Two-Stage Security Constraint Screening for Scalable SCUC Computation

Sushant Varghese, Shubo Zhang, Hamid Davoudi, Matthew Musto, Kanchan Upadhyay, Jack Garrett, and Pradip G. Kumar
New York Independent System Operator, Rensselaer, NY 12144, USA

***Abstract*—Security-Constrained Unit Commitment (SCUC) models are central to ISO market clearing, ensuring that transmission constraints are satisfied and power flows remain feasible under N-1 and N-2 contingencies. The large number of network security constraints can make SCUC computationally intractable, particularly when implementing complex model changes that are computationally demanding, such as dynamically calculated nodal reserves that enforce reserve deliverability. This paper introduces a two-stage constraint screening framework to identify and remove redundant constraints. Stage 1 applies a bound-based test to detect constraints that cannot become active under conservative system limits. Stage 2 incorporates both bound and constraint interactions via a linear programming formulation to capture interdependencies among candidate constraints. The approach is tested on NYISO's market clearing engine for i) the current production system and ii) a prototype system with nodal reserve requirements. Results demonstrate significant reductions in day-ahead SCUC solution times while maintaining model accuracy, enabling improved scalability as NYISO incorporates renewable generation and explores potential reserve modeling enhancements.**

***Index Terms*—security-constrained unit commitment, constraint screening, umbrella constraints, network security analysis, dynamic reserves, NYISO.**

I. Introduction

The New York Independent System Operator (NYISO) operates competitive wholesale electricity markets and ensures reliable system operation in the state of New York. Central to this is the Security-Constrained Unit Commitment (SCUC) model, which optimally commits and dispatches generation while maintaining reliability under normal and contingency conditions [1].

To ensure network security, the SCUC integrates with the Network Security Analysis (NSA) process, which validates each candidate solution against both base-case and post-contingency conditions. After each SCUC iteration, the current commitment and dispatch schedules are evaluated through alternating-current (AC) power flow simulations for all monitored contingencies (typically N-1 or N-2). Any identified thermal or voltage limit violations are translated into security constraints, which are linearized representations of network limits and are fed back into the next SCUC iteration. This iterative feedback loop continues until no further violations are detected, guaranteeing that the final commitment schedule is both economically optimal and secure [2].

While this approach ensures operational reliability, it also introduces significant computational overhead. Each iteration can generate thousands of security constraints, with even more added when reserve deliverability and contingency relationships are included [3]. Many of these constraints are redundant, meaning they never bind or influence the feasible region but still persist in the model, inflating data size and solver time.

Although NYISO's current SCUC implementation remains computationally tractable, future market design enhancements that incorporate higher renewable penetration, energy limited resources and dynamic reserve deliverability requirements will substantially increase model dimensionality and computational complexity [4]. These model improvements are necessary to handle emerging system changes but will also magnify the scalability challenges inherent to large-scale mixed-integer optimization. Addressing redundant constraint growth is therefore crucial for maintaining reliable and efficient market clearing as NYISO transitions to more complex, data-intensive SCUC formulations.

Recent literature using heuristic or machine-learning–based filtering techniques can reduce problem size but lack the theoretical guarantees needed for production use, as they risk removing critical security constraints [5]–[7]. This work instead introduces a deterministic and theoretically robust two-stage constraint-filtering framework that enhances computational scalability while fully preserving system security and solution integrity. The method draws from the analytical congestion conditions proposed in [8], [9] and combines them with an optimization-based dominance test similar in spirit to umbrella-constraint screening [1].

II. Problem Definition

In NYISO's SCUC formulation, the NSA module generates linear constraints to enforce post-contingency transmission flow limits. These security constraints, along with other unit commitment (UC) and economic dispatch (ED) constraints, define the feasible region of the commitment problem, but not all are necessary. Many security constraints remain inactive across all feasible operating conditions, providing no benefit to solution quality while contributing to model size and leading to longer solve times [3].

