Demand Response for Home Energy Management Using Reinforcement Learning and Artificial Neural Network

Renzhi Lu, *Member, IEEE*, Seung Ho Hong, *Senior Member, IEEE*, and Mengmeng Yu, *Member, IEEE*

***Abstract*—Ever-changing variables in the electricity market require energy management systems (EMSs) to make optimal real-time decisions adaptively. Demand response (DR) is the latest approach being used to accelerate the efficiency and stability of power systems. This paper proposes an hour-ahead DR algorithm for home EMSs. To deal with the uncertainty in future prices, a steady price prediction model based on artificial neural network is presented. In cooperation with forecasted future prices, multi-agent reinforcement learning is adopted to make optimal decisions for different home appliances in a decentralized manner. To verify the performance of the proposed energy management scheme, simulations are conducted with non-shiftable, shiftable, and controllable loads. Experimental results demonstrate that the proposed DR algorithm can handle energy management for multiple appliances, minimize user energy bills, and dissatisfaction costs, and help the user to significantly reduce its electricity cost compared with a benchmark without DR.**

***Index Terms*—Artificial intelligence, reinforcement learning, artificial neural network, demand response, home energy management.**

I. Introduction

CONTINUAL changes in variables relevant to the electricity market, such as the electricity price, energy consumption and dynamic operation, require the energy management systems (EMSs) to take optimal real-time actions instantly and adaptively [1]. With the modern advances in information and communication technologies (ICTs), and smart metering infrastructure, demand response (DR) becomes a significant role in terms of promoting the reliability and efficiency of energy systems, as it affords the capacity to balance electricity supply and demand incongruity by regulating elastic loads from the demand side [2]. A well-designed DR scheme in an EMS can have significant positive effects on society, such as improving human comfort levels, facilitating the accommodation of renewable resources, reducing worldwide energy expenditure, and reducing reliance on fuel resources associated with high carbon emissions [3], [4].

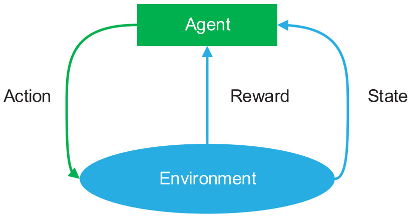


Fig. 1. Reinforcement learning setup.

Price-based DR programs considering the energy consumption of domestic electric appliances have been widely studied recently. For example, in [5]–[7], several home energy management system (HEMS) structures under an hourly pricing DR were developed to determine the optimal appliance scheduling based on a mixed-integer linear programming(MILP) approach, with the aim of reducing user costs and enhancing energy efficiency. The work described in [8] proposed a price-based DR algorithm for obtaining optimal control of appliances in a home via virtual electricity trading using a Stackelberg game, in which the energy management center of home is the virtual retailer and offers virtual retail prices, according to which appliances can buy energy. In [9], [10], two further price-based DR schemes for HEMSs were proposed to minimize the total energy cost and maximize user comfort, considering the uncertainties in energy storage system availability and small-scale renewable energy generation. At present, most HEMSs still rely on traditional methods, i.e., deterministic rules or abstract models such as MILP, which can be criticized in three main respects: (1) Deterministic rules for the handling of volatile energy systems cannot assure optimality, since any alterations in variables could lead to financial losses; (2) Abstract models are commonly only estimations of real situations and thus may be unrealistic compared to actual energy systems, as the effectiveness of abstract models is greatly limited by the operator's skill; (3) MILP-based or game theoretic-based optimization solutions are centralized, restricted and lacking in scalability, in terms of the high number of integer or binary variables in large-scale energy systems.

Manuscript received October 26, 2018; revised December 17, 2018 and March 3, 2019; accepted March 30, 2019. Date of publication April 4, 2019; date of current version October 30, 2019. This work was supported in part by the National Research Foundation of Korea through the Framework of International Cooperation Programs (Korea–China) under Grant NRF-2016K2A9A2A11938310 and Grant NRF-2018K1A3A1A61026320. Paper no. TSG-01619-2018. *(Corresponding author: Seung Ho Hong.)*

The authors are with the Department of Electronic Engineering, Hanyang University, Ansan 15588, South Korea (e-mail: lofky7@gmail.com; shhong@hanyang.ac.kr; ymm929@gmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2019.2909266

Over the past few years, with the rapid evolution of artificial intelligence (AI), much attention has been devoted to the use of AI for optimal decision-making. Some breakthroughs in applying AI to solve realistic problems have been reported: two representative successful cases are AlphaGO and AlphaGO Zero, introduced by Google, which have demonstrated that reinforcement learning (RL) has excellent decision-making ability because of its capacity to solve problems in the absence of initial knowledge of the environment. Fig. 1 shows the general architecture of RL; an agent and its environment interact via a sequence of discrete time steps and, at each step, the agent selects an action to be sent to the environment. Accordingly, the agent obtains a reward and also the environment changes to a new state. RL seeks to create a map between agent actions and states to maximize the total rewards, relying on the knowledge of the agent, obtained via direct interaction with the environment, in an unsupervised manner [11]. According to the distinct features of "model-free" and "no need for prior domain knowledge" paradigms, RL has become a powerful tool to optimize the control of energy systems that must deal with continous changes in several variables, e.g., intermittent availability of renewable resources, dynamic electricity prices; and the changes in energy consumption amounts. A practical pioneer of this approach is Google DeepMind who successfully applied a RL-based scheme to reduce the electricity bill associated with data center cooling by 40% [12]; which is another key factor for motivating the application of RL technology to energy systems.

Some research has also been done on adopting RL for solving decision-making problems in energy management. For example, in [13], [14], the authors applied RL-based algorithms to energy trading games among different strategic players with incomplete information, enabling each player used the learning scheme to choose a strategy to trade energy in an independent market, so as to maximize the average revenue. The studies in [15], [16] proposed batch RL algorithms to schedule controllable loads, such as the electric water heater or heat pump thermostat, under a day-ahead pricing scheme without any expert knowledge of the system dynamics or solutions. In [17]–[19], RL algorithms were used to obtain an energy consumption plan for electric vehicles (EVs) and the plan was then put into operation; wherein EV charging was controlled during operation using a heuristic scheme, and the outcomes of the charging policy acquired via RL. Most recently, the authors of [20], [21] presented a price-based DR scheme linking the service provider to its customers via RL methodology, where the scheme was modeled as a Markov decision process (MDP), then Q-learning was used to make optimal decisions. However, despite these efforts, there still exist two significant limitations. First, most studies focused on only one kind of appliance such as thermostatically control load or EV that did not consider how the proposed learning algorithms would enable decision-making when dealing with multiple types of appliances. Second, all studies considered day-ahead energy management, but the hour-ahead DR exhibits greater potential for balancing power systems due to the dynamic constraints associated with energy generation and the uncertainty in prediction [22].

Given the above mentioned issues, this paper proposes an hour-ahead DR algorithm for multiple appliances in a HEMS. Specifically, because of the inherent nature in hour-ahead electricity price market, the customer accesses only one price for the current hour. To deal with the uncertainty in future prices, a stable price forecasting model is presented, which is implemented by artificial neural network (ANN). Price forecasting has become an important topic in electrical engineering over the past few years, and several implementation methods have been attempted. The ANN approach is comparatively easy to implement and shows good performance, being less time-consuming than other techniques, such as the ARIMA model [23]. Each time the new electricity price is obtained, the ANN model predicts future prices, and this process is repeated hourly to the end of the day. Furthermore, in cooperation with the forecasted future prices, multi-agent RL is adopted to make optimal decisions for different appliances in a decentralized manner, to minimize the user energy bill and degree of discomfort. Here, each appliance has an agent, and RL is used for decision-making in the context of uncertainty regarding the price information and load demand of the appliances. Thereby, the computational load is also shifted from a central optimizer to a set of intelligent agents that collectively arrive at an optimal solution. There are several advantages to employ RL for optimal decision making. First, RL is model-free; the agent does not require a predefined rule or prior knowledge about how to select an action. Instead, it discovers optimal actions by "learning" directly whilst interacting with the environment. Second, RL is adaptive; the agent can acquire the optimal decisions autonomously, in an online fashion adapted to different appliances, taking into account the uncertainty and flexibility of the EMS. Third, RL is concise; the entire computation is based on a look-up table, which is much easier to apply in the real world than conventional optimization methods.

