Dynamic Pricing for Smart Grid with Reinforcement Learning

Byung-Gook Kim*, Yu Zhang†, Mihaela van der Schaar†, and Jang-Won Lee‡
*Samsung Electronics, Suwon, Korea
†Department of Electrical Engineering, UCLA, Los Angeles, USA
‡Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea

Abstract—**In the smart grid system, dynamic pricing can be an efficient tool for the service provider which enables efficient and automated management of the grid. However, in practice, the lack of information about the customers' time-varying load demand and energy consumption patterns and the volatility of electricity price in the wholesale market make the implementation of dynamic pricing highly challenging. In this paper, we study a dynamic pricing problem in the smart grid system where the service provider decides the electricity price in the retail market. In order to overcome the challenges in implementing dynamic pricing, we develop a reinforcement learning algorithm. To resolve the drawbacks of the conventional reinforcement learning algorithm such as high computational complexity and low convergence speed, we propose an approximate state definition and adopt virtual experience. Numerical results show that the proposed reinforcement learning algorithm can effectively work without a priori information of the system dynamics.**

I. Introduction

In the smart grid system, thanks to the real-time information exchange through communication networks, customers can schedule the operation of their appliances according to the change of electricity price via the automated energy management system equipped in households, which we refer to as *demand response* [1]. From the customers' perspective, previous works on the load scheduling focus on directly controlling the energy consumption of the residential appliances. For example, in our previous work [2], we proposed two different load scheduling algorithms for a collaborative and a non-collaborative smart grid system by taking into account the customers' bidirectional energy trading capability via electric vehicles. Most of these works aim at maximizing the social welfare of the smart grid system assuming that the pricing policies are predetermined in the electricity market by the service provider. Consequently, the service provider is regarded as a passive and indifferent entity whose role in the smart grid system is significantly limited.

On the contrary, from the service provider's perspective, dynamic pricing is an attractive tool that enables efficient grid operation in terms of both efficient energy consumption and automated management. Our paper is related to this second strand of literature. Specifically, we focus on a scenario where the service provider can adaptively decide the retail electricity price based on the customers' load demand level and the wholesale price such that it minimizes either the customers' disutility (in the case of a benevolent service provider) or its own cost (in the case of a profit-making service provider). Although dynamic pricing does not directly control each customer's load scheduling, the appropriate pricing can give considerable benefits to the smart grid system by encouraging the customers to consume energy in a more efficient way.

Recently, there have been several works on dynamic pricing for smart grid [3][4][5][6][7][8]. In [3] and [4], dynamic pricing problems were studied aiming at maximizing the social welfare. Considering a smart grid system with multiple residences and a single service provider, optimal dynamic pricing schemes were proposed based on the dual decomposition approaches. The authors in [5] focused on the smart grid system with non-cooperative customers where the conventional optimization approach cannot be applied to as in [3]. To overcome the lack of cooperation of customers, a simulated annealing-based dynamic pricing algorithm was developed. In a similar context, the authors in [6] modeled a dynamic pricing problem as a Stackelberg game where the service provider decides the retail price and each selfish customer decides the schedule for its appliances according to the price. In [7], the authors developed an incentive-based dynamic pricing scheme which allows the service provider to decide the incentive for the customers who shift their appliances' usage from peak hours to off-peak hours. In [8], the authors introduced a two-timescale dynamic pricing scheme to incorporate both the customers with day-ahead scheduling and the customers with real-time scheduling. To take into account the uncertainties of energy supply and demand, the authors formulated a Markov decision process (MDP) problem and developed an online algorithm.

Despite those previous efforts, there still exist several critical challenges in implementing dynamic pricing for demand response. First, in the practical smart grid system, it is not easy for the service provider to obtain the customer-side information such as their current load demand levels and the transition probability of the demand levels, and the customer-specific utility models including the willingness to purchase electric energy given their load demand level and retail price. Second, even if the service provider can obtain those information, it will surfer from various system dynamics and uncertainties. The service provider which lies between the utility company and the customers may not obtain the perfect information of those system dynamics a priori. Finally, the service provider is required to have the ability to estimate the impact of its current pricing decision on the customers' future behavior. In fact, the current price influences not only the customers' current energy consumption but also their energy consumption for the next several hours or the next day. However, it is not easy for the service provider to calculate the optimal price considering the future influence of the current price without the detail customer-side information. Thus, most of existing

This work was supported in part by Mid-career Researcher Program through NRF grant funded by the MSIP, Korea (2013R1A2A2A01069053), and by USA National Science Foundation (NSF) grant CCF-1218136.

works on dynamic pricing for smart grid have been studied in myopic approaches where the algorithms for dynamic pricing and demand side load scheduling are conducted within a given time period without considering the long-term performance of the smart grid system.

