A novel optimization approach for sub-hourly unit commitment with large numbers of units and virtual transactions

Jianghua Wu, Student Member, IEEE, Peter B. Luh, Life Fellow, IEEE, Yonghong Chen, Senior Member, IEEE, Mikhail A. Bragin, Member, IEEE, Bing Yan, Member, IEEE

Abstract — Unit Commitment (UC) is an important problem in power system operations. It is traditionally scoped for 24 hours with one-hour time intervals. To improve system flexibility by accommodating the increasing net-load variability, sub-hourly UC has been suggested. Such a problem is larger and more complicated than hourly UC because of the increased number of periods and reduced unit ramping capabilities per period. The computational burden is further exacerbated for systems with large numbers of virtual transactions leading to dense transmission constraints matrices. Consequently, the state-of-the-art and practice method, branch-and-cut (B&C), suffers from poor performance. In this paper, our recent Surrogate Absolute-Value Lagrangian Relaxation (SAVLR) is enhanced by embedding ordinal-optimization concepts for a drastic reduction in subproblem solving time. Rather than formally solving subproblems by using B&C, subproblem solutions satisfying SAVLR’s convergence condition are obtained by modifying solutions from previous iterations or solving crude subproblems. All virtual transactions are included in each subproblem to reduce major changes in solutions across iterations. A parallel version is also developed to further reduce the computation time. Testing on MISO’s large cases demonstrates that our ordinal-optimization embedded approach obtains near-optimal solutions efficiently, is robust, and provides a new way of solving other MILP problems.

Index Terms -- Ordinal Optimization, Parallel Processing, Surrogate Absolute-Value Lagrangian Relaxation, Sub-hourly Unit Commitment

I. INTRODUCTION

UNIT Commitment (UC) is an important problem in power system operations – it identifies how to meet the system demand by committing units and deciding generation levels while minimizing the total cost of production subject to individual unit constraints and system-wide reserve and transmission capacity constraints. A UC problem is generally formulated as a Mixed-Integer Linear Programming (MILP) problem over a 24-hour horizon with one hour as the time interval. Increasing dynamics on the grid is prompting the industry to consider whether sub-hourly intervals in UC could increase efficiencies in managing the power system [1]. Sub-hourly UC with 15, 20, or 30 minutes time intervals has been suggested as a way to improve system flexibility and reliability because it can accommodate greater intra-hour net load variability [2][3][4]. It has also been evaluated for potential efficiencies under high levels of intermittent renewable penetration (e.g., solar and wind) by enabling quick response to uncertainties [5][6]. Sub-hourly UC, however, is much more complex than hourly UC because of 1) the increased number of periods leading to larger problem sizes, and 2) much-reduced unit ramping capabilities per period resulting in more complicated convex hulls (the smallest convex set enclosing all feasible solutions) as presented in Figure 9 of [7]. The computational burden is further exacerbated for systems with large numbers of virtual transactions which cause dense transmission capacity constraint matrices [8].

As will be reviewed in Section II, the state-of-the-art and practice branch-and-cut (B&C) [9] method for hourly UC problems cannot handle the increased complexity and suffers from poor performance. Lagrangian Relaxation (LR) [10][11] was one of the earlier methods to solve UC problems. It reduces complexity by relaxing coupling constraints and decomposing the relaxed problem into subproblems, which are coordinated by iteratively updating Lagrangian multipliers. Standard LR, however, has several major difficulties, such as high computational requirements, zigzagging of multipliers, and the need to adaptively guesstimate the unknown optimal dual value. These major difficulties have recently been overcome by the Surrogate Lagrangian Relaxation (SLR) method [12]. The convergence of SLR has also been accelerated by adding absolute-value penalties on constraint violation in our recent Surrogate Absolute-Value Lagrangian Relaxation (SAVLR) method [13]. Within SAVLR, MILP subproblems are normally solved by using B&C. Subproblem solving, however, may still be time-consuming for sub-hourly UC problems with large numbers of conventional units and virtual transactions.

In this paper, the formulation of sub-hourly UC, which is the same as that of hourly UC but with shorter time intervals, is briefly presented in Section III. System demand constraints and
reserve requirements should be strictly satisfied. Transmission capacity constraints, however, are modeled as “soft” and allowed to be violated with a predetermined penalty coefficient. Moreover, however, are obvious LR major improvements. Moreover, however, are obvious LR major improvements. Moreover, however, are obvious LR major improvements.

In Section IV, solution methodology is presented. To avoid introducing too many multipliers, only system demand constraints are relaxed. Soft transmission capacity constraints are not relaxed, and system reserve constraints are also converted to soft and are not relaxed following the approach of [14]. More importantly, inspired by the Ordinal Optimization (OO) concept that an “order” is easier to obtain than “values” and a problem with a softened goal is easier to solve than the original problem, a novel approach is developed by embedding the OO concepts within SAVLR. Specifically, rather than formally solving a subproblem by using B&C, “good enough” feasible subproblem solutions that satisfy the SAVLR convergence condition are obtained by modifying solutions from previous iterations or solving crude subproblems following OO concepts [15][16]. B&C is called to solve a subproblem only when such a good-enough solution cannot be obtained. This novel idea leads to a drastic reduction in CPU times because B&C is rarely called. Virtual transactions and dispatchable demand bids are included in all subproblems to reduce multiplier zigzagging and improve convergence. Finally, a parallel version is also developed to further reduce the CPU time.

