Effects of Tightening Unit-level and System-level Constraints in Unit Commitment

Bing Yan, *Member*, *IEEE*, Peter B. Luh, *Fellow*, *IEEE*, Eugene Litvinov, *Fellow*, *IEEE*, Tongxin Zheng, *Senior Member*, *IEEE*, Dane Schiro, *Member*, *IEEE*, Mikhail A. Bragin, *Member*, *IEEE*, Feng Zhao, *Member*, *IEEE*, Jinye Zhao, *Member*, *IEEE*, Izudin Lelic, *Member*, *IEEE*

***Abstract*—Unit Commitment is an important problem faced by independent system operators. It is usually formulated as a Mixed Binary Linear Programming (MBLP) problem, and is believed to be NP hard. To solve UC problems efficiently, an idea is through formulation tightening. If constraints can be transformed to directly delineate an MBLP problem's convex hull during data preprocessing, then the problem can be solved by using linear programming methods. The resulting formulation can be reused for other data sets, tremendously reducing computational requirements. To achieve the above goal, both unit- and system-level constraints are tightened with synergistic combination in this paper. Unit-level constraints are tightened based on existing cuts and novel "constraint-and-vertex conversion" and vertex projection processes. To tighten system-level constraints, selected cuts are applied and some potentially powerful cuts are identified. Numerical results demonstrate the effectiveness of tightening unit- and system-level constraints.**

***Index Terms*—Unit commitment, mixed binary linear programming, convex hull, formulation tightening.**

I. Introduction

Unit Commitment (UC) is an important problem faced by independent system operators. It is to minimize the total commitment and dispatch cost by committing appropriate units while satisfying system demand and other constraints [1]. The problem is usually formulated as a Mixed Binary Linear Programming (MBLP, with binary and continuous variables) problem, and is believed to be NP hard. For such difficult problems to be solved daily within short amount of time, the state-of-the-practice in the industry is to use commercial solvers generally based on the branch-and-cut method. For problems with a large number of units, the solvers may experience difficulties.

To solve the UC problem efficiently, most research focuses on solution methodology, and limited results have been reported on problem formulation. Formulation, however, is critically important since if constraints directly delineate an MBLP problem's convex hull (i.e., the formulation is "tight"), the problem can be solved by using linear programming (LP) methods without combinatorial difficulties. Formulation tightening, the process of transforming constraints to delineate the convex hull during data preprocessing, thus has a great potential. If it can be done, then the resulting formulation can be reused for other data sets, tremendously reducing computational requirements.

In this paper, both unit-level and system-level constraints are tightened. The problem, however, is challenging. For a given formulation, it is difficult to obtain the convex hull, even for a single-unit problem. Also it is hard to get tight constraints in generic forms in terms of unit parameters explicitly. If a specific unit is considered and tightened constraints are obtained in terms of numerical values, the tightened constraints obtained cannot apply to other units. As for system-level constraints such as system demand and reserve requirements, they only contain continuous variables, and cannot be tightened alone. Also there are a lot of candidates for existing cuts, and most of them are not helpful.

Section II of this paper presents a literature review, where a few tightened formulations for single-units were presented without explaining how they were generated. Very few studies were reported on tightening system-level constraints. In Section III, unit-level constraints are tightened without system-level constraints. The idea is first to apply data independent and easily implementable existing cuts based on constraint characteristics. More importantly, tightened constraints are established based on novel "constraint-and-vertex conversion" and vertex projection processes. For a problem with given unit parameters, the idea is to relax integrality requirements; generate vertices from constraints; project vertices onto the original convex hull; convert these vertices back to tightened constraints; and represent the numerical coefficients in terms of unit parameters. Tightened formulations for individual units can then be obtained through simple table lookup. To tighten system-level constraints, unit on/off variables and unit capacity constraints are incorporated. Selected cuts are applied and some potential powerful cuts (e.g., cover cuts for peak load hours) are identified.

