CONTEXTUAL LEARNING FOR UNIT COMMITMENT WITH RENEWABLE ENERGY SOURCES

Hyun-Suk Lee[⋆] *Cem Tekin*[†] *Mihaela van der Schaar*[‡] *Jang-Won Lee*[⋆]

[⋆] Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea
[†] Electrical and Electronics Engineering Department, Bilkent University, Turkey
[‡] Department of Electrical Engineering, UCLA, Los Angeles, USA

ABSTRACT

In this paper, we study a unit commitment (UC) problem minimizing operating costs of the power system with renewable energy sources. We develop a contextual learning algorithm for UC (CLUC) which learns which UC schedule to choose based on the context information such as past load demand and weather condition. CLUC does not require any prior knowledge on the uncertainties such as the load demand and the renewable power outputs, and learns them over time using the context information. We characterize the performance of CLUC analytically, and prove its optimality in terms of the long-term average cost. Through the simulation results, we show the performance of CLUC and the effectiveness of utilizing the context information in the UC problem.

Index Terms— Unit commitment, uncertainty, learning, renewable energy

1. INTRODUCTION

Using renewable energy sources such as wind and solar has many advantages, e.g., low economic costs and reducing carbon footprint from fossil fuels. In general, to efficiently use renewable energy sources in power systems, uncertainties in power systems such as load demands and renewable power outputs should be addressed. Thus, recently, such uncertainties have been considered in unit commitment (UC) problems which determine the on/off states of thermal generation units in power systems and their power outputs, i.e., UC schedule, to minimize operating costs.

In [1–3], uncertainties are modeled as scenarios each of which represents the sequence of the realizations of uncertainties over the optimization horizon (e.g. 24 hours). Then, UC scheduling problems are formulated as stochastic optimization problems minimizing the expected operating costs over the probability distribution of scenarios. The UC schedule is determined by solving the problems. In [4], the load demand is modeled as a Markov-modulated Poisson process and the renewable power output is modeled as a hidden Markov models. Then, the UC scheduling problem is formulated as a partially observable Markov decision process and its structural results to determine the UC schedule are derived. In [5, 6], the uncertainties are modeled by bounded closed intervals. Then, UC scheduling problems are formulated as interval optimization problems with the constraints considering the bounded intervals. By solving the problems, the UC schedule is obtained.

Although the prior works [1–6] determine the UC schedule with considering uncertainties in different ways, they commonly need *a priori knowledge* of uncertainties such as their probability distributions and forecasts. In general, such a knowledge can be always provided from the past. However, their performances deteriorate when the knowledge is inaccurate, and an additional computational cost is needed to obtain such a priori knowledge with a certain accuracy [7,8]. Thus, to overcome these problems, a UC algorithm which does not need any a priori knowledge of uncertainties is needed.

In smart grids, there are many other issues considering uncertainties, e.g., storage management, load scheduling, and dynamic pricing. They have been widely studied [9–12], and due to the uncertainties such as load demand and electricity price, they have the same problems with UC. In [11, 12], learning algorithms are proposed to overcome the problems. The algorithms does not need a priori knowledge since they learn the dynamics of the uncertainties. Thus, as the prior works on smart grids, we can adopt learning methods to develop the UC algorithm which does not need a priori knowledge.

In this paper, we study a UC problem with uncertainties, i.e., load demands and renewable power outputs, minimizing the average total operating cost. We develop a contextual learning algorithm for UC (CLUC) which does not need any a priori knowledge of uncertainties. It learns which UC schedule to choose based on the context information such as the current time, the past load demand and the weather condition. To evaluate CLUC, we use the learning regret from the complete information benchmark given the probability distributions of the uncertainties. We then show the regret bound of CLUC is sublinear in time, i.e., the average cost of CLUC converges to the average cost of the complete information benchmark. Through numerical results, the performance of CLUC and the effectiveness of using the context information in the UC problem are shown.

The rest of this paper is organized as follows. Section 2 provides the system model. In Section 3, we develop a contextual learning algorithm for UC, and provide its regret bound. We provide numerical results in Section 4. Finally, we conclude in Section 5.