In Section V, multiple Midcontinent ISO (MISO)’s “hard” cases, whose solutions are difficult to obtain within 20 or 30 minutes even for hourly UC by using B&C [8], are tested with 15 minutes as a time interval over a horizon of 36 hours. Compared with the original version of SAVLR that uses B&C to solve subproblems [13], the new method takes roughly the same number of iterations to obtain a solution with similar quality but much-reduced CPU times. Results demonstrate that our approach obtains high-quality solutions in a computationally efficient way, significantly outperforms existing methods, and is robust.

This manuscript is a major improvement over our preliminary results presented at the 2020 IEEE Power and Energy Society General Meeting [17]. Key enhancements include: 1) the specific use of OO concepts in SAVLR for sub-hourly UC is elaborated; 2) a parallel version is developed to further reduce the CPU time; 3) more MISO’s hard cases are tested and analyzed to demonstrate the performance and robustness of our methods; and 4) the reasons why B&C is rarely used to solve subproblems are examined.

II. LITERATURE REVIEW

In subsection II.A, the standard method B&C to solve hourly UC is presented. Decomposition and coordination approaches based on Lagrangian relaxation are reviewed in subsection II.B.

A. Branch-and-cut (B&C)

UC problems are generally formulated as MILP problems, and solved by using B&C [9]. For a given problem, the method applies “valid cuts” and tries to delineate the convex hull of feasible solutions. If the convex hull or its facets adjacent to the optimal solution are obtained, then the optimal solution can be quickly obtained by solving the corresponding Linear Programming (LP) problem. If the above is difficult to achieve as explained in Section I, then the method relies on time-consuming branch-and-bound. B&C is currently used to solve small sub-hourly UC problems with less than 100 units [4][6]. For large sub-hourly UC problems, however, direct application of B&C suffers from poor performance. In [18], solutions of hourly UC from B&C are used to provide initial solutions for MISO sub-hourly UC. Good solutions can then be obtained by a “polishing” method, which adaptively fixes binary and continuous variables while filtering out the constraints that are unlikely to be violated.

B. Decomposition and coordination approaches

Lagrangian Relaxation (LR) is a price-based decomposition and coordination method, and was one of the early methods to solve UC problems [10][11]. It reduces complexity by relaxing coupling constraints and decomposing the relaxed problem into subproblems, which are coordinated by iteratively updating Lagrangian multipliers with subgradient directions. The standard LR methods, however, have several major difficulties: 1) significant efforts to obtain a subgradient – requiring solving all subproblems optimally; 2) zigzagging of multipliers in view of the geometry of the dual function for MILP problems; and 3) the need to guessestimate the unknown optimal dual value.

All the major difficulties mentioned above have recently been overcome in the Surrogate Lagrangian relaxation (SLR) method [12]. SLR updates Lagrangian multipliers based on “surrogate” subgradients [19], which can be obtained by solving one or a few subproblems without requiring optimality as long as the “surrogate optimality condition” is satisfied. Therefore, SLR obtains smooth directions towards the optimal multipliers with much-reduced efforts as compared to that of standard LR. Moreover, unlike previous LR-based methods, SLR does not require the knowledge of the unknown optimal dual value for convergence proof as well as for practical implementations. Very recently, the convergence of SLR has been improved by introducing exact linearizable absolute-value terms to penalize the violations of relaxed system-wide constraints in the Surrogate Absolute-Value Lagrangian Relaxation (SAVLR) method [13]. Subproblems in SLR and SAVLR are generally solved as MILP problems by using B&C. This, however, may still take a long time for sub-hourly UC problems with large numbers of units and virtual transactions.

III. PROBLEM FORMULATION

This section presents the formulation of the sub-hourly UC problem with I conventional units, V virtual transactions, Y dispatchable demand bids, N nodes, Z types of reserves, and L transmission lines following [20]. The formulation is the same as that for the hourly UC, except that 15 minutes are used as the time interval over T periods (or 4T hours). Constraints include (1) system-wide demand, reserve, and transmission capacity constraints; (2) individual unit-level constraints, e.g., generation capacity and ramp-rate constraints for conventional units; and capacity constraints for virtual transactions and dispatchable demand bids as presented below.

Constraints

System Demand Constraints. The total generation from all resources should equal system demand at each period, i.e.,

\[ \sum_{j=1}^{I} p_{j,t} + \sum_{v=1}^{V} m_{v,t} - \sum_{j=1}^{V} d_{j,t} = \sum_{n=1}^{N} D_{n,t}, \forall t, \] (1)
where the generation level of unit $i$ ($1 \leq i \leq I$) at time $t$ ($1 \leq t \leq T$) is denoted as $p_{i,t}$, the generation level of virtual transaction $v$ ($1 \leq v \leq V$) at $t$ is denoted as $m_{v,t}$, and the power required by dispatchable demand bid $y$ ($1 \leq y \leq Y$) at $t$ is denoted as $d_{y,t}$. The system demand at node $n$ ($1 \leq n \leq N$) at $t$ is denoted as $D_{n,t}$.

**System Reserve Constraints.** To maintain reliability, the system is required to have certain levels of reserves following [21]. Three types of reserves, regulating, regulating plus spinning, and operating reserve constraints, are considered, indexed by $z = 1, 2$ and 3, respectively:

$$
\sum_{i=1}^{I} r_{i,z,t} \geq R_{z,t}, \forall z, \forall t. \tag{2}
$$

For unit $i$ at time $t$, the amount of reserve contribution of type $z$ is denoted as $r_{i,z,t}$. The required amount for reserve type $z$ at time $t$ is denoted as $R_{z,t}$.