This redundancy creates a cascading effect: each NSA constraint is linked to multiple auxiliary constraints representing contingency interactions, reserve deliverability, and other network relationships. As a result, even a modest reduction in redundant security constraints can translate into tens of

979-8-3315-8138-1

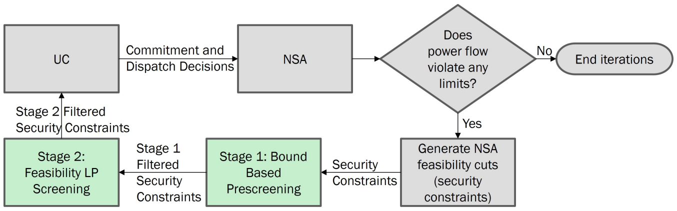

Fig. 1. SCUC framework with redundant constraint filtering

thousands of auxiliary constraints being removed from the overall SCUC model. The outcome is a smaller optimization problem without compromising physical realism or reliability.

The challenge lies in efficiently distinguishing between constraints that are essential for maintaining system feasibility and those that can be safely eliminated. The objective of this work is to develop a structured, two-stage constraint-filtering framework that identifies and removes redundant NSA-generated constraints with theoretical rigor. Stage 1 employs analytical bounds based on necessary congestion conditions [8], [9], while Stage 2 solves feasibility-based linear programs to certify dominance among interacting constraints [1]. By targeting redundancy at the network-security level, the proposed approach significantly improves computational efficiency and enables the scalability required for next-generation SCUC formulations incorporating renewable variability, uncertainty, and dynamic reserve deliverability [2].

Fig. 1 demonstrates how this implementation is integrated into the UC-NSA loop. In contrast to prior screening approaches, this work contributes an ISO-scale, deterministic implementation with commitment-consistent bounds, reverse-direction validation, and pre-generation screening. The design reduces both solver time and data-handling cost while preserving feasibility, yielding an auditable, operator-grade process suitable for day-ahead timelines.

III. Methodology

To address the scalability limitations introduced by redundant security constraints, we develop a two-stage constraint-filtering framework that deliberately balances speed and rigor. Stage 1 rapidly removes clear redundancies using analytical bounds, while Stage 2 applies an optimization-based dominance test to capture subtle network interactions that analytical screening alone cannot. This complementary design allows most redundant cuts to be discarded almost instantly, with a small number of survivors examined more deeply to guarantee accuracy. The method is general to large-scale UC and designed for NYISO's production environment, where both computational determinism and reliability are critical.

Let $\mathcal{L}$ be the set of monitored lines and $\mathcal{K}$ the controllable injections (generators, phase shifters, dispatchable loads/virtuals). The set of monitored lines may include all transmission lines in the system or just a subset of lines that need to be secured for contingencies. $\mathcal{K}$ must include all resources and loads in the system that can inject into or withdraw power from the grid.

p_k is a decision variable and limits $P_k^{\min} \le p_k \le P_k^{\max}$. Let H_{ik} be the direct-current (DC) shift factor from injection k to line i, and F_i the (one-sided) flow limit. The set $\mathcal{L}$ includes line definitions in both directions such that reverse direction constraints are handled symmetrically. A typical NSA cut is

$$\sum_{k\in\mathcal{K}} H_{ik}\, p_k \le F_i, \quad \forall i \in \mathcal{L}. \tag{1}$$

A. Stage 1: Bound-based prescreening

This stage performs a coarse yet safe elimination of constraints using analytical bounds derived from power-flow sensitivities. It evaluates whether each constraint could ever bind given conservative bounds on generation, demand, and network parameters. Those that cannot possibly bind are removed immediately. Because this test is purely algebraic and fully vectorizable, it completes in less than $0.1\,\mathrm{s}$ per case while removing roughly $15\,\%$ of all constraints.

Define $H_{ik}^{+} = \max(H_{ik}, 0)$ and $H_{ik}^{-} = \min(H_{ik}, 0)$. A conservative upper bound for the left-hand side (LHS) of (1) is

$$\overline{f}_i = \sum_{k\in\mathcal{K}} \left(H_{ik}^{+} P_k^{\max} + H_{ik}^{-} P_k^{\min}\right). \tag{2}$$

If $\overline{f}_i < F_i$, constraint i is provably redundant and can be safely removed.

