JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 105, No. 3, pp. 707–730, JUNE 2000

Solving the Unit Commitment Problem by a Unit Decommitment Method[1,2]

C. L. TSENG,[3] C. A. LI,[4] AND S. S. OREN[5]

Abstract. In this paper, we present a unified decommitment method to solve the unit commitment problem. This method starts with a solution having all available units online at all hours in the planning horizon and determines an optimal strategy for decommitting units one at a time. We show that the proposed method may be viewed as an approximate implementation of the Lagrangian relaxation approach and that the number of iterations is bounded by the number of units. Numerical tests suggest that the proposed method is a reliable, efficient, and robust approach for solving the unit commitment problem.

Key Words. Power system scheduling, unit commitment, unit decommitment, mixed-integer programming, Lagrangian relaxation, heuristic procedures.

1. Introduction

A problem that must be solved frequently by a power utility is to determine economically a schedule of what units will be used to meet the forecasted demand and operating constraints, such as spinning reserve requirements, over a short time horizon. This problem is commonly referred to as the unit commitment (UC) problem. The UC problem is a mixed-integer programming problem and is in the class of NP-hard problems (Ref. 1).

Because of its size and NP-hardness, the true optimal solution of the UC problem is normally difficult to obtain. Many optimization methods

[1]This paper is dedicated by Shmuel S. Oren to David G. Luenberger, his thesis adviser, inspiring teacher, and colleague who introduced him to the field of optimization.

[2]This work was supported in part by the National Science Foundation, Grant IRI-9120074.

[3]Assistant Professor, Department of Civil Engineering, University of Maryland, College Park, Maryland.

[4]Consultant, Pacific Gas and Electric Company, San Francisco, California.

[5]Professor, Department of Industrial Engineering and Operations Research, University of California at Berkeley, Berkeley, California.

have been proposed to solve the UC problem. For example, we mention the priority list method (Ref. 2), branch-and-bound methods (Refs. 3–5), dynamic programming approaches (Refs. 6–8), and Lagrangian relaxation (LR) methods (e.g. Refs. 9–11). For a detailed review, the reader is referred to Ref. 12. Among them, the LR methods are the most advanced and widely used approaches. Though popular, the LR approaches are known to require many heuristics which strongly influence their performance (Refs. 12–13).

There are also heuristic methods. For example, Lee develops in Ref. 14, a method which sequentially determines the commitment of the next most-advantageous unit to commit; the decision making involves a price adjustment, which resembles a bidding process. Also in Ref. 15, Li et al. proposed a method which mimics the LR approach; the multipliers are taken from the economic dispatch phase, rather than updated by the subgradient iteration. In Ref. 16, a unit decommitment (UD) method was developed as a postprocessing tool to improve the solution quality of the existing UC algorithms.

In this paper, we consolidate the approaches presented in Refs. 15–16 and extend them to a more general formulation. The proposed method is a unified unit decommitment method. This method starts with a solution having all available units online at all hours in the planning horizon and determines an optimal strategy for decommitting units, one at a time. We show that the proposed method may be viewed as an approximate implementation of the LR approach. The multiplier updating rule is similar to that in Ref. 15. Furthermore, we show that the number of iterations required by the method is bounded by the number of units. Empirical tests suggest that the proposed method is efficient and robust.

This paper is organized as follows. In Section 2, the UC problem is formulated. Section 3 presents some properties of economic dispatch. We generalize the UD method and propose an algorithm for solving the UC problem using the UD method in Section 4. The relation between the proposed method and the LR approach is discussed in Section 5. Finally, we generate random instances of UC problems and solve them by the proposed method. Numerical test results and conclusions are given in Section 6.

2. Problem Formulation

In this paper, the following standard notations will be used. Additional symbols will be introduced when necessary.

i = index for number of units, $i = 1, \ldots, I$;
t = index for time, $t = 0, \ldots, T$;

u_{it} = zero–one decision variable indicating whether unit i is up or down in time period t;
x_{it} = state variable indicating the length of time that unit i has been up or down in time period t;
t_i^{on} [t_i^{off}] = minimum number of periods unit i must remain on [off] after it has been turned on [off];
t_i^{cold} = number of periods required for the boiler of unit i to cool down;
p_{it} = state variable indicating the amount of power unit i is generating in time period t;
$p_i^{\min}$ [$p_i^{\max}$] = minimum [maximum] rated capacity of unit i;
$r_i^{\max}$ = maximum reserve for unit i;
$r_i(p_{it}) \equiv \min(r_i^{\max}, p_i^{\max} - p_{it})$, reserve available from unit i in time period t;
$C_i(p_{it})$ = fuel cost for operating unit i at output level p_{it} in time period t, assumed to be smooth, increasing, and strictly convex;
$S_i(x_{i,t-1}, u_{it}, u_{i,t-1})$ = startup cost associated with turning on unit i at the beginning of time period t;
D_t = forecast demand in time period t;
R_t = spinning reserve requirement in time period t.

The unit commitment (UC) problem is formulated as the following mixed-integer programming problem:

$$\min_{u,x,p} \sum_{t=1}^{T} \sum_{i=1}^{I} [C_i(p_{it})u_{it} + S_i(x_{i,t-1}, u_{it}, u_{i,t-1})], \tag{1}$$

subject to the demand constraints

$$\sum_{i=1}^{I} p_{it}u_{it} = D_t, \qquad t = 1, \dots, T, \tag{2}$$

and the spinning reserve constraints

$$\sum_{i=1}^{I} r_i(p_{it})u_{it} \geq R_t, \qquad t = 1, \dots, T, \tag{3}$$

where

$$r_i(p_{it}) \equiv \min(r_i^{\max}, p_i^{\max} - p_{it}).$$

There are other unit constraints such as the unit capacity constraints

$$p_i^{\min} \leq p_{it} \leq p_i^{\max}, \qquad i = 1, \dots, I \text{ and } t = 1, \dots, T, \tag{4}$$

the state transition equations for $i = 1, \dots, I$,

$$x_{it} = \begin{cases} \min(t_i^{\mathrm{on}}, \max(x_{i,t-1}, 0) + 1), & \text{if } u_{it} = 1, \\ \max(-t_i^{\mathrm{cold}}, \min(x_{i,t-1}, 0) - 1), & \text{if } u_{it} = 0, \end{cases} \tag{5}$$

the minimum up/down time constraints for $i = 1, \dots, I$,

$$u_{it} = \begin{cases} 1, & \text{if } 1 \le x_{i,t-1} < t_i^{\mathrm{on}}, \\ 0, & \text{if } -1 \ge x_{i,t-1} > -t_i^{\mathrm{off}}, \\ 0 \text{ or } 1, & \text{otherwise}, \end{cases} \tag{6}$$

and the initial conditions on x_{it} at $t = 0$ for $\forall i$.

In the objective function, we assume that the fuel cost C_i of a unit to be a smooth, increasing, and strictly convex function of the power output [MWh] of the unit. For each unit, the startup cost $S_i(x_{i,t-1}, u_{it}, u_{i,t-1})$ is an increasing function of the length of time that the unit has been off [i.e., $x_{i,t-1}$]. The state transition diagram is given in Fig. 1. To limit the size of

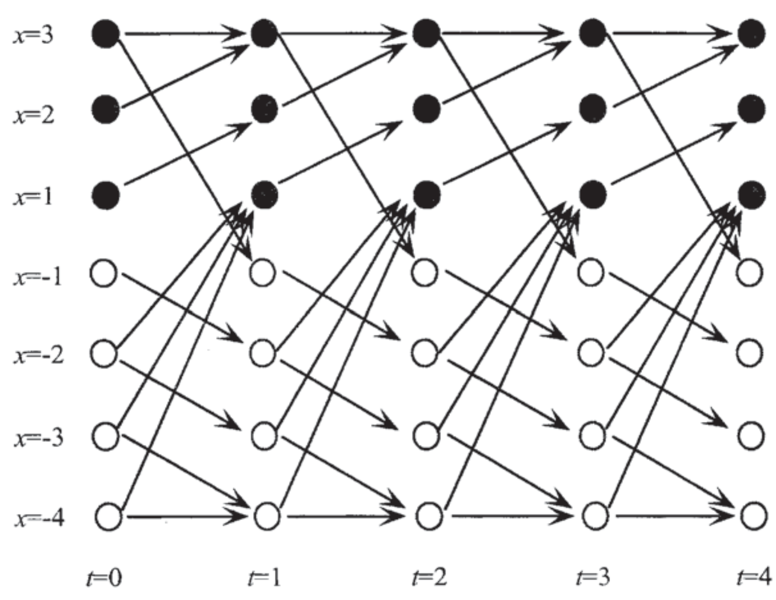

Fig. 1. Example of state transition diagram: $t^{\mathrm{on}} = 3$, $t^{\mathrm{off}} = 2$, $t^{\mathrm{cold}} = 4$.

