Building Energy Management for Demand Response Using Kernel Lifelong Learning

Sunyong Kim[1], Rami Mowakeaa[2], Seung-Jun Kim[2], and Hyuk Lim[3]

[1]School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology (GIST), Gwangju, Republic of Korea, (e-mail: sunyongkim@gm.gist.ac.kr)

[2]Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County (UMBC), Baltimore, MD, USA (e-mail: ramo1@umbc.edu; sjkim@umbc.edu)

[3]AI Graduate School, Gwangju Institute of Science and Technology (GIST), Gwangju, Republic of Korea, (e-mail: hlim@gist.ac.kr)

Corresponding author: Hyuk Lim (hlim@gist.ac.kr)

ABSTRACT Demand response (DR) aims at improving the reliability and efficiency of the power grids by shaping the power demand over time. Given that building energy consumption constitutes a significant portion of the overall grid load, building energy management is a critical component for the DR portfolio. In this study, DR control policies for lighting and air-conditioner systems for the individual spaces in buildings are proposed. The policies are designed to achieve the energy reduction amount specified in the DR request while minimizing the user discomfort. A significant challenge is to cope with the uncertainty of various environmental factors such as the solar illuminance and ambient temperature, as well as the psycho-economic factors such as the energy usage preferences of the occupants. We employ a data-driven machine learning approach to tackle this challenge. Our novel idea is to take advantage of the structural similarity of the control policies across the spaces in a lifelong multi-task learning framework. To accommodate significant nonlinearity in efficient policies, a kernel-based learning approach is pursued. The dual decomposition method is employed to relax the constraint coupled across the spaces, which allows solving the overall learning problem via a series of unconstrained subproblems. The efficacy of the proposed method is verified by numerical experiments based on semi-real data sets.

INDEX TERMS Smart grid, demand response, building energy management, multi-task learning, lifelong learning, kernel-based learning, dual decomposition

I. INTRODUCTION

In recent years, the demand response (DR) has become a key component of smart grid systems due to its significant potential for enhancing grid economy and reliability [1]. DR encourages the energy users to adjust their energy consumption by offering financial rewards or imposing penalties in order to elicit a desirable balance between the supply and demand of electrical power. It is worthwhile to note that the energy consumption of buildings accounts for more than 20% of the global energy consumption, which is expected to continue to rise. Thus, developing effective energy management strategies for buildings is a major DR challenge [2],

[3]. A building can be viewed as a collection of numerous spaces, each of which contains controllable loads. Therefore, an energy management system (EMS) for buildings should be capable of controlling a large number of controllable loads effectively, while meeting the given DR requests from the utility, saving overall energy costs, and guaranteeing the comfort of the users occupying the individual rooms and spaces. Much research effort has been devoted to developing DR algorithms for controlling building energy consumption [4]–[16].

In this work, our goal is to develop a DR policy capable of achieving a given amount of reduction in building en-

ergy consumption, specified in the DR request signal, while minimizing the dissatisfaction levels of the occupants of each room. There are many types in commercial buildings such as offices, schools, hospitals, and so on, and the major energy consumption appliances can be different highly depending on the type of building. In this work, we consider the building, where the energy consumption of the lighting and air conditioning (AC) systems constitutes a dominant portion of the total building energy usage [17], [18]. A key challenge is that one must cope with the stochasticity of various environmental variables such as the ambient sunlight and temperature. Furthermore, it is desired that the policy also respects different energy usage preferences of the occupants. Acquiring the exact distributional information for such random processes can be challenging, especially when it is prone to change over time.

To tackle this challenge, a data-driven machine learning (ML) approach is taken, where the optimal policy is learned from the collected data. For efficient training and good generalization performance of the learned policies, one has to incorporate any prior information in the learning formulation. Our key idea is to recognize that the policies for different rooms share certain structural similarities, although there are variations from room to room due to the particularities of individual spaces. For example, the control policies for campus classrooms should have a lot in common, although there would be differences due to varying room sizes, directions/locations, and usage patterns.

Such shared structures can be captured in a multi-task learning (MTL) framework. In MTL, the classifiers or regressors are learned to solve multiple related tasks jointly by leveraging their similarities [19], [20]. Thus, MTL can attain a performance that exceeds what can be achieved by learning each of the tasks in isolation. In particular, MTL is useful when the numbers of data samples for the tasks are relatively small (or when their distributions are continuously changing so that the effective numbers for a given state are small), as the knowledge can be shared across tasks, and thus pooling all tasks' data facilitates the learning for the individual tasks.