Commitment-consistent bounds: When screening for unit commitment, resources may be offline. To ensure that filtering remains conservative, we adopt commitment-consistent bounds in (2): for any unit that can be off, we set $P_k^{\min} = 0$, while keeping its maximum achievable output $P_k^{\max}$. This adjustment preserves security across the commitment states. The same relaxation is unnecessary for economic dispatch, but crucial for mixed-integer SCUC. Units that have $P_k^{\min} < 0$ and $P_k^{\max} < 0$ such as loads and virtuals are also relaxed such that their $P_k^{\max} = 0$ to account for load shedding events. Constraints passing this step form the survivor set $\mathcal{S} \subseteq \mathcal{L}$.

Stage 1 thus provides a fast, analytical first cut, removing only those constraints that are provably non-binding under any feasible injections, yet retaining every potentially active constraint for deeper verification in Stage 2.

B. Stage 2: Feasibility-LP screening

Stage 2 enforces rigor by explicitly modeling the interactions among surviving constraints. For each $i \in \mathcal{S}$, a feasibility linear program (LP) maximizes the monitored flow while respecting all other network limits, demand balance, and bounds. If even under these most favorable conditions the flow remains below its limit, the constraint is redundant.

Formally, for each $i \in \mathcal{S}$ we solve the LP, Q_i:

$$\max_{p} \sum_{k\in\mathcal{K}} H_{ik}\, p_k \tag{3}$$

979-8-3315-8138-1

$$\text{s.t.} \quad \sum_{k\in\mathcal{K}} H_{jk}\, p_k \le F_j,\ \forall j \in \mathcal{S}\setminus\{i\}, \tag{4}$$

$$\sum_{k\in\mathcal{K}} p_k = D, \tag{5}$$

$$P_k^{\min} \le p_k \le P_k^{\max},\ \forall k \in \mathcal{K}. \tag{6}$$

Here D aggregates fixed loads and loss-delivery adjustments. The bounds in (6) use the same commitment-consistent values as Stage 1. If the optimum $f_i^\star$ satisfies $f_i^\star < F_i$ (and likewise for the reverse direction), then line i is redundant given feasible network interactions. This can be proved as follows. Let X^{MIP} denote the feasible region of the full unit commitment problem. For each candidate constraint i, let X_i^{LP} be the relaxation defined by equations (4)–(6). Because X_i^{LP} is constructed using only a subset of the constraints that define X^{MIP}, it follows that $X^{MIP} \subseteq X_i^{LP}$. By definition of optimality,

$$\sum_{k\in\mathcal{K}} H_{ik}\, p_k \le f_i^\star \quad \forall p \in X_i^{LP} \tag{7}$$

Since $X^{MIP} \subseteq X_i^{LP}$, the same bounds applies to all feasible SCUC dispatches. Thus if $f_i^\star < F_i$:

$$\sum_{k\in\mathcal{K}} H_{ik}\, p_k \le f_i^\star < F_i \quad \forall p \in X^{MIP} \tag{8}$$

Therefore, if $f_i^\star < F_i$, then no dispatch feasible to the true SCUC problem can violate constraint i.

Each LP (3)–(6) is independent, enabling straightforward parallelization and batched factorization. Because each LP is substantially smaller and simpler than the full unit commitment (UC) problem, which is a large-scale mixed-integer linear programming (MILP), the screening framework achieves strong computational scalability, aided by the elimination of auxiliary constraints and parallel execution.

C. Complementary roles of Stage 1 and Stage 2

The two stages operate in a complementary and sequential manner. Stage 1 is performed first and is designed for speed and broad coverage, rapidly screening thousands of constraints in milliseconds. Constraints that are not eliminated in Stage 1 are passed to Stage 2 for further screening. Stage 2 emphasizes precision and reliability, validating the remaining constraints through explicit feasibility optimization. As the screening operates on DC shift-factor constraints, these guarantees apply to the DC formulation. In practice, the AC-based NSA validation remains the final safeguard, reintroducing any binding constraints that the DC approximation may not capture. The overall implementation of this framework is summarized in Algorithm 1.

D. Complexity, Generalizability and Implementation Details

Stage 1 has computational complexity $O(|\mathcal{L}||\mathcal{K}|)$ and is fully vectorizable, allowing rapid elimination of clearly redundant constraints. Stage 2 solves $|\mathcal{S}|$ independent linear programs (LPs) of identical structure, which are orders of magnitude smaller and simpler than the full MILP SCUC problem. Warm starts and shared factorization further amortize computational effort, making Stage 2 efficient even when processing hundreds of security constraints. Contingency screening is performed independently for each time period, enforcing all bounds and demand balance. This time-decoupled approach is more computationally efficient than time-coupled filtering.