the state space, we assume that the increase of $S_i(x_{i,t-1}, u_{it}, u_{i,t-1})$ is negligible when $x_{i,t-1} < -t_i^{\text{cold}}$, where $t_i^{\text{cold}} > t_i^{\text{off}}$ is the unit cold time. To further simplify notation, we let

$$S_i(u, t) = S_i(x_{i,t-1}, u_{it}, u_{i,t-1}).$$

3. Reserve-Constrained Economic Dispatch

Given a known commitment $\tilde{u} = \{\tilde{u}_{it}\}$ satisfying (5)–(6), the UC problem is reduced to a nonlinear program called economic dispatch (ED). ED is the problem of allocating system demand among all online generating units while satisfying (2)–(4) at any time over the planning horizon, i.e., to determine the corresponding $\tilde{p} = \{\tilde{p}_{it}\}$. In this paper, a tilde superscript denotes fixed realization of the corresponding variable. Since ED is separable in time, it can be solved sequentially by hour t. If the spinning reserve constraints (3) are not considered, at each time t the ED problem is a conventional resource allocation (RA) problem (e.g. Ref. 17), which has the following form:

$$\min \quad \sum_{i \in J} C_i(p_i), \tag{7a}$$

$$\text{s.t.} \quad \sum_{i \in J} p_i = D, \tag{7b}$$

$$p_i^{\min} \le p_i \le p_i^{\max}, \qquad \forall i \in J, \tag{7c}$$

where J is some unit index set. Optimality of such RA-type ED problem requires that all generators operate at a marginal cost that either equals same fixed value λ (Lagrange multiplier) or equals the marginal cost corresponding to the upper or lower bound of a generator output level, whichever is closer to λ. Let $\{\tilde{p}_i\}$ be the optimal dispatch; there exists a $\tilde{\lambda}$ such that

$$C_i'(\tilde{p}_i) = \tilde{\lambda}, \qquad \text{for } p_i^{\min} < \tilde{p}_i < p_i^{\max}, \tag{8a}$$

$$C_i'(\tilde{p}_i) \ge \tilde{\lambda}, \qquad \text{for } \tilde{p}_i = p_i^{\min}, \tag{8b}$$

$$C_i'(\tilde{p}_i) \le \tilde{\lambda}, \qquad \text{for } \tilde{p}_i = p_i^{\max}. \tag{8c}$$

This property is commonly referred to as the equal-lambda rule (e.g. Ref. 8). Equations (8) define also an equal-lambda mapping (also called a lambda iteration in this paper) from a given $\tilde{\lambda}$ to a unit generation denoted by $p_i(\tilde{\lambda})$,

$$p_i(\tilde{\lambda}) = \begin{cases} (C_i')^{-1}(\tilde{\lambda}), & \text{if } C_i'(p_i^{\min}) \le \tilde{\lambda} \le C_i'(p_i^{\max}), \\ p_i^{\min}, & \text{if } \tilde{\lambda} < C_i'(p_i^{\min}), \\ p_i^{\max}, & \text{if } \tilde{\lambda} > C_i'(p_i^{\max}). \end{cases} \tag{9}$$

Another approach for solving the RA-type ED problem is called the dual approach, which performs iterations on the lambda domain using the mapping (9) and terminates when the relation

$$\sum_{i \in J} p_i(\tilde{\lambda}) = D \tag{10}$$

is satisfied. In this paper, we will use the term "equal-lambda method" to refer to a method which solves the RA-type ED problem using the dual approach. An equal-lambda method can be implemented very efficiently such that it obtains the optimal solution within strongly polynomial time if the functions $C_i(\cdot)$ are quadratic convex (Ref. 18).

With the presence of the reserve constraints (3), the problem is a reserve-constrained economic dispatch (RCED). Methods for obtaining approximate solutions for RCED were proposed in Refs. 19–20. In this section, we state some mathematical properties of RCED. Define the index set of online units at time t with respect to this feasible commitment,

$$J(t;\tilde{u}) \equiv \{i \,|\, \tilde{u}_{it} = 1\}.$$

For simplicity, let

$$\tilde{J}_t = J(t;\tilde{u}).$$

The RCED problem in time t is denoted by

$$\text{(RCED)} \quad \text{rced}(\tilde{J}_t, t) \equiv \min \quad \sum_{i \in \tilde{J}_t} C_i(p_{it}), \tag{11a}$$

$$\text{s.t.} \quad \sum_{i \in \tilde{J}_t} p_{it} = D_t, \tag{11b}$$

$$\sum_{i \in \tilde{J}_t} r_{it}(p_{it}) \geq R_t, \tag{11c}$$

$$p_i^{\min} \leq p_{it} \leq p_i^{\max}, \qquad \forall i \in \tilde{J}_t. \tag{11d}$$

Also, assume that $\tilde{p} = \{\tilde{p}_{it}\}$ solves $\text{rced}(\tilde{J}_t, t)$, if the solution exists.

Proposition 3.1. The solution of RCED exists if and only if the following conditions hold:

$$\sum_{i \in \tilde{J}_t} p_i^{\min} \leq D_t \leq \sum_{i \in \tilde{J}_t} p_i^{\max} - R_t, \tag{12a}$$

$$\sum_{i \in \tilde{J}_t} r_i^{\max} \geq R_t. \tag{12b}$$

Proof. The "only if" part is obvious. To show the "if" part, note that (12b) implies that there exist $\{\tilde{r}_i\}$ such that

$$\sum_{i\in\tilde{J}_t} \tilde{r}_i = R_t, 0\le\tilde{r}_i\le r_i^{\max}, \qquad \forall i\in\tilde{J}_t.$$

Since

$$\sum_{i\in\tilde{J}_t} p_i^{\min}\le D_t\le \sum_{i\in\tilde{J}_t} p_i^{\max} - R_t = \sum_{i\in\tilde{J}_t} (p_i^{\max} - \tilde{r}_i),$$

there exist $\{\tilde{p}_{it}\}$ such that

$$\sum_{i\in\tilde{J}_t} \tilde{p}_{it} = D_t,$$

and

$$p_i^{\min}\le\tilde{p}_{it}\le p_i^{\max} - \tilde{r}_i, \qquad \forall i\in\tilde{J}_t.$$

Note that

$$\sum_{i\in\tilde{J}_t} r_i(\tilde{p}_{it})\ge R_t;$$

so, $\{\tilde{p}_{it}\}$ is a feasible solution for RCED. □

In the sequel, we assume always that the conditions stated in Proposition 3.1 are satisfied; therefore, the optimal solution of RCED exists for all t.

Proposition 3.2. Given a commitment $\tilde{u}$, assume that $\{\tilde{p}_{it}\}$ is an optimal solution of the corresponding RCED. Then, there exist $\tilde{\Omega}_t$ and $\tilde{\Lambda}_t$, two mutually exclusive and exhaustive subsets of $\tilde{J}_t$, i.e.,

$$\tilde{\Omega}_t\cup\tilde{\Lambda}_t = \tilde{J}_t \quad \text{and} \quad \tilde{\Omega}_t\cap\tilde{\Lambda}_t = \varnothing,$$

and Lagrange multipliers $\tilde{\lambda}_t$, $\tilde{\alpha}_t$, $\tilde{\mu}_t$,, $t = 1, \ldots, T$, such that, for $\forall i\in\tilde{\Omega}_t$,

$$C_i'(\tilde{p}_{it}) = \tilde{\alpha}_t, \qquad \text{for } p_i^{\max} - r_i^{\max} < \tilde{p}_{it} < p_i^{\max}, \tag{13a}$$

$$C_i'(\tilde{p}_{it})\le\tilde{\alpha}_t, \qquad \text{for } \tilde{p}_{it} = p_i^{\max}, \tag{13b}$$

and for $\forall i\in\tilde{\Lambda}_t$,

$$C_i'(\tilde{p}_{it}) = \tilde{\lambda}_t, \qquad \text{for } p_i^{\min} < \tilde{p}_{it} < p_i^{\max} - r_i^{\max}, \tag{14a}$$

$$C_i'(\tilde{p}_{it})\le\tilde{\lambda}_t, \qquad \text{for } \tilde{p}_{it} = p_i^{\max} - r_i^{\max}, \tag{14b}$$

$$C_i'(\tilde{p}_{it})\ge\tilde{\lambda}_t, \qquad \text{for } \tilde{p}_{it} = p_i^{\min}, \tag{14c}$$

with

$$\tilde{\mu}_t\left(\sum_{i\in\mathcal{J}_t}\min(p_i^{\max}-\tilde{p}_{it}, r_i^{\max})-R_t\right)=0, \tag{15}$$

$$\tilde{\mu}_t=\tilde{\lambda}_t-\tilde{\alpha}_t, \tag{16}$$

$$\tilde{\lambda}_t\geq 0, \qquad \tilde{\alpha}_t\geq 0, \qquad \tilde{\mu}_t\geq 0, \tag{17}$$

for $t=1,\dots,T$.