As the number of tasks increases, the computational burden can become significant for the MTL training, with all tasks' data processed together. Furthermore, the classifiers need to be continuously updated when the data distribution is non-stationary. In some applications, new tasks can arrive, or additional data can be collected for existing tasks, after the classifiers have been trained. In this case, re-training the entire set of classifiers just to incorporate the new arrivals is not well justified. These issues can be addressed by the lifelong learning approach, where the MTL problems are solved in an online manner [21], [22]. Lifelong learning can transfer the knowledge acquired from novel tasks/data to all other tasks by continuously refining the basis of the shared representation of the classifiers. In our building DR setup, the framework can exploit the structural similarities of DR policies for different rooms, while reducing computational complexity, incorporating sequentially arriving mea-

surements, and tracking slow variations of optimal policies over time.

For designing ML methods, one needs to specify the space of functions in which the best approximation of the optimal policy is searched for. A simplest example would be the space of linear functions, which map the features linearly to the target labels. Nonlinear relationship can be accommodated by adopting a parameterized family of functions, but selecting a proper parameterization often requires detailed domain knowledge and educated guesses. Kernel-based learning allows one to explore very flexible non-parametric classes of nonlinear functions, namely, reproducing kernel Hilbert spaces (RKHSs). A kernel-based efficient lifelong learning algorithm (KELLA) was developed recently, which is adapted here for our problem [23]. In particular, as our problem contains a constraint, whereas typical classification problems are formulated as unconstrained ones, a Lagrange duality-based method is employed to transform the overall problem into a series of smaller unconstrained learning problems.

Summarizing, the main contributions of this work are:

- A DR problem involving major controllable loads in building EMS is formulated, where the occupants' discomfort levels are minimized while meeting a specified DR target.
- The lifelong learning approach is employed to exploit structural similarities in the control policies for individual spaces in the building, accommodating the nonlinearities in a kernel-based learning framework.
- The coupling DR constraint is relaxed via the dual decomposition technique so that the lifelong learning method, originally developed for unconstrained ML problems, can be readily adapted to the optimization problem at hand.
- The proposed algorithm is tested by numerical experiments designed partly based on real data sets.

The rest of this paper is organized as follows. Related works are reviewed in Sec. II. The system model for the building EMS and the DR problem formulation are given in Sec. III. The overall solution architecture based on the dual decomposition method is described in Sec. IV. The proposed lifelong learning solution is derived in Sec. V. The proposed method is tested using semi-real data sets in Sec. VI. The conclusions are provided in Sec. VII.

II. RELATED WORKS

Existing building DR algorithms are summarized here. They can be categorized as optimization-based and ML-based.

A. OPTIMIZATION-BASED BUILDING DR ALGORITHMS

The building DR problems can be formulated as optimization problems of load schedules. A load dispatch model for residential buildings was formulated as a mixed integer linear program to achieve peak reduction and to meet DR requirements [4]. The role of heating, ventilation and air

conditioning (HVAC) systems in commercial buildings as DR and ancillary service resources was explored in [5]. The control strategy for HVAC systems in commercial buildings was investigated in [6], where a transactive market mechanism was employed based on a convex optimization formulation. A load shedding strategy for lighting systems was studied, where the dimming levels were adjusted to meet the DR requirements without violating the minimum illuminance requested by users, through a constrained quadratic optimization formulation [7]. Based on the predicted load profiles, an optimal scheduling strategy for HVAC and energy storage systems was designed using model predictive control (MPC) to reduce the building energy bill [8]. In these works, however, the uncertainty due to various stochastic variables were often neglected by assuming the availability of accurate sensor readings and/or their predictions.

A number of studies focused on energy management for DR under uncertainty. A robust commercial building DR strategy was proposed employing a genetic algorithm under cooling load prediction uncertainty [9]. Limitations of conventional DR methods designed for individual buildings were observed and a simple coordinated DR strategy was developed for demonstrating the need to develop a coordinated DR strategy for improving the building group-level performance [10].

B. ML-BASED DR ALGORITHMS
Recently, there have been numerous efforts to incorporate ML techniques to DR algorithms, in order to cope with the uncertainty present in realistic DR scenarios. They range from using ML techniques simply to improve the prediction of stochastic variables [11]–[13] to incorporating ML models directly to the optimization formulations [14]–[16].

A neural network (NN)-based prediction model for the energy consumption of air conditioning systems in office buildings was proposed in [11], where the averaging effect of the prediction errors was reported for aggregated curtailment for DR. A support vector regression (SVR)-based building load prediction method was proposed for commercial buildings, where the HVAC set points capturing energy consumption trend were used as features [12]. A framework for forecasting the reduction in the individual users' energy consumption during the DR period was proposed using various ML methods including ordinary least-squares (OLS) regression, K-nearest neighbors (KNN), and support vector regression (SVR) algorithms [13]. A reinforcement learning approach was taken for the residential building EMS, where the running time and energy allocation of each scheduled device were determined by learning individual consumers' preferences and price variation [14].