**Transmission Capacity Constraints.** DC power flow is considered, and the flow in each line ($1 \leq l \leq L$), $f_{l,t}$, cannot exceed the line’s capacities at each period:

$$
f_{l,t} - \bar{s}_{l,t} \leq F_{l}, \forall t, \forall l, \tag{3}
$$

$$
f_{l,t} + \bar{s}_{l,t} \geq F_{l}, \forall t, \forall l, \tag{4}
$$

$$
f_{l,t} = \sum_{n=1}^{N} \alpha_{n,l} \left( \sum_{i:d_{i,t}} p_{i,t} + \sum_{v=0}^{V} m_{v,t} - \sum_{y=0}^{Y} d_{y,t} - D_{n,t} \right), \forall t, \forall l. \tag{5}
$$

In the above, transmission capacities of line $l$ are denoted as $F_{l}$ and $\bar{F}_{l}$; the sets of units, virtual transactions and dispatchable demand bids at bus $n$ are denoted as $I_{n}$, $V_{n}$, and $Y_{n}$, respectively. The generation shift factor $\alpha_{n,l}$ indicates the change of power flow through line $l$ with respect to a change in injection at bus $n$. Following [14], the above transmission capacity constraints are modeled as soft constraints, and are allowed to be violated by non-negative variables $\bar{s}_{l,t}$ and $s_{l,t}$ with a fixed penalty coefficient $\epsilon$ as will be seen in (11).

**Individual unit-level constraints.** Conventional units have unit capacity constraints, ramping constraints, three types (hot, intermediate and cold) of start-ups indexed by $s$ ($1 \leq s \leq S$) and shut-downs following [20]. Also, minimum up/down-time constraints follow Equations (1)-(2) in [22]; and unit reserve capacities for three types of reserves follow [23]. For certain units, the energy generated within 24 hours (or 96 periods when using 15 minutes as the time interval) is limited by maximum daily energy $TE_{i}$:

$$
\sum_{i=1}^{I} \frac{p_{i,t}}{4} \leq TE_{i}. \tag{6}
$$

Similarly, the maximum number of daily start-up may be limited by $TS_{i}$:

$$
\sum_{i=1}^{I} u_{s_{i,t}} \leq TS_{i}, \tag{7}
$$

where start-up status for unit $i$ at time $t$ is denoted as $u_{s_{i,t}}$.

Virtual transactions include virtual generations and virtual demands. They are subject to capacity constraints:

$$
0 \leq m_{v,t} \leq \bar{M}_{v,t}, v \in VG, \forall t, \tag{8}
$$

$$
M_{y,t} \leq m_{v,t} \leq 0, v \in VG, \forall t, \tag{9}
$$

where VG and VD are the sets of virtual generations and virtual demands, respectively. The generation level (or demand) of virtual transaction $v$ at time $t$ is denoted by $m_{v,t}$, and it is limited by $\bar{M}_{v,t}$ (or $M_{y,t}$). Similarly, dispatchable demand bid $y$ has a maximum limit $\bar{D}_{y,t}$ on its level $d_{y,t}$ at period $t$.

$$
0 \leq d_{y,t} \leq \bar{D}_{y,t}, y \in [1,Y], \forall t. \tag{10}
$$

Virtual transactions and dispatchable demand bids are related to continuous variables only and have linear costs $C_{x}^{m} m_{v,t}$ and $C_{y}^{v} d_{y,t}$, respectively.

**Objective Function**

The objective of a sub-hourly UC problem is to minimize the total cost including costs from conventional units (start-ups costs $C_{i,s,t}^{\text{Start}} b_{i,s,t}$) related to different types of start-ups $b$, no-load costs $C_{i,t}^{\text{NL}} x_{i,t}$ related to the commitment status $x$, piece-wise linear generation costs $C(p_{i,t})$, and linear reserve costs $C_{v}^{N}(m_{v,t})$; linear costs from virtual transactions $C_{y}^{V}(m_{v,t})$ and dispatchable demands $C_{y}^{y}(d_{y,t})$; and linear penalties with a fixed coefficient $\epsilon\tau$ for the violations of transmission capacity constraints, i.e.,

$$
\min_{\delta,\alpha,\rho,\tau,y} \left\{ \sum_{i=1}^{I} \sum_{t=1}^{T} \sum_{s=1}^{S} C_{i,s,t}^{\text{Start}} b_{i,s,t} + C_{i,t}^{\text{NL}} x_{i,t} + C^{E}(p_{i,t}) \right. \\
+ \sum_{z=1}^{Z} C_{i,z,t}^{\text{NL}} r_{i,z,t} + \left. \sum_{v=1}^{V} C_{v}^{V} m_{v,t} - \sum_{y=1}^{Y} C_{y}^{y} d_{y,t} \right\} + \epsilon\tau \sum_{l=1}^{L} \left( \bar{s}_{l,t} + s_{l,t} \right). \tag{11}
$$

The minimization is subject to system demand constraints (1), system reserve constraints (2), transmission capacity constraints (3)-(5), and all individual unit-level constraints. The overall problem is an MILP problem.

**IV. SOLUTION METHODOLOGY**

Subsection IV.A presents the key steps of decomposing the problem into subproblems based on SAVLR with a few major modifications. In subsection IV.B, ordinal optimization concepts are introduced to provide “good enough” feasible subproblem solutions by avoiding solving subproblems as MILP problems. In subsection IV.C, coordination of subproblem solutions, algorithm initialization, and finding feasible solutions are presented. Subsection IV.D presents a parallel version of the method to further reduce the CPU time.