In order to overcome the aforementioned challenges of dynamic pricing, in this paper, we use reinforcement learning to allow the service provider to learn the behaviors of customers and the change of wholesale price to make an optimal pricing decision. We consider various stochastic dynamics of the smart grid system including the customers' dynamic demand generation and energy consumptions, and wholesale price changes. Based on the considered system model, we formulate an MDP problem where the service provider decides the retail electricity price based on the observed system state transition to minimize its expected total cost or the customers' disutility. Contrary to the previous works [3]-[8] with simplified models, in this paper, we consider a more realistic system model to incorporate the customers' demand generation and load demand change as well as the wholesale market dynamic where the wholesale electricity price can be changed by the utility company at each time-slot. To solve the MDP problem without a priori information about the change of the customers' load demand level, we adopt the Q-learning algorithm, and to resolve the existing drawbacks of the conventional Q-learning algorithm, we propose the following two improvements. First, to reduce the complexity of the Q-learning algorithm which mainly comes from the large number of customers, we propose an alternative state definition based on the observed total energy consumption. Second, to improve the learning speed, we adopt virtual experience in Q-learning updates.

The rest of this paper is organized as follows. In Section II, the system model is presented. In Section III, we define the dynamic pricing problem and develop the reinforcement learning-based dynamic pricing algorithm. We provide numerical results in Section IV and finally conclude in Section V.

II. System Model

We consider a smart grid system which consists of one service provider and a set of customers $\mathcal{I}$ as in Fig. 1. The smart grid system operates in a time-slotted fashion, where each time-slot has an equal duration. At each time-slot t, the service provider buys electric energy from the utility company through a wholesale electricity market and provides it to the customers through a retail electricity market. In the retail electricity market, at each time-slot t, the service provider determines the retail pricing function $a^t : \mathbb{R}_+ \to \mathbb{R}_+$ and charges each customer i an electricity bill $a^t(e_i^t)$, where e_i^t denotes customer i's energy consumption at time-slot t. We define the set of retail pricing functions as $\mathcal{A}$ and assume that the number of retail pricing functions, $|\mathcal{A}|$, is finite.

At each time-slot, each customer generates its electricity load demand and decides the amount of energy consumption based on its current load demand level and the retail pricing function. We assume that the customers' average demand generation rate and the wholesale pricing function at a time-slot can vary depending on its actual time in a day. Moreover, this time-dependency of customers' energy consumption may influence the utility company's decision on the wholesale pricing function. To model this time-dependency of the demand generation rate and the wholesale pricing function, we introduce a set of periods $\mathcal{H} = \{0, 1, \cdots, H-1\}$ each of which represents an actual time in a day. We map each time-slot t to one period $h \in \mathcal{H}$ denoting the period at time-slot t by h^t. We assume that the sequence of periods $h^t, t = 0, 1, 2, \cdots$ is predetermined and repeated every day. For example, if one day consists of $H = 24$ periods (i.e., 24 hours), each time-slot t is mapped to one period in $\mathcal{H} = \{0, 1, \cdots, 23\}$ and the mapping between time-slots and periods can be represented as

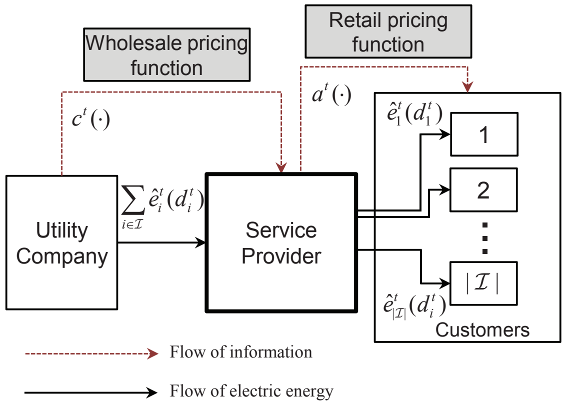

Fig. 1. Smart grid system.

$$h^t = \mod (t, H), \quad \forall t \geq 0. \tag{1}$$

A. Model of Customer's Response

In each time-slot, each customer has an *accumulated* load demand [1], which is defined as the total amount of energy that it wants to consume for its appliances in that time-slot. We denote the amount of the accumulated load demand of customer i at time-slot t by $d_i^t \in \mathcal{D}_i$, where $\mathcal{D}_i$ is the set of customer i' accumulated load demand levels. Once customer i consumes energy e_i^t at time-slot t, it implies that the corresponding amount of customer i's load demand is satisfied and the rest of the accumulated load demand $d_i^t - e_i^t$ is not satisfied and we call it the *remaining* load demand. When the remaining load demand of a customer is greater than 0, it causes some degree of dissatisfaction to the customer at that time-slot. To capture this dissatisfaction from the remaining load demand, we introduce a disutility function for $u_i : \mathbb{R}_+ \to \mathbb{R}_+$ for each customer i. We assume that $u_i(\cdot)$ is an increasing function of the remaining load demand $d_i^t - e_i^t$.