Several studies proposed pricing schemes for DR. A dynamic pricing algorithm for DR in a hierarchical energy market was proposed to simultaneously optimize the profit of service providers and the costs of consumers, where the uncertainty in the load demand and the wholesale energy price was tackled using a reinforcement learning method [15].
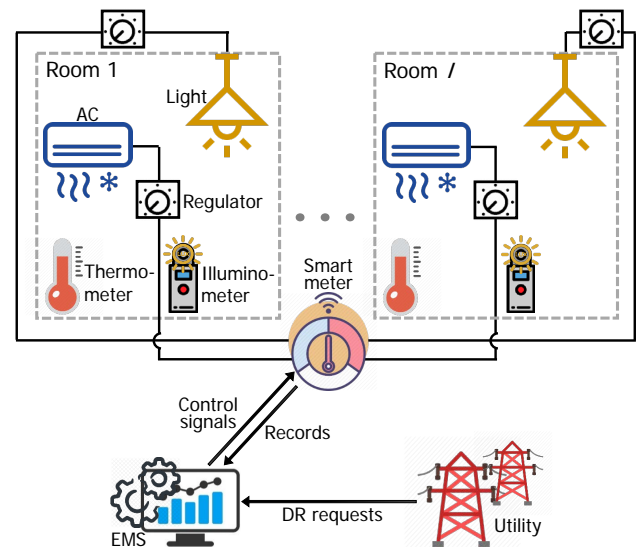

FIGURE 1: Architecture of the considered building EMS for DR, where each room is equipped with lighting and AC systems, a thermometer, and an illuminometer.

An optimal dynamic pricing strategy for retail energy was developed, where an online learning-based algorithm was proposed to learn the behaviors of customers by observing their responses to varying prices [16].

III. SYSTEM MODEL AND PROBLEM FORMULATION
First, the overall architecture of the considered building EMS is explained. In this work, since we consider the building, where the dominant electric energy consumption is ascribed to lighting and AC systems, we focus on controlling these two appliances for DR. Extensions for including additional types of appliances are straightforward. Fig. 1 illustrates a building with each of the I rooms equipped with lighting and AC systems. An illuminometer and a thermometer in each room measure the ambient illuminance due to sunlight and indoor temperature of the room, respectively. The smart meter records the energy consumption of the controllable appliances and communicates the information to the processing unit of the EMS. In addition, the smart meter controls the power consumption of the appliances according to the control signals sent from the EMS controller. The building EMS needs to meet a DR request, which is the overall reduction in the energy consumption in the building, by controlling the appliances in the individual rooms.

The energy consumption of the controllable appliances can be modeled as follows. Let $E_i^{\text{light}*}$ ($E_i^{\text{ac}*}$, respectively) be the energy consumption of the lighting (AC) system for room $i \in \{1, \ldots, I\}$ at the *desired* user comfort setting. Let l_i^* denote the corresponding desired illuminance in lux, l_i^{sun} the ambient illuminance due to natural sunlight in lux, η_i the luminous efficacy of the lighting system of room i in lm/kW, S_i the surface area of room i in m^2, and τ the time duration in hour (h). In this paper, it is assumed for

simplicity that whenever a DR request arrives, the EMS makes the control decisions for the next hour, i.e., $\tau = 1$ is used. Similarly, let t_i^* represent the desired temperature setting for room i in °C, t_i^{cur} the current indoor temperature in °C, A_i the energy consumption variation with respect to the temperature variation during an hour in kWh/°C, and B_i the minimum energy consumption required to maintain the current temperature for an hour in kWh. Then, $E_i^{\mathrm{light}*}$ can be modeled as [24]

$$E_i^{\mathrm{light}*} = \eta_i^{-1} S_i (l_i^* - l_i^{\mathrm{sun}}) \tau \qquad (1)$$

while $E_i^{\mathrm{ac}*}$ can be expressed as [25]–[27][1]

$$E_i^{\mathrm{ac}*} = B_i - A_i (t_i^* - t_i^{\mathrm{cur}}). \qquad (2)$$

Note that the illuminance due to sunlight, l_i^{sun}, and the current indoor temperature, t_i^{cur}, are stochastic variables. However, they can be measured by an illuminometer and a thermometer, respectively. Since the desired illuminance and indoor temperature of each room often depend on the user preferences and the room type, we assume that l_i^* and t_i^* cannot be measured directly. In addition, we assume that η_i, A_i, and B_i are unknown to the controller as well since they can be influenced by various factors such as physical characteristics of the lighting and the AC systems, which may be difficult to estimate and prone to change. Consequently, we model $E_i^{\mathrm{light}*}$ and $E_i^{\mathrm{ac}*}$ themselves as stochastic variables.