Proof. The Lagrangian of RCED can be expressed as

$$\begin{aligned}\mathscr{L}(p,\lambda,\mu,\xi,\zeta)&=\sum_{i\in\mathcal{J}_t}C_i(p_{it})+\lambda_t\left(D_t-\sum_{i\in\mathcal{J}_t}p_{it}\right)\\&\quad+\mu_t\left[R_t-\sum_{i\in\mathcal{J}_t}r_{it}(p_{it})\right]\\&\quad+\xi_t(p_{it}-p_i^{\max})+\zeta_t(p_i^{\min}-p_{it})\\&=\sum_{i\in\mathcal{J}_t}C_i(p_{it})+\lambda_t\left(D_t-\sum_{i\in\mathcal{J}_t}p_{it}\right)\\&\quad+\mu_t\left[R_t-\sum_{i\in\tilde{\Omega}_t}(p_i^{\max}-p_{it})-\sum_{i\in\tilde{\Lambda}_t}r_i^{\max}\right]\\&\quad+\xi_t(p_{it}-p_i^{\max})+\zeta_t(p_i^{\min}-p_{it}),\end{aligned} \tag{18}$$

where

$$\tilde{\Omega}_t=\{i\,|\,\tilde{p}_{it}>p_i^{\max}-r_i^{\max}\},$$
$$\tilde{\Lambda}_t=\{i\,|\,\tilde{p}_{it}\leq p_i^{\max}-r_i^{\max}\},$$

such that

$$\tilde{\Omega}_t\cup\tilde{\Lambda}_t=\tilde{J}_t.$$

Since $\{\tilde{p}_{it}\}$ is an optimal solution of RCED, there exist associated nonnegative multipliers $\tilde{\lambda}$, $\tilde{\mu}$, $\tilde{\xi}$, $\tilde{\zeta}$ such that

$$0=\partial\mathscr{L}/\partial p_{it}(\tilde{p},\tilde{\lambda},\tilde{\mu},\tilde{\xi});$$

hence,

$$0=\begin{cases}C_i'(\tilde{p}_{it})-\tilde{\lambda}_t+\tilde{\mu}_t+\tilde{\xi}_t-\zeta_t, & \text{if } i\in\tilde{\Omega}_t,\\ C_i'(\tilde{p}_{it})-\tilde{\lambda}_t+\tilde{\xi}_t-\tilde{\zeta}_t, & \text{if } i\in\tilde{\Lambda}_t.\end{cases} \tag{19}$$

Note that at least one of $\tilde{\xi}_t$ and $\tilde{\zeta}_t$ must equal zero, depending on whether $\tilde{p}_{it} \neq p_i^{\min}$ or $\tilde{p}_{it} \neq p_i^{\max}$ is met, respectively. Let

$$\tilde{\alpha}_t = \tilde{\lambda}_t - \tilde{\mu}_t;$$

thus, (19) reduces to (13) and (14). Because of the assumption that C_i is increasing, $\tilde{\alpha}_t \geq 0$. Equation (15) is the so-called complementary slackness condition. □

An intuitive way to interpret the optimality condition is to divide the units into two categories: $\tilde{\Lambda}_t$ is the set of units with cheap reserve but expensive generation, and $\tilde{\Omega}_t$ is the counterpart. Based on the result of Proposition 3.2, we have the following modified equal-lambda method for solving RCED.

Modified Equal-Lambda Method for Solving RCED. Initially, set lambda equal to zero; by (9), all units generate at their minimum rated capacity, i.e.,

$$\tilde{p}_{it} = p_i^{\min}, \qquad i \in \tilde{J}_t.$$

The system now is overreserved, but undergenerated. Gradually increasing lambda will increase system generation and simultaneously decrease system reserve. Let lambda be gradually increased until either the demand constraint (2) or the equality of the reserve constraint (3) is satisfied, whichever occurs first. At this point, assume that $\{\tilde{p}_{it}\}$ is the corresponding unit generation. There are two cases.

Case 1. Demand Constraint Is Met First. This corresponds to an overreserved system. Let

$$\tilde{\Omega}_t = \{i \mid \tilde{p}_{it} > p_i^{\max} - r_i^{\max}\}, \qquad \tilde{\Lambda}_t = \{i \mid \tilde{p}_{it} \leq p_i^{\max} - r_i^{\max}\}.$$

Denote the value of lambda by $\tilde{\lambda}_t$. By (15), $\tilde{\mu}_t = 0$, i.e., $\tilde{\alpha}_t = \tilde{\lambda}_t$. It can be seen easily that the optimality conditions stated in Proposition 3.2 are satisfied.

Case 2. Equality of Reserve Constraint Is Met First. Denote the value of lambda by $\tilde{\alpha}_t$. Let $\tilde{\Omega}_t$ and $\tilde{\Lambda}_t$ be defined as in Case 1. In order to induce more generation to satisfy the demand constraint while maintaining the reserve, the generations of units in $\tilde{\Omega}_t$ should remain fixed and only the generations of units in $\tilde{\Lambda}_t$ are increased by increasing lambda, until the demand constraint is satisfied. Note that, during the lambda iterations applied to units in $\tilde{\Lambda}_t$, the capacities of units in $\tilde{\Lambda}_t$ are further bounded from above by $p_i^{\max} - r_i^{\max}$ in order to maintain reserve. Denote the final value of

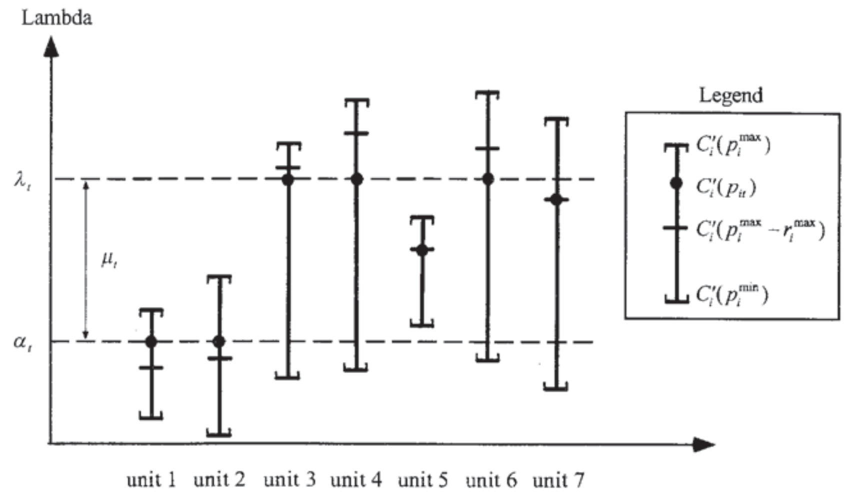

Fig. 2. Illustration of the modified equal lambda method, $\tilde{\Omega}_t = \{1, 2\}$.

lambda by $\tilde{\lambda}_t$. By definition,

$$\tilde{\mu}_t = \tilde{\lambda}_t - \tilde{\alpha}_t \geq 0. \qquad \square$$

In Fig. 2, an illustration of the modified equal-lambda method (Case 2) is given. Seven units are considered. For each unit, the p_{it} axis is suppressed and only the $C'(p_{it})$ axis is shown, since they are in one-to-one correspondence. When lambda is gradually increased from 0 to α_t, equality of the reserve constraint is met first. Units 1 and 2 belong to $\tilde{\Omega}_t$, and the other units are in $\tilde{\Lambda}_t$. For the units in $\tilde{\Lambda}_t$, the value of lambda continues to be raised until λ_t. Since the upper bound of the generation capacity of unit $i \in \tilde{\Lambda}_t$ has been reduced to $p_i^{\max} - r_i^{\max}$, though $\lambda_t > C_i'(p_i^{\max} - r_i^{\max})$, units 4 and 7 still generate at the level of $p_i^{\max} - r_i^{\max}$.

Proposition 3.3. The modified equal-lambda method guarantees to find a set of generation $\{\tilde{p}_{it}\}$, associated with multipliers $\{\tilde{\lambda}_t\}$, *and* $\{\tilde{\mu}_t\}$, satisfying the optimality condition stated in Proposition 3.2.

Proof. First, we need to show that, in Case 2, the lambda iterations would terminate with the demand constraint (2) satisfied. Assume on the contrary that this would not happen. It implies that

$$\begin{aligned} D_t &> \sum_{i\in\tilde{\Omega}_t} \tilde{p}_{it} + \sum_{i\in\tilde{\Lambda}_t} (p_i^{\max} - r_i^{\max}) \\ &\geq \sum_{i\in\tilde{\Omega}_t} (p_i^{\max} - r_i^{\max}) + \sum_{i\in\tilde{\Lambda}_t} (p_i^{\max} - r_i^{\max}) \\ &\geq \sum_{i\in\tilde{J}_t} p_i^{\max} - \sum_{\tilde{J}_t} r_i^{\max} \\ &\geq \sum_{i\in\tilde{J}_t} p_i^{\max} - R_t \qquad \text{[by (12b)].} \end{aligned} \tag{20}$$

Inequality (20) violates (12). This is a contradiction. Secondly, we need to show that the obtained $\{\tilde{p}_{it}\}$ and $\{\tilde{\lambda}_t\}$, $\{\tilde{\mu}_t\}$, $\{\tilde{\alpha}_t\}$ satisfy the optimality conditions in Proposition 3.2. This part is straightforward and is omitted. Since the RCED problem involves the minimization of a strictly convex function over a convex set, the solution obtained is the unique and global one. □

Note that, in the modified equal-lambda method, we do not emphasize implementation issues. We intend to interpret how one arrives at the optimality conditions. The purpose is to develop the proposition given in the following section.