To the best of our knowledge, this is the first paper to deal with the DR problem in HEMSs using RL and ANN. The main contributions of this paper are as follows:

(1) An hour-ahead DR algorithm for HEMSs by using AI techniques is proposed, considering the costs of both user electricity and dissatisfaction.

(2) A multi-agent RL methodology is adopted to ensure optimal and decentralized decision-making for multiple home appliances. And, this approach is benchmarked with a centralized MILP solver; the evaluation results show that the proposed method learns a cost effective schedules for multiple appliances, and even achieves a superior performance than the MILP solver.

(3) A stable price forecasting model based on ANN is presented to overcome the uncertainty in future prices. By carefully selecting adequate input training data though a number of accuracy tests, this ANN model is capable of making reasonable and accurate predictions compared to the existing studies.

(4) The electricity costs under two different cases without and with AI based DR are compared, indicating the proposed DR algorithm can help the user to reduce its electricity cost, significantly.

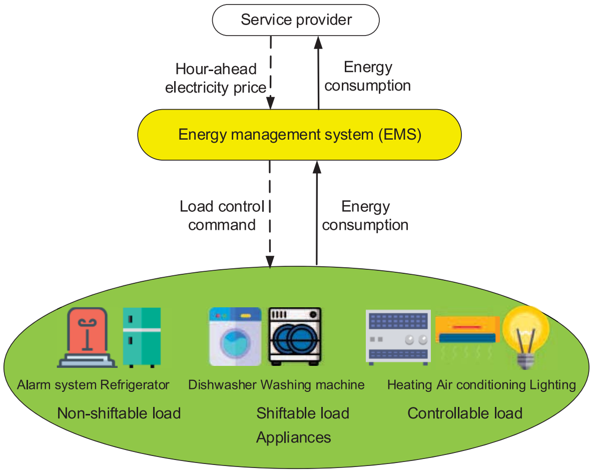

Service provider
Hour-ahead electricity price
Energy consumption
Energy management system (EMS)
Load control command
Energy consumption
Alarm system Refrigerator
Dishwasher Washing machine
Heating Air conditioning Lighting
Non-shiftable load
Shiftable load
Controllable load
Appliances

Fig. 2. Home energy management system.

The rest of this paper is organized as follows. Section II describes the problem formulations of the HEMS. In Section III, the ANN and multi-agent RL methodology are presented in detail. Section IV provides the numerical simulation results and evaluates the system performance. Finally, conclusions and future work are discussed in Section V.

II. Problem Formulation

We consider a HEMS wherein the household is equipped with one EMS and different kinds of appliances, with the requirement to optimize their energy consumption, as shown in Fig. 2. The EMS is connected to the service provider though a two-way communication network that enables exchange of price and appliance energy consumption information. The EMS receives the hour-ahead price information from the service provider, and then manages the energy consumption of each appliance in response to the prices. Usually, the domestic electric appliances are separated into three main types based on their characteristics and priorities, including non-shiftable, shiftable and controllable loads [6], [24]. The mathematical formulations of the HEMS, including the various operation constraints for the three categories of appliances and the objective function, are described in detail in the following subsections.

A. Non-Shiftable Load

Non-shiftable loads have critical demand requests that must be met during the energy distribution process, e.g., the refrigerators (REFGs) or certain alarm systems for security. Once a non-shiftable load begins operation, it must work continuously, and cannot be scheduled. The energy consumption of such loads is always equal to the energy demand.

$$E_{n,h}^{non} = e_{n,h}^{non} \tag{1}$$

where $n \in \{1, 2, 3, ...N\}$ represents the appliance n, N is the total number of appliances, $h \in \{1, 2, 3, ...H\}$ denotes the hour h, and H is the final hour of a day, i.e., $H = 24$ if the price is updated every 1 hour. $e_{n,h}$ and $E_{n,h}$ indicate the electricity demand and the actual energy consumption of appliance n at hour h, respectively.

The cost of this kind appliance is just the electricity bill of energy consumption. So, the utility function of a non-shiftable appliance n is:

$$U_{n,h}^{non} = P_h \cdot E_{n,h}^{non} \tag{2}$$

where P_h indicates the electricity price at hour h.

B. Shiftable Load

Shiftable loads can schedule their energy demand to off-peak hours when the prices are low in the schedule horizon, so that not only peak energy usage is avoided, but also the energy bill is reduced. Shiftable loads have two available operating points, “on” and “off”.

$$E_{n,h}^{shift} = I_{n,h} \cdot e_{n,h}^{shift} \tag{3}$$

where $I_{n,h}$ is a binary variable for appliance n, i.e., $I_{n,h} = 1$ if the appliance works at hour h; otherwise $I_{n,h} = 0$.

For this kind of appliance, there are two types of cost: the electricity bill of consuming energy, and the dissatisfaction of waiting time for a device to begin and then complete its operation. For example, the washing machines (WMs) usually operate during the working period $[T_{n,ini}, T_{n,end}]$ (i.e., 6 pm–11 pm), but the time of operation can be shifted from high electricity price periods to low price periods, if the WM starts to work at $T_{n,w}$ (i.e., 9 pm), in this case, the waiting time would be $T_{n,w} - T_{n,ini}$ (i.e., 3 h).

Thus, the utility function of a shiftable appliance n is [25]:

$$U_{n,h}^{shift} = P_h \cdot E_{n,h}^{shift} + k_n \cdot \left(T_{n,w} - T_{n,ini}\right) \tag{4}$$
$$T_{n,ini} \leq T_{n,w} \leq \left[T_{n,end} - T_{n,ne}\right] \tag{5}$$
$$T_{n,ne} \leq T_{n,end} - T_{n,ini} \tag{6}$$

where the first term represents the electricity cost and the second term denotes the cost of waiting time in Eq. (4). k_n is a coefficient that depends on the system, $T_{n,ini}$ and $T_{n,end}$ refer to the initial time and end time of the working period, $T_{n,w}$ denotes the operation starting time, and $T_{n,ne}$ indicates the time required for the shiftable load to complete its operation.

C. Controllable Load

Different from shiftable loads, the controllable loads can be operated with a flexible power consumption between the minimum power demand and maximum power demand denoted by $e_{n,\min}$ and $e_{n,\max}$, respectively, i.e., the lights (Ls) and air conditioners (ACs) can regulate their power consumption from $e_{n,\min}$ to $e_{n,\max}$ in response to price changes.

$$E_{n,h}^{con} = e_{n,h}^{con} \tag{7}$$
$$e_{n,\min} \leq e_{n,h}^{con} \leq e_{n,\max} \tag{8}$$

The objective of this kind of appliance should be to minimize the customer electricity bill by decreasing the power demand during the scheduling slots, however, the reduced power can cause dissatisfaction for the user.

Hence, the utility function of a controllable appliance n is:

$$U_{n,h}^{con} = P_h \cdot E_{n,h}^{con} + \beta_n \cdot \left(E_{n,h}^{con} - e_{n,\max}\right)^2 \tag{9}$$

where the first term denotes the electricity cost, and the second term indicates the customer dissatisfaction cost, which is defined by a quadratic function adopted from [26]. β_n is a device-dependent dissatisfaction cost parameter: a device with a greater β_n prefers consume more energy to improve the satisfaction level, and vice versa.