In Section IV, two examples are presented. The first 5-bus problem is to illustrate the idea of tightening unit- and system-level constraints. The second IEEE 118-bus problem is to demonstrate the performance of tightening. Numerical results demonstrate effectiveness of tightening unit- and system-level constraints, and great potential for tightening MBLP problems.

This work was supported in part by the National Science Foundation under Grant ECCS- 1810108 and by a project funded by ISO New England.

Bing Yan, Peter B. Luh, and Mikhail A. Bragin are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269-4157, USA (e-mail: bing.yan@uconn.edu).

Eugene Litvinov, Tongxin Zheng, Feng Zhao, Jinye Zhao, Dane Schiro, and Izudin Lelic are with Business Architecture and Technology, ISO New England, Holyoke, MA, 01040-2841, USA.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of National Science Foundation or ISO New England.

978-1-7281-1981-6/19/$31.00 ©2019 IEEE

II. Literature Review

In the following, tightened unit-level and system-level formulations are reviewed. Our recent work is also presented.

Tightened unit-level formulations. In the literature, a few tightened formulations were presented without explaining how they were generated. Simplified UC with start-up and minimum up/down-time constraints was considered in [3]. New start-up/shut-down constraints were developed by analyzing a 7-hour formulation, and proved to be tight for the simplified UC problem. UC with capacity, ramp rate, and minimum up/down-time constraints was studied in [4], and new ramp rate constraints were established for two-hour problems. In [5], similar tightened constraints were presented for a three-hour problem. For a two/three-hour problem, combined minimum-up/down time and ramp rate constraints were established under different parameter settings in [6]. In [4-6], with specific assumptions on unit parameters, formulations were proved tight for short-period problems, and shown computationally efficient for overall UC problems.

Tightened system-level formulations. Very few studies were reported on tightening system-level constraints and the overall UC formulation. In [7], after projecting the power generation level onto [0, 1], the overall formulation without considering transmission capacity constraints was tightened iteratively online by using the lift-and-project method. Numerical results demonstrated the effectiveness and validity of the method.

Our recent work. In [8], a systematic approach was developed based on "constraint-and-vertex conversion" and "vertex projection" processes. The focus is on single units assuming system-wide constraints are relaxed. Innovative aspects also include elimination of dependence on initial conditions, and handling of different types of units. Numerical simulation demonstrates effectiveness of the approach.

III. Formulation Tightening

In Subsection A, a standard UC formulation is presented. Unit-level constraints are tightened by applying existing cuts and generating tightened constraints by "constraint-and-vertex conversion" and vertex projection processes as to be discussed in Subsection B. System-level constraints are then tightened by applying selected cuts in Subsection C.

A. UC formulation [1]

For unit k at node i, at each time t, major decision variables include unit on/off status x (binary), startup decision u (binary), and generation level p (continuous). Unit-level and system-level constraints are presented in first two subsubsections. The objective function is discussed in Subsubsection c.

a) Unit-level constraints

Unit-level constraints include generation capacity, offer price block, startup, ramp rate, minimum up/down-time, and reserve capability. Indices i and k are omitted for brevity.

1). Generation capacity: If a unit is online, its generation level should be within its minimum value P^{min} and maximum value P^{max}; otherwise, its generation level has to be zero, i.e

$$x(t)P^{\min} \le p(t) \le x(t)P^{\max}, \forall t. \quad (1)$$

2). Offer price block: Energy generation cost of a unit is usually a piecewise-linear function of p. To maintain linearity, a few offer price blocks are considered, where the price is a constant in each block (assume offer prices are monotonically non-decreasing). For each block, a new continuous decision variable p_b is needed, and their sum equals p, i.e.,

$$p_b(t) \le P_b^{\max}, \sum_b p_b(t) = p(t), \forall t, \quad (2)$$

where $P^{max}{}_b$ (MW) is the maximum generation of block b.