3.1. Postoptimality Analysis. In Proposition 3.2, $\{\tilde{\lambda}_t\}$ and $\{\tilde{\mu}_t\}$ are the Lagrange multipliers for the constraints (2) and (3), respectively. Given that $\{\tilde{p}_{it}\}$ associated with $\{\tilde{\lambda}_t\}$ and $\{\tilde{\mu}_t\}$ solves rced$(\tilde{J}_t, t)$, suppose that $j\in\tilde{J}_t$. To know whether $\tilde{J}_t\backslash\{j\}$ is a more economic commitment in time t (ignoring other physical constraints, e.g., minimum uptime constraints at this point), one can either directly evaluate rced$(\tilde{J}_t\backslash\{j\}, t)$ or estimate the increased dispatch cost due to the decommitment of unit j in time t using the Lagrange multipliers $\{\tilde{\lambda}_t\}$ and $\{\tilde{\mu}_t\}$, and then compare with the saved cost $C_j(\tilde{p}_{jt})$. We investigate next the latter approach.

Proposition 3.4. Suppose that $\{\tilde{p}_{it}\}$ is the optimal solution to rced$(\tilde{J}_t, t)$ and $j\in\tilde{J}_t$. After decommitting unit j in time t, suppose that $\tilde{p}_{it}+\Delta\tilde{p}_{it}$, for $\forall i\in\tilde{J}_t\backslash\{j\}$, solve rced$(\tilde{J}_t\backslash\{j\}, t)$. If at time t, the system is not overreserved (i.e., the reserve constraint is achieved at equality), the following properties of $\{\Delta\tilde{p}_{it}\}$ are true:

(i) $\displaystyle\sum_{i\in\tilde{J}_t\backslash\{j\}} \Delta\tilde{p}_{it} = \tilde{p}_{jt}$;

(ii) $\Delta\tilde{p}_{it}\leq 0, \qquad \text{for } \forall i\in\tilde{\Omega}_t\backslash\{j\}$,

$$\Delta\tilde{p}_{it} \geq 0, \qquad \text{for } \forall i \in \tilde{\Lambda}_t \backslash \{j\};$$

(iii) $\tilde{p}_{it} + \Delta\tilde{p}_{it} = p_i^{\max} - r_i^{\max}$, for $i \in \tilde{\Lambda}'_t \backslash \tilde{\Lambda}_t = (\tilde{\Omega}_t \backslash \{j\}) \backslash \tilde{\Omega}'_t$;

(iv) $\sum_{i \in \tilde{\Omega}_t \backslash \{j\}} \Delta\tilde{p}_{it} + r_j(\tilde{p}_{jt}) = 0.$

Proof. Since the modified equal-lambda method presented in the previous section has been shown to be able to locate the unique solution of RCED, we shall use it to prove these properties. Property (i) is due to the load balance equation. Since it is assumed that the system is not over-reserved, applying the modified equal-lambda method to the system with unit j decommitted, Case 2 will continue to prevail. Let the primed variables (e.g. $\tilde{p}'_{it}$, $\tilde{\alpha}'_t$, $\tilde{\Lambda}'_t$, $\tilde{\Omega}'_t$) be the obvious corresponding notations in the decommitted system (e.g. $\tilde{p}'_{it} = \tilde{p}_{it} + \Delta\tilde{p}_{it}$). Because the decommitted system has less reserve capability, $\tilde{\alpha}'_t \leq \tilde{\alpha}_t$ and $\tilde{\Omega}'_t \subseteq \tilde{\Omega}_t$ in order to induce more reserve. Therefore, $\tilde{p}'_{it} \leq \tilde{p}_{it}$, i.e.

$$\Delta\tilde{p}_{it} \leq 0, \qquad \text{for } \forall i \in \tilde{\Omega}'_t;$$

and $\tilde{p}'_{it} \leq p_i^{\max} - r_i^{\max} \leq \tilde{p}_{it}$, i.e.

$$\Delta\tilde{p}_{it} \leq 0, \qquad \text{for } i \in \tilde{\Omega}_t \backslash \tilde{\Omega}'_t.$$

Similarly, to satisfy the load balance equation, $\tilde{\lambda}'_t \geq \tilde{\lambda}_t$ and $\tilde{\Lambda}_t \backslash \{j\} \subseteq \tilde{\Lambda}'_t$. So,

$$\Delta\tilde{p}_{it} \geq 0, \qquad \text{for } i \in \tilde{\Lambda}_t \backslash \{j\},$$

and (ii) is proved. Property (iii) follows immediately from the fact that

$$\tilde{\lambda}'_t \geq \tilde{\lambda}_t \geq \tilde{\alpha}_t \geq C'_i(p_i^{\max} - r_i^{\max}), \qquad \text{for } \forall i \in \tilde{\Lambda}'_t \backslash \tilde{\Lambda}_t = (\tilde{\Omega}_t \backslash \{j\}) \backslash \tilde{\Omega}'_t.$$

To prove property (iv), note that

$$\begin{aligned}
R_t &= r_j(\tilde{p}_{jt}) + \sum_{i \in \tilde{\Omega}_t \backslash \{j\}} (p_i^{\max} - \tilde{p}_{it}) + \sum_{i \in \tilde{\Lambda}_t \backslash \{j\}} r_i^{\max} \\
&= \sum_{i \in \tilde{\Omega}'_t} (p_i^{\max} - \tilde{p}'_{it}) + \sum_{i \in \tilde{\Lambda}'_t} r_i^{\max} \\
&= \sum_{i \in \tilde{\Omega}'_t} (p_i^{\max} - \tilde{p}'_{it}) + \sum_{i \in \tilde{\Lambda}'_t \backslash \tilde{\Lambda}_t} r_i^{\max} + \sum_{i \in \tilde{\Lambda}_t \backslash \{j\}} r_i^{\max} \\
&= \sum_{i \in \tilde{\Omega}'_t} (p_i^{\max} - \tilde{p}'_{it}) + \sum_{i \in (\tilde{\Omega}_t \backslash \{j\}) \backslash \tilde{\Omega}'_t} r_i^{\max} + \sum_{i \in \tilde{\Lambda}_t \backslash \{j\}} r_i^{\max} \\
&= \sum_{i \in \tilde{\Omega}'_t} (p_i^{\max} - \tilde{p}'_{it}) + \sum_{i \in (\tilde{\Omega}_t \backslash \{j\}) \backslash \tilde{\Omega}'_t} (p_i^{\max} - \tilde{p}'_{it}) + \sum_{i \in \tilde{\Lambda}_t \backslash \{j\}} r_i^{\max} \\
&= \sum_{i \in \tilde{\Omega}_t \backslash \{j\}} (p_i^{\max} - \tilde{p}'_{it}) + \sum_{i \in \tilde{\Lambda}_t \backslash \{j\}} r_i^{\max}.
\end{aligned} \tag{21}$$

Comparing the first and last lines of (21), property (iv) follows. □

Again, properties (ii) and (iv) can be interpreted intuitively as follows: when decommitting unit j, the units in $\tilde{\Omega}_t \backslash \{j\}$, those with expensive reserve, decrease generation to make up the loss of reserve originally provided by unit j, while the units in $\tilde{\Lambda}_t \backslash \{j\}$, those with expensive generation, increase generation to balance the load.

Since all the fuel cost functions C_i are assumed smooth and convex, we can estimate the increased cost due to the decommitment:

$$\begin{aligned}
&\sum_{i \in \tilde{J}_t \backslash \{j\}} C_i(\tilde{p}_{it} + \Delta\tilde{p}_{it}) - C_i(\tilde{p}_{it}) \\
&\approx \sum_{i \in \tilde{J}_t \backslash \{j\}} C_i'(\tilde{p}_{it}) \Delta\tilde{p}_{it} \\
&\approx \tilde{\lambda}_t \sum_{i \in \tilde{\Lambda}_t \backslash \{j\}} \Delta\tilde{p}_{it} + (\tilde{\lambda}_t - \tilde{\mu}_t) \sum_{i \in \tilde{\Omega}_t \backslash \{j\}} \Delta\tilde{p}_{it} \\
&= \tilde{\lambda}_t \tilde{p}_{jt} - \tilde{\mu}_t \sum_{i \in \tilde{\Omega}_t \backslash \{j\}} \Delta\tilde{p}_{it} \\
&= \tilde{\lambda}_t \tilde{p}_{jt} + \tilde{\mu}_t r_j(\tilde{p}_{jt}).
\end{aligned} \tag{22}$$

Note that the approximation in the third line of (22) uses the result from Proposition 3.2, and that (22) is due to property (iv) in Proposition 3.4. Although (22) is derived under the assumption that the system is not overreserved, it remains a good approximation when the system has excessive reserve capability. If the system is overreserved, Case 1 in the modified equal-lambda method will be encountered with $\tilde{\mu}_t = 0$. In this case, (22) reduces to the first-order approximation of the conventional ED problem (without reserve constraints). As will be shown in a later section, if the decommitment process is repeated for multiple units, one at a time, eventually Case 2 prevails, and obtains nonzero $\tilde{\mu}_t$. Also, it can be further shown that the two approximate relations ($\approx$) in (22), second and third lines, can be replaced by inequalities ($\geq$).