D. Objective Function

The objective function of the user is to not only minimize the electricity cost, but also minimize the dissatisfaction cost that can be expressed as follows:

$$\min \sum_{n=1}^{N} \sum_{h=1}^{H} \left\{ \begin{array}{l} (1-\rho) \cdot P_h \cdot \left(E_{n,h}^{non} + E_{n,h}^{shift} + E_{n,h}^{con}\right) \\ +\rho \cdot \begin{bmatrix} k_n \cdot \left(T_{n,w} - T_{n,ini}\right) \\ + \beta_n \cdot \left(E_{n,h}^{con} - e_{n,\max}\right)^2 \end{bmatrix} \end{array} \right\} \tag{10}$$

where the first term denotes the electricity cost, and the second term indicates the dissatisfaction cost. ρ is a balance parameter for achieving the trade-off between the electricity and dissatisfaction costs [25], which is determined by the user preference.

III. Artificial Neural Network and Multi-Agent Reinforcement Learning Methodology

In this work, we propose an hour-ahead energy management scheme for different types of appliances within a HEMS using multi-agent RL and ANN approach as shown in Fig. 3. To achieve the hour-ahead energy management scheme, ANN is used to predict the future electricity prices. After that, multi-agent RL is adopted to make the optimal decisions for different types of residential appliances.

Next, the details of the ANN and multi-agent RL methodology are introduced.

A. Price Forecasting With ANN

To deal with the uncertainty in future prices, ANN is used to predict these prices when the EMS receives hour-ahead price from the service provider. ANN is a computing system comprising highly interconnected processing units designed to replicate the performance of a human brain engaged in a particular task. The processing units of ANN are organized in sequential layers, containing one input layer, at least one hidden layer, and one output layer, as shown on the left side of Fig. 3. Each of these units, forms a weighted (W_i) sum of its inputs and a constant term called bias (b_i), and the sum is passed from one layer to the next layer through a transfer function. ANN is widely used in electrical fields such as load forecasting [27] and electricity price forecasting [22] due to its ability to handle non-linear relationship problems more accurately.

Adequate selection of inputs for ANN is highly important to the success of forecasting. The input data must contain maximally correlated historical data that are appropriately styled and formatted. In this paper, the inputs of ANN are chosen based on correlation analysis, some empirical guidelines described in [28]–[30], and a number of accuracy tests and comparisons in the simulation. In the price forecasting model of this study, detailed inputs are listed in Table I, and the output is the forecasted prices. In general, price parameters may depend on several factors. In particular, they depend on both near- and long-term historical electricity prices and power demands. The relevant near-term demands and prices information include data from 1, 2, and 3 hours earlier. The relevant long-term demands and prices information include those for the current hour, as well as 1 and 2 hours earlier, from 1 day and 2 days ago. Additionally, it is usually expected that the prices will be higher during the afternoon, and prices can be expected to vary depending on whether it is a work day or the weekend, which are listed in the first three inputs of Table I. These pieces of information can potentially help in predicting the prices in an hour-ahead environment. In this study, Sigmoid function [31] is adopted as the transfer function of each layer, and Levenberg-Marquardt algorithm [28] is used as the training algorithm for the model. The performance of this model is evaluated by mean absolute error (MAE) and mean absolute percentage error (MAPE) between the forecasted and actual values, as shown in Eqs. (11) and (12).

TABLE I
Inputs to the Artificial Neural Network

Input index	Description
1	Day of week (1-7)
2	Hour stage of day (1-24)
3	Is holiday (0 or 1)
4	Electricity demand of hour $h-1$
5	Electricity demand of hour $h-2$
6	Electricity demand of hour $h-3$
7	Electricity demand of hour $h-24$
8	Electricity demand of hour $h-25$
9	Electricity demand of hour $h-26$
10	Hour-ahead price of hour $h-1$
11	Hour-ahead price of hour $h-2$
12	Hour-ahead price of hour $h-3$
13	Hour-ahead price of hour $h-24$
14	Hour-ahead price of hour $h-25$
15	Hour-ahead price of hour $h-26$
16	Hour-ahead price of hour $h-48$
17	Hour-ahead price of hour $h-49$
18	Hour-ahead price of hour $h-50$

$$MAE = \frac{1}{T} \sum_{t=1}^{T} \left| RTP_t - RTP_t^f \right| \tag{11}$$

$$MAPE = \frac{100}{T} \sum_{t=1}^{T} \frac{\left| RTP_t - RTP_t^f \right|}{RTP_t} \tag{12}$$

where RTP_t represents the actual price, and RTP_t^f denotes the forecasted price.

While we are interested in accurate price predictions, our main focus in this study is to develop an efficient and scalable decision-making algorithm for HEMS using multi-agent RL with price predictors that have low computational complexity, and can be implemented easily in residential smart meters along with the energy scheduling unit. In recent years, price

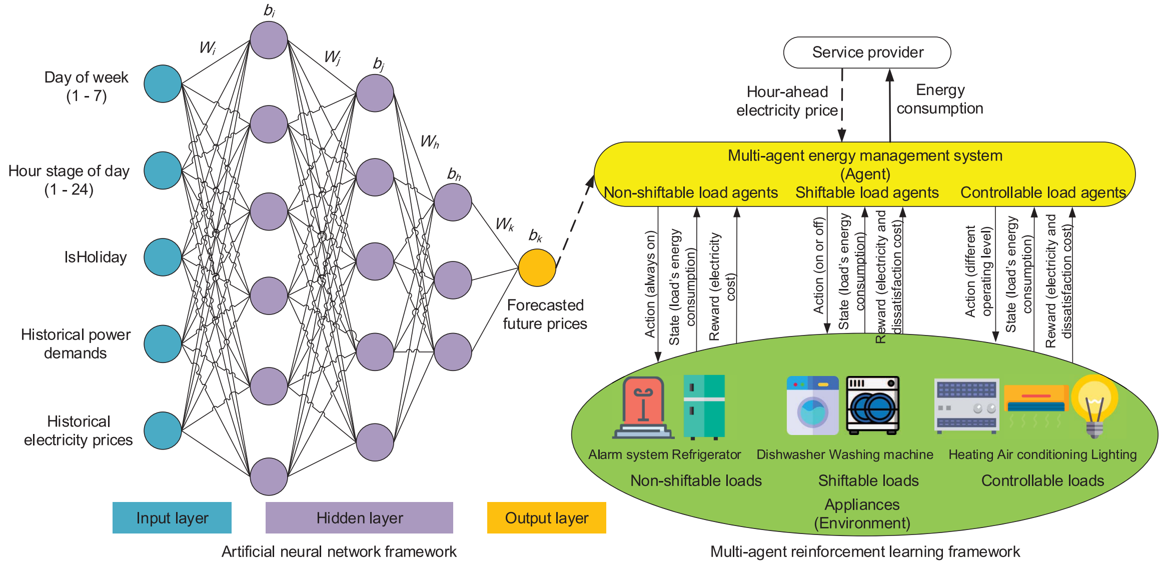

b_i
W_i
W_j
b_j
W_h
b_h
W_k
b_k
Day of week (1 - 7)
Hour stage of day (1 - 24)
IsHoliday
Historical power demands
Historical electricity prices
Forecasted future prices
Input layer
Hidden layer
Output layer
Artificial neural network framework
Service provider
Hour-ahead electricity price
Energy consumption
Multi-agent energy management system (Agent)
Non-shiftable load agents
Shiftable load agents
Controllable load agents
Action (always on)
State (load's energy consumption)
Reward (electricity cost)
Action (on or off)
State (load's energy consumption)
Reward (electricity and dissatisfaction cost)
Action (different operating level)
State (load's energy consumption)
Reward (electricity and dissatisfaction cost)
Alarm system
Refrigerator
Dishwasher
Washing machine
Heating
Air conditioning
Lighting
Non-shiftable loads
Shiftable loads
Controllable loads
Appliances (Environment)
Multi-agent reinforcement learning framework

Fig. 3. Artificial neural network and multi-agent reinforcement learning methodology.

forecasting has been a popular topic, and has already been examined by many existing studies. More information about price forecasting by different techniques, the readers can refer to [29], [32]. Once the future prices are acquired, they will be regarded as the inputs to the multi-agent RL methodology described in the following subsection.