**A. Problem decomposition**

This subsection presents the decomposition process based on SAVLR. The system-wide constraints are firstly relaxed or softened, and then the relaxed problem is decomposed into subproblems by properly grouping conventional units, virtual transactions and dispatchable demand bids.

**Relaxing or softening system-wide constraints**

Unlike the approach presented in [13], not all system-wide constraints are relaxed here. Instead, only system demand constraints are relaxed by using the Lagrangian multipliers $\{\lambda_{t}\}$, and their violations are penalized with the adjustable
penalty coefficient $c^D$ [13]. To avoid introducing additional multipliers, soft transmission capacity constraints are not relaxed, but are allowed to be violated with a fixed penalty coefficient $c^R$ following the approach of [14]. As for system reserve constraints, although they are modeled as hard constraints in (2), they are also treated as soft during the iterative process and are not relaxed to avoid introducing additional multipliers. Specifically, non-negative slack variables $\{g^{s^R}_{i,j,t}\}$ are introduced, and the original system reserve constraints (2) are softened as:

$$
\sum_{i=1}^{I} f_{i,z,t} + g^{s^R}_{i,z,t} \geq R_{z,t}, 1 \leq z \leq Z, \forall t.
$$

Here, an adjustable penalty coefficient $c^R$ is used to penalize positive values of slack variables $\{g^{s^R}_{i,j,t}\}$, slightly different from the approach of [14]. This is to emphasize feasibility by dynamically increasing $c^R$ when the original system reserve constraints are violated. At the last stage of the solution process to find feasible solutions, the original system reserve constraints (2) are required to be satisfied.

With the above, the relaxed problem is:

$$
\min_{\lambda, \delta, a, p, r, s, y, z, t} \left[ \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{t=1}^{T} C_{i,j,t}^h p_{i,j,t} + C_{i,j,t}^N x_{i,j,t} + C^E(\lambda_{i,j,t}) + \sum_{z=1}^{Z} \sum_{t=1}^{T} C_{i,j,t}^R \delta_{i,j,t} \right]
+ \sum_{z=1}^{Z} \sum_{t=1}^{T} C_{i,j,t}^s \delta_{i,j,t} - \sum_{i=1}^{I} \sum_{t=1}^{T} C_{i,t}^q \delta_{i,t} + c^D \sum_{i=1}^{I} \sum_{t=1}^{T} \sum_{l=1}^{L} \delta_{i,t,l} + \delta_{i,t} \right]
$$

where

$$
g(p, m, d) = \sum_{n=1}^{N} D_{n,t} - \sum_{i=1}^{I} p_{i,t} - \sum_{v=1}^{V} m_{v,t} + \sum_{y=1}^{Y} d_{y,t}, \forall t, \forall l.
$$

is the violation of demand constraints, and is penalized with the coefficient $c^D$. The relaxed problem is subject to transmission capacity constraints (3)-(5), softened system reserve constraints (12), and all individual unit-level constraints.

### Formulating subproblems

Following [13], conventional units in the relaxed problem (13) are divided into subproblems based on areas. Virtual transactions and dispatchable demand bids can also be divided into these subproblems based on their areas. This, however, will cause subproblem solutions to drastically change across iterations because virtual transactions and dispatchable demand bids do not have discrete decision variables and are only subject to simple bounds (8)-(10). Consequently, solutions of virtual transactions and dispatchable demand bids are sensitive to the values of Lagrangian multipliers. This, in turn, may cause significant changes of multipliers across iterations, resulting in slow convergence. Therefore, different from conventional units, all virtual transactions and dispatchable demand bids are included in every subproblem. The objective function of a subproblem is formed by collecting all the terms in (13) associated with decision variables belonging to that subproblem while fixing decision variables of conventional units belonging to other subproblems at their latest available values. For compactness of expression, subscripts “$j$” and “$f$” are used to indicate whether variables belong to subproblem $j$ or not. For example, $I_j$ is the set of conventional units belonging to subproblem $j$, and $L_j$ is the set of conventional units not belonging to subproblem $j$. The formulation of subproblem $j$ at iteration $k$ is thus as follows:

$$
\begin{align*}
\min_{\delta, a, p, r, s, y, z, t} & \left[ \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{t=1}^{T} C_{i,j,t}^h p_{i,j,t} + C_{i,j,t}^N x_{i,j,t} + C^E(\lambda_{i,j,t}) + \sum_{z=1}^{Z} \sum_{t=1}^{T} C_{i,j,t}^R \delta_{i,j,t} \right]
+ \sum_{z=1}^{Z} \sum_{t=1}^{T} C_{i,j,t}^s \delta_{i,j,t} - \sum_{i=1}^{I} \sum_{t=1}^{T} C_{i,t}^q \delta_{i,t} + c^D \sum_{i=1}^{I} \sum_{t=1}^{T} \sum_{l=1}^{L} \delta_{i,t,l} + \delta_{i,t} \right]

\text{subject to:} \\
\sum_{i=1}^{I} f_{i,z,t} + g^{s^R}_{i,z,t} \geq R_{z,t}, 1 \leq z \leq Z, \forall t.
\end{align*}
$$

where

$$
g(p, m, d, j) = \sum_{n=1}^{N} D_{n,t} - \sum_{i=1}^{I} p_{i,t} - \sum_{v=1}^{V} m_{v,t} + \sum_{y=1}^{Y} d_{y,t}, \forall t, \forall l.
$$

is the surrogate subgradient obtained after solving subproblem $j$. Non-negative variables $q^D_{t} - q^D_{t}$ are used to linearize the absolute-value penalty terms following a standard way explained on the page 63 of [24].