Algorithm 1 Two-Stage Redundant-Constraint Filtering

Input: NSA cuts $\{(H_i, F_i)\}_{i\in\mathcal{L}}$, bounds $\{P_k^{\min}, P_k^{\max}\}$, demand D
Output: Survivor set $\mathcal{S} \subseteq \mathcal{L}$
$\mathcal{S} \leftarrow \emptyset$
for each $i \in \mathcal{L}$ **do** ▷ Stage 1: fast analytical prescreen
  compute $\overline{f}_i$ via (2) (commitment-consistent bounds)
  if $\overline{f}_i \ge F_i$ **then**
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{i\}$
  end if
end for
for each $i \in \mathcal{S}$ in parallel **do** ▷ Stage 2: rigorous LP validation
  solve LP Q_i
  if $f_i^\star < F_i$ **then**
    $\mathcal{S} \leftarrow \mathcal{S} \setminus \{i\}$
  end if
end for
return $\mathcal{S}$

The method is model- and ISO-agnostic and can be applied to any subset of constraints. It is most effective for large sets of linear constraints, where solving many LP subproblems is faster than solving a single large MILP. The two-stage filter is applied before SCUC model generation to reduce data volume, constraint creation, and presolve time. Survivor sets, bounds, and LP results are logged to ensure reproducibility, auditability, and transparency.

IV. Case Study

The proposed two-stage constraint filtering framework was evaluated using two formulations of NYISO's SCUC: the production SCUC model and a complex prototype SCUC model. These case studies demonstrate both the reliability of the method in current market operations and its potential to improve scalability for future enhancements. All numerical simulations were carried out on NYISO's market clearing engine using © Gurobi 9.1.1 on Red Hat Enterprise Linux 7.5 with an Intel Xeon E5@2.60 GHz CPU and 256 GB RAM. Stage 2 constraint filtering is parallelized across 8 cores, and the relative MIP-gap tolerance is set to 10^{-6}.

A. Production SCUC Model Validation

The production SCUC model represents the formulation currently used in NYISO's day-ahead market clearing process. The SCUC model has a 24-hour time horizon with 1-hour intervals and determines unit commitment and economic dispatch decisions for all generating units in the system. It accounts for generator operating characteristics, including

979-8-3315-8138-1

minimum up and down times, ramp rates, startup and shutdown costs, and generator operating costs. The model also enforces system-wide energy balance and meets deterministic reserve requirements to ensure reliability under contingency scenarios. The reserve requirements are 655 MW for 10-min spinning reserve, 1310 MW for 10-min total reserve and 2620 MW for 30-min total reserve.

It includes transmission security constraints generated through NSA to enforce N-1 and N-2 reliability across critical transmission elements, capturing the limits on line flows under credible contingencies. The model explicitly considers N-2 contingencies rather than N-1-1 events. Unlike N-1-1 contingencies, which allow for corrective actions between sequential outages, the N-2 contingencies modeled here assume the simultaneous loss of two components with no opportunity for intermediate recourse and are therefore more conservative. The SCUC formulation integrates these operational and network constraints into a mixed-integer linear programming problem that is solved each day to provide the day-ahead market schedule. To validate that constraint filtering does not compromise feasibility or accuracy, thirty representative 2025 day-ahead cases were analyzed.

Each case was solved twice: with the full NSA constraint set and with the filtered set. Commitment and dispatch schedules were identical for the LP and within the same optimality tolerance for the MILP. All filtered solutions passed post-contingency validation, confirming that no active constraints were removed. Because the production SCUC already meets runtime targets, no performance gains were expected; this study instead verifies that filtering preserves model integrity and can be safely deployed in production.

B. Complex SCUC Prototype Model

To evaluate the computational benefits of constraint filtering, the framework was tested on a complex SCUC prototype model developed by NYISO [10]. This prototype extends the current SCUC formulation to reflect anticipated market design enhancements, including dynamic reserve deliverability, enhanced network modeling, and nodal reserve constraints.