2. SYSTEM MODEL

We consider a UC problem in a power system which has thermal and renewable power generation units. The power system has J thermal power generation units each of which is denoted with an index $j \in \mathcal{J} = \{1, 2, ..., J\}$.[1] In addition, it has also renewable power generation units. The power system schedules the on/off status and power outputs of its thermal power generation units, i.e., a

The work of H.-S. Lee and J.-W. Lee was supported in part by Mid-career Researcher Program through NRF grant funded by the MSIP, Korea (2013R1A2A2A01069053).

[1] In this paper, unit j implies thermal power generation unit j.

UC schedule, over a discrete time horizon, e.g., an hour, where each time period has a fixed equal duration. Let t be an index of time periods of the time horizon. The set of time periods is denoted by $\mathcal{T} = \{0, 1, 2, ...\}$. At the beginning of time period t, the power system schedules its thermal power generation units for time period $t + T_{sc}$, where T_{sc} is the number of necessary time periods to prepare the operation of the thermal power generation units according to the UC schedule. The on/off status of unit j during time period t is denoted by $u_j(t) \in \{0, 1\}$, where 1 represents the on state and 0 represents the off state. The vector of the on/off states of all thermal power generation units during time period t is denoted by $\mathbf{u}(t) = \{u_j(t)\}_{j\in\mathcal{J}}$. The up time of unit j at time period t, which represents the number of consecutive time periods that unit j has been in the on state at the *end* of time period t, is denoted by $T_{j,on}(t)$, and it is obtained by

$$T_{j,on}(t) = \begin{cases} T_{j,on}(t-1) + 1, & \text{if } u_j(t) = 1 \\ 0, & \text{if } u_j(t) = 0 \end{cases}.$$

Similarly, the down time of unit j at time period t, which represents the number of consecutive time periods that unit j has been in the off state at the *end* of time period t, is denoted by $T_{j,off}(t)$, and it is obtained by

$$T_{j,off}(t) = \begin{cases} T_{j,off}(t-1) + 1, & \text{if } u_j(t) = 0 \\ 0, & \text{if } u_j(t) = 1 \end{cases}.$$

We denote the vectors of $T_{j,on}(t)$'s and $T_{j,off}(t)$'s of all thermal power generation units as $\mathbf{T}_{on}(t) = \{T_{j,on}(t)\}_{j\in\mathcal{J}}$ and $\mathbf{T}_{off}(t) = \{T_{j,off}(t)\}_{j\in\mathcal{J}}$, respectively. When a thermal power generation unit is turned on, it cannot be turned off for the next specific number of time periods, i.e., for each unit j,

$$1 \le T_{j,on}(t-1) < MUT_j \Rightarrow u_j(t) = 1, \quad (1)$$

where MUT_j is the minimum up time of unit j. Similarly, when it is turned off, it cannot be turned on for the next specific number of time periods, i.e., for each unit j,

$$1 \le T_{j,off}(t-1) < MDT_j \Rightarrow u_j(t) = 0, \quad (2)$$

where MDT_j is the minimum down time of unit j.

The power output of unit j at time period t is denoted by $p_j(t)$, and it is bounded by $p_j(t) \in \left[p_j^{min}, p_j^{max}\right]$, where p_j^{min} and p_j^{max} are the minimum and maximum power outputs of unit j, respectively. The vector of the power outputs of all thermal power generation units at time period t is denoted as $\mathbf{p}_{ther}(t) = \{p_j(t)\}_{j\in\mathcal{J}}$. Due to the ramp rate limit, the power output of unit j at time period t should satisfy the following constraint:

$$p_j(t-1) - RR_j \le p_j(t) \le p_j(t-1) + RR_j, \quad (3)$$

where RR_j is the ramp rate limit of unit j. Moreover, the spinning reserve requirement in the power system for the critical loads should be guaranteed as

$$\sum\nolimits_{j\in\mathcal{J}} u_j(t) \left(p_j^{max} - p_j(t)\right) \ge SR, \quad (4)$$

where SR is the spinning reserve requirement.

