,

F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[116] S. Zhang, H. Ye, F. Wang, Y. Chen, S. Rose, and Y. Ma, "A Data-aided Security Constraint Prescreening Technique and Application to Real-world System," in *2019 North American Power Symposium (NAPS)*, Oct. 2019, pp. 1–6.

[117] ——, "Data-Aided Offline and Online Screening for Security Constraint," *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 2614–2622, May 2021, conference Name: IEEE Transactions on Power Systems.

APPENDIX

Code Documentation

1.1 General Requirement and Setup

This section discusses the codes that were designed to run the proposed algorithms and simulations. The codes written for the two-step security constraint screening and static analysis were tested on Python 2.7/3.6. Machine learning applications were tested on Python 3.6 with Tensorflow 2.0. Use the requirement.txt and pip to install the required packages. The following packages are necessary to run the code.

- Nypmy/Pandas/Matplotlib : matrix calculation, creating figures and reports
- Numba : Cython based numerical acceleration for Numpy
- Igraph : Required for HIPPO
- Xlrt/Xlwt : Output csv/xls files
- Gurobipy (8.0.5) : Default MIP solver for HIPPO and security constraints screening package
- Keras/Scikitlearn/Tensorflow : Machine learning packages
- Imblearn : Package that performs imbalance learning sampling, i.e. oversampling, undersampling.

1.2 Security Constraint Screening and Congestion Forecasting Packages

The codes developed for the project mainly consisted in three packages, i.e. *watchlist_proto, sft* and *machine_learning_models*. The scripts that run through the entire congestion forecasting process are located in *run_example folder*. The following section discusses in detail the purpose of each script and how to implement them. Table 1.1 lists the steps, actions and top level scripts associated with security constraint screening in DAM and congestion forecasting.

1.2.1 Watchlist Screening Module

watchlist_proto – watchlist constraints screening functions. The package was originally designed for the watchlist security constraint screening project. Prior to solving day ahead SCUC cases, watchlist constraints were hand selected from a pool of highly credible contingency pool. The watchlist security constraints screening tool filters through the selected constraints and determines the list set of constraints that needs to be modeled in the day ahead SCUC case. Usually, 50% security constraints are determined to be redundant, and DAM SCUC solution time is shortened by 50% on average.

1.2.2 Enhanced SFT Module

sft - is a standard power flow violation calculation after solving DAM SCUC. Hundreds of credible contingencies, line outage/generator outage, are pre-selected. SFT module use the contingency topology to calculate the sensitivity factor matrix. SCUC resulted dispatches are fed into SFT to calculate the power flow on each constraint. Power flow that exceeds its limit will mark the constraint as potentially congested. The constraint will be added back to the SCUC model for the next MIP optimization. There are three function packages in this module,

1. **read_data.py** – read data from case file (developed by PNNL)

2. **preprocessing.py** – read addition information from case file and establish indexing for efficient numerical programming (developed by PNNL)

3. **sft_calculation.py** – calculate sensitivity factors for all credible contingencies, determine power flow upper/lower bounds given the SCUC solution, and mark the potentially binding security constraints.

Other functions include and not limited to,

nearestPD(A): for Cholesky solve, A matrix must be positive definite. However, in realty, the admittance matrix can be slightly off to a true positive definite matrix. This function rounds it up, so calculations can continue.

isPD(B): determine whether a matrix is positive definite.

read_pinj_from_file(file): read power injection from a SCUC solution file

build_pinj_interval(sft_data, pinj_data): match the power injections read from file, to the corresponding sensitivity matrix location.

sft(object): sft main function

1. **compute_base_ptdf**: compute the base case sensitivity factor matrix

2. **calculate_lo_power_flow_deviation**: calculate line outage power flow deviations. Note that this function only calculates the difference admittance matrix

3. **calculate_lo_power_flow_dev_pct**: calculate the power flow deviation as a percentage of power flow limit.

4. **build_sc_ptdf**: build the complete sensitivity matrix for all contingencies,

$$\mathbf{\Gamma} = \Gamma_{\text{base}} + \mathbf{\Delta\Gamma}^{\text{c}}, \quad c \in \mathcal{C}$$

5. **sc_prescreen**: two-step security constraint screening

6. **match_pnode_with_market**: contingency topology is different to base case topology, some pnode are not present in either/or base. This function attempt

to match the pnodes.