3). Ramp rate constraints: The generation level change between two consecutive hours cannot exceed hourly ramp rate R. Also it cannot exceed P^{min} plus 30-minute ramp upon starting up or at shutting down following standard industrial practice. The above is formulated in a linear way below,

$$p(t) - p(t-1) \le (R/2 - P^{\min})x(t-1) + (P^{\min} + R/2)x(t), \forall t,$$

$$p(t-1) - p(t) \le (P^{\min} + R/2)x(t-1) + (R/2 - P^{\min})x(t), \forall t. \quad (3)$$

4). Start up constraints: Binary startup variable u(t) equals 1 if the unit is turned on from offline at hour t, i.e.,

$$u(t) \ge x(t) - x(t-1). \quad (4)$$

5). Minimum up/down-time: The unit must remain online or offline for its minimum up or down time, respectively. The minimum up time is modeled as follows [9],

$$\sum_{\tau=1}^{T^{MOn}} x(\tau) = T^{MOn},$$

$$\sum_{\tau=t}^{t+T^{MU}-1} x(\tau) \ge T^{MU}\left(x(t) - x(t-1)\right), t \in [1+T^{MOn}, T - T^{MU} + 1],$$

$$\sum_{\tau=t}^{T}\left(x(\tau) - \left(x(t) - x(t-1)\right)\right) \ge 0, t \in [T - T^{MU} + 2, T], \quad (5)$$

In the above, T^{MU} denotes the minimum up time, and T^{MOn} denotes the number of hours the unit must be on at the beginning of the time horizon (assume given). The modeling of minimum down time is similar.

6). Reserve capability: To ensure system reliability under contingencies, ten-minute spinning reserve (TMSR) and thirty-minute operating reserve (TMOR) are considered. For TMSR, designation p^{TMSR} cannot exceed capability P^{TMSR} (calculated based on R) and is zero when the unit is off, i.e.,

$$p^{TMSR}(t) \le P^{TMSR}, p^{TMSR}(t) + p(t) \le P^{\max}x(t), \forall t. \quad (6)$$

The modeling of TMOR is similar, and the difference is that the unit can provide TMOR when it is off.

b) System-level constraints

System-level constraints include system demand, reserve requirement, and transmission capacity.

1). System demand: Total generation equals total demand, i.e.,

$$\sum_{i,k} p_{i,k}(t) = \sum_i P_i^D(t), \forall t. \quad (7)$$

where $P_i^D(t)$ is the demand of node i at time t.

2). Reserve requirement: The total TMSR of all units has to satisfy the requirement $P^{S,TMSR}$, i.e,

$$\sum_{i,k} p_{i,k}^{TMSR}(t) \ge P^{S,TMSR}(t), \forall t. \quad (8)$$

TMOR is modeled similarly.

3). Transmission capacity: For line l, DC power flow $f_l(t)$ is a linear combination of nodal injections $P_i(t)$ from all nodes weighted by generation shift factors $\alpha_{i,l}$, and it cannot exceed capacity f_l^{max} at every t, i.e.,

$$-f_l^{\max} \le f_l(t) \le f_l^{\max}, f_l(t) = \sum_i a_{i,l} P_i(t),$$
$$P_i(t) = \sum_k p_{i,k}(t) - P_i^D(t), \forall t. \quad (9)$$

c) Objective function

The total cost to be minimized is the commitment cost plus the dispatch cost, i.e.,

$$\sum_t \sum_i \sum_k \left(u_{i,k}(t) S_{i,k} + x_{i,k}(t) S_{i,k}^{NL} \right) + \sum_t \sum_i \sum_k \sum_b C_{i,k,b} p_{i,k,b}(t), \quad (10)$$

where $S_{i,k}$, $S_{i,k}^{NL}$, and $C_{i,k,b}$ are start-up, no-load, and generation costs, respectively.

The above UC problem (1)-(10) is an MBLP problem.

B. Tighten unit-level constraints

In this subsection, unit-level constraints are tightened by applying existing cuts and by "constraint-and-vertex conversion" and vertex projection processes.

a) Apply existing cuts

Based on the characteristics of unit-level constraints, implied bound and mixed integer rounding cuts are applied.