The critical difference between the two models that contributes to the complexity of the model is the reserve deliverability constraints (9). They serve two purposes: i) ensure that the procured reserves are deliverable (reserve deployment does not exceed transmission line limits) and ii) ensure that there are sufficient reserves procured to meet generator contingencies that can overload transmission lines. In addition to the line outages modeled in NSA, these constraints model possible generator outages and secure the system to N-1 for 10-min spinning and 10-min total reserves and N-2 for 30-min total reserves for both generator and line outages. For each secured line, a set of generator outages are modeled and $o \in \mathcal{O}$ denotes the set of generator outages that are modeled. r_k denotes the reserves procured from unit k and $\mathcal{K}_o^{OUT}$ denotes the set of generators that are out in outage scenario o. Since N-1 and N-2 contingencies are modeled, this set can consist of one or two generators. The sets are defined such that no more than 2 contingencies (line or generator) are modeled, i.e., if a line contingency is already modeled in NSA, no more than 1 generator contingency is modeled.

$$\begin{aligned}\sum_{k \in \mathcal{K}} H_{ik}\, p_k - \sum_{k \in \mathcal{K}_o^{OUT}} H_{ik}\, p_k \\ + \sum_{k \in \mathcal{K} \setminus \mathcal{K}_o^{OUT}} H_{ik}\, r_k \leq F_i \quad \forall i \in \mathcal{L},\, o \in \mathcal{O}\end{aligned} \tag{9}$$

Unlike the production model, the prototype explicitly links reserve requirements to network deliverability and transmission constraints, substantially increasing the number of variables, contingencies, and auxiliary constraints. For each secured line, there are $|\mathcal{O}|$ constraints added to the set of constraints. The set $\mathcal{O}$ can be large since it includes a combination of 2 generators in the N-2 contingency. While this formulation captures system flexibility more accurately, it also introduces significant computational burden.

For the complex SCUC prototype test case, fifteen representative 2024 day-ahead cases were used to test the filtering method. The two-stage process was applied as a preprocessing step prior to SCUC model construction, ensuring that only essential NSA constraints were passed to the solver. This setup allows for a clear measurement of structural and performance improvements attributable to constraint filtering.

C. Results

All results presented in this section correspond to the complex SCUC prototype model described earlier. Fifteen representative 2024 day-ahead cases were analyzed, and the reported values reflect averages across all fifteen test cases. The results quantify the impact of the two-stage filtering framework on constraint reduction, model structure, and computational performance.

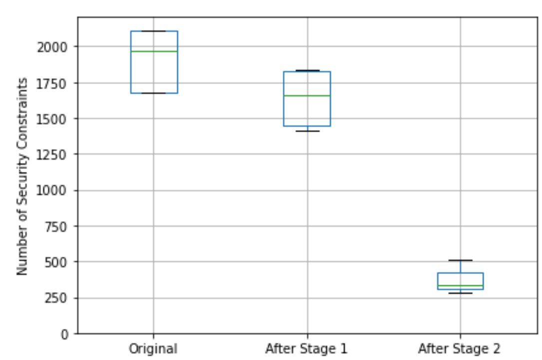

Fig. 2. Reduction in the number of security constraints after each screening stage across all test days.

1) Reduction in Constraints: The two-stage filtering process achieved substantial reductions in NSA-generated constraints without compromising system security. Fig. 2 demonstrates the number of constraints that remain after each stage of the screening process across all the test days.

979-8-3315-8138-1

- **Stage 1 (Bound-based prescreening)** removed approximately 15% of constraints within 0.1 seconds per case on average.
- **Stage 2 (Feasibility-LP screening)** eliminated an additional 65% within about 10 seconds per case.

Combined, these stages reduced the total number of NSA-generated constraints by over 80%. Because each NSA constraint corresponds to multiple auxiliary constraints in the SCUC formulation, this reduction also led to an average 12% decrease in total model size and a 74% reduction in auxiliary constraints such as 9. These results, summarized in Table I, confirm that the filtering framework effectively streamlines the problem formulation before optimization begins. Using Stage 2 alone would produce comparable results but would require higher LP effort.