7. **calculate_ctg_power_deviation** implements (3.35) in a modified form, shown as the following equation. If the change of sensitivity factor from base multiplied by the power injection bounds is larger than a threshold, it is considered to be a potentially binding constraint, resulted from enhanced sft module.

$$|\mathbf{\Delta\Gamma}^{\mathrm{c}}|\cdot(\overline{P}^{\mathrm{inj}} - \underline{P}^{\mathrm{inj}}) \ \leq \ \gamma * F_l, \quad l \in \mathcal{L}$$

8. **build_amcey_matrix**: build the A, M, C, E, Y matrices for Cholesky decomposition.

9. **cholesky_solve**: use decomposition method to solve the matrix inverse

10. **solve_ctg**: calculate final values of sensitivity factor

11. **check_violation_from_data**: check power flow violation with the difference matrix calculated by the decomposition method.

12. **two_step_check_violation**: use the two-step security constraint screening method to determine power flow upper/lower bounds

13. **one_step_max_min**: use the max value of power injection if the corresponding sensitivity is a positive number, and if the sensitivity is a negative value, then use the minimum power injection value. However, if the product is a negative number, then use zero to truncate the result. The matrix calculate is very efficient.

1.2.3 Machine Learning Module

The machine learning module aims to breach the gap between day-ahead market

and real-time market congestion status. Utilizing historical congestion information, the machine learning module predicts a pool of potentially congested lines. When new information becomes available, the machine learning module will calculate the probabilities of a line is binding in real time. The machine learning model has two major functions (Process is shown in figure 1.1).

1. Classification: determine the probability of a line becomes binding in real time using, logistic regression, knn, svm, kernal SVC, naïve bayes and random forest

2. Deep neural network: using DNN to predict the upper and lower bounds of power flow of a line

- Data Parsing

  Learn_ctg_pf_ub.py: the script can calculate the power flow upper/lower bound of security constraints in IEEE-118 bus test system. It can also read previous calculated results and perform offline-online calculation comparison. Note that this script was originally written to test Step 2: Linear Programming Based Offline Power Flow Maximization and Step 2: Dual Problem Based Online Screening. Since step 2 calculates all the power flow upper/lower bounds of N-1 contingency, we used this script to generate training data set for the DNN machine learning algorithm

- Build and train model

  classify_dom.py: classification algorithm to predict dominating security constraints. Input is sensitivity factor matrix, and output is whether a constraint is dominating or not

  treat_imba: treat data imbalance problem. Only 5% of the total security constraints are dominating, using ensemble method/bootstrapping to create a balanced dataset

  try_other_methods: the best one we found is kernel SVC. If set try_other_methods

to true, the script will run logistic regression, knn, svm, naive bayes and decision tree methods

Grid_search: enumeration method to find optimal hyper perimeter.

train_ann_n1: the script trains DNN to learn power flow upper/lower bounds

GridSearch_table_plot: plot loss against hyper-parameter sets

Create_regressor: create regression model

GridSearchCV: grid search cross validation

dump_model: output model

- Testing and verification

  classify_dom.py script uses k-fold validation, see cross validation section

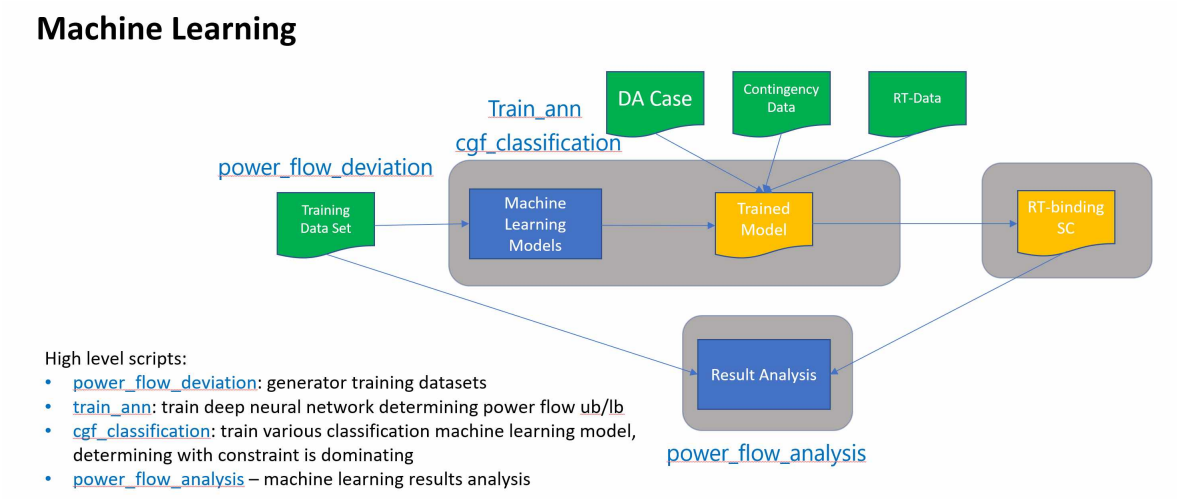

Figure 1.1: Congestion Forecasting Framework (online)