Based on the disutility $u_i(d_i^t - e_i^t)$ and the electricity bill $a^t(e_i^t)$ that customer i has to pay to the service provider, we define customer i's cost at each time-slot t as

$$\phi_i^t(d_i^t, e_i^t) = u_i(d_i^t - e_i^t) + a^t(e_i^t). \tag{2}$$

We assume that each customer tries to minimize its cost at each time-slot by deciding the amount of its energy consumption and let $\hat{e}_i^t(d_i^t)$ denote customer i's energy consumption decision that minimizes its cost [2], i.e.,

$$\hat{e}_i^t(d_i^t) = \underset{0 \leq e_i^t \leq \min(e_i^{max}, d_i^t)}{\operatorname{argmin}} \phi_i^t(d_i^t, e_i^t), \tag{3}$$

[1] For the convenience, 'accumulated load demand' and 'load demand' are used interchangeably in the rest of this paper.

[2] Customer i's energy consumption decision is also a function of the retail pricing function a^t. However, for the simple expression, we represent it as $\hat{e}_i^t(d_i^t)$.

where e_i^{max} is the maximum amount of energy that customer i can consume at each time-slot due to physical limitations of the grid.

We assume that a portion of each customer's remaining load demand at a time-slot is carried forward to the next time-slot. We call the corresponding load demand the *demand backlog* and represent it as $\lambda_i(d_i^t - \hat{e}_i^t(d_i^t))$, where $0 \leq \lambda_i \leq 1$ is the backlog rate of load demand that determines the amount of demand backlog of customer i. At each time-slot t, each customer i randomly generates its *new* load demand, $D_i^t(h^t)$, and its distribution is assumed to be dependent on the current period h^t. At the beginning of each time-slot $t+1$, according to the demand backlog at the previous time-slot t, $\lambda_i(d_i^t - \hat{e}_i^t(d_i^t))$, and the newly generated load demand, $D_i^{t+1}(h^{t+1})$, customer i's accumulated load demand d_i^{t+1} is updated as

$$d_i^{t+1} = \lambda_i(d_i^t - \hat{e}_i^t(d_i^t)) + D_i^{t+1}(h^{t+1}). \tag{4}$$

Here, if $\lambda_i = 0$, no remaining demand at a time-slot is carried forward generating no demand backlog, whereas if $\lambda_i = 1$, all the remaining demand at a time-slot is carried forward to the next time-slot.

It is worth noting that the transition probability of the load demand from d_i^t to d_i^{t+1} depends only on the accumulated load demand d_i^t, the period h^t, and the retail pricing function a^t at time-slot t, and we represent it as $p_{d_i}(d_i^{t+1}|d_i^t, h^t, a^t)$.

B. Wholesale Electricity Market

At each time-slot t, the service provider buys electric energy, which corresponds to the total amount of energy consumption of customers, $\sum_{i\in\mathcal{I}} \hat{e}_i^t(d_i^t)$, from the utility company in the wholesale electricity market as illustrated in Fig. 1. The utility company charges the service provider an wholesale electricity cost based on a wholesale pricing function $c^t : \mathbb{R}_+ \to \mathbb{R}_+$, where c^t is a function of the total amount of energy consumption $\sum_{i\in\mathcal{I}} \hat{e}_i^t(d_i^t)$. We assume that c^t is selected among a finite number of wholesale pricing functions in set $\mathcal{C}$ and its transition probability from c^t to c^{t+1} depends on the current wholesale pricing function, c^t, and the current period, h^t, and thus it can be represented as $p_c(c^{t+1}|c^t, h^t)$.

We define the service provider's cost function at each time-slot t as a function of the customers' load demand vector $\bar{d}^t = [d_i^t]_{i\in\mathcal{I}}$, the wholesale electricity pricing function c^t, and the retail pricing function a^t, i.e.,

$$\psi^t(\bar{d}^t, c^t, a^t) = c^t\Big(\sum_{i\in\mathcal{I}} \hat{e}_i^t(d_i^t)\Big) - \sum_{i\in\mathcal{I}} a^t(\hat{e}_i^t(d_i^t)), \tag{5}$$

where the first term denotes the total wholesale electricity cost that the service provider has to pay to the utility company and the second term denotes the service provider's revenue from selling energy to the customers.

III. Reinforcement Learning Algorithm

In this section, based on the smart grid system introduced in the previous section, we first formulate a dynamic pricing problem in the framework of MDP. Then, by using reinforcement learning, we develop an efficient and fast dynamic pricing algorithm which does not require the information about the system dynamics and uncertainties.