B. Decision Making With Multi-Agent RL

RL is a type of machine learning algorithm dealing with how artificial agents should choose the ideal behaviors in a stochastic environment, so as to maximize the cumulative rewards. In practice, the agent controls a dynamic system through executing a sequence of actions. The dynamic system, also termed the environment, is characterized by certain states and a reward function that describes the evolution of states given the actions chosen by the agent. After executing an action, the agent obtains a numerical reward, and then moves to the next state; the objective of the agent is to gather as many rewards as possible. To fulfil this objective, the agent has to learn a policy dictating how actions are chosen, to maximize the expected total rewards in the long term.

In this work, each residential appliance represents the environment, and each has its own agent with different actions and rewards, as shown on the right side of Fig. 3.

The non-shiftable load only has one *action*, "on", because it cannot be scheduled.

The shiftable load has two available *actions*, "on" and "off".

The controllable load has a set of *actions*, i.e., 1, 2, 3,…, to represent various power ratings at different levels within the power demand range defined in Section II.

The *states* are denoted by the appliances' energy consumption information.

The inverse of the utility function (electricity and dissatisfaction costs defined in Section II) represents the *reward*.

To perform the optimal action-making, the RL control problem is considered as a discrete finite horizon MDP, which exhibits the Markov property that the state transitions are dependent only on the current state and the current action taken and, thus are independent of all previous environmental states and agent actions. The key elements in this RL include: a discrete hour h, an action a_h, a state s_h, and a reward $r(s_h, a_h)$. We use υ to denote the policy mapping states to actions, i.e., $\upsilon : a_h = \upsilon(s_h)$. The objective of this RL problem is to discover an optimal policy υ for each state s_h so that the action a_h selected maximizes the reward $r(s_h, a_h)$.

Q-learning [33], which is a technique in RL, is adopted to obtain the optimal policy υ (a sequence of operating actions for each appliance in this work). The basic mechanism of Q-learning is the assigning of a Q-value $Q(s_h, a_h)$ to each state-action pair at hour h, and updating of this value at each iteration, in a manner that optimizes the result. The optimal Q-value $Q_\upsilon^*(s_h, a_h)$, denotes the maximum discounted future reward $r(s_h, a_h)$ when taking action a_h at state s_h, and while continuing to follow the optimal policy υ, which satisfies the Bellman equation below:

$$Q_\upsilon^*(s_h, a_h) = r(s_h, a_h) + \gamma \cdot \max Q(s_{h+1}, a_{h+1}) \tag{13}$$

In which $\gamma \in [0, 1]$ is a discounting factor indicating the relative importance of future versus current rewards. Specifically, when $\gamma = 0$, the agent is shortsighted and only consider the current reward, while a factor of 1 will make the agent strive for the future rewards. If the agent wishes to balance the current and future rewards, γ is set to a fraction between 0 and 1.

The Q-value $Q(s_h, a_h)$ is stored in a state-action table, with each cell corresponding to the performance of carrying out a specific action in a specific state. Each hour, an agent performs an action, and the Q-value of the corresponding cell is updated based on the Bellman equation, Eq. (13), as follows:

$$Q(s_h, a_h) \leftarrow Q(s_h, a_h) + \theta \begin{bmatrix} r(s_h, a_h) \\ + \gamma \cdot \max Q(s_{h+1}, a_{h+1}) \\ - Q(s_h, a_h) \end{bmatrix} \tag{14}$$

TABLE II
COMBINING ARTIFICIAL NEURAL NETWORK AND MULTI-AGENT REINFORCEMENT LEARNING

```
Algorithm: hour-ahead DR with ANN and multi-agent RL
Initialize each appliance's power rating, working time,
        dissatisfaction cost parameter, and the balance parameter
1: For each hour $h$ do
2:   %%ANN for price forecasting
3:   DofWeek $\leftarrow$ updateDayofWeek()
4:   HofDay $\leftarrow$ updateHourStageofDay()
5:   IsHol $\leftarrow$ updateHoliday()
6:   DeData $\leftarrow$ updateHistoricalDemandData()
7:   PrData $\leftarrow$ updateHistoricalPriceData()
8:   FuturePrices $\leftarrow$ ANN(DofWeek, HofDay, IsHol, DeData, PrData)
9:   %%Multi-agent RL for decisions making
10:  For each agent do in parallel
11:    Initialize Q-value arbitrarily
12:    Repeat (for each iteration $i$)
13:    Initialize $s_{n,h}$
14:      Repeat (for each step in $i$)
15:        Choose $a_{n,h}$ from current $s_{n,h}$ using $\varepsilon$-greedy policy
16:        Take action $a_{n,h}$, observe $r\left(s_{n,h}, a_{n,h}\right)$ and next $s_{n,h+1}$
17:        $Q\left(s_{n,h}, a_{n,h}\right) \leftarrow Q\left(s_{n,h}, a_{n,h}\right) + \theta\begin{bmatrix} r\left(s_{n,h}, a_{n,h}\right) \\ +\gamma \cdot \max Q\left(s_{n,h+1}, a_{n,h+1}\right) \\ -Q\left(s_{n,h}, a_{n,h}\right)\end{bmatrix}$
18:      Until $s_{n,h+1}$ is terminal
19:    Until Q-value is converged, such that $\left|Q^{i} - Q^{i-1}\right| \leq \phi$
20:    Outputs the optimal actions
       (Only the actions for the current hour will be executed)
21:  End for
22:End for
```

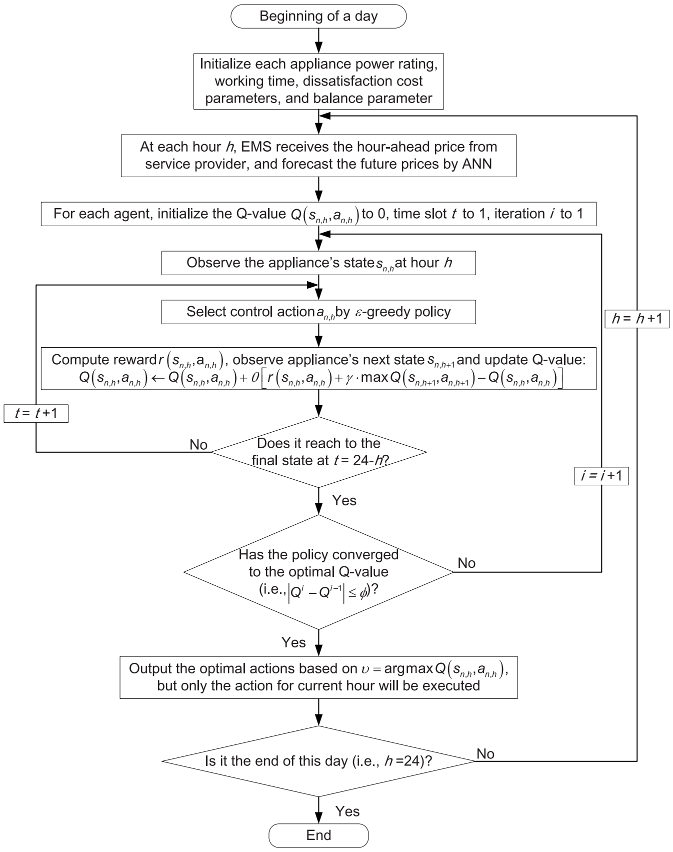

Beginning of a day
Initialize each appliance power rating, working time, dissatisfaction cost parameters, and balance parameter
At each hour h, EMS receives the hour-ahead price from service provider, and forecast the future prices by ANN
For each agent, initialize the Q-value $Q(s_{n,h}, a_{n,h})$ to 0, time slot t to 1, iteration i to 1
Observe the appliance's states $s_{n,h}$ at hour h
Select control action $a_{n,h}$ by ε-greedy policy
Compute reward $r(s_{n,h}, a_{n,h})$, observe appliance's next state $s_{n,h+1}$ and update Q-value:
$Q(s_{n,h}, a_{n,h}) \leftarrow Q(s_{n,h}, a_{n,h}) + \theta[r(s_{n,h}, a_{n,h}) + \gamma \cdot \max Q(s_{n,h+1}, a_{n,h+1}) - Q(s_{n,h}, a_{n,h})]$
$t = t+1$
No
Does it reach to the final state at $t = 24$-h?
Yes
$h = h+1$
$i = i+1$
Has the policy converged to the optimal Q-value (i.e., $|Q^i - Q^{i-1}| \leq \phi$)?
No
Yes
Output the optimal actions based on $\upsilon = \arg\max Q(s_{n,h}, a_{n,h})$, but only the action for current hour will be executed
Is it the end of this day (i.e., h =24)?
No
Yes
End

Fig. 4. Flowchart for implementing the algorithm in Table II.