In our system model, we use the current time, the weather condition, and the past load demands as the context information which the power system considers.[2] To model the current time, we introduce a set of time indices for a circular time duration, e.g., a day, a month, and a year, $\mathcal{H} = \{0, 1, ..., H-1\}$, where each index represents an actual time in the time duration. Then, each time period t is mapped to the corresponding current time index $h(t) \in \mathcal{H}$ as $h(t) = \mathrm{mod}(t, H)$. Let $w(t)$ be the weather condition which is observed by the power system at the beginning of time period t. The set of weather conditions is denoted by $\mathcal{W}$. In addition, during each time period, the uncertainties in our system, i.e., the load demand and the power outputs of the renewable power generation units, are realized, and they can be observed by the power system. The uncertainties during each time period have a strong correlation with the context at the time period. The load demand at time period t is denoted by $M(t)$ and is assumed to be bounded by $M(t) \in \mathcal{M} = [M_{min}, M_{max}]$, where M_{min} and M_{max} are the minimum and maximum load demands, respectively. The sum of power outputs of all renewable power generation units during time period t is denoted by $p_{re}(t)$.

With the UC schedule and the realization of the uncertainties, the total operating cost of the power system during time period t, $C_{tot}(t)$, is obtained as

$$C_{tot}(t) = \sum\nolimits_{j\in\mathcal{J}} \left(C_{j,fu}(t) + C_{j,su}(t)\right) + C_{sh}(t) + C_{cu}(t), \quad (5)$$

where $C_{j,fu}(t)$ is the fuel cost of unit j that supplies power $p_j(t)$ during time period t, $C_{j,su}(t)$ is the start-up cost of unit j at time period t, $C_{sh}(t)$ is the load shedding cost during time period t, and $C_{cu}(t)$ is the power curtailment cost during time period t. The fuel cost can be modeled as a non-linear function of the power output [13] as

$$C_{j,fu}(t) = C_{j,fu}^{(0)} \cdot u_j(t) + C_{j,fu}^{(1)} \cdot p_j(t) + C_{j,fu}^{(2)} \cdot p_j(t)^2, \quad (6)$$

where $C_{j,fu}^{(0)}$, $C_{j,fu}^{(1)}$, and $C_{j,fu}^{(2)}$ are the cost coefficients of unit j. The start-up cost can be modeled as follows [13, 14]:

$$C_{j,su}(t) = CM_j + CSC_j \left\{1 - e^{\left(-\frac{T_{j,off}(t-1)}{CST_j}\right)}\right\}, \quad (7)$$

where CM_j is the crew start-up cost and maintenance cost of unit j, CSC_j is the cold start-up cost of unit j, and CST_j is the cold start-up time of unit j. The load shedding cost during time period t, $C_{sh}(t)$, is given by

$$C_{sh}(t) = LSP \cdot \left[M(t) - \sum\nolimits_{j\in\mathcal{J}} p_j(t) - p_{re}(t)\right]^+, \quad (8)$$

where LSP is the load shedding price and $[\cdot]^+ = \max[0, \cdot]$. The power curtailment cost during time period t, $C_{cu}(t)$, is given by

$$C_{cu}(t) = PCP \cdot \left[\sum\nolimits_{j\in\mathcal{J}} p_j(t) + p_{re}(t) - M(t)\right]^+, \quad (9)$$

where PCP is the power curtailment price.

3. CONTEXTUAL LEARNING ALGORITHM

3.1. Problem Formulation

The context at time period t is defined by $\mathbf{x}(t) := \{h(t), \mathbf{M}(t, T_M), \mathbf{w}(t, T_W))\}$, where $\mathbf{M}(t, T_M)$ is the vector of load demands of the past T_M time periods and $\mathbf{w}(t, T_W)$ is the vector of weather conditions of the past T_W time periods, and the context space is defined by $\mathcal{X} = \mathcal{H} \times \mathcal{M}^{T_M} \times \mathcal{W}^{T_W}$. We introduce a projection function ϕ which projects the context $\mathbf{x}$ into a low dimensional space.

[2] It is worth noting that any other related information such as the past weather condition and the weather forecast can be used.