4. Solving the Unit Commitment Using the Unit Decommitment

4.1. Unit Decommitment Method. In Section 3.1, given a feasible and economically dispatched schedule $(\tilde{u}, \tilde{p})$, we discussed how to estimate the increased cost due to the decommitment of one unit. Now, we incorporate other physical constraints and present the problem of optimality improving a unit's schedule by decommitment. That is, in the time periods when the unit is already offline, it remains offline. The unit may be turned off in some online periods only if doing so can result in cost saving and does not cause problem infeasibility. While we are improving a unit's schedule, say unit j's schedule, by decommitment, other units' commitments are kept fixed. For example, $\tilde{u}_{it}$ remains unchanged for $\forall i \neq j$, $\forall t$, but $\tilde{p}_{it}$ are subject to change due to the commitment of unit j. The formulation is as follows:

$$(\mathrm{P}_j) \quad \min \sum_{t=1}^{T} [C_j(\tilde{p}_{jt})u_{jt} + (\tilde{\lambda}_t \tilde{p}_{jt} + \tilde{\mu}_t r_j(\tilde{p}_{jt}))(1 - u_{jt}) + S_j(u, t)], \tag{23}$$

subject to

$$u_{jt} = \begin{cases} 0, & \text{if } \tilde{u}_{jt} = 0, \\ 1, & \text{if } \tilde{u}_{jt} = 1 \text{ and the removal of } j \text{ from } \tilde{J}_t \\ & \quad \text{results in violation of (12)}, \\ 0 \text{ or } 1, & \text{otherwise}, \end{cases} \tag{24}$$

and subject to the minimum uptime, downtime constraints and the initial conditions for unit j.

Note that in, (23), u_{jt} and $1 - u_{jt}$ are two mutually exclusive decisions. If unit j is already online in time t ($u_{jt} = 1$), the generation cost is $C_j(\tilde{p}_{jt})$; if unit j is decommitted, the increased cost of the other units is approximated by $\tilde{\lambda}_t \tilde{p}_{jt} + \tilde{\mu}_t r_j(\tilde{p}_{jt})$. The startup cost of unit j is imposed whenever applicable. (P_j) is an integer programming problem and can be solved using the following dynamic programming recursive equations:

$$F(u_{jT}, x_{jT}) = 0, \tag{25a}$$

$$\begin{aligned} F(u_{jt}, x_{jt}) = \min_{(u_{j,t+1}, x_{j,t+1}) \in W_j} & [C_j(\tilde{p}_{jt})u_{jt} + (\tilde{\lambda}_t \tilde{p}_{jt} + \tilde{\mu}_t r_j(\tilde{p}_{jt}))(1 - u_{jt}) \\ & + S_j(u, t) + F(u_{j,t+1}, x_{j,t+1})], \\ & t = 0, \ldots, T-1, \end{aligned} \tag{25b}$$

where the decision space W_j is given by

$$\begin{aligned} W_j = &\{(1, x_+) \,|\, 1 \leq x_+ \leq t_j^{\mathrm{on}}, x_+ \in Z\} \\ &\cup \{(0, x_-) \,|\, -1 \geq x_- \geq -t_j^{\mathrm{cold}}, x_- \in Z\}. \end{aligned} \tag{26}$$

The optimal solution of (P_j) is obtained from the last step of the dynamic programming algorithm as $F(\tilde{u}_{j0}, \tilde{x}_{j0})$. In the above recurrence relation, $F(u_{jt}, x_{jt})$ is known as the cost-to-go and defines the optimal cost for the remaining t periods, $t = 0, \ldots, T-1$. In this paper, the solution of (P_j) is called the tentative commitment of unit j.

In the following algorithm, the superscript k denotes the kth iteration of the algorithm. Let $\tilde{\Theta}_i^k$, $i = 1, \ldots, I$, be the total generating cost (fuel cost and startup cost) of unit i of the feasible schedule $(\tilde{u}^k, \tilde{p}^k)$; and let Θ_i^k, $i = 1, \ldots, I$, be the optimal objective value of (P_i^k) solved with respect to the feasible solution $(\tilde{u}^k, \tilde{p}^k)$. We now state the decommitment algorithm.

Algorithm 4.1. UD Algorithm.

Data. Feasible solution $(\tilde{u}^0, \tilde{p}^0)$ is given.
Step 0. Set $k \leftarrow 0$, evaluate $\tilde{\Theta}_i^0$, $i = 1, \ldots, I$.
Step 1. Solve (P_i^k) with respect to $(\tilde{u}^k, \tilde{p}^k)$ and obtain Θ_i^k for all $i = 1, \ldots, I$.
Step 2. Select a unit m such that $\tilde{\Theta}_m^k - \Theta_m^k > 0$. If there is no such a unit, stop; otherwise, update the commitment of unit m in $\tilde{u}^k$ by the tentative commitment obtained in (P_m^k). The resultant unit commitment is assigned to be $\tilde{u}^{k+1}$.
Step 3. Perform RCED on $\tilde{u}^{k+1}$ to obtain $\tilde{p}^{k+1}$ and evaluate $\tilde{\Theta}_i^{k+1}$, the total generating cost of unit i, $i = 1, \ldots, I$.
Step 4. Set $k \leftarrow k+1$, go to Step 1. □

The algorithm in Step 2 chooses the tentative commitment of a unit which can yield savings to replace the original commitment. That is, the method corrects the unit commitment, one unit at a time. The following proposition states some properties of the algorithm[6].

Proposition 4.1. At time t, before unit i is decommitted, the following statements are true:

(i) If unit i is in $\tilde{\Lambda}_t$ at some iteration, it will remain in $\tilde{\Lambda}_t$ at subsequent iterations. If unit i is in $\tilde{\Omega}_t$, it may leave $\tilde{\Omega}_t$ to join $\tilde{\Lambda}_t$ at some iteration and then remains there afterward.

(ii) $\{\tilde{\lambda}_t^k\}\uparrow, \{\tilde{\mu}_t^k\}\uparrow, \{\tilde{\alpha}_t^k\}\downarrow$, as $k\uparrow$.

(iii) $\tilde{p}_{it}^{k+1} \geq \tilde{p}_{it}^k$, if $i \in \tilde{\Lambda}_t^k$;
$\tilde{p}_{it}^k \geq \tilde{p}_{it}^{k+1}$, if $i \in \tilde{\Omega}_t^k$ and $i \in \tilde{\Omega}_t^{k+1}$;
$\tilde{p}_{it}^{k+1} = p_i^{\max} - r_i^{\max}$, if $i \in \tilde{\Omega}_t^k$ and $i \in \tilde{\Lambda}_t^{k+1}$.

(iv) $\{C_i(\tilde{p}_{it}^k) - (\tilde{\lambda}_t^k \tilde{p}_{it}^k + \tilde{\mu}_t^k r_i(\tilde{p}_{it}^k))\}\downarrow$, as $k\uparrow$.

Proof. Statements (i) to (iii) are direct extensions of the properties shown in Proposition 3.4. To prove (iv), consider two cases:

[6]We use the notations ↑ and ↓ to represent nondecreasing and nonincreasing sequences, respectively.