$$
q^D_{t} - q^D_{t} = g(p, m, d, j), \forall t, \forall l.
$$

For system reserve and transmission capacity constraints, the values of decision variables of conventional units belonging to other subproblems are fixed at their latest available values, and constraints (3)-(5), and (12) are rewritten as

$$
\tilde{f}_{i,t} - \tilde{g}_{i,t} \geq \tilde{f}_{i,t} - \tilde{g}_{i,t}, \forall t, \forall l, (18)
$$

$$
\tilde{f}_{i,t} + \tilde{g}_{i,t} \geq \tilde{f}_{i,t} + \tilde{g}_{i,t}, \forall t, \forall l, (19)
$$

$$
\tilde{f}_{i,t} = \sum_{n=1}^{N} \alpha_{n,t} \left( \sum_{i \in I(t)} p_{i,t} + \sum_{i \in I(t)} p_{i,t} + \sum_{v \in V} m_{v,t} - \sum_{y \in Y} d_{y,t} \right), \forall t, \forall l, (20)
$$

$$
\tilde{g}_{i,t} + \tilde{g}_{i,t} \geq \tilde{g}_{i,t} + \tilde{g}_{i,t}, \forall t, \forall l, (21)
$$

Subproblem (15) is subject to the updated system reserve and transmission capacity constraints (18)-(21), and all individual unit-level constraints. It is still an MILP problem.

### B. Quick searching process for good-enough feasible subproblem solutions

Within SAVL, subproblems are not required to be optimally solved. Rather, they should be sufficiently optimized to satisfy the following surrogate optimality condition to ensure convergence:

$$
\tilde{L}(\lambda^{x^k}, b^k, x^k, p^k, p^{k-1}, m^k, d^k) < \tilde{L}(\lambda^{x^k}, b^k, x^k, p^k, p^{k-1}, m^k, d^{k-1}),
$$

(22)
where \( \tilde{L}(\lambda^{s, k}, x^s, y^s, p^s, m^k, d^k) \) is the “surrogate dual value” at iteration \( k \) defined as

\[
\tilde{L}(\lambda^{s, k}, x^s, y^s, p^s, m^k, d^k) = \sum_{i,j} \sum_{l} C_{i,j,l} b_{i,j,l}^k + C_{i,j}^N x^s_{i,j} + C_E(p_i^s) + \sum_{y=1}^{\gamma} \sum_{t=1}^{\gamma} C_{i,j,L}^y m_{L} - \sum_{y=1}^{\gamma} \sum_{t=1}^{\gamma} C_{i,j,0}^y d_{y,t}(23)
\]

The right-hand side of (22) is defined similarly.

To satisfy the above surrogate optimality condition, subproblems are normally solved by using B&C. This is generally fine since subproblems are much smaller than the original problem. However, for large sub-hourly subproblems, e.g., MISO’s, B&C may suffer from poor performance. This difficulty is resolved by a novel exploitation of the Ordinal Optimization concepts as explained next.

Ordinal Optimization (OO) has been effectively used in computationally intensive simulation-based optimization, and has two major concepts [16]. The first is that “order” is easier to obtain than “value.” Take two objects A and B as an example, it is easier to know which object is heavier than to know the weights of A and B. Second, a problem with a softened goal is easier to solve than the original problem. For example, it is easier to obtain a solution that is within top 5% of all solutions than to obtain the optimal solution. OO thus uses crude models and quick simulation runs to roughly order solution candidates, and then select solutions that are good enough with high probabilities for further exploration. OO has recently been used to solve subproblems in generalized assignment problems [25].

Inspired by the OO concepts mentioned above, a novel idea to accelerate SAVLR is as follows. Rather than solving a subproblem by using an MILP method such as B&C, good-enough feasible solutions that satisfy subproblem constraints and the surrogate optimality condition (22) can be quickly obtained through “ordering” and checking solution candidates. These candidates can be derived, for example, by modifying solutions obtained in the previous iterations for feasibility using heuristics, e.g., neighborhood search [18]. They can also be obtained by solving a crude subproblem, e.g., an LP relaxed version, and then making solutions feasible using heuristics.

Solution candidates are then “ordered” in the ascending order of the surrogate dual value (23) to obtain good-enough feasible subproblem solutions through checking (22). These ways to obtain subproblem solutions are much more computationally efficient than by using B&C. Only when good-enough feasible subproblem solutions cannot be obtained, B&C is used. This approach is therefore much faster than solving subproblems exclusively by using B&C. Furthermore, with the convergence condition (22) satisfied at most iterations, the quality of the feasible solution obtained at the end is generally good – similar to that of solutions obtained by using B&C to solve subproblems, even though the quality of subproblem solutions may not be as good as that obtained by using B&C. This will be demonstrated in the next numerical testing section.

C. Coordination of subproblem solutions, initialization and finding feasible solutions

This subsection presents the coordination of subproblem solutions by updating multipliers and penalty coefficients; initialization of subproblem solutions, multipliers and penalty coefficients; and finding feasible solutions at the termination of iterative subproblem solving and multiplier updating process.

Updating multipliers and penalty coefficients

When the surrogate optimality condition (22) is satisfied as a strict inequality, multipliers \( \lambda^D \) are updated based on the surrogate subgradient (16) following (17)-(19) of [13], and the penalty coefficient \( c^D \) is updated based on (20) of [13]. When (22) is not met as a strict inequality but as an equality, the above updating process is skipped, and the next subproblem is solved. However, if (22) cannot be satisfied for all the subproblems within a major iteration (i.e., all subproblems are solved once), then the penalty coefficient \( c^D \) is deemed to be too large, and is reduced by following (21) of [13].