Then, we denote the projected context from the context $\mathbf{x}$ by ϕ, i.e., $\phi(\mathbf{x})$, as $\mathbf{x}_\phi$ and the projected context space by ϕ as $\mathcal{X}_\phi$ which has $D_\mathcal{X}$-dimensions. For example, a weighted average function of load demands and weather conditions can be used. Note that the projection function is not necessary to our algorithm but it helps our algorithm learn faster if necessary. In addition to the projected context, $\mathbf{x}_\phi$, the down time of units, $\mathbf{T}_{off}$, should be considered when choosing the action since the start-up cost in (7) depends on it. For the sake of analysis, we define the bounded down time of unit j at time period t, $\tilde{T}_{j,off}(t)$, bounded by PDT_j, i.e., $\tilde{T}_{j,off} \in \tilde{\mathcal{T}}_{j,off} = \{0, 1, ..., PDT_j\}$, where PDT_j is the maximum bounded down time of unit j. Note that since the start-up cost becomes almost constant for large down times, it is enough to consider down times in a bounded region. Then, the bounded down time space is defined by $\tilde{\mathcal{T}}_{off} = \prod_{j\in\mathcal{J}} \tilde{\mathcal{T}}_{j,off}$. We denote the vector of $\tilde{T}_{j,off}(t)$'s of all units as $\tilde{\mathbf{T}}_{off}(t) = \{\tilde{T}_{j,off}(t)\}_{j\in\mathcal{J}}$. Then, we define an extended context at time period t by $\mathbf{z}(t) := \{\mathbf{x}_\phi(t), \tilde{\mathbf{T}}_{off}(t-1)\}$, and define the extended context space by $\mathcal{Z} = \mathcal{X}_\phi \times \tilde{\mathcal{T}}_{off}$. We now define the state for units at the beginning of time period t as $\mathbf{s}(t) := \{\mathbf{u}(t-1), \mathbf{p}_{ther}(t-1), \mathbf{T}_{on}(t-1), \mathbf{T}_{off}(t-1)\}$.

At the beginning of each time period t, an action which is denoted by $a(t) = \{\mathbf{u}(t), \mathbf{p}_{ther}(t)\}$, is chosen. Then, the action space which represents all actions is defined by $\mathcal{A} = \{0,1\}^J \times \prod_{j\in\mathcal{J}} [p_j^{min}, p_j^{max}]$. The set of actions at time period t is constrained by the unit status at time period t, $\mathbf{s}(t)$, due to the constraints. Thus, the set of feasible actions at time period t with the unit status $\mathbf{s}(t)$, $\mathcal{A}(\mathbf{s}(t))$, is obtained as

$$\mathcal{A}(\mathbf{s}(t)) = \{\mathbf{u}(t), \mathbf{p}_{ther}(t) |\ (1), (2), (3), \text{and } (4)\}.$$

We denote a UC policy which depends on the extended context with given unit status $\mathbf{s}$ as $\pi : \mathcal{Z} \to \mathcal{A}(\mathbf{s})$. For given extended context $\mathbf{z}(t)$ and unit status $\mathbf{s}(t)$, the UC policy π chooses the action denoted by $\pi_{\mathbf{s}(t)}(t, \mathbf{z}(t))$ from the set of feasible actions, $\mathcal{A}(\mathbf{s}(t))$. For convenience, we denote the action for time period t, $\pi_{\mathbf{s}(t)}(t, \mathbf{z}(t))$, as $\pi(t)$. Then, the UC problem is formally defined as follow.

$$\underset{\pi:\mathcal{Z}\to\mathcal{A}(\mathbf{s})}{\text{argmin}}\ \mathbb{E}\left[\lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T} C_{tot}(\pi(t), t)\right], \tag{10}$$

where $C_{tot}(\pi(t), t)$ is the total operating cost during time period t given action $\pi(t)$.

