Transmission Constraint Filtering in Large-Scale Security-Constrained Unit Commitment

Álinson S. Xavier, *Member, IEEE*, Feng Qiu, *Senior Member, IEEE*, Fengyu Wang, *Member, IEEE*, Prakash R. Thimmapuram

***Abstract*—When solving the security-constrained unit commitment (SCUC) problem, one of the most complicating factors is handling the large number of transmission constraints, corresponding to both base case and N-1 line contingency scenarios. Although it is well known that only a few of these constraints need to be enforced, identifying this critical subset of constraints efficiently remains a challenge. In this paper, we propose a novel and simple iterative contingency-screening procedure that is able to eliminate 99.4% of the constraints selected by existing iterative methods, allowing for the solution of much larger-scale problems. We report computational results in realistic instances with up to 6,468 buses and 9,000 transmission lines. The method was also independently implemented and evaluated at MISO, where it performed faster than alternative methods.**

Keywords—Mixed-integer linear programming (MIP), security constrained unit commitment (SCUC), contingency screening.

I. Introduction

SCUC is one of the most fundamental optimization problems in energy systems, being solved a number of times daily by major reliability coordinators, independent system operators (ISOs) and regional transmission organizations (RTOs). The objective of SCUC is to determine the most cost-effective operating schedule of generating units, while ensuring that the load is satisfied over a given operational horizon and that the operations are secured even if there is a transmission line contingency. Additional constraints, such as ramping constraints and system-wide operating reserves, are also usually enforced.

Mathematically, SCUC is nowadays most often formulated as a mixed-integer linear programming problem (MIP). One factor that significantly complicates the solution of SCUC is the requirement that any solution must satisfy a large set of transmission constraints, corresponding not only to the base case, but also to each N-1 line contingency scenario. While adding all these constraints to the MIP formulation can quickly make SCUC computationally intractable, it has been observed that enforcing only a small subset of these constraints is already sufficient to guarantee that all the remaining ones are automatically satisfied [1]. Identifying this critical subset of constraints efficiently, however, remains a challenge.

Á. S. Xavier, F. Qiu and P. R. Thimmapuram are with Energy Systems Division, Argonne National Laboratory, Lemont, IL 60439 USA (e-mail: axavier@anl.gov, fqiu@anl.gov, prakash@anl.gov). F. Wang is with Midcontinent Independent System Operator, Inc. (MISO), Carmel, IN 46032 USA. (email: fwang@misoenergy.org). Final published article: https://doi.org/10.1109/TPWRS.2019.2892620

In practice, system operators still rely on experience and external ad-hoc processes to decide what transmission constraints to enforce [2]. Previous research proposed identifying a critical set of constraints through auxiliary optimization problems [3]. However, such approach is not yet scalable to real-size transmission networks. More recently, researchers showed that a good subset of constraints can be found through a simple iterative procedure [2], [4]. Their approach is the following. First, SCUC is solved with only the pre-contingency transmission constraints. Then, after a solution is obtained, every post-contingency transmission constraint that turns out to be violated is added to the relaxation. The process repeats, and new constraints are added until no violations are found.

In this paper, we propose a refinement to this procedure. The main novelty is that, instead of adding all the violated post-contingency transmission constraints to the relaxation, as done in previous methods, we develop a fast and effective heuristic procedure to filter down this list. Numerical experiments show that our method is able to eliminate 99.4% of the constraints selected by previous methods, leading to significant improvements in solution time. More importantly, the dramatic reduction in the number of constraints allows us to solve significantly larger problems, and we present computational results on large-scale realistic instances, with up to 6,468 buses and 9,000 transmission lines. The proposed method was also independently implemented and evaluated at MISO, the Midcontinent Independent System Operator. The proposed method presented better performance when compared to MISO's own constraint filtering methods.

II. Mathematical Formulation

A number of MIP formulations have been proposed for the security-constrained unit commitment problem. In our experiments, we have modeled SCUC using formulation [5]. The formulation enforces (i) minimum and maximum production limits; (ii) ramping restrictions; (iii) minimum uptime and downtime; and (iv) system-wide reserves. It includes startup costs, as well as piecewise linear production costs. For transmission, the DC power flow-model was used, and the constraints were modeled using Injection Shift Factors (ISF).

III. Constraint Selection Algorithm

In this section, we present a refinement of the procedures presented in [2] and [4]. Our refinement is based on two main observations. First, commercial MIP solvers have become surprisingly effective at solving the unit commitment problem when no transmission constraints are included. In fact, these solvers are often able to find very high quality solutions at the root node of the branch-and-bound tree through primal heuristics and cutting planes alone. It would be helpful to take advantage of these advancements, even when solving SCUC. Secondly, as already observed by [6], even when multiple N-1 security constraints are violated, it is seldom necessary to add all of them to the relaxation, as done before. Enforcing a small number of constraints is, most times, already sufficient to guarantee that many others are automatically satisfied.