1.3 Top Level Scripts

MIP_full_scenarios.py: security constraint screening example script. The script takes the day ahead SCUC case, and filters through the watchlist security constraints and determines the least set that needs to be included in the case without jeopardizing model integrity (feasibility region remains the same). There are 12 scenarios that can be implemented individually. It is a combination of three types of algorithms, i.e.

Table 1.1: Top Level Scripts of Congestion Forecasting

Step	Action	Script
1	Build day ahead market object	MIP_full_scenarios.py
2	Run two-step security constraint screening	
3	Solve DAM SCUC	
4	Run enhanced SFT	screen_sft_offline.py
5	Run deep learning networks	train_ann1.py
6	Run Classification to identify binding constraints	classify_dom.py

- **PRES** : two-step security constraints screening
- **CI** : confidence interval of previous commitment
- **LZ** : lazy constraint setting

PRES: two-step security constraints screening should be applied right after building the market object, where watchlist security constraints are modeled. Choosing this option will implement the two-step method. See 3.2 for details.

CI: MISO studies have found that the current day commitment is very similar to previous day commitment (roughly 70-95%). Using the previous day commitment as a warm start can possibly shorten the DAM SCUC solution time. Two sets of values need to be fed into the engine, 1st is commitment information (binary variables, generator commit status), and 2nd is the confident interval of the actual dispatch. The dispatch confidence interval is derived from seasonal data sets. However, using ramp rate to constraint the generating units with high volatility is also viable.

LZ: lazy constraint settings. Not all dominating constraints (constraints that shapes the feasibility region of the solution set) are binding at the optimal solution. We cannot simply remove the constraints that we think would not bind, because the feasibility region will be altered if not all dominating constraints are present in the model. However, the lazy constraint concept in the optimization engine allows us to "sudo remove" the nonbinding constraint for faster calculation. Lazy constraints are user defined constraint that are believed to not bind at optimal solution. The solver

will optimize the problem without lazy constraints at the start. Once a solution is found, it is checked against the lazy constraints to see if any violation occurs. If so, the violated constraints are pulled into the active model, and optimized again. And iteration continues until no more violation occurs. The question is how to pick which constraints are likely to bind. We know that a relaxed MIP model takes little time to solve, even though the solution is inappropriate to what the true MIP solution is. It offers insight where the true solution may reside. We proposed two methods, integer relaxation (IR) and initial commitment (CI). Both methods require solving a relaxed version of the MIP model, and using the solution to determine whether security constraints needs to be set as lazy. Work flow and implementation are illustrated in figure 3.2. For technical details, see 3.3.

Table 1.2 lists all the combinations we applied to the experiment data, the detailed explanations are as follows,

0. base case solution. Build and optimize the DAM SCUC directly.

1. Build market model, relax integer variables, solve the LP model. Use the LP solution and determine the power injections on each node. Introduce uncertainty margin to create a power injection bounds. Determines the power flow bounds on all the security constraints (two-step), if the bounds are outside of power flow limits, it will be kept in the MIP model, otherwise, set them as lazy constraints. Optimize the MIP model with lazy constraints.

2. Build market model, relax integer variables, solve the LP model. Use the LP solution to determine the power flow on each security constraints. Introduce a threshold value, say 80%, if the power flow is greater than 80% of the power flow limit, it has the potential of becoming binding, and it will be kept untouched in the MIP model. Power flow with 80% or less of the power flow limit will be set as lazy in the MIP model. Optimize the model.