1) Implied bound cuts (IBC) [10]

Implied bound cuts reflect the relationship between binary and continuous variables when the binary imply bounds on the continuous. Consider the continuous offer price block variable p_b with an upper bound $P^{max}{}_b$, i.e., $p_b \le P^{max}{}_b$. In addition, the on/off variable x implies a new upper bound on p_b, e.g., $x = 0 \Rightarrow p_b \le 0$ (= 0). The idea is to lift x into $p_b \le P^{max}{}_b$ as follows,

$$p_b(t) + \left(P_b^{\max} - 0\right)\left(1 - x(t)\right) \le P_b^{\max}, \forall t. \quad (11)$$

Eq. (11) guarantees that when $x(t) = 1$, $p^b(t) \le P^{max}{}_b$; and when $x(t) = 0$, $p^b(t) \le 0$ (= 0).

Implied bound cuts can also be applied to the reserve capability constraints.

2) Mixed-integer rounding cuts (MIRC) [11]

Mixed-integer rounding cuts apply integer rounding on coefficients of integer variables and the constant of a constraint. Consider a constraint $z - y \le b$ with integer variable z and continuous variable y ($z \ge 0$, $y \ge 0$). After relaxing the integrality requirement on z, (b, 0) is a vertex of the convex hull of the LP-relaxation problem. If constant b is not an integer, the vertex is a non-integer vertex. To avoid this, a mixed-integer rounding cut that goes through the two points ($\lfloor b \rfloor$, 0) and ($\lfloor b \rfloor + 1$, $\lfloor b \rfloor + 1 - b$) ($\lfloor\ \rfloor$ is the floor function that outputs the greatest integer less than or equal to b) on the convex hull of the LP-relaxation problem is applied below,

$$z - \frac{1}{1 - \left(b - \lfloor b \rfloor\right)} y \le \lfloor b \rfloor. \quad (12)$$

For the UC problem, mixed integer rounding cuts can be applied to the offer price block constraints as follows,

$$\lfloor P^{\min} \rfloor x(t) - \sum_b p_b(t) \le 0, \forall t. \quad (13)$$

b) Establish tight constraints [8]

Tight constraints are established through four steps following our approach developed in [8]. For a unit with given parameters in numerical values, the first step is to relax integrality requirements, and generate vertices of the convex hull of the LP-relaxation problem from constraints (constraint-to-vertex conversion). The second step is to project this set of vertices onto the original convex hull. For MBLP problems, this projection is simply done by dropping vertices with fractional values for binary variables. The projected vertices are the vertices of the original convex hull. The third step is to convert those vertices back to tight constraints (vertex-to-constraint conversion). In the last step, to make those tight constraints reusable for other units, their numerical coefficients are represented in terms of unit parameters. It is done through analyzing these constraints and the relationship between numerical coefficients and unit parameters. To guarantee that resulting constraints are valid under all possible unit statuses (on, off, start-up or shut-down), the parameterization process may impose conditions on unit parameters (e.g., P^{min}, P^{max}; and R). A few sets of tightened constraints are thus developed based on unit parameters. Tightened formulations for individual units can then be obtained through simple table lookup. For practical applications, our goal is to obtain "near-tight" formulations by analyzing short time horizon problems, e.g., 3 hours.

C. Tighten system-level constraints

As mentioned earlier, system-level constraints only contain continuous variables (generation level and reserve), and cannot be tightened alone. To tighten them, unit on/off variables and unit capacity constraints are incorporated. Given these characteristics, cover and flow cover cuts are applied.

a) Cover cuts (CC) [10]

Cover cuts apply to constraints take the form of a knapsack constraint with a set of binary variables z shown as follows:

$$\sum_{j \in N} a_j z_j \le b, z_j \in \{0,1\}, a_j \ge 0, b \ge 0. \quad (14)$$

A minimal cover C is a subset of z such that if all the subset z are set to one, the knapsack constraint would be violated, but if any one subset z is excluded, the constraint would be satisfied. A cover cut is defined as,

$$\sum_{j \in C} z_j \le |C| - 1, C \subseteq N, \Delta \equiv \sum_{j \in C} a_j - b > 0, a_j \ge \Delta, \forall j \in C. \quad (15)$$