As mentioned in subsection IV.A, the penalty coefficient \( c_T \) is a fixed value, and the penalty coefficient \( c_R \) is dynamically increased to minimize the violation of original system reserve constraints. If any slack variable \( s_{i,j}^R \) is positive, \( c_R \) is increased by multiplying a constant \( \alpha (>1) \); and remains the same otherwise, i.e.,

\[
c_R = \begin{cases} c_R \times \alpha & \text{if } s_{i,j}^R \neq 0, \\ c_R & \text{otherwise}. \end{cases}
\]

Initializing subproblems solutions, multipliers and penalty coefficients

The initialization of SAVLR is also implemented by using the “good enough” concept. Before the iterative subproblem solving process, the hourly LP relaxed UC problem is solved. Its solution is rounded and duplicated as the initial subproblem solutions, which will be modified to provide good-enough feasible subproblem solution for the first major iteration as
presented in subsection IV.B. Lagrangian multipliers are initialized by using the results obtained from hourly LP relaxed UC problem as well. The initial penalty coefficients are set to be an order of magnitude higher than multiplier values.

**Finding feasible solutions**

The iterative subproblem solving and multiplier updating process terminates when stopping criteria are satisfied, e.g., the gap calculated against a lower bound is less than a certain percentage, or the time limit is reached, or each subproblem has been solved for a certain number of times. With system-level constraints relaxed or softened, subproblem solutions, when put together, may not satisfy the original constraints (1)-(10). A feasible solution is then constructed by using heuristics. For example, subproblem solutions are adjusted by using neighborhood search (e.g., the one embedded in Gurobi or CPLEX); or a portion of the binary variables is fixed at subproblem solution values, and the remaining decision variables are solved by using B&C. To measure the quality of a feasible solution, the best known lower bound obtained by using B&C in advance is used to calculate the optimality gap following (17) in [14].

The above approach synergistically incorporates SAVLR, Ordinal Optimization, and B&C (SAVLR+OO+B&C), and the flow chart is presented in Figure 1.

**D. Parallelization of the method**

To further reduce the CPU time, a parallel version of the approach is developed. The idea is to build subproblem models in parallel, and solve them in parallel at each iteration. Results from subproblems at each iteration are then merged to form a combined solution to update Lagrangian multipliers and penalty coefficients. There are, however, several difficulties. First, solving all subproblems in parallel at an iteration may lead to significant zigzagging of multipliers. This is precisely one of the major difficulties of the traditional LR: when all subproblems are solved, subgradient, rather than surrogate subgradient, are obtained. With “ridges” in the dual function, subgradient may change drastically across iterations, leading to multiplier zigzagging across ridges and slow convergence. Second, as explained in subsection IV.A, all virtual transactions and dispatchable demand bids are included in each subproblem. There are thus multiple values for each transaction or bid after solving subproblems in parallel. Which one should be used? Finally, even if each subproblem solved in parallel satisfies the surrogate optimality condition, the merged solution might not, leading to convergence difficulties.

To overcome the above-mentioned difficulties, the following steps are taken. First, a small subset of subproblems (10% to 40% based on testing experience) is solved in parallel in a round robin manner following the suggestion of [14]. Second, when merging subproblem results to form a combined solution, multiple ways are considered in sequence. For example, suppose that three subproblems are solved in parallel. A combined solution obtained by merging three subproblem results is first checked to see if the surrogate optimality condition is satisfied. If so, multipliers are updated, and the next batch of three subproblems are solved in parallel. If not, a combined solution obtained by merging any two subproblem solutions is checked to see if the surrogate optimality condition is satisfied, and the process repeats. If no merged solution satisfies the surrogate optimality condition, then the solution of the subproblem with the lowest surrogate dual value is selected to update multipliers and penalty coefficients. Third, values of virtual transactions and dispatchable demand bids are determined by solving an extra LP problem with all units’ variables fixed, except when no merged solution satisfies the surrogate optimality condition. The flow chart of the parallel version is presented in Figure 2.

**V. NUMERICAL TESTING**

Our methods, both the sequential and the parallel versions, have been implemented by using Gurobi 7.5.0 and Python 2.7. Testing has been performed on the HIPPO platform at MISO’s HPC cluster, and each node on the cluster has Intel Xeon @2.3GHz, 64GB RAM and 24 cores with Linux Redhat 6.6. Two examples of MISO’s UC problems are considered with 15 minutes as the time interval over 36 hours. Example 1 is used to demonstrate the performance of our new method. In Example 2, three additional MISO cases with different numbers of units and locations of virtual transactions are tested to demonstrate the robustness of our method. For both examples, high quality solutions are difficult to obtain by using B&C alone within 20 minutes (1200s) or 30 minutes (1800s) even with hourly time intervals.