Let $\mathbf{a}_i := [a_i^{\mathrm{light}}, a_i^{\mathrm{ac}}]^\top$, where $0 \le a_i^{\mathrm{light}} \le 1$ and $0 \le a_i^{\mathrm{ac}} \le 1$, denote the actions that control the energy consumption of the lighting and the AC systems of room i, respectively, where $[\cdot]^\top$ represents the vector/matrix transposition. Then, the actual energy consumption of the lighting system, denoted as E_i^{light}, and the same for the AC system, denoted as E_i^{ac}, can be expressed as

$$E_i^{\mathrm{light}} = a_i^{\mathrm{light}} E_i^{\mathrm{light}\dagger} \qquad (3)$$

and

$$E_i^{\mathrm{ac}} = a_i^{\mathrm{ac}} E_i^{\mathrm{ac}\dagger} \qquad (4)$$

respectively, where $E_i^{\mathrm{light}\dagger}$ and $E_i^{\mathrm{ac}\dagger}$ are the baseline energy consumption of the lighting and AC systems of room i, respectively, which are determined by the EMS according to the typical energy consumption of these systems. That is, a_i^{light} and a_i^{ac} capture the fractions of the energy expended relative to the baseline energy consumption. When a_i^{light} and a_i^{ac} are less than 1, reductions in the energy consumption of the lighting and AC systems from the baseline levels occur.

In this study, the user discomfort level is quantified simply by the amount of deviation (reduction) in the energy con-

[1]As our work develops a ML-based DR algorithm, the proposed method can accommodate a broad range of appliance models. Thus, in this work, relatively simple yet commonly employed models are adopted to set the stage for the algorithm development. It is also worth noting that the energy consumption breakdowns among the loads can depend on the size and purpose (residential/commercial) of the buildings/rooms. Such variations can be readily captured by setting the model parameters S_i, η_i, A_i, and B_i appropriately [26].

sumption from the *desired* comfort setting. By quantifying the discomfort levels in terms of energy, combining the discomfort for two different systems into a single cost function becomes straightforward. Specifically, the difference in the energy consumption for system $j \in \{\mathrm{light}, \mathrm{ac}\}$ can be expressed as

$$\Delta E_i^j = E_i^{j*} - E_i^j = E_i^{j*} - a_i^j E_i^{j\dagger}. \qquad (5)$$

Upon defining $\mathbf{E}_i^* := [E_i^{\mathrm{light}*}, E_i^{\mathrm{ac}*}]^\top$, the total user discomfort can then be modeled simply as the sum of the squared deviations.

$$D_i(\mathbf{a}_i, \mathbf{E}_i^*) = (\Delta E_i^{\mathrm{light}})^2 + (\Delta E_i^{\mathrm{ac}})^2 \qquad (6)$$
$$= (E_i^{\mathrm{light}*} - a_i^{\mathrm{light}} E_i^{\mathrm{light}\dagger})^2$$
$$+ (E_i^{\mathrm{ac}*} - a_i^{\mathrm{ac}} E_i^{\mathrm{ac}\dagger})^2. \qquad (7)$$

The squared deviations are used here based on the intuitive notion that the user discomfort can increase quickly as the deviation grows. However, any other discomfort functions can be employed in our algorithm, as long as they are twice differentiable. For example, on can alternatively model the discomfort level to be zero in a certain comfort range, and outside the range a step or a linear penalty can be assessed. In such cases, a step function $u(x - a)$ can be approximated by a twice differentiable function $\frac{1}{1 + e^{-(x-a)/b}}$, and the hinge loss function $\max\{0, x - a\}$ by $\frac{x-a}{1 + e^{-(x-a)/b}}$ with a small positive b. Finally, in order to combine the user discomfort levels for different rooms in a balanced way, the normalized user discomfort for each room i is defined as

$$\hat{D}_i(\mathbf{a}_i, \mathbf{E}_i^*) = \frac{D_i(\mathbf{a}_i, \mathbf{E}_i^*)}{\|\mathbf{E}_i^*\|_2^2}. \qquad (8)$$

Let E^{DR} be the DR request from the utility, which is a mandated reduction amount in the total building energy consumption. The energy reduction amount for room i can be expressed as

$$E_i^{\mathrm{red}}(\mathbf{a}_i) := (1 - a_i^{\mathrm{light}}) E_i^{\mathrm{light}\dagger} + (1 - a_i^{\mathrm{ac}}) E_i^{\mathrm{ac}\dagger}. \qquad (9)$$

The sum of E_i^{red} over all rooms must be larger than or equal to E^{DR} to satisfy the DR request.