A. Problem Formulation

We formulate the dynamic pricing problem in the smart grid system as an MDP problem, which is defined by a set of decision maker's actions, a set of system states and their transition probabilities, and a system cost function for the decision maker. In our MDP problem, the decision maker is the service provider whose action is choosing a retail pricing function $a^t \in \mathcal{A}$ at each time-slot t. We define the state of our smart grid system at time-slot t as the combination of the accumulated load demands vector, $\bar{d}^t$, the current period h^t, and the wholesale pricing function, c^t, i.e.,

$$s^t = (\bar{d}^t, h^t, c^t) \in \mathcal{S}, \tag{6}$$

where $\mathcal{S} = \prod_{i\in\mathcal{I}} \mathcal{D}_i \times \mathcal{H} \times \mathcal{C}$. Since the transition of each customer's load demand (from d_i^t to d_i^{t+1}), that of the period (from h^t to h^{t+1}), and that of the wholesale pricing function (from c^t to c^{t+1}) depend only on the state s^t and action a^t at time-slot t, the sequence of states $\{s^t, t = 0, 1, 2, \cdots\}$ follows a Markov decision process with action a^t. The transition probability from state $s^t = (\bar{d}^t, h^t, c^t)$ to state $s^{t+1} = (\bar{d}^{t+1}, h^{t+1}, c^{t+1})$ with given action a^t can be represented as

$$\begin{aligned} p_s(s^{t+1}|s^t, a^t) =& p_h(h^{t+1}|h^t) p_c(c^{t+1}|c^t, h^t) \\ &\times \prod_{i\in\mathcal{I}} p_{d_i}(d_i^{t+1}|d_i^t, h^t, a^t), \end{aligned} \tag{7}$$

where $p_h(h^{t+1}|h^t)$ denotes the transition probability of the period from h^t to h^{t+1}. We define the system cost for the service provider at each time-slot t as the weighted sum of the service provider's cost and the customers' cost at the time-slot:

$$r^t(s^t, a^t) = (1-\rho)\psi^t(\bar{d}^t, c^t, a^t) + \rho \sum_{i\in\mathcal{I}} \phi_i^t(d_i^t, \hat{e}_i^t), \tag{8}$$

where ρ denotes the weighting factor that determines the relative importance between the service provider's cost and the customers' cost.

We denote the stationary policy that maps states to actions (retail pricing functions) by $\pi : \mathcal{S} \to \mathcal{A}$, i.e, $a^t = \pi(s^t)$. The objective of our dynamic pricing problem is to find an optimal policy π^* for each state $s \in \mathcal{S}$ that minimizes the expected discounted system cost of the service provider as in the following MDP problem $(\mathbf{P})$:

$$(\mathbf{P}) : \min_{\pi:\mathcal{S}\to\mathcal{A}} E\left[\sum_{t=0}^{\infty} (\gamma)^t r^t(s^t, \pi(s^t))\right], \tag{9}$$

where $0 \leq \gamma < 1$ is the discount factor which represents the relative importance of the future system cost compared with the present system cost.

The optimal stationary policy π^* can be well defined by using the optimal *action-value function* $Q^* : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ which satisfies the following Bellman optimality equation:

$$Q^*(s, a) = r(s, a) + \gamma \sum_{s'\in\mathcal{S}} p(s'|s, a) V^*(s'), \tag{10}$$

where $V^*(s')$ is the optimal *state-value function* [9], which is defined as

$$V^*(s') = \min_{a\in\mathcal{A}} Q^*(s', a), \forall s' \in \mathcal{S}. \tag{11}$$

Since $Q^*(s,a)$ is the expected discounted system cost with action a in state s, we can obtain the optimal stationary policy as

$$\pi^*(s) = \underset{a \in \mathcal{A}}{\operatorname{argmin}}\, Q^*(s,a). \tag{12}$$

In this paper, we use the well-known Q-learning algorithm to solve our MDP problem $(\mathbf{P})$ without information of state transition probabilities. We refer the readers to [10] for more detail on the Q-learning algorithm. In the following subsections, in order to resolve the critical issues which make it difficult to apply the conventional Q-learning algorithm to our smart grid system, we propose an alternative state definition based on the observed total energy consumption as well as adopt virtual experience in Q-learning updates.

B. Energy Consumption-Based Approximate State (EAS)

When the size of the state space is large, the Q-learning algorithm requires not only a large memory space to store the state-action function $Q(s,a)$, but also a long time to converge. Moreover, in the practical smart grid system, it is difficult for the service provider to acquire or use the information about the customers' current load demands due to privacy. In order to resolve these difficulties, in this section, we propose an alternative definition of the system state, which is based on the observed total energy consumption, $\sum_{i\in\mathcal{I}} \hat{e}_i^{t-1}$, and the previously chosen action a^{t-1}. For notational convenience, we will omit d_i^t in $\hat{e}_i^t(d_i^t)$ in the rest of this section.