In which $\theta \in [0, 1]$ is a learning rate representing to what degree the new overrides the old Q-values. A value of 0 indicates that the agent learns nothing, exploiting prior knowledge exclusively, whereas a value of 1 denotes that the agent takes into account only the current estimate, ignoring prior knowledge to explore possibilities. For trading off the newly acquired information and old information, θ should be set to a decimal between 0 and 1.

In the Q-learning mechanism, the artificial agent directly interacts with the dynamic environment by executing actions. Afterwards, the agent obtains a reward and moves to a new environmental state. Learning occurs through trials and errors during this course. And in this learning process, the Q-value of every state-action pair is stored and updated. After updating an adequate number of iterations, the Q-value eventually converges to the maximum value. Detailed proof of convergence is provided in [34], [35].

When RL is applied in a multi-agent context in this study, each agent of the residential appliance acts independently to identify its optimal policy. During the learning process, each agent maintains its own Q-values, and reaches a policy based solely on the effects occurring in the environment caused by its own actions. Each policy is implemented as a separate Q-learning process with its own state space. When each agent reaches the optimal Q-value, all agents have obtained the maximum reward, meaning that the sum of the rewards is also at a maximum, and the system has reached the global optimal Q-value [36].

Since the Q-value represents the maximum reward with action a_h at state s_h for each agent, we can obtain the optimal policy

$$\upsilon = \arg\max Q(s_h, a_h) \tag{15}$$

wherein the optimal actions for each appliance are acquired.

C. Combining ANN and Multi-Agent RL

Table II shows the detailed DR algorithm that combines the ANN and multi-agent RL: every hour, the EMS receives the hour-ahead price, and uses the ANN to predict the future prices. Then, multi-agent RL is adopted to obtain the optimal decisions for different kinds of residential appliances.

To help the reader better understand the algorithm presented in Table II, a flowchart in Fig. 4 is plotted to show how the methodologies are implemented to obtain the optimal actions. Specifically, the algorithm starts running at the beginning of a day, i.e., $h = 1$. The EMS first initializes each appliance's power rating, working time, dissatisfaction cost parameter and balance parameter. Upon setting these parameters, at each hour h, the EMS will update the inputs of the price forecasting model (i.e., hour stage of the day, historical power demands and electricity prices), then use the pre-trained ANN model to predict the future prices for the following hours (lines 3-8 of Table II). Afterwards, in cooperation with the forecasted future prices, the EMS will compute the optimal decisions for each appliance in an iterative way (lines 10-21 of Table II), i.e., at each iteration i, each agent observes the appliance's energy information $s_{n,h}$, and then selects an operating action

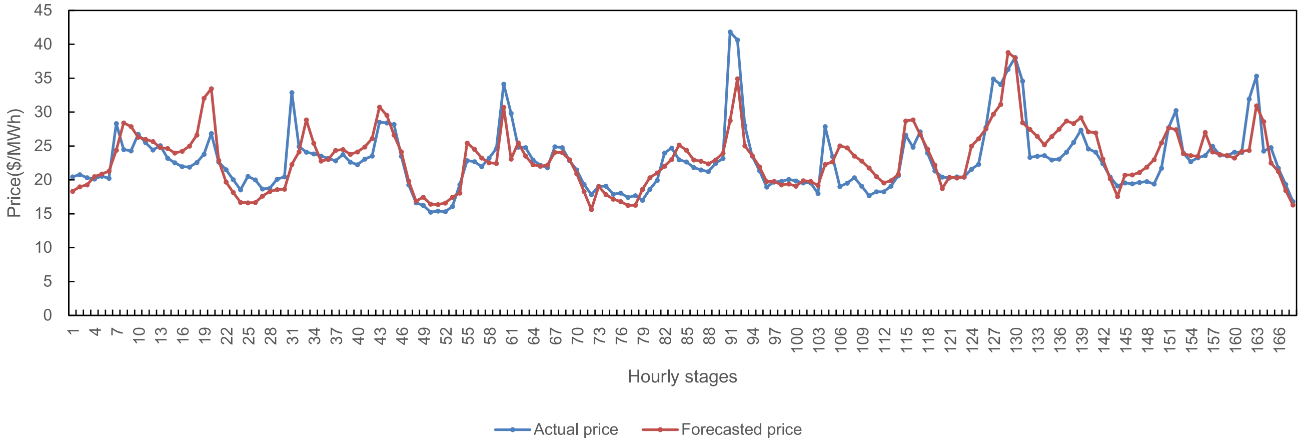


Fig. 5. Comparison of the actual and forecasted prices from February 22-28, 2017.

$a_{n,h}$ based on the exploration and exploitation mechanism. Exploration means the evaluation of the values on available actions, and exploitation refers to the utilization of current knowledge on action values to maximize the return. The most common way to realize exploration and exploitation is via a ε-greedy policy ($\varepsilon \in [0, 1]$) [11], where the agent can either select a random action with probability ε, or take an action with probability 1-ε by reference to the Q-value table. Here, random selection indicates that the agent chooses an operating action randomly from the set of available actions at that state, and selection from the Q-value table represents that the agent selects the operating action whose current Q-value is "maximum". However, the current "maximum" Q-value could be overridden in future iterations. This allows the agent to explore the action spaces with some degree of randomness, whereas forbids it from being totally random. After choosing the operating action, each agent gains an immediate reward $r(s_{n,h}, a_{n,h})$, observes the appliance's next state $s_{n,h+1}$ and updates the Q-value $Q(s_{n,h}, a_{n,h})$ using Eq. (14); this process is repeated until state $s_{n,h+1}$ is terminal. After that, each agent compares the current and last Q-value to confirm whether convergence to the maximum Q-value has occurred; if not, the agent moves to the next iteration $i+1$ and repeats this process.

The iteration termination criterion is calculated as $|Q^i - Q^{i-1}| \leq \phi$; if the difference between the Q^i and Q^{i-1} is no more than ϕ, the Q^i has converged to the maximum value, wherein the ϕ is a system-dependent parameter [37]. At last, each agent will access the optimal operating action for the coming $H - h$ hours, but only the optimal action for the current hour h will be executed.

Then, the system moves to the next hour $h + 1$, and repeats the above procedure until it attains the end hour H.

IV. Performance Evaluation

This section presents the numerical simulation results used to evaluate the performance of the price forecasting model with ANN and DR algorithm with RL.

A. Performance of the Price Forecasting Model

The price and energy data used to train and test this model were obtained from the PJM [38] electricity market. Specifically, the data from January 1, 2016 to February 21, 2017 were used to train the model, and then the model was used to predict the prices for February 22-28, 2017. The neural network toolbox of MATLAB was selected to train and learn the model due to its flexibility and simplicity. After several accuracy tests, the ANN model finally included five layers, containing one input layer with 18 neurons, three hidden layers with 40, 20 and 10 neurons, and one output layer with 1 neuron.