Here the cover cuts cannot be directly applied to system-level constants as they do not follow the structure of cover cuts. The idea of cover cuts can still be explored. For example, without considering other constraints, a cover U^{Cap} is a subset of units such that if all the subset units are set to be on with P^{max}, the demand can be covered (may exceed), but if any one subset unit is excluded, the demand cannot be satisfied. In practice, U^{Cap} can be obtained by sorting units based on their generation capacities in an ascending order during the data preprocessing stage, and $|U^{Cap}|$ represents the smallest number of units that is needed to satisfy demand. This type of "cover cuts" is defined as follows (t is omitted, and j is used as the unit index instead of (i, k) for brevity),

$$\sum_{j \in U} x_j \ge \left|U^{Cap}\right|, U^{Cap} \subseteq U, \Delta^{Cap} \equiv \sum_{j \in U^{Cap}} P_j^{\max} - \sum_i P_i^D > 0,$$
$$P_j^{\max} \ge \Delta^{Cap}, \forall j \in U^{Cap}. \quad (16)$$

In the above, U is the set of units.

The above cut can be also applied to TMSR with U^{TMSR} obtained by sorting units based on their TMSR capacities ascendingly. The total number of online units should be no less than the larger of $|U^{Cap}|$ and $|U^{TMSR}|$.

There is also another way to explore the idea of cover cuts. Consider $U^{TMSR'}$ as a subset of units. If all the subset units are set to be on with reserve capability as P^{TMSR}, the reserve requirement cannot be satisfied. Also there exists at least one unit in the remaining unit set $U\backslash U^{TMSR'}$: if the unit is added to $U^{TMSR'}$, the demand can be covered. Therefore at least one unit in $U\backslash U^{TMSR'}$ must be on, i.e.,

$$\sum_{j\in U\backslash U^{TMSR'}} x_j \geq 1, U^{TMSR'} \subseteq U, \Delta^{TMSR'} \equiv P^{S,TMSR} - \sum_{j\in U^{TMSR'}} P_j^{TMSR} > 0,$$
$$P_j^{TMSR} \geq \Delta^{TMSR'}, \exists j \in U \setminus U^{TMSR'}. \tag{17}$$

Here two types of $U^{TMSR'}$ can be obtained by sorting units based on their TMSR capacities ascendingly and descendingly.

b) Flow cover cuts (FCC) [12]

Flow cover cuts are for constraints that contain continuous variables with their lower and upper bounds depending on the values of binary variables. The idea is to treat such a constraint as a single node in a network where continuous variables are in/out-flows that can be on or off based on binary variables. Flows and demand imply a knapsack constraint to generate cuts on the flows. In UC, the TMSR capability constraints and reserve requirements can be rewritten as follow,

$$0x_j \leq p_j^{TMSR} \leq P_j^{TMR} x_j, -\sum_j p_j^{TMSR} \leq -P^{S,TMSR}. \tag{18}$$

In the above, the lower and upper bounds of continuous p^{TMSR} depend on binary x, and p^{TMSR} is the out-flow in the network. This fits the format of flow cover cuts, and one type of flow cover cuts established in [12] is defined as,

$$-\sum_{j\in U\backslash U^{TMSR'}} p_j^{TMSR} - \sum_{j\in U^{TMSR'}} P_j^{TMSR} x_j \leq -P^{S,TMSR}. \tag{19}$$

The above flow cover cuts and cover cuts in Subsubsection III-C-a work well for peak load hours.