**Example 1**

In this example, a MISO 15-min interval UC problem is considered. There are 1,105 conventional units, 15,843 virtual...
transactions, 75 dispatchable demand bids, and 227 transmission lines. Following the process in Section IV, the problem is decomposed into 10 subproblems, each with roughly 110 units and all virtual transactions and dispatchable demand bids. The initialization of our method is as follows: the average initial multiplier is $12.56/MW; the initial penalty coefficient is $125.6/MW; and the initial adjustable penalty coefficient for not meeting system reserve constraints is $500/MW with the growth rate a equal to 1.01. The problem is solved by using B&C, SAVLR+B&C (solving subproblems sequentially by using B&C), and both sequential and parallel versions of the new method. In the parallel version, 3 subproblems are solved in parallel in a round robin manner. The stopping criterion is set as 1% of the gap (calculated by the feasible solution cost and the best known lower bound obtained by B&C in advance).

| Methods          | Solving Time (s) | CPU Time (s) | Gap (%) |
|------------------|------------------|--------------|---------|
| B&C              | 5211             | 5443         | 0.90    |
| SAVLR +B&C       | 2985             | 4086         | 0.90    |
| SAVLR+OO+B&C     | 1484             | 3237         | 0.77    |
| SAVLR+OO+B&C (parallel) | 979             | 1639         | 0.84    |

Gaps are calculated by the obtained feasible solution costs and the best known lower bound obtained by B&C in advance.

Feasible solutions over time

Figure 3 compares the feasible solutions obtained by using different methods over time. In the testing, the new method (both sequential and parallel versions) obtains good feasible solutions after solving each subproblem only twice (or after two “major iterations”), same as that of the SAVLR+B&C. Within the new method, B&C was never called to solve a subproblem. Rather, good-enough feasible subproblem solutions are always obtained by using heuristics to modify existing solutions obtained from previous iterations (as part of the Gurobi “presolving” process). By doing this, the average time to obtain a good-enough feasible solution (both sequential and parallel versions) is 53s, which is much less than solving a subproblem by using B&C of 162s. Moreover, by applying parallelization, the overhead of model building and miscellaneous time is reduced from 1,753s to 660s, and the total solving time is reduced from 1,484s to 979s. Only the parallel version of our approach satisfies the stopping criterion of 1% gap within the required 1800s (i.e., 30 minutes). These results show that even though our subproblems are not solved by using B&C, our new methods, both sequential and parallel versions, obtain high quality overall solutions (within 1% of the gap) with the same number of major iterations as that of SAVLR+B&C but with much reduced CPU times. Our new method thus significantly outperforms B&C and SAVLR+B&C.

| Case  | # of units | # of virtual transactions | # of transmission constraints each interval |
|-------|------------|---------------------------|------------------------------------------|
| 1     | 1,109      | 16,504                    | 220                                      |
| 2     | 1,118      | 14,955                    | 226                                      |
| 3     | 1,102      | 14,704                    | 235                                      |

| Methods          | Solving Time (s) | CPU Time (s) | Gap (%) |
|------------------|------------------|--------------|---------|
| B&C              | 2548             | 3000         | 2.00    |
| Our approach (parallel) | 990             | 1409         | 1.10    |
| B&C              | 2787             | 3120         | 4.31    |
| Our approach (parallel) | 638             | 993          | 3.09    |
| B&C              | 3089             | 3600         | 76.00   |
| Our approach (parallel) | 619             | 1016         | 1.60    |

Gaps are calculated by using the feasible solution costs and the best known lower bound obtained by B&C in advance.

Example 2

To demonstrate the robustness of our methods, three additional cases roughly of the size of Example 1 but with different days of the MISO system are tested. Characteristics of test cases are summarized in Table II.

These three cases are solved by using B&C and the parallel version of our new method. Similar to that of Example 1, the problem is decomposed into 10 subproblems; 3 subproblems are solved in parallel by modifying existing solutions obtained.
in previous iterations using Gurobi’s heuristics. The initial values of multipliers and penalty coefficients are close to those of Example 1. With 2 major iterations as the stopping criterion for our method, and 3,600s (1 hour) as the time limit for B&C, the testing results are summarized in Table III.

As can be seen from Table III, feasible solutions obtained by using B&C have a gap of 2% after 3,000s for Case 1; a gap of 4.31% after 3,120s for Case 2; and a gap of 76% after 3,600s for Case 3. By using our method, feasible solutions have a gap of 1.10% after 1,409s for Case 1; a gap of 3.09% after 993s for Case 2; and a gap of 1.60% after 1,016s for Case 3. Similar to that of Example 1, B&C was never called to solve subproblems in Cases 2 and 3. For Case 1, B&C was used only twice to obtain good-enough feasible solutions at the beginning of iterations. The above results thus demonstrate that our new approach obtains near-optimal solutions in a computationally efficient manner for different sub-hourly UC cases, and significantly outperforms B&C.

VI. CONCLUSION

This paper presents a novel decomposition and coordination approach. Instead of formally solving subproblems by using MILP methods, the new approach obtains good-enough feasible subproblem solutions by modifying existing subproblem solutions or solving crude subproblems based on the OO concept. The new approach has high computational efficiency while requiring the same number of major iterations as that of SAVLR+B&C to obtain near-optimal solutions of similar quality. Our method thus represents a new optimization concept, and will have vital implications on solving other complicated MILP problems in power systems and beyond.

VII. ACKNOWLEDGMENT

The authors would like to express their sincere appreciation to Professor Yu-Chi Ho of Harvard University for his inspiration on combining mathematical optimization with Ordinal Optimization; and to Mr. Anbang Liu of Tsinghua University in Beijing for his work on combining mathematical optimization with OO to solve generalized assignment problems.

REFERENCES

[1] “FERC Order 764,” Tech. Rep. Accessed: May, 03, 2021. [Online]. Available: https://www.ferc.gov/sites/default/files/2020-05/F-01_70.pdf.

[2] J. Cochran, M. Miller, O. Zinaman, M. Milligan, D. Arent, B. Palminteri, M. O’Malley, S. Mueller, E. Lannoye, A. Tuohy, B. Kujala, M. Sommer, H. Holttinen, J. Kiviuluoma, and S. K. Soonee, “Flexibility in 21st Century Power Systems,” 21st Century Power Partnership, May 2014.