Fig. 5 shows a comparison of forecasted prices and actual prices for the last 7 days of February 2017, where the blue line represents the actual prices and the red line denotes the forecasted prices. From the figure, we can see that the trend in forecasted prices is quite similar to the actual prices. In this case, the MAE is 2.12, and the MAPE is 8.59. Compared with the price forecasting results in [22], [28], the MAE and MAPE in this work are lower, indicating the ANN model in this paper can make accurate and reasonable price forecasting.

B. Performance of the Demand Response Algorithm

A smart home is considered for a simulation scenario featuring non-shiftable, shiftable and controllable loads. To have an ease of exemplification, the simulations were conducted on seven appliances including one non-shiftable load (REFG), five controllable loads (three ACs: AC1, AC2, AC3; and two Ls: L1, L2), and one shiftable load (WM). However, without loss of generality, the proposed DR scheme can easily be extended with more appliances due to the scalable architecture based on multi-agent RL methodology used in this study.

Naturally, the user could have different power requirements with regard to ACs and Ls depending on where they are in the house, e.g., bedroom or sitting room. In this situation, the dissatisfaction cost parameter β is appliance-specific. The parameters of each appliance are listed in Table III, derived from [8], [24], [25], [39].

All parameter values in this study are particular, and they can vary according to the characteristics of the appliances or users. However, this does not affect the analysis and interpretation of the simulation results.

To demonstrate the performance of the proposed DR scheme, the detailed simulations for February 24, 2017 are

TABLE III
PARAMETERS OF EACH APPLIANCE

Device type	Non-shiftable	Shiftable	Controllable				
ID	REFG	WM	AC1	AC2	AC3	L1	L2
k_n	-	0.1	-	-	-	-	-
β_n	-	-	2.0	2.5	3.0	2.2	2.8
Power rating (kWh)	0.2	0.7	0 - 1.4	0 - 1.4	0 - 1.4	0.2 - 0.8	0.2 - 0.8
Working time	24 h	6 pm - 11 pm	24 h	24 h	24 h	6 am - 11 pm	6 am - 11 pm
$T_{n,ne}$	-	2 h	-	-	-	-	-

REFG: refrigerator; WM: washing machine; AC: air conditioner; L: light.

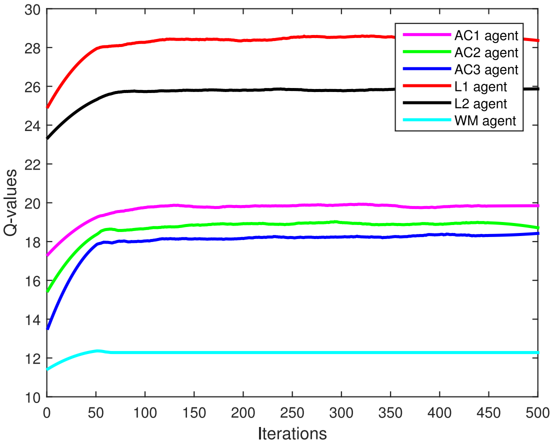
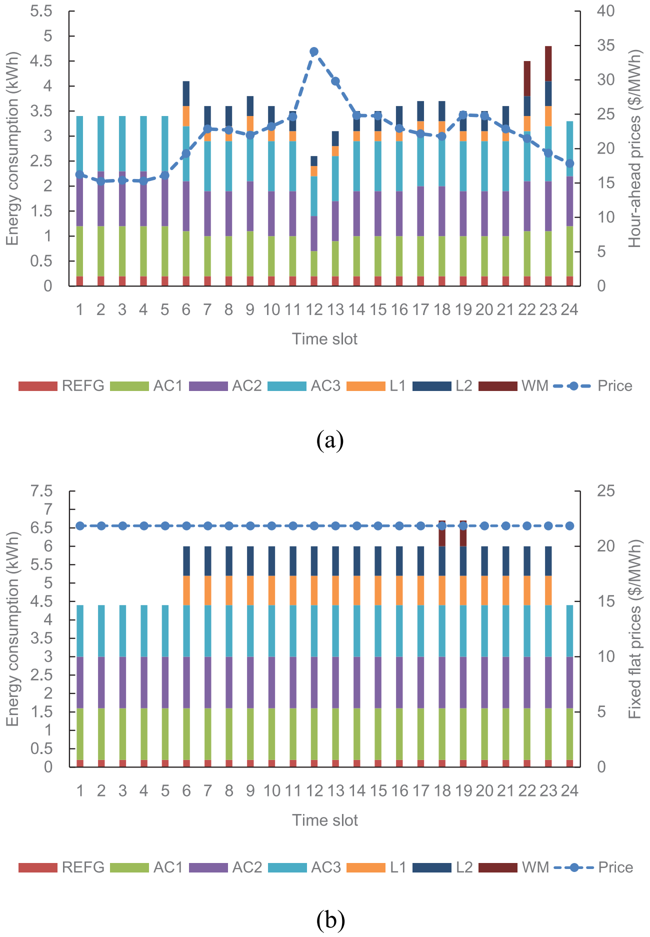


Fig. 6. Convergence of the Q-value for each agent on February 24, 2017.

discussed. In order to keep the agent visiting all the state-action pairs and learning new knowledge from the system, the tuning parameter ε of ε-greedy policy is set to 0.2. To update $Q(s_{n,h}, a_{n,h})$ from the experimental experience, we set the discounting factor γ to 0.95, and the learning rate θ to 0.1. After executing the simulation, each agent can converge to the maximum Q-value, as shown in Fig. 6. It can be seen that, at the outset, the agents chose poor actions yielding lower Q-values, however, with each successive iteration, the Q-value increased as the agents discovered the actions yielding higher Q-values by learning them through trials and errors, and finally achieving the maximum Q-values.

Once the maximum Q-value is obtained, the optimal energy consumption of each load can be determined. Fig. 7 shows the aggregated energy consumption of all loads in two different cases with and without AI based DR, along with the electricity prices on February 24, 2017. As shown in Fig. 7a, when the proposed DR algorithm in this work was deployed, the loads consume more energy when the prices are low, and then reduce their demand when prices are high such that energy consumption at peak times is avoided. Specifically, all the energy demands of controllable and shiftable loads are scheduled to off-peak slots, i.e., all the controllable loads consume less energy during time slots 10-15, and consume more energy during time slots 1-6 and 22-24; the shiftable load (WM) operates in time slots 22 and 23, wherein the prices are the lowest in its working period. Thus, the overall energy consumption is maintained at a low level during peak slots, confirming that the established DR algorithm can handle energy management well. In comparison, for the case without DR, it was assumed that the electricity price is fixed flat price in each time interval, and equals to the average of the dynamic price. Clearly, the user had no incentive to reduce or shift its energy consumption when the fixed flat prices were applied, as shown in Fig. 7b.



(a)

(b)

Fig. 7. Aggregated energy consumption of all loads on February 24, 2017. (a) With DR. (b) Without DR.

To gain insights into the effectiveness of the proposed DR algorithm, Fig. 8 shows the energy consumption of five controllable loads during each time slot. When looking into Fig. 8, it can be seen that as the price starts to increase in time slot 5, the energy consumption of the three ACs decreases. As

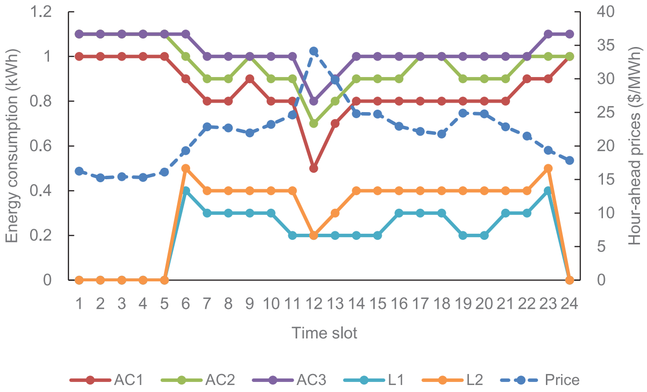


Fig. 8. Energy consumption of five controllable loads during each time slot.