In addition, the key idea of flow cover cuts can be used to filter transmission lines with large capacities. Without considering energy balance, the largest possible power flow for line l is calculated as follows,

$$f_l^{\max,NPB} = \sum_{i:a_{i,l}\geq 0} a_{i,l}\left(\sum_k P_{i,k}^{max} - \min_t P_i^D(t)\right)$$
$$+ \sum_{i:a_{i,l}<0} a_{i,l}\left(-\max_t P_i^D(t)\right), \forall l. \tag{20}$$

If $f_l^{max,NPB}$ is smaller than the line capacity f_l^{max} for every t, then the transmission capacity constraint is always satisfied at the positive flow direction, no matter how the units are committed and dispatched. The transmission capacity constraint for this line can be thus removed. The transmission constraints in the opposite flow direction can be handled in a similar way.

IV. Testing Results

The method developed above has been implemented by using IBM ILOG CPLEX Optimization Studio V 12.8.0.0 on a PC with 2.90GHz Intel Core(TM) i7 CPU and 16G RAM. Two examples are presented. The first small 5-bus problem illustrates the idea of tightening unit- and system-level constraints. The second IEEE 118-bus problem demonstrates the performance of our constraint tightening approach.

Example 1. 5-bus system: One hour

This example is based on a 5-bus system with 9 units. Unit parameters P^{min}, P^{max}, R, and P^{TMSR} are shown in Table I below. For the peak hour, total load is 1422 MW, and it is assumed that the TMSR requirement is 130 MW.

TABLE I 5-Bus System: Unit Parameters

	1	2	3	4	5	6	7	8	9
P^{min}	80	102	139	76	123	426	158	158	105.6
P^{max}	181	183	240	245	123	426	295	295	338
R	72.4	73.2	96	98	123	426	118	118	135.2
P^{TMSR}	12.07	12.2	16	16.33	20.5	71	19.67	19.67	22.53

To tighten unit-level constraints, consider a two-hour (t - 1, t) problem for unit 4. Capacity, ramp rate and start up constraints are considered as shown at the top left corner of Fig. 1 ($x(t$-1), $u(t$-1), $p(t$-1), $x(t)$, $u(t)$, $p(t)$ (the first three are initial conditions)): x_1-x_6). Other constraints are not considered for simplicity, and ranges for binary variables ($0 \leq x \leq 1$) are not presented for brevity. By constraint-to-vertex conversion, the vertices obtaind are shown at the bottom of Fig. 1 (only the last 10 vertices are shown). By keeping only binary vertices, the tight constraints that directly delineate the problem convex hull is shown at the top right corner of Fig.1.

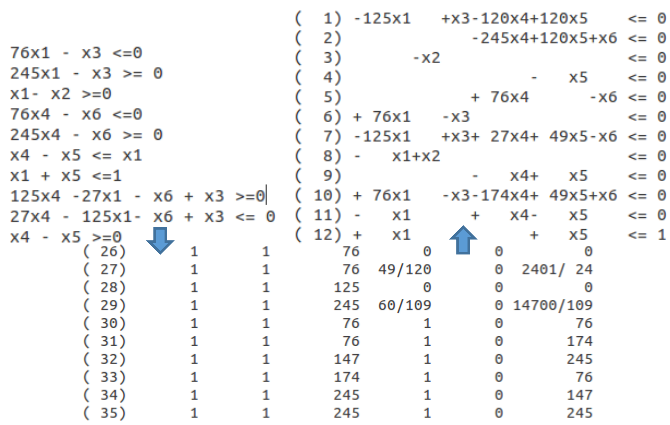

Figure 1: Constraints-and-vertex conversion

There are four new constraints (1, 2, 10, and 11) as compared to the original. Take (2) as an example, it can be generalized to the following constraint (24),

$$p(t) \leq P^{\max} x(t) - (P^{\max} - P^{\min} - R/2)u(t), \forall t. \tag{21}$$

In the above, when $x(t) = u(t) = 0$, $p(t) = 0$; when $x(t) = 1$ and $u(t) = 0$, it represent the maximum generation level; and when $x(t) = u(t) = 1$, it represents the maximum generation limit upon starting up. With Eq. (21), the right-hand side of capacity constraint Eq. (1) can be deleted.