The main idea of this alternative state definition comes from the fact that, since each customer's disutility function u_i is a decreasing function of $\hat{e}_i^t$, from the load demand update process in (4), the retail pricing function and each customer i's energy consumption at time-slot $t-1$ in a tuple $(\hat{e}_i^{t-1}, a^{t-1})$ characterizes the past accumulated load demand d_i^{t-1}. Hence, if the service provider knows new load demand $D_i^t(h^t)$ at time-slot t, it can regard a different tuple $(\hat{e}_i^{t-1}, a^{t-1}, D_i^t(h^t))$ as a different actual accumulated load demand of customer i at time-slot t. Similarly, once a tuple $(\sum_{i\in\mathcal{I}} \hat{e}_i^{t-1}, a^{t-1}, \sum_{i\in\mathcal{I}} D_i^t(h^t))$ is observed by the service provider, it approximately reflects the customers' overall load demands at time-slot t. It is worth noting that since $D_i^t(h^t)$ is independent random variable for each customer i, by the law of the large number, the average of the sum of new load demands, $\sum_{i\in\mathcal{I}} D_i^t(h^t)/|\mathcal{I}|$, goes to its expected value as the number of customers gets larger. This implies that in the practical smart grid system with a large number of customers, a tuple $(\sum_{i\in\mathcal{I}} \hat{e}_i^{t-1}, a^{t-1})$ provides enough information for the service provider to infer the customers' overall load demand level at time-slot t. Hence, instead of the original state s^t in (6), we can use a new state definition based on the observed energy consumption by which the service provider does not need to know either each customer's load demand or its disutility function.

To reduce the number of system states, we discretize the observed energy consumption $\sum_{i\in\mathcal{I}} \hat{e}_i^{t-1}$ into a finite number of energy levels in $\mathcal{E}$ [3] The set of observed energy consumption level by using a quantization operation $q_{\mathcal{E}}(\cdot)$. Then, we refer to tuple $(q_{\mathcal{E}}(\sum_{i\in\mathcal{I}} \hat{e}_i^{t-1}), a^{t-1})$ as the *approximate demand* at time-slot t and represent it as

$$d_{app}^t = \Big(q_{\mathcal{E}}\Big(\sum_{i\in\mathcal{I}} \hat{e}_i^{t-1}\Big), a^{t-1}\Big). \tag{13}$$

Algorithm 1 Q-Learning Algorithm with Virtual Experience

```
Initialize $Q$ arbitrarily, $t = 0$
for each time-slot $t$
    Choose $a^t$ according to policy $\pi(x^t)$
    Take action $a^t$, observe system cost $r(x^t, a^t)$ and next state $x^{t+1}$
    Obtain experience tuple $\sigma^{t+1} = (x^t, a^t, r^t, x^{t+1})$
    Generate set of virtual experience tuples $\theta(\sigma^{t+1})$
    for each virtual experience tuple $\tilde{\sigma}^{t+1} \in \theta(\sigma^{t+1})$
        $v = r(x^t, a^t) + \gamma \max_{a' \in \mathcal{A}} Q(x^{t+1}, a') - Q(x^t, a^t)$
        $Q(x^t, a^t) \leftarrow Q(x^t, a^t) + \alpha^t v$
    end
end
```

Based on the approximate demand, we now define the *energy consumption-based approximate state (EAS)* of the smart grid system as

$$x^t = (d_{app}^t, h^t, c^t) \in \mathcal{X}, \tag{14}$$

where $\mathcal{X} = \mathcal{E} \times \mathcal{A} \times \mathcal{H} \times \mathcal{C}$ denotes the set of the EASs. Note that EAS extremely reduces the number of states from $|\mathcal{S}| = |\prod_{i\in\mathcal{I}} \mathcal{D}_i \times \mathcal{H} \times \mathcal{C}|$ to $|\mathcal{X}| = |\mathcal{E} \times \mathcal{A} \times \mathcal{H} \times \mathcal{C}|$, while allowing the service provider to easily infer the customers' current load demands level without using direct signaling from the customers. Now, we can simply substitute the original state definition s^t by EAS x^t in the Q-learning algorithm.

C. Accelerated Learning using Virtual Experience

Although the EAS x^t significantly reduces the state space, the learning speed of the Q-learning algorithm might be seriously limited by its inherent structure in which only one state-action pair is updated at each time-slot. In this subsection, in order to improve the speed of the Q-learning algorithm, we adopt virtual experience which was introduced in [11]. The Q-learning algorithm with virtual experience enables the service provider to update multiple state-action pairs at each time-slot by exploiting a priori known partial information of the state transition probability $p_x(x^{t+1}|x^t, a^t)$. In this subsection, we consider the case where the service provider knows the transition probability of the wholesale pricing function $p_c(c^{t+1}|c^t, h^t)$ a priori. Note that this is not a too restrictive assumption because the service provider can gather sufficient data for the transition probability of the wholesale pricing function while participating in the wholesale electricity market.