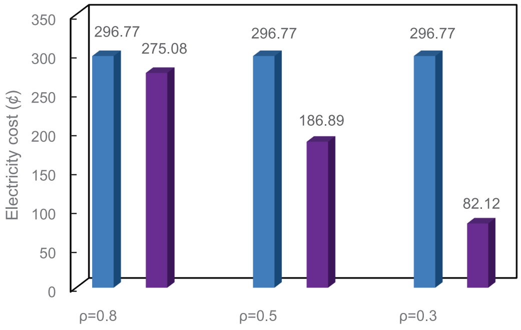


Fig. 9. Electricity cost comparison without and with DR under different ρ.

the price continues to increase, the other two Ls also decrease their energy consumption from time slot 10. Finally, the energy consumption of the five controllable loads is reduced to their minimum as the price reaches its maximum value in time slot 12. After time slot 12, the energy consumption begins to increase as the price goes down.

It can also be observed that AC3 consumes more energy than AC2, and AC2 consumes more energy than AC1 for the whole time horizon. This is because AC3 has a larger dissatisfaction cost parameter β, promoting a greater energy consumption to suffer less incurred dissatisfaction when the proposed DR algorithm is in effect. In contrast, an AC with a smaller dissatisfaction cost parameter β will prefer to use less energy. The same phenomenon is evident in L1 and L2.

Fig. 9 shows the electricity cost comparison of two cases under different balance parameter ρ on February 24, 2017, where blue bar represents the case without DR, and purple bar represents the case with DR. The daily electricity costs of the case when proposed DR in this work was deployed were reduced significantly by 7.3%, 37.0% and 72.3%, compared with the case when no DR was applied, which serves core motivation for the users to participate in the proposed DR program. From the figure, we can also see that a bigger balance parameter ρ (e.g., ρ=0.8) causes a greater electricity cost in this proposed DR scheme. It is because with a larger ρ, the dissatisfaction cost becomes relatively more crucial compared with the electricity cost in the objective function (10), which results in a higher energy consumption, leading to a greater electricity cost.

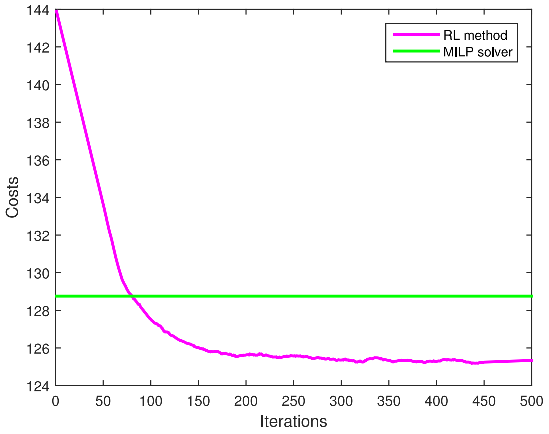


Fig. 10. Total costs of RL method compared to MILP solver.

To evaluate the superiority of the proposed algorithm with decentralized multi-agent RL methodology, a benchmark without learning is considered, in which the optimization problem was solved by a centralized commercial MILP Gurobi solver [40]. This benchmark uses the exact model and knows all information of the system, to minimize the electricity and dissatisfaction costs defined in Eq. (10) with shortsighted actions. On the contrary, the multi-agent RL strategy allowing utilizing learning capacity to choose different actions to maximize the *reward* defined in Section III. Fig. 10 shows the total costs (sum of the electricity and dissatisfaction costs) of the seven appliances under these two methods. It can be observed that the RL approach actually does not work well at the beginning since it is still undergoing trial and error as part of its learning process. However, with more experience obtained through running more iterations and for a longer time, the RL starts to autonomously adapt to the market characteristics and adjust its policy by exploration and exploitation mechanism explained in Section III. In the long run, it will outperform the MILP solver that has no learning capability as depicted in Fig. 10. This is because when selecting actions during the learning process, the RL agent considers not only current reward but also future rewards in Eq. (13).

Finally, Table IV gives the computational statistics for the simulation scenario on February 24, 2017. The computation time for training the price forecasting model with ANN, and obtaining the optimal actions with multi-agent RL, is on average about 1 min and 20 s, respectively, which can fully satisfy the time requirement for deploying the hour-ahead DR scheme in a HEMS. And the time to solve the optimization problem by centralized MILP Gurobi solver is on average 50 s, which is longer than that taken by the decentralized multi-agent RL method. In summary, all the experiments show that the proposed approach with multi-agent RL and ANN is able to learn a cost effective schedule for different residential appliances under varying circumstances (i.e., dynamic hour-ahead prices), without using prior information about the system.

V. Conclusion and Future Work

In this paper, we proposed an hour-ahead DR algorithm for HEMS, taking an AI approach with the aim of minimizing

TABLE IV
COMPUTATIONAL STATISTICS FOR THE CASE STUDY

	Hardware	Software	Computation time
Price forecasting with ANN	Windows PC,	Matlab	1 min
Decision making with multi-agent RL	4-core i5-6600 CPU 3.30GHz,	Java programming, Eclipse IDE	20 s
Decision making with MILP solver	8GB RAM	C++ programming, Visual Studio IDE	50 s

the user energy bill and degree of discomfort. Specifically, to overcome future price uncertainties, a steady price forecasting model based on ANN is presented. In cooperation with the forecasted prices, multi-agent RL is adopted to make optimal decisions for different appliances. Simulation results showed that the proposed DR algorithm can handle the energy management and minimize the user energy bill and dissatisfaction costs. Furthermore, the electricity costs of two different cases without and with DR are compared, indicating that the DR algorithm in this work can help the user to reduce its energy cost, significantly.

In future work, the uncertainty of appliance energy usage patterns and working time will be taken into account as an extension of the current work. Also, the impacts of network congestion should be considered when implementing the presented DR algorithm in real physical applications.

REFERENCES

[1] M. Muratori and G. Rizzoni, "Residential demand response: Dynamic energy management and time-varying electricity pricing," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1108–1117, Mar. 2016.

[2] M. Yu, R. Lu, and S. H. Hong, "A real-time decision model for industrial load management in a smart grid," *Appl. Energy*, vol. 183, pp. 1488–1497, Dec. 2016.

[3] Y. M. Ding, S. H. Hong, and X. H. Li, "A demand response energy management scheme for industrial facilities in smart grid," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2257–2269, Nov. 2014.

[4] Y.-C. Li and S. H. Hong, "Real-time demand bidding for energy management in discrete manufacturing facilities," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 739–749, Jan. 2017.

[5] F. De Angelis *et al.*, "Optimal home energy management under dynamic electrical and thermal constraints," *IEEE Trans. Ind. Informat.*, vol. 9, no. 3, pp. 1518–1527, Aug. 2013.

[6] N. G. Paterakis, O. Erdinc, A. G. Bakirtzis, and J. P. Catalão, "Optimal household appliances scheduling under day-ahead pricing and load-shaping demand response strategies," *IEEE Trans. Ind. Informat.*, vol. 11, no. 6, pp. 1509–1519, Dec. 2015.

[7] S. Pal and R. Kumar, "Electric vehicle scheduling strategy in residential demand response programs with neighbor connection," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 980–988, Mar. 2018.

[8] M. Yu and S. H. Hong, "A real-time demand-response algorithm for smart grids: A Stackelberg game approach," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 879–888, Mar. 2016.

[9] L. Park, Y. Jang, S. Cho, and J. Kim, "Residential demand response for renewable energy resources in smart grid systems," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3165–3173, Dec. 2017.

[10] M. Shafie-Khah and P. Siano, "A stochastic home energy management system considering satisfaction cost and response fatigue," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 629–638, Feb. 2018.

[11] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[12] J. Gao and R. Jamidar, "Machine learning applications for data center optimization," Mountain View, CA, USA, Google, White Paper, 2014.