3.2. Contextual Learning Algorithm for UC

In this subsection, we present a contextual learning algorithm for UC (CLUC). For simple presentation, we normalize the extended context[3] space to be $\mathcal{Z} = [0, 1]^D$,[4] where D is the dimension of the context space, i.e., $D_\mathcal{X} + J$. Note that normalizing the context is used for the regret analysis in Section 3.3, and the regret bound of CLUC can always be achieved by a proper scaling of the context. At the beginning of CLUC, the context space and the action space are uniformly partitioned and discretized, respectively. We denote the slicing parameter for the context space, which is a positive integer, by m_Z. The context space $\mathcal{Z}$ is partitioned into $(m_Z)^D$ sets where each set is a D-dimensional hypercube with $1/m_Z$ edge length. We denote the partition of the context space by $\mathcal{P}_\mathcal{Z}$ which contains $(m_Z)^D$ sets. Let p_z be an index of sets in $\mathcal{P}_\mathcal{Z}$, and let $p_z(\mathbf{z})$ be the index of the set where context $\mathbf{z}$ belongs. We also uniformly discretize the power output of unit j using the slicing parameter for the power output m_A which is a positive integer. The set of the discretized power outputs of unit j is denoted by $\bar{p}_j(t) \in \bar{P}_j \{p_j^{min} + p_j^{m_A}, p_j^{min} + 2p_j^{m_A}, ..., p_j^{max}\}$, where $p_j^{m_A} = (p_j^{max} - p_j^{min})/m_A$. We denote the vector of the discretized power outputs of all units during time period t as $\bar{\mathbf{p}}_{ther}(t) = \{\bar{p}_j(t)\}_{\forall j\in\mathcal{J}}$. The discretized action space is given by $\bar{\mathcal{A}} = \{0,1\}^J \times \prod_{j\in\mathcal{J}} \bar{P}_j$. Then, we define the set of discretized feasible actions with unit status $\mathbf{s}$, $\bar{\mathcal{A}}(\mathbf{s})$, as

$$\bar{\mathcal{A}}(\mathbf{s}) := \{\mathbf{u}(t), \bar{\mathbf{p}}_{ther}(t) | (1), (2), (3), \text{and } (4)\}.$$

Algorithm 1 CLUC

Create context partition $\mathcal{P}_\mathcal{Z}$ with m_Z
Discretize action space $\bar{\mathcal{A}}$ with m_A
$N(a, p) \leftarrow 0, \hat{c}(a, p) \leftarrow \infty, \forall a \in \bar{\mathcal{A}}, \forall p \in \mathcal{P}_\mathcal{Z}$
while TRUE **do**
  Observe context $\mathbf{z}$ and unit status $\mathbf{s}$
  $p \leftarrow p_z(\mathbf{z})$, $a \leftarrow \text{argmin}_{a'\in\bar{\mathcal{A}}(\mathbf{s})} \hat{c}(a', p)$
  Operate units with a and observe C_{tot}
  $\hat{c}(a, p) \leftarrow \frac{\hat{c}(a,p)N(a,p)+C_{tot}}{N(a,p)+1}$
  Virtually update $\hat{c}(a', p), \forall a' \in \bar{\mathcal{A}}(\mathbf{s}) \setminus \{a\}$
  $N(a, p) \leftarrow N(a, p) + 1,\ \forall a \in \bar{\mathcal{A}}(\mathbf{s})$
end while

We denote the number of times that action a is chosen with a context in set p_z as $N(a, p_z)$. We also define the estimated cost of action a on set p_z, $\hat{c}(a, p_z)$, which represents the sample mean of the total operating cost observed from action a on set p_z. At the beginning of each time period t, the power system observes its context $\mathbf{z}(t)$ and unit status $\mathbf{s}(t)$. Then, it checks the corresponding set to the context, $p_z(\mathbf{z}(t))$, and the set of actions with the unit status, $\bar{\mathcal{A}}(\mathbf{s}(t))$. With $p_z(\mathbf{z}(t))$, it chooses the action $\hat{\pi}(t) \in \bar{\mathcal{A}}(\mathbf{s}(t))$ with the lowest total operating cost estimation. During the time period, the power system operates its thermal power generation units according to the chosen action $\hat{\pi}(t)$. At the end of the time period, the power system observes the realization of the uncertainties with which the total operating cost during the time period, $C_{tot}(t)$, is obtained as in (5). Then, the power system updates the estimated cost $\hat{c}(\hat{\pi}(t), p_z)$ by using the cost during the time period. Moreover, the estimated costs of the other actions, i.e., $a \in \bar{\mathcal{A}}(\mathbf{s}(t)) \setminus \{\hat{\pi}(t)\}$, also can be updated even they were not chosen, since choosing the action does not affect to the uncertainties. This *virtual* update of the estimated costs can accelerate the learning speed of the algorithm. Then, the number of times that action a is (virtually) chosen with a context in set p_z, $N(a, p_z)$, is updated for $a \in \bar{\mathcal{A}}(\mathbf{s}(t))$. CLUC is described in Algorithm 1.