Table 1.2: Security Constraint Screening Scenario List

#	scenario	PRES	CI	LZ
0	base_noci_nolz	FALSE	FALSE	nolz
1	base_noci_pinj	FALSE	FALSE	pinj
2	base_noci_flow	FALSE	FALSE	flow
3	base_ci_nolz	FALSE	TRUE	nolz
4	base_ci_pinj	FALSE	TRUE	pinj
5	base_ci_flow	FALSE	TRUE	flow
6	pres_noci_nolz	TRUE	FALSE	nolz
7	pres_noci_pinj	TRUE	FALSE	pinj
8	pres_noci_flow	TRUE	FALSE	flow
9	pres_ci_nolz	TRUE	TRUE	nolz
10	pres_ci_pinj	TRUE	TRUE	pinj
11	pres_ci_flow	TRUE	TRUE	flow

3. Apply previous commitment information to the model, i.e. unit commitment and dispatch confidence interval. The option to set commitment is in the solution "hint". The MIP is still bounded by DAM case file input, just the initial node to be explored is set by previous commitment.

4. Combining 3 and 1. Set initial commitment before optimization, solve LP, use pinj method to set lazy constraint. Optimize MIP model.

5. Combining 3 and 2. Set initial commitment before optimization, solve LP, use flow method to set lazy constraints. Optimize MIP model.

6. Build the DAM SCUC model. Apply two-step security constraint screening method, determine the redundant constraints and remove them prior to optimizing the MIP model. This method, when compared with 0, shows how effective the two-step method is.

7. Combining 6 and 1. Prescreen and remove redundant constraints. Use "pinj" method to set constraints as lazy in the MIP model prior to optimization.

8. Combining 6 and 2. Prescreen and remove redundant constraints. Use "flow" method to set constraints as lazy in the MIP model prior to optimization.

9. Combining 6 and 3. In 6, the power injection bounds are deterministic, i.e. using the asset's physical limit. This conservative approach is relaxed in this method. Instead of using the absolute bounds of uncertainty, we can use dispatch confidence interval, the result will be a subset of 6. And the subset difference can be set as lazy constraints prior to solving the MIP model.

10. Apply two-step security constraint screening before 4.

11. Apply two-step security constraint screening before 5.

screen_sft_offline: After solving DAM SCUC, the resulted dispatch is fed into the SFT model to test credible contingency power flows. We took the same input, i.e. dispatch and credible contingency topologies and applied security constraint screening. Remaining security constraints would be a robust set of potentially congested constraints. See section VI for details. The work flow is summarized as follows,

1. Build DAM model and GUROBI MIP model, *SCUC = set_model(market)*

2. Solve MIP, *mc.optimize()*

3. Build SFT model, i.e. contingencies that were deemed to be credible *sft_test = sft(sft_data)*

4. Separate contingency sensitivities factors into base case and contingency matrix, the operation is only done on the contingency matrix to take advantage of sparse matrix calculation. *sft_test.compute_base_ptdf()*

5. Input deviation (generator, renewable, load), calculate uncertainty margin, *pinj_bounds = get_pnode_pinj_from_mc_with_dev_or_std(·).* Note that the generator deviation can take a percentage increase, i.e. say 7%, it will increase the uncertainty bound on generator dispatch to be between 93% and 107% around the SCUC solution. This bound is capped by generator limits and ramp rate. One can also use generator deviation standard deviation. The code will calculate the 95% confidence interval as the uncertainty bound. Note that we only consider normal distribution. Load deviation input is a fixed percentage and default value is 5%. Wind generation uncertainty default value is 20%. We can improve this in the future to treat it as a distribution.

6. Determine power flow bounds based on contingencies and power injection bounds. Note that, due to the large number of security constraints, we applied 3 layers of screening, described as follows,

    6.1. Step 0: coarse scan, max power injection on all the nodes with positive sensitivity factors, and minimize power injection on all the nodes with negative sensitivity factors. Calculate power flow. This step is numerical, which takes very little time to solve. However, it does not take into consideration realistic dispatch

    6.2. Step 1: use unique function on the remaining constraints from step 0. DAM will only model 1 constraint per line (this is due to the financial settling scheme). This step is done after step 0, because operating unique operation on large matrix is very expensive

    6.3. Step 2: apply uncertainty induced power injections to DAM model solu-

tion. Use the two-step security constraint screening method to determine the potential binding security constraints

6.4. Step 3: Introduce a tolerance value, which is the max MW allowed to be violated before it is deemed to be binding (see chapter IV for details.)

7. Determine/output potential binding constraints (see chapter V for details)

power_flow_deviation: apply generator deviation statistics to the security constraint screening tool, and determine the uncertainty bounds of power flows on the selected constraints. See chapter V for details.