[13] H. Wang, T. Huang, X. Liao, H. Abu-Rub, and G. Chen, "Reinforcement learning in energy trading game among smart microgrids," *IEEE Trans. Ind. Electron.*, vol. 63, no. 8, pp. 5109–5119, Aug. 2016.

[14] H. Wang, T. Huang, X. Liao, H. Abu-Rub, and G. Chen, "Reinforcement learning for constrained energy trading games with incomplete information," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3404–3416, Oct. 2017.

[15] F. Ruelens *et al.*, "Reinforcement learning applied to an electric water heater: From theory to practice," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3792–3800, Jul. 2018.

[16] F. Ruelens *et al.*, "Residential demand response of thermostatically controlled loads using batch reinforcement learning," *IEEE Trans. Smart Grid*, vol. 8, no. 5, pp. 2149–2159, Sep. 2017.

[17] S. Vandael, B. Claessens, D. Ernst, T. Holvoet, and G. Deconinck, "Reinforcement learning of heuristic EV fleet charging in a day-ahead electricity market," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1795–1805, Jul. 2015.

[18] T. Liu, Y. Zou, D. Liu, and F. Sun, "Reinforcement learning of adaptive energy management with transition probability for a hybrid electric tracked vehicle," *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7837–7846, Dec. 2015.

[19] A. Chiş, J. Lundén, and V. Koivunen, "Reinforcement learning-based plug-in electric vehicle charging with forecasted price," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 3674–3684, May 2017.

[20] R. Lu, S. H. Hong, X. Zhang, X. Ye, and W. S. Song, "A perspective on reinforcement learning in price-based demand response for smart grid," in *Proc. Int. Conf. Comput. Sci. Comput. Intell. (CSCI)*, 2017, pp. 1822–1823.

[21] R. Lu, S. H. Hong, and X. Zhang, "A dynamic pricing demand response algorithm for smart grid: Reinforcement learning approach," *Appl. Energy*, vol. 220, pp. 220–230, Jun. 2018.

[22] X. Huang, S. H. Hong, and Y. Li, "Hour-ahead price based energy management scheme for industrial facilities," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 2886–2898, Dec. 2017.

[23] N. Chaâbane, "A hybrid ARFIMA and neural network model for electricity price prediction," *Int. J. Elect. Power Energy Syst.*, vol. 55, pp. 187–194, Feb. 2014.

[24] X. H. Li and S. H. Hong, "User-expected price-based demand response algorithm for a home-to-grid system," *Energy*, vol. 64, pp. 437–449, Jan. 2014.

[25] K. Ma, T. Yao, J. Yang, and X. Guan, "Residential power scheduling for demand response in smart grid," *Int. J. Elect. Power Energy Syst.*, vol. 78, pp. 320–325, Jun. 2016.

[26] M. Yu and S. H. Hong, "Incentive-based demand response considering hierarchical electricity market: A Stackelberg game approach," *Appl. Energy*, vol. 203, pp. 267–279, Oct. 2017.

[27] N. G. Paterakis, A. Taşcıkaraoğlu, O. Erdinç, A. G. Bakirtzis, and J. P. Catalao, "Assessment of demand-response-driven load pattern elasticity using a combined approach for smart households," *IEEE Trans. Ind. Informat.*, vol. 12, no. 4, pp. 1529–1539, Aug. 2016.

[28] I. P. Panapakidis and A. S. Dagoumas, "Day-ahead electricity price forecasting via the application of artificial neural network based models," *Appl. Energy*, vol. 172, pp. 132–151, Jun. 2016.

[29] J. Nowotarski and R. Weron, "Recent advances in electricity price forecasting: A review of probabilistic forecasting," *Renew. Sustain. Energy Rev.*, vol. 81, pp. 1548–1568, Jan. 2018.

[30] R. Lu and S. H. Hong, "Incentive-based demand response for smart grid with reinforcement learning and deep neural network," *Appl. Energy*, vol. 236, pp. 937–949, Feb. 2019.

[31] H. S. Sandhu, L. Fang, and L. Guan, "Forecasting day-ahead electricity prices using data mining and neural network techniques," in *Proc. 11th Int. Conf. Service Syst. Service Manag. (ICSSSM)*, 2014, pp. 1–6.

[32] R. Weron, "Electricity price forecasting: A review of the state-of-the-art with a look into the future," *Int. J. Forecast.*, vol. 30, no. 4, pp. 1030–1081, 2014.

[33] C. J. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.

[34] T. Jaakkola, M. I. Jordan, and S. P. Singh, "On the convergence of stochastic iterative dynamic programming algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 703–710.

[35] F. S. Melo, "Convergence of Q-learning: A simple proof," Inst. Syst. Robot., Zürich, Switzerland, Rep., pp. 1–4, 2001.

[36] L. Busoniu, R. Babuŝka, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 38, no. 2, pp. 156–172, Mar. 2008.

[37] L. A. Hurtado, E. Mocanu, P. H. Nguyen, M. Gibescu, and R. I. Kamphuis, "Enabling cooperative behavior for building demand response based on extended joint action learning," *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 127–136, Jan. 2018.

[38] PJM. (2018). *Data Miner 2*. Accessed: Feb. 2018. [Online]. Available: http://dataminer2.pjm.com/list

[39] P. Siano and D. Sarno, "Assessing the benefits of residential demand response in a real time distribution energy market," *Appl. Energy*, vol. 161, pp. 533–551, Jan. 2016.

[40] Gurobi. (2018). *Gurobi Optimization*. Accessed: Dec. 2018. [Online]. Available: http://www.gurobi.com/

Renzhi Lu (M'18) received the B.S. degree from the School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, China, in 2014. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Hanyang University, Ansan, South Korea.

He has published several papers of applying state-of-the-art in artificial intelligence for energy management. His research interests include artificial intelligence (deep reinforcement learning), smart grid (demand response), and smart manufacturing. He is a member of the IEEE Industrial Electronics Society, the IEEE Computational Intelligence Society, and the IEEE Power and Energy Society.

Seung Ho Hong (M'89–SM'10) received the B.S. degree in mechanical engineering from Yonsei University, Seoul, South Korea, in 1982, the M.S. degree in mechanical engineering from Texas Tech University, Lubbock, TX, USA, in 1985, and the Ph.D. degree in mechanical engineering from Pennsylvania State University, University Park, PA, USA, in 1989.

He was the Director of the Ubiquitous Sensor Networks Research Center, Hanyang University, Ansan, South Korea, a subsidiary of the Gyeonggi Regional Research Center Program. He was a Visiting Scholar with the National Institute of Standards and Technology, USA, the Vienna University of Technology, Austria, and Zhejiang University, China. He is currently a Professor with the Department of Electronic Engineering and the Director of the Connected Smart Systems Laboratory, Hanyang University. He is also a Visiting Professor with the Shenyang Institute of Automation, Chinese Academy of Sciences, the Chongqing University of Posts and Telecommunications, and the Wuhan University of Science and Technology, China. He is also a Foreign Expert with the Tianjin University of Technology through the Tianjin Thousand Talents Program. His research interests include the areas of smart manufacturing, smart grid, industrial IoT, cyber-physical systems, and artificial intelligence.

Mengmeng Yu (M'16) received the B.S. degree in communication engineering from the Wuhan University of Technology, Wuhan, China, in 2008, the M.S. degree in detection technique and automation equipment from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2011, and the Ph.D. degree in electronic systems engineering from Hanyang University, Ansan, South Korea, in 2015.

She was a Post-Doctoral Researcher under the BK21 PLUS Program (BK21+) with Hanyang University from 2015 to 2018, where she is currently a Research Professor. Her research interests include smart grid, game theory, machine learning, and smart manufacturing. She was a recipient of the Outstanding Researcher Award of BK21+ and the Best Paper Award from the Workshop on Smart City Infrastructure and Applications, in 2016.