Case A. $i \in \tilde{\Lambda}_t^k$. In this case,

$$C_i(\tilde{p}_{it}^k) - [\tilde{\lambda}_t^k \tilde{p}_{it}^k + \tilde{\mu}_t^k r_i(\tilde{p}_{it}^k)] = C_i(\tilde{p}_{it}^k) - \tilde{\lambda}_t^k \tilde{p}_{it}^k - \tilde{\mu}_t^k r_i^{\max}. \tag{27a}$$

We shall show that the first two terms $\{C_i(\tilde{p}_{it}^k) - \tilde{\lambda}_t^k \tilde{p}_{it}^k\}\downarrow$, since the third term $\{-\tilde{\mu}_t^k r_i^{\max}\}\downarrow$, as $k\uparrow$. By convexity of $C_i(\cdot)$, $\{C_i(\tilde{p}_{it}^k) - C_i'(\tilde{p}_{it}^k)\tilde{p}_{it}^k\}\downarrow$ as $k\uparrow$, since $\{\tilde{p}_{it}^k\}\uparrow$ as $k\uparrow$ from (iii). This would be the case when

$$C_i'(p_i^{\min}) \le \tilde{\lambda}_t^k = C_i'(\tilde{p}_{it}^k) \le C_i'(p_i^{\max} - r_i^{\max}).$$

When $\tilde{\lambda}_t^k > C_i'(p_i^{\max} - r_i^{\max})$ [or $< C_i'(p_i^{\min})$],

$$C_i(\tilde{p}_{it}^k) - \tilde{\lambda}_t^k \tilde{p}_{it}^k = C_i(p_i^{\max} - r_i^{\max}) - \tilde{\lambda}_t^k(p_i^{\max} - r_i^{\max})$$

[or $C_i(p_i^{\min}) - \tilde{\lambda}_t^k(p_i^{\min})]\downarrow$, as $k\uparrow$ since $\{\tilde{\lambda}_t^k\}\uparrow$.

Case B. $i \in \tilde{\Omega}_t^k$. In this case,

$$C_i(\tilde{p}_{it}^k) - [\tilde{\lambda}_t^k \tilde{p}_{it}^k + \tilde{\mu}_t^k r_i(\tilde{p}_{it}^k)] = C_i(\tilde{p}_{it}^k) - (\tilde{\alpha}_t^k \tilde{p}_{it}^k + \tilde{\mu}_t^k p_i^{\max}) \tag{27b}$$

and $\{\tilde{p}_{it}^k\}\downarrow$, as $k\uparrow$. We shall show that $\{\tilde{\alpha}_t^k \tilde{p}_{it}^k + \tilde{\mu}_t^k p_i^{\max}\}\uparrow$, as $k\uparrow$. This is obvious because $p_i^{\max} > \tilde{p}_{it}^k$ and the increase of $\tilde{\mu}_t$ between two consecutive iterations is greater than the decrease amount of $\tilde{\alpha}_t$. This is true even at the iteration that i switches from $\tilde{\Omega}_t$ to $\tilde{\Lambda}_t$. □

Theorem 4.1. The UD algorithm terminates within I iterations, where I is the number of units.

Proof. We shall show that, once a unit has been selected in Step 2 in the UD algorithm, it will not be selected again in Step 2 at any subsequent iteration. Therefore, the UD algorithm terminates within I iterations. Suppose that unit j is selected at iteration k', and its tentative commitment is

$$\{\hat{u}_{jt}^{k'}\} = \{\tilde{u}_{jt}^{k'+1}\},$$

so that

$$\sum_{t=1}^{T} \{[C_j(\tilde{p}_{jt}^{k'}) + S_j(\hat{u}^{k'}, t)]\hat{u}_{jt}^{k'} + [\tilde{\lambda}_t^{k'} \tilde{p}_{jt}^{k'} + \tilde{\mu}_t^{k'} r_j(\tilde{p}_{jt}^{k'})](1 - \hat{u}_{jt}^{k'})\}$$
$$\le \sum_{t=1}^{T} \{[C_j(\tilde{p}_{jt}^{k'}) + S_j(u, t)]u_{jt} + [\tilde{\lambda}_t^{k'} \tilde{p}_{jt}^{k'} + \tilde{\mu}_t^{k'} r_j(\tilde{p}_{jt}^{k'})](1 - u_{jt})\}, \tag{28}$$

for any $\{u_{jt}\}$ satisfying (24).

On the contrary, assume that unit j is selected again in Step 2 of the UD algorithm for the first time at iteration $k'' > k'$ with the tentative commitment

$$\{\hat{u}_{jt}^{k''}\} = \{\tilde{u}_{jt}^{k''+1}\}.$$

Then,

$$\sum_{t=1}^{T} \{[C_j(\tilde{p}_{jt}^{k''}) + S_j(\hat{u}^{k''}, t)]\hat{u}_{jt}^{k''} + [\tilde{\lambda}_t^{k''}\tilde{p}_{jt}^{k''} + \tilde{\mu}_t^{k''} r_j(\tilde{p}_{jt}^{k''})](1 - \hat{u}_{jt}^{k''})\}$$

$$< \sum_{t=1}^{T} \{[C_j(\tilde{p}_{jt}^{k''}) + S_j(\hat{u}^{k'}, t)]\hat{u}_{jt}^{k'} + [\tilde{\lambda}_t^{k''}\tilde{p}_{jt}^{k''} + \tilde{\mu}_t^{k''} r_j(\tilde{p}_{jt}^{k''})](1 - \hat{u}_{jt}^{k'})\}. \tag{29}$$

Note that, in the right-hand side of (29), the commitment is the one at iteration k' because unit j has not been selected again until iteration k'', but the dispatch is updated at every iteration. Let

$$\Gamma = \{t \,|\, \hat{u}_{jt}^{k'} \neq \hat{u}_{jt}^{k''}, t = 1, \ldots, T\};$$

i.e.,

$$\hat{u}_{jt}^{k'} = 1, \text{ but } \hat{u}_{jt}^{k''} = 0, \qquad \text{for } \forall t \in \Gamma,$$

since the algorithm only does decommitment. With

$$u_{jt} = \hat{u}_{jt}^{k''}$$

substituted into (28), we have

$$\sum_{t\in\Gamma} C_j(\tilde{p}_{jt}^{k'}) \leq \sum_{t\in\Gamma} [\tilde{\lambda}_t^{k'}\tilde{p}_{jt}^{k'} + \tilde{\mu}_t^{k'} r_j(\tilde{p}_{jt}^{k'})] + \Delta S_j, \tag{30}$$

where

$$\Delta S_j = \sum_{t=1}^{T} (S_j(\hat{u}^{k''}, t)\hat{u}_{jt}^{k''} - S_j(\hat{u}^{k'}, t)\hat{u}_{jt}^{k'})$$

is a constant. Arranging (29) in a similar way, we have

$$\sum_{t\in\Gamma} C_j(\tilde{p}_{jt}^{k''}) > \sum_{t\in\Gamma} [\lambda_t^{k''}\tilde{p}_{jt}^{k''} + \tilde{\mu}_t^{k''} r_j(\tilde{p}_{jt}^{k''})] + \Delta S_j. \tag{31}$$

From Proposition 4.1(iv), that $\{C_i(\tilde{p}_{it}^k) - (\tilde{\lambda}_t^k\tilde{p}_{it}^k + \tilde{\mu}_t^k r_i(\tilde{p}_{it}^k))\}\downarrow$, as $k\uparrow$, and from (30), we have that

$$\sum_{t\in\Gamma} C_j(\tilde{p}_{jt}^{k''}) \leq \sum_{t\in\Gamma} [\tilde{\lambda}_t^{k''}\tilde{p}_{jt}^{k''} + \tilde{\mu}_t^{k''} r_j(\tilde{p}_{jt}^{k''})] + \Delta S_j, \tag{32}$$

because $k'' > k'$. This contradicts (31); unit j should not be selected again. Therefore, the number of iterations of the UD algorithm is bounded by the number of units. □

To prove Theorem 4.1, only the property in Proposition 4.1(iv) is needed. It does not depend on any selection rule in Step 2 of the algorithm. The term $C_i(\tilde{p}_{it}^k) - [\tilde{\lambda}_t^k\tilde{p}_{it}^k + \tilde{\mu}_t^k r_i(\tilde{p}_{it}^k)]$ can be interpreted as the net profit of decommitting unit i at time t, which decreases as the iteration proceeds.

When a unit, say j, is selected in Step 2 to improve its commitment (by decommitting it at some hours, with the other units commitments fixed) at some iteration, the obtained tentative commitment for unit j will remain optimal for unit j at subsequent iterations, since decommitting unit j becomes less and less attractive as the iteration proceeds. Should there be a better (de)commitment for unit j at a future iteration, this commitment must have been obtained earlier. Therefore, the algorithm terminates within I iterations.

4.2. Unified UD Algorithm for Solving UC. The UD method was originally proposed in Ref. 16 to serve as a postprocessing tool to improve the solution quality for existing UC methods. Therefore, it starts with an initial feasible solution of the UC problem. In this section, we shall investigate the possibility of solving UC by means of UD. Initially as many units as possible are turned on in all hours without violating the minimum uptime and downtime constraints. A schematic algorithm for implementing the outlined procedure is given below.

Algorithm 4.2. Unified UD Algorithm for Solving UC.

Data. $\tilde{u}_{i0}$ and $\tilde{x}_{i0}$ are given for $\forall i$.

Step 0. Set $i \leftarrow 1$.

Step 1. If $i > I$, go to Step 4 and $\tilde{u}$ is the initial commitment. Otherwise, set $t \leftarrow 1$ and go to Step 2.