To tighten system-level constraints, the units are sorted in an ascending order based on their generation capacities, and U^{Cap} = {6, 9, 7, 8, 4}. The order of units based on reserve happens to be the same: U^{TMSR} = {6, 9, 7, 8}; $U^{TMSR'}_1$ = {6, 9, 7}; and $U^{TMSR'}_2$ = {7, 8, 4, 3, 2, 1, 5}. Given these, the three cover cuts and two flow cover cuts are shown as follows,

$$\sum_{j\in U} x_j \geq 5, \sum_{j\in U\backslash U_1^{TMSR'}} x_j \geq 1, \sum_{j\in U\backslash U_2^{TMSR'}} x_j \geq 1,$$
$$-\sum_{j\in U\backslash U_m^{TMSR'}} p_j^{TMSR} - \sum_{j\in U_m^{TMSR'}} P_j^{TMSR} x_j \leq -P^{S,TMSR}, m = 1,2. \tag{22}$$

Example 2. IEEE 118-bus system: 96-hour

This example is based on the IEEE 118-bus system with 54 units. To test the performance of our unit-level tightened constraints, ramp rates of units are reduced by 2/3. The problem is solved by using branch-and-cut with: (1) standard formulations presented in Subsection III-A; (2) additional unit-level cuts and tightened constraints obtained in Subsection III-B; (3) additional system-level cuts and transmission constraint filter obtained in Subsection III-C; and (4) additional constraints from (2) and (3). The stop time is 600 seconds and the stop gap is 0.01%. Results are compared in Table II below. CPU time is the total time including data and model loading, problem solving and solution outputting time, and the solving time excludes loading and outputting time.

TABLE II IEEE 118-BUS SYSTEM: NOMINAL CASE

	(1): Standard formulation	(2): Tighten unit-level	(3): Tighten system-level	(4) Tighten both
CPU time (s)	120.82	17.38	65.17	17.32
Solving time (s)	112.3	11.98	55.69	11.88
Cost ($)	3,925,757	3,925,896	3,925,767	3,925,795
Gap (%)	0.01	0	0.01	0
# of IBC	4,428	0	5,154	0
# of MIRC	482	79	801	68
# of CC	255	2	82	6
# of FCC	3,170	1	3,289	1
# of other cuts	261	19	275	9

As shown in Table II, both CPU and solving time are reduced by tightening unit- and system-level constraints, while the former contributes much more than the latter. In addition, the numbers of the cuts are significantly reduced by tightening.

The problem is also solved with different reserve requirements. CPU and solving time are compared in Fig. 2 (Other = CPU - Solving), and the numbers of various cuts generated by the solver are show in Table III below.

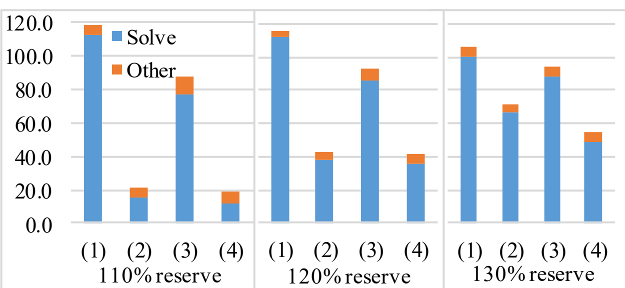

Figure 2: CPU and solving time under different reserve requirements

TABLE III NUMBER OF CUTS UNDER DIFFERENT RESERVE REQUIREMENTS

	110% Reserve				120% Reserve				130% Reserve			
	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)
IBC	4428	0	5410	0	5626	0	4857	0	5995	0	5188	0
MIRC	482	79	743	68	468	97	433	49	409	77	371	52
CC	255	2	186	6	155	2	235	5	166	167	182	138
FCC	3170	1	3284	1	3759	26	2862	10	4050	24	2677	7
Other	261	19	285	9	394	11	556	3	360	30	370	22

According to the results, CPU and solving time are dramatically reduced by constraint tightening. In addition, the numbers of different types of cuts are reduced. Especially, implied bound cuts are all gone, and the total number of flow cover cuts is reduced by more than 99%. Results demonstrate great potential of our approach to tighten MBLP problems.