We first define the *experience tuple* (ET) observed by the service provider at time-slot $t+1$ as $\sigma^{t+1} = (x^t, a^t, r^t, x^{t+1})$, where r^t is the observed system cost. Then, given the actual ET σ^{t+1}, we define a set of *virtual experience tuples* (virtual ETs) $\theta(\sigma^{t+1})$, which are statistically equivalent [4] to the actual ET σ^{t+1}, i.e.,

$$\theta(\sigma^{t+1}) = \left\{ \tilde{\sigma}^{t+1} \,\middle|\, \begin{array}{l} \tilde{d}_{app}^t = d_{app}^t, \tilde{h}^t = h^t, \\ \tilde{a}^t = a^t, \tilde{r}^t = r(\tilde{c}^t), \\ p_c(\tilde{c}^{t+1}|\tilde{c}^t, \tilde{h}^t) = p_c(c^{t+1}|c^t, h^t) \end{array} \right\}, \tag{15}$$

where $r(c)$ represents the system cost that is virtually calculated by using an arbitrary wholesale pricing function c. In our

[3] In our system model, the method of discretization of the observed energy consumption is not limited to a specific method. A simple example is to use an equally divided levels between the maximum and minimum amounts of the energy consumption.

[4] An ET $\tilde{\sigma}^{t+1} = (\tilde{x}^t, \tilde{a}^t, \tilde{r}^t, \tilde{x}^{t+1})$ is said to be statistically equivalent to ET $\sigma^{t+1} = (x^t, a^t, r^t, x^{t+1})$ if $p_x(\tilde{x}^{t+1}|\tilde{x}^t, \tilde{a}^t) = p_x(x^{t+1}|x^t, a^t)$ and the system cost $\tilde{r}^t$ can be calculated by using σ^{t+1}.

TABLE I
COMPLEXITY COMPARISON OF THREE DIFFERENT ALGORITHMS.

	Learning update complexity per iteration	Memory complexity
Q-learning with original state	$O(\vert\mathcal{A}\vert)$	$O(\prod_{i\in\mathcal{I}} \vert\mathcal{D}_i\vert\vert\mathcal{H}\vert\vert\mathcal{C}\vert\vert\mathcal{A}\vert)$
Q-learning with EAS	$O(\vert\mathcal{A}\vert)$	$O(\vert\mathcal{E}\vert\vert\mathcal{H}\vert\vert\mathcal{C}\vert\vert\mathcal{A}\vert^2)$
Q-learning with EAS and virtual experience	$O(\vert\theta(\hat{\sigma})\vert\vert\mathcal{A}\vert)$	$O(\vert\mathcal{E}\vert\vert\mathcal{H}\vert\vert\mathcal{C}\vert\vert\mathcal{A}\vert^2)$

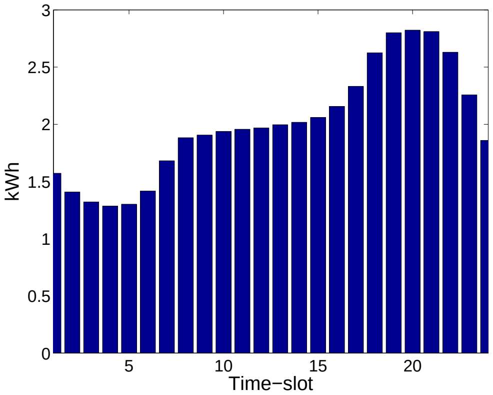

Fig. 2. Load demand profile.

smart grid system, if d^t_{app} and h^t are fixed, the system cost r^t can be easily calculated for an arbitrary wholesale pricing functions $c'^t \in \mathcal{C}$ by applying the same energy consumption $\sum_{i\in\mathcal{I}} \hat{e}^t_i$ to (8). Moreover, since the transition probability of the wholesale pricing function $p_c(c^{t+1}|c^t, h^t)$ is independent of approximate demand d^t_{app} and retail electricity pricing function a^t, we can easily generate the set of virtual ETs $\theta(\sigma^{t+1})$ in (15) from the observed actual ET σ^{t+1}. While the observed ET σ^{t+1} is used to update only one state-action function $Q(x^t, a^t)$ in the conventional Q-learning, by using the virtual ETs, the Q-learning algorithm can update multiple state-action pairs at each time-slot. The Q-learning algorithm with virtual experience is outlined in Algorithm 1. Lines 3-5 describe the operation of the conventional Q-learning where the service provider obtains the experience tuple σ^{t+1}. Then, in line 6, based on σ^{t+1}, a set of its virtual experiences is generated. In lines 7-10, the action-value function $Q(x^t, a^t)$ is updated for all virtual experiences in σ^{t+1}.

The complexity of the proposed reinforcement learning algorithms are summarized in Table I. Although the Q-learning algorithm with virtual experience has a higher update complexity than the Q-learning algorithm without virtual experience, as we will show in Section IV, it significantly reduces the number of time-slots needed to converge, which is regarded as a more important aspect than the computational complexity at each time-slot in reinforcement learning algorithms.

IV. NUMERICAL RESULTS

In this section, we provide numerical results to evaluate the performance of our dynamic pricing algorithm. One day consists of 24 time-slots each of which lasts for one hour. Hence, the set of periods is given as $\mathcal{H} = \{0, 1, \cdots, 23\}$ and the mapping between time-slots and periods is given as $h^t = \text{mod }(t, 24)$.