Step 2. If $t > T$, then set $i \leftarrow i+1$ and go to Step 1. Otherwise, set

$$\tilde{u}_{it} = \begin{cases} 0, & \text{if } -1 \geq \tilde{x}_{i,t-1} > -t_i^{\text{off}}, \\ 1, & \text{otherwise}, \end{cases} \tag{33}$$

$$\tilde{x}_{it} = \begin{cases} \min[t_i^{\text{on}}, \max(x_{i,t-1}, 0) + 1], & \text{if } \tilde{u}_{it} = 1, \\ \max[-t_i^{\text{cold}}, \min(x_{i,t-1}, 0) - 1], & \text{if } \tilde{u}_{it} = 0. \end{cases} \tag{34}$$

Step 3. Set $t \leftarrow t+1$, go to Step 2.

Step 4. Apply the RCED procedure with respect to $\tilde{u}$ to obtain $\tilde{p}$.

Step 5. Apply the UD algorithm with respect to $(\tilde{u}, \tilde{p})$. □

In the algorithm, the loop between Step 1 and Step 3 is to commit as many units as possible at all hours. However, such commitment tends to violate (12), i.e., the so-called minimum load conditions. In other words, the RCED phase in Step 4 of the UD algorithm may not be feasible. Since RCED is also a subroutine required at each iteration of the UD algorithm stated in Step 5 of the above algorithm, we need to extend the RCED subroutine to handle also cases where the minimum load conditions are not satisfied.

A possible modification is to dispatch the online generators so as to equalize the marginal costs to the extent possible, even if the minimum load conditions are not satisfied. That is, when

$$\sum_{i \in \mathcal{J}_t} p_i^{\min} > D_t, \tag{35}$$

all the online units are dispatched to their minimum capacities, respectively,

$$\tilde{p}_{it} \leftarrow p_i^{\min}, \qquad \forall i, \tag{36a}$$

and the corresponding lambda is the minimum of the marginal costs of the corresponding dispatches in (36a),

$$\tilde{\lambda}_t = \tilde{\alpha}_t \leftarrow \min_{i \in \mathcal{J}_t} C'(p_i^{\min}), \tag{36b}$$

$$\tilde{\mu}_t = \tilde{\lambda}_t - \tilde{\alpha}_t \leftarrow 0. \tag{36c}$$

This can be viewed as Case 1 in the modified equal-lambda method. Though the demand constraint is not met exactly, it is met at the closest possibility. In this case, $\tilde{\Omega}_t = \varnothing$ and $\tilde{\Lambda}_t = \{1, 2, \ldots, I\}$.

Although the UD method starts with an initial feasible solution of the UC problem, the unified UD algorithm, using the modified RCED procedure, may not always obtain a feasible schedule initially. The modification given in (36) in the RCED phase above is based on the expectation that, as the decommitment procedure proceeds, the commitment obtained will eventually satisfy the minimum load conditions, thus producing a feasible schedule. While in theory obtaining a feasible solution of the UC problem is an NP-hard problem (Ref. 1), it is a relatively easy task in real-world instances of that problem. In extensive numerical tests, we have found that the above approach worked satisfactorily. In all observed cases, the UD method performed well as a UC algorithm and obtained feasible solutions.

Finally, in the unified UD algorithm, properties (i) to (iii) in Proposition 4.1 may not be valid due to the modified steps (36). Fortunately, property (iv) remains valid; therefore the unified UD algorithm still terminates within I iterations. To see why property (iv) in Proposition 4.1 is valid, note that, when the minimum load conditions are not satisfied, $\tilde{p}_{it}^k = p_i^{\min}$ is fixed, $\tilde{\mu}_t^k = 0$, and $\tilde{\lambda}_t^k$ is increasing; therefore, property (iv) of Proposition 4.1 holds. When $\tilde{p}_{it}^k > p_i^{\min}$, the minimum load conditions must have been satisfied. Proposition 4.1 becomes applicable again.

Theorem 4.2. The unified UD algorithm, with the modification given in (36) in the RCED phase, terminates within I iterations, where I is the number of units.

5. Unit Decommitment vs Lagrangian Relaxation

In this section, we present an intuitive discussion on the relationship between the UD method and the LR method for solving the UC problem. Let $\{\lambda_t\}$ and $\{\mu_t\}$, $t=1,\ldots,T$, be the corresponding nonnegative Lagrange multipliers to (2) and (3). Conventional LR approaches solve the following dual problem (D):

$$\text{(D)} \quad \max_{\lambda,\mu\geq 0} \quad d(\lambda,\mu), \tag{37a}$$

$$\begin{aligned}\text{s.t.} \quad d(\lambda,\mu) &= \min_{u,p} \sum_{i=1}^{I}\sum_{t=1}^{T} C_i(p_{it})u_{it} + S_i(u,t) \\ &\quad + \sum_{t=1}^{T}\lambda_t\left(D_t - \sum_{i=1}^{I} p_{it}u_{it}\right) \\ &\quad + \sum_{t=1}^{T}\mu_t\left[R_t - \sum_{i=1}^{I} r_i(p_{it})u_{it}\right] \\ &= \sum_{i=1}^{I} d_i(\lambda,\mu) + \sum_{t=1}^{T}(\lambda_t D_t + \mu_t R_t),\end{aligned} \tag{37b}$$

$$\begin{aligned} d_i(\lambda,\mu) = \min_{u_{it},\,p_{it}} \sum_{t=1}^{T} [C_i(p_{it})u_{it} + S_i(u,t) \\ - \lambda_t p_{it}u_{it} - \mu_t r_i(p_{it})u_{it}].\end{aligned} \tag{37c}$$

The minimization problem (37c) is subject to (4)–(6) and the initial conditions. In (37c), when $u_{it}=1$, the optimal p_{it} can be obtained by solving

$$(\mathrm{Q}_i) \quad \min \quad C_i(p_{it}) - \lambda_t p_{it} - \mu_t r_i(p_{it}), \tag{38a}$$

$$\text{s.t.} \quad p_i^{\min} \leq p_{it} \leq p_i^{\max}. \tag{38b}$$

Equivalently, this defines a mapping

$$(\lambda_t,\mu_t) \overset{\mathrm{Q}_i}{\mapsto} p_{it}.$$

For simplicity, we use $p_{it}(\lambda_t,\mu_t)$ to denote the mapping. After all subproblems $d_i(\lambda,\mu)$, $i=1,\ldots,I$, are solved, the multipliers λ and μ are then updated by the subgradient rule (e.g. Ref. 21) so as to maximize $d(\cdot,\cdot)$. It is well known that, even if (D) is completely solved, it would not yield a feasible schedule.

With p_{it} in (37c) substituted by $p_{it}(\lambda_t, \mu_t)$, the dual subproblem (37c) can be rewritten as

$$d_i(\lambda, \mu) = \min_{u_{it} \in \{0,1\}} \sum_{t=1}^{T} [C_i(p_{it}(\lambda_t, \mu_t)) - \lambda_t p_{it}(\lambda_t, \mu_t) - \mu_t r_i(p_{it}(\lambda_t, \mu_t))] u_{it} + S_i(u, t). \tag{39}$$

Note that $d_i(\lambda, \mu)$ in (39) is now a 0–1 integer programming problem. Furthermore, solving $d_i(\lambda, \mu)$ in (39) is equivalent to solving the following problem:

$$\hat{d}_i(\lambda, \mu) = \min_{u_{it} \in \{0,1\}} \sum_{t=1}^{T} C_i(p_{it}(\lambda_t, \mu_t)) u_{it} + [\lambda_t p_{it}(\lambda_t, \mu_t) + \mu_t r_i(p_{it}(\lambda_t, \mu_t))](1 - u_{it}) + S_i(u, t). \tag{40}$$

To see the equivalence between (39) and (40), add $0 \cdot (1 - u_{it})$ to (39), and then add $\lambda_t p_{it}(\lambda_t, \mu_t) + \mu_t r_i(p_{it}(\lambda_t, \mu_t))$ to the payoffs of the decision variables u_{it} and $1 - u_{it}$, which yields (40).

Solving $\hat{d}_i(\lambda, \mu)$, or equivalently solving a unit subproblem $d_i(\lambda, \mu)$, is like solving a UD problem (23). This implies that a solution of a LR subproblem is already optimally decommitted in some sense. Given an economically dispatched schedule $(\tilde{u}, \tilde{p}, \tilde{\lambda}, \tilde{\mu})$ [if $\tilde{u}$ is not dispatchable, use the modified steps (36)], it can be shown that

$$(\tilde{\lambda}_t, \tilde{\mu}_t) \overset{Q_i}{\mapsto} \tilde{p}_{it}.$$

Starting from this schedule $(\tilde{u}, \tilde{p}, \tilde{\lambda}, \tilde{\mu})$, performing either the UD step (23) with respect to $\tilde{u}$ or the LR subproblem (37c) with respect to $(\tilde{\lambda}, \tilde{\mu})$ for any unit would result in the same (tentative) commitment for that unit. However, at the next iteration, the LR subproblem would consider the tentative commitments of all units, while the UD method corrects only the commitment of one unit (by adopting its tentative commitment). Also, the updating of the multipliers $(\tilde{\lambda}, \tilde{\mu})$ is different: UD performs RCED, and LR uses subgradients. The multipliers $(\tilde{\lambda}, \tilde{\mu})$ obtained from either approach are intended to capture the marginal cost of the demand and reserve constraints, respectively. The unit commitment updating of the LR approaches results in overcorrection of the commitments, which leads to its primal infeasibility at all iterations, while the UD method adjusts the commitment, one unit at a time, toward a feasible commitment and then maintains feasibility thereafter. It is fair to say that the UD method is a LR-like method and that the LR approaches perform UD-type operations.