Processes discussed in this section follows the work flow shown in figure 1.2. Minor scripts are discussed next,

cgs_read_market: function that calls security constraint screening step 1 and 2 with variations.

cgs_read_mc: functions that read the results of DAM solution and determines the true domination constraints. Some functions are used for verification purposes.

cgf_deviation: This script contains functions that create the price constraints. Add uncertainty to the results of DAM solution and resolve the power flow upper and lower bounds of selected security constraints. Mapping uncertainty limits of power injections to uncertainty limits of power flow. Also introduce price constraints (budge of uncertainty) to congestion forecasting process, refer to price constraints (budget of uncertainty) section.

step1fcn: For security constraint screening step1, derivation and explanation, see methodology section.

Step 1 is numerical method, the idea is to sort sensitivity factor on a constraint in descending order, if the sum of power injection on the respective node multiplied by the sensitivity factor can be greater or equal to the power flow limit, the constraint have the potential of becoming binding. This method is the necessary (but not sufficient) condition for line congestion. Step1 decouples individual constraints from

the MIP model, which is parallel processing friendly. We used python numerical calculation acceleration package NUMBA to help with computation efficiency.

step2fcn: For security constraints screening step2, derivation and explanation, see methodology section.

Essentially, there are three variations of step2. In short, step2 creates an LP problem with the objective as the security constraint under consideration, the constraints are all the rest of the security constraints in the SCUC model, load balance. Then, to optimize the model, if the objective, maximized power flow, is greater than the limit, the security constraint has the potential to become binding. The first variable is called the feasibility check, which is set the objective of the LP problem to be zero, set the targeted constraint as a constraint, and it is greater than the power flow limit, all other constraints remain the same. If this problem is feasible, meaning the target constraint can be greater than its limit, the constraint is marked as dominating. We will get the upper and lower bounds of the power flow by solving the LP problem completely. This will be the 2nd variation. However, the complete solution of the LP problem takes quite long to solve. The 3rd variation is the complete solution to the problem and retrieve the dual variables of the LP problem. This is needed for Online method.

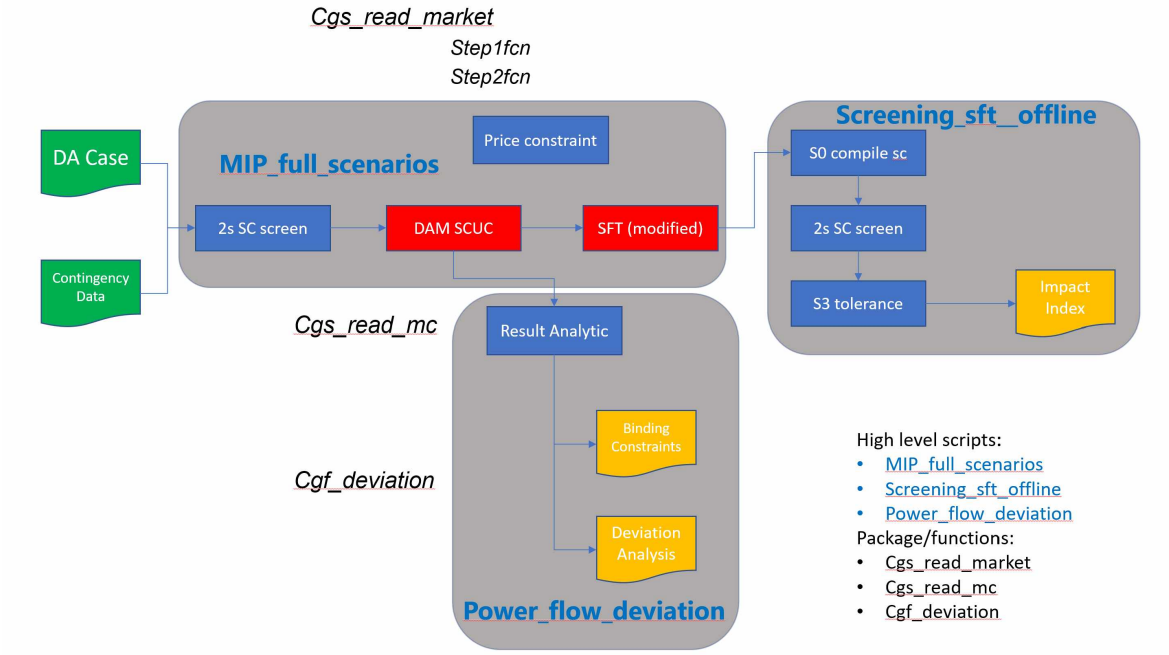

Figure 1.2: Congestion Forecasting Framework (offline)