[3] G. Morales-Espona, L. Ramirez-Elizondo, and B. F. Hobbs, “Hidden power system inflexibilities imposed by traditional unit commitment formulations,” Applied Energy, Vol. 191, pp. 223–238, April 2017.

[4] S. Pineda, R. Fernández-Blanco, and J. Miguel Morales, “Time-adaptive unit commitment,” IEEE Transactions on Power Systems, Vol. 34, No. 5, pp. 354–359, May 2019.

[5] J. P. Deane, G. Drayton, and B. P. Ó Gallachóir, “The impact of sub-hourly modeling in power systems with significant levels of renewable generation,” Applied Energy, Vol. 113, pp. 152–158, January 2014.

[6] M. Kazemi, P. Siano, D. Sarno, and A. Goudarzi, “Evaluating the impact of sub-hourly unit commitment method on spinning reserve in presence of intermittent generators,” Energy, Vol. 113, pp. 338–354, October 2016.

[7] B. Yan, P. B. Luh, T. Zheng, D. Schiro, M. A. Bragin, F. Zhao, J. Zhao, and I. Lelic, “A Systematic Formulation Tightening Approach for Unit Commitment Problems,” IEEE Transactions on Power Systems, Vol. 35, No. 1, pp. 782–794, January 2020.

[8] 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, February 2016.

[9] J. E. Mitchell, “Branch-and-cut,” Wiley encyclopedia of operations research and management science. John Wiley & Sons, Inc. 2010.

[10] S. Virmani, E. C. Adrian, K. Imhof, and S. Mukherjee, “Implementation of a Lagrangian relaxation based unit commitment problem,” IEEE Transactions on Power Systems, Vol. 4, No. 4, pp. 1373–1380, November 1989.

[11] X. Guan, P. B. Luh, H. Yan, and J. A. Amalfi, “An optimization-based method for unit commitment,” Int. J. Elec. Power Energy System, Vol. 14, No. 1, pp. 9–17, February 1992.

[12] M. A. Bragin, P. B. Luh, J. H. Yan, N. Yu, and G. A. Stern, “Convergence of the surrogate Lagrangian relaxation method,” J. Optimization Theory and Applications, Vol. 164, No. 1, pp. 173-201, April 2015.

[13] M. A. Bragin, P. B. Luh, B. Yan, and X. Sun, “A scalable solution methodology for mixed-integer linear programming problems arising in automation,” IEEE Transactions on Automation Science and Engineering, Vol. 16, No. 2, pp. 531-541, April 2019.

[14] A. Ragunathan, M. A. Bragin, B. Yan, P. B. Luh, K. Moslehi, X. Feng, Y. Yu, C. Yu, C. Tsai, “Exploiting Soft Constraints within Decomposition and Coordination Methods for Sub-hourly Unit Commitment,” Submitted and available at TechRxiv. Accessed: May 03, 2021. Available: https://www.techrxiv.org/articles/preprint/Exploiting_Soft_Constraints_within_Decomposition_and_Coordination_Methods_for_Sub-hourly_Unit_Commitment/17985414.

[15] M. Deng and Y. C. Ho, “An ordinal optimization approach to optimal control problems,” Automatica, Vol. 35, pp. 331-338, February 1999.

[16] Y. C. Ho, Q. Zhao and Q. Jia, Ordinal Optimization: Soft optimization for hard problems, Springer US, 2007.

[17] J. Wu, P. B. Luh, Y. Chen, B. Yan, and M. A. Bragin, “A Decomposition and Coordination Approach for Large Sub-hourly Unit Commitment,” 2020 IEEE Power & Energy Society General Meeting (PESGM), Montreal, QC, August 2020.

[18] Y. Chen, F. Pan, J. Holzer, E. Rothberg, Y. Ma and A. Veeramany, “A High Performance Computing Based Market Economics Driven Neighborhood Search and Polishing Algorithm for Security Constrained Unit Commitment,” IEEE Transactions on Power Systems, Vol. 36, No. 1, pp. 292-302, Jan. 2021.

[19] X. Sun, P. B. Luh, M. A. Bragin, Y. Chen, J. Wan, 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, September 2018.

[20] G. Morales-Espona, 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, November 2013.

[21] NERC BAL-001-2. [Online]. Accessed: May, 03, 2021. Available: https://www.nerc.com/files/BAL-001-2.pdf.

[22] D. Rajan and S. Takriti, “Minimum up/down polynomials of the unit commitment problem with start-up costs,” IBM, 2005. [Online]. Accessed: May, 03, 2021. Available: http://domino.research.ibm.com/library/cyberdig.nsf/1e4115aea78b6e7c85256b360066f0d4/c8c02a7c809d89e8525702300502ac0?OpenDocument.

[23] Y. Chen, P. Gribik, and J. Gardner, “Incorporating post zonal reserve deployment transmission constraints into energy and ancillary service co-optimization,” IEEE Transactions on Power Systems, Vol. 29, No. 2, pp. 537–549, March 2013.

[24] J. Bisschop, AIMMS Optimization Modeling. Morrisville, NC, USA: Lulu Press, 2006

[25] A. B. Liu, P. B. Luh, M. A. Bragin and B. Yan, “Ordinal-Optimization Concept Enabled Decomposition and Coordination of Mixed-Integer Linear Programming Problems,” IEEE Robotics and Automation Letters, Vol. 5, No. 4, pp. 5051-5058, October 2020.