3.3. Regret Bound for CLUC

In this subsection, we study the learning regret from a complete information benchmark which is the myopically optimal policy with a priori information, i.e., $f_{\mathbf{x}_\phi}$. The expected operating cost during the time period of action $a \in \mathcal{A}$ with given context $\mathbf{z} \in \mathcal{Z}$, $c(a, \mathbf{z})$, is obtained by $c(a, \mathbf{z}) := \mathbb{E}_{\hat{M}(\mathbf{x}_\phi), \hat{p}_{re}(\mathbf{x}_\phi)} [C_{tot}(a, t)]$, where $\hat{M}(\mathbf{x}_\phi)$ and $\hat{p}_{re}(\mathbf{x}_\phi)$ are the random variables for the load demand and the renewable power output during the time period, respectively. The joint probability distribution of $\hat{M}(\mathbf{x}_\phi)$ and $\hat{p}_{re}(\mathbf{x}_\phi)$ is given by $f_{\mathbf{x}_\phi}$. Then, the benchmark with given context $\mathbf{z}$ and unit status $\mathbf{s}$, $\pi^*_\mathbf{s}(\mathbf{z})$,

[3] In algorithm description, we omit "extended" from the extended context for convenience.

[4] According to the definition of the bounded down time space, $\tilde{\mathcal{T}}_{off}$, it can be normalized to be $[0, 1]^J$ by using PDT_j's.

is defined by

$$\pi_{\mathbf{s}}^*(\mathbf{z}) := \mathrm{argmin}_{a \in \mathcal{A}(\mathbf{s})} c(a, \mathbf{z}), \; \forall \, \mathbf{z} \in \mathcal{Z}. \tag{11}$$

Let $\hat{\pi}$ be the UC policy from CLUC. Then, the expected learning regret with respect to the benchmark, $\pi_{\mathbf{s}}^*(\mathbf{z})$, in (11) by time period T is given by

$$R(T) := \mathbb{E}\left[\sum_{t=0}^{T} C_{tot}(\hat{\pi}(t), t)\right] - \sum_{t=0}^{T} c(\pi_{\mathbf{s}}^*(\mathbf{z}), \mathbf{z}).$$

For the simple presentation of the analysis, we normalize the total operation cost to be $[0, 1]$. We assume that the expected load demand and renewable power outputs are similar for similar contexts, which is widely used as a similarity information [15–17]. We formalize this as a Hölder condition.

Assumption 1 *There exists $L > 0$, $\alpha > 0$ such that for all $\mathbf{x}_\phi$, $\mathbf{x}'_\phi \in \mathcal{X}_\phi$, we have $|\mathbb{E}[\hat{M}(\mathbf{x}_\phi)] - \mathbb{E}[\hat{M}(\mathbf{x}'_\phi)]| \le L\|\mathbf{x}_\phi - \mathbf{x}'_\phi\|^\alpha$ and $|\mathbb{E}[\hat{p}_{re}(\mathbf{x}_\phi)] - \mathbb{E}[\hat{p}_{re}(\mathbf{x}'_\phi)]| \le L\|\mathbf{x}_\phi - \mathbf{x}'_\phi\|^\alpha$.*

The following theorem provides the regret bound of CLUC. Due to space limitations, its proof is given in our technical report [18].

Theorem 1 *With $m_A = \left\lceil T^{\frac{2\alpha}{J(3\alpha+D)}} \right\rceil$ and $m_Z = \left\lceil T^{\frac{1}{(3\alpha+D)}} \right\rceil$, the regret bound of CLUC satisfies $R(T) = O(T^{\frac{2\alpha+D}{3\alpha+D}} \log T)$.*

The regret bound in Theorem 1 is sublinear in T, and thus, in our system model with the indefinite time periods, it is guaranteed that the average cost of CLUC converges to the myopically optimal average cost, i.e., $\lim_{T\to\infty} R(T)/T = 0$.