With these two observations in mind, we propose the usage of Algorithm 1 to solve SCUC. At start, we solve a relaxation of SCUC that does not have any transmission or N-1 security constraints. Then, pre-contingency and post-contingency flows are calculated by using ISF and Line Outage Distribution Factors (LODF), respectively. Next, instead of adding all the violated constraints to the relaxation, as done by previous methods, we filter this list even further. First, we keep at most one constraint for each monitored transmission line. That is, if the flow on a certain monitored transmission line exceeds the limit under multiple N-1 line contingency scenarios, we keep only the constraint corresponding to the scenario that causes the largest violation. Next, from all the remaining constraints, we keep only a fixed number k of constraints having the highest violation. The best value for k varies according to the network, but values between 5 and 15 presented the best results in our experiments, for networks of various sizes. The main difference between our approach and [6] or [7] is that we do not solve any auxiliary problems to decide which subset of constraints to keep. Picking the constraints that are most violated proves to be a fast and effective heuristic. We clarify that the solutions obtained by Algorithm 1 are the same as the solutions obtained by previous methods, such as [4].

Note that, because of the size of the transmission networks considered, merely identifying and generating the violated constraints can become a bottleneck to the entire procedure if not done carefully. In our implementation, pre-contingency and post-contingency flows are computed using `jblas`, a fast linear algebra library. Because of their high density, we never explicitly compute transmission constraints until it has been determined that they will be added to the relaxation. In addition, to efficiently maintain the list of the largest violations found so far, we make use of priority queues. Our implementation also takes advantage of the fact that many steps in Algorithm 1 can be performed in parallel, in particular the computation of the post-contingency flows. To improve performance even further, we solve the first iterations of SCUC with a large MIP gap tolerance of 5%. After no further violations are found, the gap tolerance is reduced to 0.1% and the problem is resolved.

Algorithm 1 Security-Constrained Unit Commitment

1: Let L^M be the set of monitored transmission lines
2: Let L^V be the set of transmission lines susceptible to disruption
3: Create a relaxation of SCUC without any transmission constraints
4: Solve the current relaxation
5: Compute pre-contingency flow f^0 using ISF
6: Compute post-contingency flow f^v using LODF, $\forall v \in L^V$
7: Let $\gamma_m^v = \max\{-f_m^v - F_m, 0, f_m^v - F_m\}, \forall v \in L^V \cup \{0\}, m \in L^M$
8: Let $\Gamma = \{(v,m) \in (L^V \cup \{0\}) \times L^M : \gamma_m^v > 0\}$ be the violations
9: **if** Γ is empty **then return**
10: **else**
11: For $m \in L^M$, keep in Γ only the pair (v,m) with highest γ_m^v
12: Keep in Γ only the k pairs (v,m) having the highest γ_m^v
13: For every violation in Γ, add the corresponding cut to the relaxation
14: **goto** step 4

TABLE I. SIZE OF SELECTED INSTANCES.

Instance	Buses	Units	Branches	Scenarios	Constraints
472-bus	472	216	752	744	1,118,976
1354-bus	1,354	260	1,991	1,288	5,128,816
1951-bus	1,951	391	2,596	1,497	7,772,424
3375-bus	3,374	596	4,161	3,245	27,004,890
6468-bus	6,468	1,295	9,000	6,094	109,692,000

IV. COMPUTATIONAL EXPERIMENTS

A. Instances and setup

Five realistic instances of varying sizes were used to evaluate the effectiveness of the proposed method. Table I presents the number of buses, generators, branches and scenarios for each instance. The table also shows the total number of transmission constraints in the model, per time period, when no filtering is performed. A 24-hour planning horizon was considered for all instances. Instance 472-bus is described in [4]. All the other instances were obtained from MATPOWER [8] and correspond to realistic, large-scale European test systems. In the MATPOWER instances, there is one contingency scenario for each transmission line, except for those lines whose removal would cause the network to become disconnected. Some generator data necessary for SCUC was missing in these instances, and was artificially generated based on real data distributions. As it is usual, sensitivity factors with small magnitudes were set to zero, in order to improve the sparsity of the constraints. The cutoffs used were 0.005 and 0.0005 for the ISF and LODF matrices, respectively.

The algorithms were implemented in Java, with IBM ILOG CPLEX 12.7.2 as MIP solver, and jblas 1.2.4 as linear algebra library. The experiments were run on an Intel Core i7-4600U (2 cores, 4 threads, 3.10 GHz) and 16.0 GB of memory. No time limit was imposed. The relative MIP gap tolerance was set to 0.1%, and all the other CPLEX parameters were left unmodified. It is well known that MIPs are often subject to dramatic variability in performance due to seemingly trivial changes [9]. To reduce this variability, each instance was solved 10 times, and CPLEX was given a different random seed at each run. The numbers obtained were then averaged.

B. Performance Evaluation

Table II shows the average wall-clock running time, number of iterations and number of constraints added (per time period), for different methods. Column "No Transmission (NT)" shows the average running-time needed to solve the model without any transmission constraints or N-1 security constraints. Column "Proposed" corresponds to Algorithm 1. Column "TSR", named after the initials of the authors, corresponds to the strategy presented in [4].