TABLE I
CONSTRAINT REDUCTION: COMPLEX SCUC CASE

Metric	Before (count)	After (count)	Change (abs, %)
Security constraints	1,920	369	$-1{,}551$ (-81%)
Auxiliary constraints	26,946	7,042	$-19{,}904$ (-74%)
Total SCUC constraints	179,392	157,937	$-21{,}455$ (-12%)

Note: "Auxiliary constraints" are non-security constraints induced by network and reserve deliverability modeling.

2) Impact on Solution Time: The filtered SCUC formulation achieved significant runtime improvements relative to the unfiltered baseline. Table II shows the impact on the MILP solver time, data processing time and the total SCUC time. Across all 15 prototype cases, the average total SCUC runtime decreased by 76%, with most gains observed during model construction and MILP solving phases. The filtering procedure itself required less than 10 seconds on average, which is negligible compared to the average runtime savings of 1,260 seconds (over 20 minutes) per case. In the baseline, solver presolve was enabled to remove constraints deemed redundant; nevertheless, the proposed model-aware filtering approach achieved significantly greater reductions in runtime.

TABLE II
COMPLEX SCUC TIME REDUCTION (SECONDS)

Phase	Before (s)	After (s)	Change (abs, %)
MILP solver	256	80	-176 (-69%)
Data handling	1411	317	-1094 (-78%)
Filtering*	0	9.7	$+9.7$ (—)
Total	1667	407	-1260 (-76%)

* Filtering is additional preprocessing not present in the baseline and is included in the *Total*.

Despite these substantial reductions, all filtered solutions remained fully feasible and had identical objective costs (within the required tolerance). Post-contingency validation confirmed that no active or potentially active constraints were removed, ensuring that system security and market reliability were fully preserved.

V. DISCUSSION AND CONCLUSION

The increasing scale of SCUC models makes enforcing thousands of security constraints computationally burdensome for ISO market clearing. We present a practical, model-aware constraint screening framework that reduces problem size while guaranteeing identical market outcomes and system reliability. The proposed two-stage approach combines analytical congestion bounds with feasibility-based LP validation to remove provably redundant security constraints. Applied to NYISO day-ahead cases, the method produced identical economic and feasibility results. This work demonstrates that screening techniques can be implemented reliably in production environments, providing a deterministic and auditable path to scaling large market models. Future work will explore more aggressive constraint filtering using dual information from Stage 2, extend the approach to stochastic SCUC formulations, refine commitment-consistent bounds, and capture intertemporal relationships.

ACKNOWLEDGMENT

The views and conclusions in this paper are those of the authors and do not necessarily reflect those of the NYISO. The methods and analyses are for research purposes only and do not imply adoption or implementation in NYISO's market design or production systems.

REFERENCES

[1] 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, 2013.

[2] S. Zhang, H. Ye, F. Wang, Y. Chen, S. Rose, and Y. Ma, "Data-aided offline and online screening for security constraint," *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 2614–2622, 2021.

[3] R. Madani, J. Lavaei, and R. Baldick, "Constraint screening for security analysis of power networks," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2732–2742, 2017.

[4] C. Tejada-Arango, S. Lumbreras, and A. Ramos, "Constraint screening in large-scale security-constrained unit commitment," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 2751–2762, 2018.

[5] S. Pineda, J. M. Morales, and A. Jimenez-Cordero, "Data-driven screening of network constraints for unit commitment," *IEEE Transactions on Power Systems*, vol. 35, no. 5, pp. 3695–3705, 2020.

[6] S. Park and P. Van Hentenryck, "Self-supervised learning for large-scale preventive security constrained dc optimal power flow," *IEEE Transactions on Power Systems*, 2024.

[7] M. I. Alizadeh and F. Capitanescu, "A tractable linearization-based approximated solution methodology to stochastic multi-period ac security-constrained optimal power flow," *IEEE Transactions on Power Systems*, vol. 38, no. 6, pp. 5896–5908, 2022.

[8] 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, 2016.

[9] 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, 2010.

[10] M. Musto, K. Upadhyay, and E. Lo, "Optimising energy and reserve schedules for post-contingency scenarios: A security constrained unit commitment approach," Presentation at the FERC Technical Conference, Jul. 2024. [Online]. Available: https://www.ferc.gov/media/presentation-optimising-energy-and-reserve-schedules-post-contingency-scenarios

979-8-3315-8138-1