4. NUMERICAL RESULTS

In this section, we provide simulation results to evaluate the performance of CLUC. The length of a time period is an hour and the time duration for the context is set to be a day, i.e., $\mathcal{H} = \{0, 1, ..., 23\}$. We consider a microgrid system with wind turbines and four identical thermal power generation units. The parameters of the thermal power generation units are adopted from [1]. The power output capacity of the wind turbines is set to be 650 kW, and their parameters and power output profile for each hour are adopted from [1]. For a load demand profile for each hour, we use the hourly average load shapes of residential electricity services in California [19] with 500 customers. We set the load shedding price, LSP, and the power curtailment price, PCP, to be 200 \$/kWh [1]. The spinning reserve requirement is set to be 10% of the total power output of the thermal power generation units. For CLUC, we set m_A and m_Z to be 4 and 10, respectively. In addition, in CLUC, we consider a context consisting of the current time, load demand context, weather context, and down time of units, where the dimensions of both load demand and weather context spaces are 1.

To evaluate the performance of CLUC, we compare it with a Q-learning-like algorithm for UC (QLUC) which does not consider both load demand context and weather context which are related to the uncertainties. We can simply implement it by neglecting both contexts in CLUC. In QLUC, we also adopt the virtual updates for fair comparison. In addition, we consider the complete information benchmark which is the optimal policy in the regret bound for CLUC. The benchmark is a kind of stochastic optimization for UC (SOUC) with a priori information, which has been widely studied.

In each time period, the load demand context and the weather context are uniformly generated between $[0.5, 1.5]$. Then, the load demand profile value and the renewable power output profile value in the time period are obtained according to the current time index of the time period. The load demand is generated by a Gaussian distribution. The mean of the distribution is set to be the value of multiplying the load demand profile value by the load demand context. The standard deviation of the distribution is set to be 2.5% of its mean.[5] Similarly, the renewable power output is also generated by a Gaussian distribution using the renewable power output profile and the weather context. For SOUC, we also consider the scenarios where the mean of the load demand is overestimated and the mean of the renewable power output is underestimated from their accurate values. In each scenario, the degree of overestimation and underestimation is given by error percentage.

Table 1. Comparison of average costs (\$)

	C_{fuel}	C_{su}	C_{sh}	C_{cu}	C_{tot}
CLUC	2,683	2,540	5,755	8,807	19,748
QLUC	2,066	2,019	13,267	9,378	26,730
SOUC	2,698	2,535	4,924	8,120	18,277
SOUC w/ 5% error	2,929	2,692	4,304	10,256	20,181
SOUC w/ 10% error	3,164	2,852	4,189	12,971	23,176

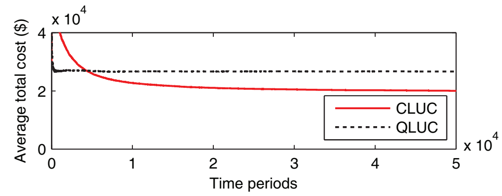

Fig. 1. Average total operation costs of CLUC and QLUC.

The average costs of CLUC, QLUC, and SOUCs are provided in Table 1. The average total operating cost of CLUC is lower than that of QLUC. From the costs of QLUC, we see that in general QLUC generates too small amount of power to support the load demand compared with CLUC since it fails to predict the uncertainties due to the lack of context information. From the results of SOUCs, we can see that CLUC achieves a close performance to that of SOUC with perfect a priori information, and it can achieve a better performance than SOUC if a priori information of SOUC is not accurate.

In Fig. 1, we compare the learning speed of CLUC with that of QLUC. The context space of CLUC has a higher dimension than that of QLUC. Hence, as shown in the figure, the learning speed of CLUC is slower than that of QLUC. However, we can see that CLUC achieves the converged average total operation cost of QLUC by relatively short time periods (about 4,200 time periods).