Because of the afore-mentioned primal infeasibility in the dual optimization, conventional LR approaches fall into a category of two-phase algorithms, with a feasibility phase following the first phase of dual optimization. In Refs. 1 and 16, Tseng et al. suggested to add a third phase, a UD phase, as the postprocessing phase, to the conventional two-phase methods. This results in a three-phase scheme. The three-phase scheme is reported not only to be able to improve the solution quality, but also to mitigate some unpredictable effect due to heuristics when integrated with the LR approaches (Ref. 16). Figure 3 gives an illustration of the typical algorithm trajectories of the conventional LR two-phase methods, the LR three-phase method (Refs. 1 and 16), and the proposed unified UD approach.

6. Numerical Results and Conclusions

We conduct numerical tests to compare the performance of UD and LR. All algorithms are implemented in FORTRAN on an HP 700 workstation. Four cases of systems with combinations of 10 or 30 units and 24 or 168 hours of planning horizon are tested. For each case, we generate randomly 100 instances of the UC problem. Detailed configuration of the random instances are available upon request to the corresponding author. Each instance is solved by the LR and UD methods. The column under DG

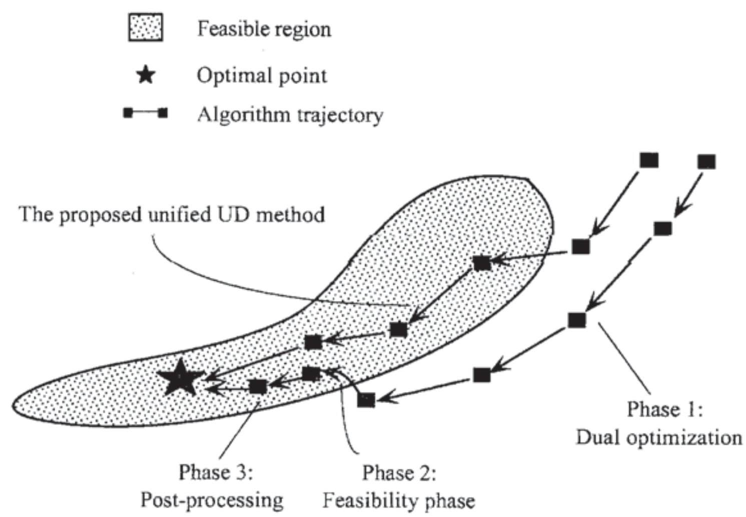

Fig. 3. Illustration of algorithm trajectories.

Table 1. Comparison of the UD and LR algorithms.

Case $I \times T$	Solution quality UD	LR	DG (%)	CPU time UD	LR
10×24	1.0010 (0.9933–1.10105)	1	0.9 (0.09–2.85)	0.2185 (0.1028–0.4773)	1
10×168	1.0008 (0.9976–1.0054)	1	0.9 (0.38–2.04)	0.1500 (0.0978–0.3446)	1
30×24	1.0013 (0.9981–1.0090)	1	0.28 (0.06–0.81)	0.5214 (0.3344–0.8608)	1
30×168	1.0017 (0.9997–1.0058)	1	0.35 (0.15–1.78)	0.2745 (0.1513–0.4152)	1

(duality gap) records the duality gap of the LR approach in terms of the percentage of the dual value. Since the comparisons are normalized to the value of the LR approach, the columns under LR consist of all ones. Also, the two numbers in parentheses define the range of the sample points. The mean of the sample points is recorded on the top of the corresponding parentheses. The test results including solution qualities and CPU times required for both methods are summarized in Table 1.

From Table 1, we know that the error between LR and UD is within 0.2% and that the UD methods take much less (save about 50% at least) CPU time than the LR approach. Besides, the only heuristic in UD is the unit selection rule, which is analogous to choosing a descent direction in continuous optimization. Our experience indicates that the algorithm performance is insensitive to the selection rule. To sum up, the numerical testing results show that UD serves as a reliable, efficient, and robust alternative to the traditional implementations of the LR approaches for solving the UC problem.

References

1. Tseng, C. L., *On Power System Generation Unit Commitment Problems*, PhD Thesis, University of California at Berkeley, 1996.
2. Chandler, W. G., Dandeno, P. L., Gilmn, A. F., and Kirchmayer, L. K., *Short-Range Operation of a Combined Thermal and Hydroelectric Power System*, AIEE Transactions, Vol. 73, No. 3, pp. 1057–1065, 1953.
3. Turgeon, A., *Optimal Unit Commitment*, IEEE Transactions on Automatic Control, Vol. 23, No. 2, pp. 223–227, 1977.
4. Turgeon, A., *Optimal Scheduling of Thermal Generating Units*, IEEE Transactions on Automatic Control, Vol. 23, No. 6, pp. 1000–1005, 1978.

5. Dillon, T. S., *Integer Programming Approach to the Problem of Optimal Unit Commitment with Probabilistic Reserve Determination*, IEEE Transactions on Power Apparatus and Systems, Vol. 97, No. 6, pp. 2154–2164, 1978.
6. Pang, C. K., and Chen, H. C., *Optimal Short-Term Thermal Unit Commitment*, IEEE Transactions on Power Apparatus and Systems, Vol. 95, No. 4, pp. 1336–1342, 1976.
7. Pang, C. K., Sheble, G. B., and Albuyeh, F., *Evaluation of Dynamic Programming Based Methods and Multiple Area Representation for Thermal Unit Commitments*, IEEE Transactions on Power Apparatus and Systems, Vol. 100, No. 3, pp. 1212–1218, 1981.
8. Wood, A. J., and Wollenberg, B. F., *Power Generation, Operation, and Control*, John Wiley and Sons, New York, NY, 1984.
9. Muckstadt, J. A., and Koenig, S. A., *An Application of Lagrangian Relaxation to Scheduling in Power-Generation Systems*, Operations Research, Vol. 25, No. 3, pp. 387–403, 1977.
10. Bertsekas, D. P., Lauer, G. S., Sandell, N. R., Jr., and Posbergh, T. A., *Optimal Short-Term Scheduling of Large-Scale Power Systems*, IEEE Transactions on Automatic Control, Vol. 28, No. 1, pp. 1–11, 1983.
11. Bard, J. F., *Short-Term Scheduling of Thermal-Electric Generators Using Lagrangian Relaxation*, Operations Research, Vol. 36, No. 5, pp. 756–766, 1988.
12. Cohen, A., and Sherkat, V., *Optimization-Based Methods for Operations Scheduling*, Proceedings of IEEE, Vol. 75, No. 12, pp. 1574–1591, 1987.
13. Zhuang, F., *Optimal Generation Unit Commitment in Thermal Electric Power Systems*, PhD Thesis, McGill University, 1988.
14. Lee, F. N., *Short-Term Thermal Unit Commitment: A New Method*, IEEE Transactions on Power Systems, Vol. 3, No. 2, pp. 421–428, 1988.
15. Li, C. A., Johnson, R. B., and Svoboda, A. J., *A New Unit Commitment Method*, IEEE Transactions on Power Systems, Vol. 12, No. 1, pp. 113–119, 1997.
16. Tseng, C. L., Oren, S. S., Svoboda, A. J., and Johnson, R. B., *A Unit Decommitment Method in Power System Scheduling*, Electrical Power and Energy Systems, Vol. 19, No. 6, pp. 357–365, 1997.
17. Ibaraki, T., and Katoh, N., *Resource Allocation Problems: Algorithmic Approaches*, MIT Press, Cambridge, Massachusetts, 1988.
18. Brucker, P., *An O(n) Algorithm for Quadratic Knapsack Problems*, Operations Research Letters, Vol. 3, No. 3, pp. 163–166, 1984.
19. Sheble, G. B., *Real-Time Economic Dispatch and Reserve Allocation Using Merit Order Loading and Linear Programming Rules*, IEEE Transactions on Power Systems, Vol. 4, No. 4, pp. 1414–1420, 1989.
20. Stadlin, W. O., *Economic Allocation of Regulating Margin*, IEEE Transactions on Power Apparatus and Systems, Vol. 89, No. 4, pp. 1776–1781, 1971.
21. Bazaraa, M. S., Sherali, H. D., and Shetty, C. M., *Nonlinear Programming: Theory and Algorithms*, John Wiley and Sons, New York, NY, 1993.