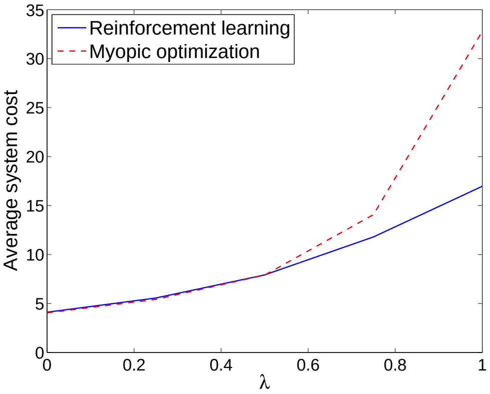

Fig. 3. Performance comparison of our reinforcement learning algorithm and the myopic optimization algorithm varying λ.

We consider a smart grid system with 20 customers. The newly generated load demand of customer i, $D^t_i(h^t)$, follows a Poisson distribution with expected value ω_{i,h^t}, which is proportional to the hourly average load shapes of residential electricity services in California [12] as shown in Fig. 2. All customers have the same backlog rate, i.e., $\lambda_i = \lambda, \forall i \in \mathcal{I}$. Each customer i's disutility function $u_i(d^t_i - e^t_i)$ is given as

$$u_i(d^t_i - e^t_i) = \kappa_i \times (d^t_i - e^t_i)^2, \tag{16}$$

where κ_i is a constant that represents customer i's disutility sensitivity to its remaining demand. Here, we let $\kappa_i = \kappa = 0.1, \forall i \in \mathcal{I}$. We model the wholesale pricing function c^t as a quadratic function of the total energy consumption $\sum_{i\in\mathcal{I}} \hat{e}^t_i$ as in [3],[4], and [6] :

$$c^t\Big(\sum_{i\in\mathcal{I}} \hat{e}^t_i\Big) = \mu^t \times \sum_{i\in\mathcal{I}} \hat{e}^t_i + \nu^t_{h^t} \times \Big(\sum_{i\in\mathcal{I}} \hat{e}^t_i\Big)^2. \tag{17}$$

We set $\mu^t = 0.02, \forall t$ and $\nu^t_{h^t}$ to be a random variable whose expected value, v_{h^t}, changes according to the corresponding period h^t based on the hourly average load shape in Fig. 2. With a given period h^t, $\nu^t_{h^t}$ is uniform randomly chosen among values in $\{0.25v_{h^t}, 0.5v_{h^t}, \cdots, 1.75v_{h^t}\}$. The discount factor γ in problem $(\mathbf{P})$ is fixed to 0.95. The retail pricing function a^t is a linear function of the energy consumption $\hat{e}^t_i$, i.e.,

$$a^t(\hat{e}^t_i) = \chi^t \hat{e}^t_i, \tag{18}$$

where the coefficient χ^t can be chosen among set $\{0.2, 0.4, \cdots, 1.0\}$ each element of which is directly mapped to one retail pricing function in $\mathcal{A}$.

We first evaluate the performance of the dynamic pricing algorithm by comparing it with that of the *myopic optimization algorithm*. In the myopic optimization algorithm, the service provider chooses an action with the lowest expected instantaneous system cost, by updating the state-action function $Q(s, a)$ similarly to the Q-learning update with a discount factor $\gamma = 0$. This implies that the myopic optimization algorithm focuses only on the immediate system cost without considering the impact of the current action on the future system cost. In Fig. 3, we show the average system costs of those two dynamic pricing algorithms by changing the backlog rate, λ, from 0 to 1. We set $\rho = 0.5$ which corresponds to the case where the service provider aims at minimizing the sum of the total disutility and the wholesale cost. We can observe that the average system costs increase as λ increases

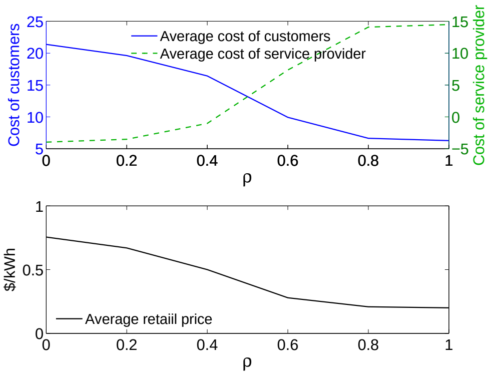

Fig. 4. Impact of the weighting factor ρ on the performances of customers and service provider.