5. CONCLUSION AND FUTURE WORK

In this paper, we developed a contextual learning for unit commitment (CLUC) minimizing the average total operating cost of the power system with renewable energy sources. CLUC does not need any a priori information of the system uncertainties, and its optimality in terms of the long-term average cost is shown. We show that using the context information is effective to minimize the average total operating cost. However, it causes a slow learning speed of CLUC compared with QLUC due to the higher dimension of the context space. Thus, one important future direction is to mitigate the slow learning speed of CLUC.

[5] Note that the Gaussian distribution is widely used to model the forecasting error [20].

6. REFERENCES

[1] A. Zein Alabedin, E. F. El-Saadany, and M. Salama, "Generation scheduling in microgrids under uncertainties in power generation," in *Proc. of IEEE EPEC*, 2012.

[2] A. Papavasiliou, S. S. Oren, and B. Rountree, "Applying high performance computing to transmission-constrained stochastic unit commitment for renewable energy integration," *IEEE Trans. Power Syst.*, vol. 30, no. 3, pp. 1109–1120, 2015.

[3] H. Quan, D. Srinivasan, A. M. Khambadkone, and A. Khosravi, "A computational framework for uncertainty integration in stochastic unit commitment with intermittent renewable energy sources," *Applied Energy*, vol. 152, pp. 71–82, 2015.

[4] S. Bu, F. R. Yu, and P. X. Liu, "Distributed unit commitment scheduling in the future smart grid with intermittent renewable energy resources and stochastic power demands," *International Journal of Green Energy*, 2014.

[5] L. Wu, M. Shahidehpour, and Z. Li, "Comparison of scenario-based and interval optimization approaches to stochastic SCUC," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 913–921, 2012.

[6] Y. Wang, Q. Xia, and C. Kang, "Unit commitment with volatile node injections by using interval optimization," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1705–1713, 2011.

[7] N. Sharma, P. Sharma, D. Irwin, and P. Shenoy, "Predicting solar generation from weather forecasts using machine learning," in *Proc. of IEEE SmartGridComm*, 2011.

[8] L. Yang, M. He, J. Zhang, and V. Vittal, *Spatio-temporal data analytics for wind energy integration*, Springer, 2014.

[9] B. G. Kim, S. Ren, M. van der Schaar, and J. W. Lee, "Bidirectional energy trading and residential load scheduling with electric vehicles in the smart grid," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1219–1234, July 2013.

[10] L. P. Qian, Y. J. A. Zhang, J. Huang, and Y. Wu, "Demand response management via real-time electricity price control in smart grids," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 7, pp. 1268–1280, July 2013.

[11] Y. Zhang and M. van der Schaar, "Structure-aware stochastic storage management in smart grids," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 6, pp. 1098–1110, Dec 2014.

[12] B. G. Kim, Y. Zhang, M. van der Schaar, and J. W. Lee, "Dynamic pricing and energy consumption scheduling with reinforcement learning," *IEEE Trans. Smart Grid*, vol. PP, no. 99, pp. 1–12, 2015.

[13] J. Zhu, *Optimization of power system operation*, John Wiley & Sons, 2009.

[14] N. P. Padhy, "Unit commitment-a bibliographical survey," *IEEE Trans. Power Syst.*, vol. 19, no. 2, pp. 1196–1205, 2004.

[15] C. Tekin and M. van der Schaar, "Distributed online learning via cooperative contextual bandits," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3700–3714, 2015.

[16] A. Slivkins, "Contextual bandits with similarity information," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2533–2568, 2014.

[17] C. Tekin and M. van der Schaar, "Active learning in context-driven stream mining with an application to image mining," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3666–3679, 2015.

[18] H.-S. Lee, C. Tekin, M. van der Schaar, and J.-W. Lee, "Online appendix for: Contextual learning for unit commitment with renewable energy sources," 2016, [Online]. Available: http://nrl.yonsei.ac.kr/tr/[GlobalSIP2016]CLUC.pdf.

[19] Pacific & Gas Electric, "Dynamic load profiles in California," [Online]. Available: http://www.pge.com/tariffs/energe_use_prices.shtml.

[20] P. Pinson, *Estimation of the uncertainty in wind power forecasting*, Ph.D. dissertation, Ecole des Mines de Paris, Paris, France, 2006.