To further test our approach, some stress cases are constructed by dropping the load by 50% to represent a high level of renewable penetration in the system. The results with different formulations are presented in Table IV.

TABLE IV IEEE 118-BUS SYSTEM: STRESS CASES

		CPU (s)	Solve (s)	Cost ($)	Gap (%)	# of cuts
100% reserve	(1)	605.97	600.27	1,923,691	0.11	9,909
	(4)	606.15	600.31	1,923,115	0.08	372
110% reserve	(1)	608.84	603.45	1,983,544	0.08	11,070
	(4)	605.82	600.34	1,983,053	0.08	390
120% reserve	(1)	608.71	603.25	2,049,100	0.08	10,947
	(4)	606.16	600.33	2,048,024	0.07	446
130% reserve	(1)	604.88	601.72	N/A	N/A	13,287
	(4)	606.21	600.38	2,139,220	0.21	473

Since the stop gap is not satisfied, all the problems are solved to the stop time, i.e., 10 minutes. According to Table IV, after tightening, better solutions are obtained within the same amount of time as compared to the standard formulation. Also the numbers of cut are significantly reduced.

V. CONCLUSION

This paper is a pioneering effort toward obtaining high quality UC solutions fast by synergistic combination of tightening unit- and system-level constraints. Unit-level constraints are tightened based on existing cuts and novel "constraint-and-vertex conversion" and vertex projection processes. To tighten system-level constraints, unit on/off variables and unit capacity constraints are incorporated. Selected cuts are applied and some potential powerful cuts are identified. Resulting formulations can be reused, tremendously reducing computational requirements. Numerical results demonstrate the effectiveness of tightening unit- and system-level constraints, and the great potential for tightening other complicated MBLP problems in power systems and beyond.

VI. REFERENCES

[1] A. J. Wood, and B. F. Wollenberg, *Power generation, operation, and control*, John Wiley & Sons, 2012.

[2] G. Morales-España, J. M. Latorre, A. Ramos, "Tight and compact MILP formulation for the thermal unit commitment problem," *IEEE T Power Syst.*, Vol. 28. no.4, pp.4897-4908, 2013.

[3] D. Rajan, and S. Tkriti, "Minimum up/down polytopes of the unit commitment problem with startup costs," Report RC23628, IBM, 2005.

[4] P. Damcı-Kurt, S. Küçükyavuz, D. Rajan, and A. Atamtürk, "A polyhedral study of production ramping," *Math. Program.*, Vol. 158, Issue 1, pp. 175-205, 2016.

[5] K. Pan, and Y. Guan, "A polyhedral study of the integrated minimum-up/-down time and ramping polytope," arXiv preprint arXiv:1604.02184, 2016.

[6] J. Ostrowski, M. F. Anjos, and A. Vannelli, "Tight mixed integer linear programming formulations for the unit commitment problem," *IEEE T Power Syst.*, Vol. 27. no.1, pp.39-46, 2012.

[7] L. Yang, J. Jian, Y. Zhu, and Z. Dong, "Tight relaxation method for unit commitment problem using reformulation and lift-and-project," *IEEE T Power Syst.*, Vol. 30. no.1, pp.13-23, 2015.

[8] B. Yan, P. B. Luh, E. Litvinov, T. Zheng. D. Schiro, M. A. Bragin, F. Zhao. J. Zhao, and I. Lelic, "A Systematical Approach to Tighten Unit Commitment Formulations," *2018 IEEE PES GM,* 2018.

[9] M. Carrión and J. M. Arroyo, "A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem," *IEEE T Power Syst.*, Vol. 21, no. 3, pp. 1371-1378, Aug. 2006.

[10] Ed Rothberg, "The CPLEX Library: Presolve and Cutting Planes."

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

[12] T. J. Van Roy and L. A. Wolsey, "Valid Inequalities for Mixed 0-1 Program," *Discrete. Appl. Math.*, Vol. 14, no. 2, pp. 199-213, 1986.