First we observe that the proposed method added significantly fewer transmission constraints to the relaxation when compared to TSR. Indeed, for instance 472-bus, the

TABLE II. PERFORMANCE COMPARISON ON IEEE AND MATPOWER INSTANCES. MISSING VALUES INDICATE OUT-OF-MEMORY.

	NT	Proposed			TSR		
Instance	Time (s)	Time (s)	Iter.	Cuts	Time (s)	Iter.	Cuts
472-bus	13.3	32.5	3.8	8.8	370.0	2.7	1,591.0
1354-bus	16.3	28.7	5.0	27.8	—	—	—
1951-bus	26.5	31.9	4.0	17.9	—	—	—
3375-bus	39.2	146.1	4.2	29.5	—	—	—
6468-bus	98.0	279.4	4.7	35.8	—	—	—

number of constraints added per time period by the proposed method was only 0.55% of TSR. This significant reduction also reflected in the total wall-clock time, with the proposed method being more than 11 times faster than TSR, on average. Interestingly, the reduction in the number of added constraints did not cause a significant increase in the number of iterations. The proposed method required on average only 1.1 iterations more than TSR.

More importantly, the dramatic reduction in the number of constraints added allowed us to solve much larger scale problems than [4]. While TSR exceeds the memory limit for all instances, with the exception of 472-bus, the proposed method is able to solve problems with up to 6468 buses in under 5 minutes of running time. We also observe that the number of cuts added did not increase as fast as the total number of transmission constraints in the model. For example, although 6468-bus has almost 100 times more transmission constraints in total than 472-bus, the proposed method added only 4 times the number of cuts. The number of iterations have also not increased significantly with instances of very large size. We conclude that the proposed method scales very well, even for large-scale instances. With the proposed algorithm, solving the unit commitment problem with transmission and N-1 security constraints was only 3 to 4 times slower than solving the traditional unit commitment problem.

C. MISO Implementation and Evaluation

The proposed method was independently implemented and evaluated at MISO, the Midcontinent Independent System Operator, which currently operates one of the largest electricity markets in the world. MISO's network model includes over 45,000 buses and 1,400 generators. When solving Day-Ahead SCUC, MISO needs to enforce a large-number of pre-selected *watch list constraints*, which are identified through external processes, but which are not always binding. To mitigate the negative performance impact of adding all these constraints simultaneously to the relaxation, MISO developed its own iterative constraint filtering procedure, named the *decomposition method*, as described in Section III.C of [2].

Table III shows a performance comparison between MISO's decomposition method and the proposed approach. Three challenging day-ahead samples were selected for this benchmark, corresponding to one winter day, one spring day, and one summer day. We note that MISO's implementation of Algorithm 1 was simplified, and, for technical reasons, did not take advantage of some features found in modern MIP solvers, such as warm starting. Nevertheless, the proposed approach was still faster than MISO's decomposition approach, requiring 14% smaller running times, while producing solutions of the same quality. Although the proposed approach required more iterations, each iteration was solved faster, leading to a net reduction in running time.

TABLE III. PERFORMANCE COMPARISON ON MISO INSTANCES.

	MISO			Proposed		
Instance	Time (s)	Iter.	Gap	Time (s)	Iter.	Gap
Case 1	1352	3	0.07%	1120	4	0.08%
Case 2	1877	4	0.09%	1537	5	0.06%
Case 3	1410	3	0.03%	1335	4	0.04%
Average	1546	3.3	0.06%	1331	4.3	0.06%

V. CONCLUSION

In this paper, we presented a simple and novel iterative contingency-filtering scheme for SCUC. In computational experiments, the proposed method eliminated 99.4% of the constraints selected by earlier iterative methods, leading to significant performance improvements. The method was able to solve instances with up to 6,468 buses and 9,000 transmission lines in under 5 minutes of running time. The method was also implemented and evaluated independently MISO, where it performed faster than MISO's existing constraint filtering methods, while producing solutions of same quality.

ACKNOWLEDGMENT

This work is supported by the U.S. Department of Energy under contract number DE-AC02-06CH11357.

REFERENCES

[1] F. Bouffard, F. D. Galiana, and J. M. Arroyo, "Umbrella contingencies in security-constrained optimal power flow," in *15th Power systems computation conference, PSCC*, vol. 5, 2005.

[2] 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, 2016.

[3] A. J. Ardakani and F. Bouffard, "Acceleration of umbrella constraint discovery in generation scheduling problems," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 2100–2109, 2015.

[4] D. A. Tejada-Arango, P. Sánchez-Martın, and A. Ramos, "Security constrained unit commitment using line outage distribution factors," *IEEE Transactions on Power Systems*, vol. 33, no. 1, pp. 329–337, 2018.

[5] G. Morales-España, J. M. Latorre, and A. Ramos, "Tight and compact MILP formulation for the thermal unit commitment problem," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4897–4908, 2013.

[6] 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.

[7] 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.

[8] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on power systems*, vol. 26, no. 1, pp. 12–19, 2011.

[9] A. Lodi and A. Tramontani, "Performance variability in mixed-integer programming," *TutORials in Operations Research: Theory Driven by Influential Applications*, pp. 1–12, 2013.