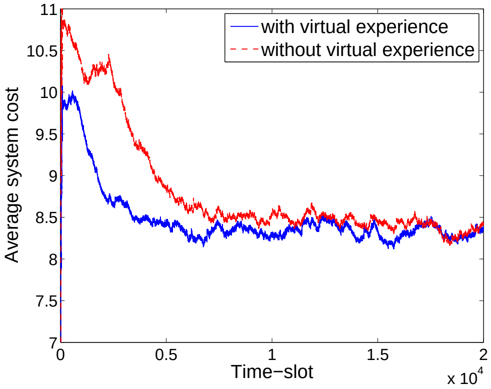

Fig. 5. Impact of virtual experience on the learning speed of the Q-learning algorithm.

in both dynamic pricing algorithms because with a higher backlog rate, the accumulated load demand causes a higher disutility. We also observe that the performance gap between two algorithms increases as λ increases. With a low backlog rate, the remaining backlog is not carried forward to the next time-slot. In this case, the solution of our reinforcement learning algorithm is the same as that of the myopic optimization algorithm. On the contrary, in the case with a higher backlog rate, the remaining backlog is carried forward to the next time-slot. Hence, the service provider's pricing decision at a time-slot influences the accumulated load demand in the future, and thus its future system cost. Due to this difference, especially when λ is large, our algorithm achieves better performance than the myopic optimization problem which considers only the current system cost.

To study the impact of the weighting factor ρ, in Fig. 4, we show the cost of customers, that of the service provider, and the average retail price with varying ρ from 0 to 1. We set $\lambda = 0.5$. We can observe that as ρ increases, the service provider reduces the average retail price, the cost of customers decreases, and the cost of the service provider increases. For example, in the case with $\rho = 0$, the service provider aims at minimizing its own cost. Hence, the service provider does not consider the customers' disutility and chooses relatively high prices to reduce the wholesale cost which contributes to most of its own cost. In the case with $\rho = 1$, the service provider aims at minimizing the customers' cost. Hence, the service provider chooses relatively low prices to provide electric energy to the customers at a low retail price as possible.

In Fig. 5, we compare the learning speed of our reinforcement learning algorithm with virtual experience to the conventional Q-learning algorithm without virtual experience. We set $\lambda = 0.5$ and $\rho = 0.5$. We can observe that our algorithm with virtual experience achieves the near optimal average system cost after about 3,000 time-slots, while the conventional Q-learning algorithm shows a worse learning speed. This means that even if the stochastic characteristic of the system dynamics vary in time, the proposed reinforcement learning algorithm can quickly adapt to the time-varying environment by exploiting virtual experience update.

V. Conclusion

In this paper, we studied a dynamic pricing problem for the smart grid system where the service provider can adaptively decide the electricity price according to the customers' load demand levels and the wholesale price. We developed a reinforcement learning-based dynamic pricing algorithm that enables efficient dynamic pricing without requiring the perfect information about the system dynamics a priori. To resolve the existing drawbacks of the conventional reinforcement learning algorithm, we proposed two improvements: energy consumption-based approximate state definition and the adoption of virtual experience update in the conventional Q-learning algorithm. Numerical results show that the reinforcement learning-based dynamic pricing achieves a higher long-term performance compared to the myopic optimization approach especially in the system where the customers have a high demand backlog rate. The results also show that our algorithm results in an improved learning speed due to the alternative state definition and virtual experience implying that our dynamic pricing algorithm can be applied to the practical smart grid system.

References

[1] M. H. Albadi and E. El-Saadany, "Demand response in electricity markets: An overview," in *IEEE Power Engineering Society General Meeting*, 2007.

[2] 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 Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1219–1234, 2013.

[3] P. Samadi, A.-H. Mohsenian-Rad, R. Schober, V. Wong, and J. Jatskevich, "Optimal real-time pricing algorithm based on utility maximization for smart grid," in *IEEE SmartGridComm*, 2010.

[4] P. Tarasak, "Optimal real-time pricing under load uncertainty based on utility maximization for smart grid," in *IEEE SmartGridComm*, 2011.

[5] L. P. Qian, Y. Zhang, J. Huang, and Y. Wu, "Demand response management via real-time electricity price control in smart grids," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1268–1280, 2013.

[6] C. Chen, S. Kishore, and L. Snyder, "An innovative RTP-based residential power scheduling scheme for smart grids," in *IEEE ICASSP*, 2011.

[7] C. Joe-Wong, S. Sen, S. Ha, and M. Chiang, "Optimized day-ahead pricing for smart grids with device-specific scheduling flexibility," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 6, pp. 1075–1085, 2012.

[8] M. He, S. Murugesan, and J. Zhang, "Multiple timescale dispatch and scheduling for stochastic reliability in smart grids with wind generation integration," in *IEEE INFOCOM*, 2011, pp. 461–465.

[9] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, May 1996.

[10] C. Watkins, "Learning from delayed rewards," Ph.D. dissertation, Cambridge University, 1989.

[11] N. Mastronarde and M. van der Schaar, "Joint physical-layer and system-level power management for delay-sensitive wireless communications," *IEEE Transactions on Mobile Computing*, vol. 12, no. 4, pp. 694–709, 2013.

[12] *Dynamic Load Profiles in California*. Pacific Gas & Electric. [Online]. Available: http://www.pge.com/tariffs/energy_use_prices.shtml