Thus, our goal is to come up with a control policy that determines actions $\mathbf{a}_i$ based on the available measurements l_i^{sun} and t_i^{cur} for each room $i = 1, 2, \ldots, I$ so that the DR request is met and the overall user discomfort levels are minimized. Upon defining $\boldsymbol{\xi}_i := [l_i^{\mathrm{sun}}, t_i^{\mathrm{cur}}]^\top$, the optimal control policy for each i is a mapping $\pi_i : \boldsymbol{\xi}_i \mapsto \mathbf{a}_i$. Note that the actions can be dependent on $\boldsymbol{\xi}_i$, which is readily available from sensor readings, but not on $\mathbf{E}_i^*$. A stochastic optimization problem for obtaining the optimal policies can
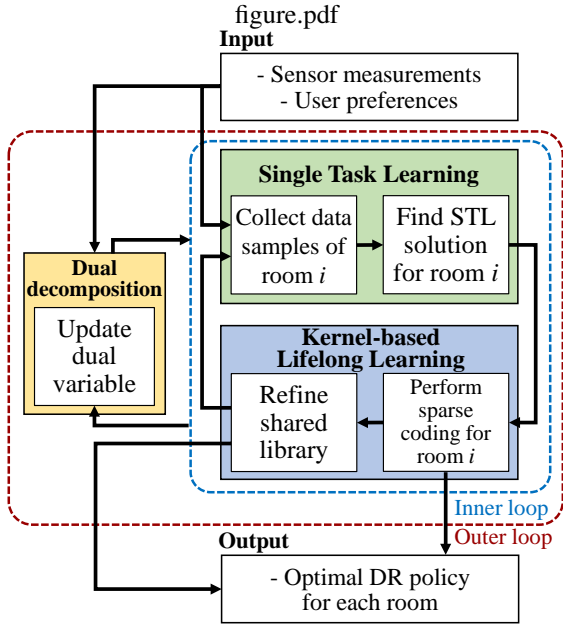
FIGURE 2: The structure of the proposed approach for building DR.

thus be formulated as

$$\underset{\{\boldsymbol{\pi}_i\}}{\text{minimize}} \sum_{i=1}^{I} \mathbb{E}[\hat{D}_i(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i), \mathbf{E}_i^*)] \tag{10a}$$

$$\text{subject to} \sum_{i=1}^{I} \mathbb{E}[E_i^{\text{red}}(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i))] \geq E^{\text{DR}} \tag{10b}$$

where $\mathbb{E}[\cdot]$ denotes the expectation with respect to the random vectors $\{\boldsymbol{\xi}_i\}$ and $\{\mathbf{E}_i^*\}$.

Remark: The goal of our building DR is to to implement the energy reduction in the controllable loads according to the given DR request at the minimum discomfort for the occupants. Thus, it is assumed that the background load fluctuations are taken into account by the utility when generating the DR request signal. On the other hand, it is worth noting that our ML-based framework can seamlessly incorporate the background load uncertainty with minor modifications in the formulation. Specifically, the change in the background load ΔL_i in the i-th room can be augmented in the energy reduction in (9) as

$$E_i^{\text{red}}(\mathbf{a}_i, \Delta L_i) := (1 - a_i^{\text{light}})E_i^{\text{light}\dagger} + (1 - a_i^{\text{ac}})E_i^{\text{ac}\dagger} + \Delta L_i$$

where ΔL_i is random. Then, formulation (10) can be modified by replacing $E_i^{\text{red}}(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i))$ in (10b) as $E_i^{\text{red}}(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i), \Delta L_i)$, where the expectation is now taken with respect to not only $\{\boldsymbol{\xi}_i\}$ and $\{\mathbf{E}_i^*\}$, but also $\{\Delta L_i\}$. ∎

In the following sections, our solution approach for (10) is described. The structure of our method is shown in Fig. 2, which consists of three parts. First, the overall problem (10) is divided to the subproblems involving individual rooms using a dual decomposition method (Sec. IV-A). Then, the

subproblems are tackled independently via single task learning (STL), coordinated by the dual variable (Sec. IV-B). The shared skills from the STL policies are collected in a kernel-based lifelong learning framework (Sec. V). The algorithm iterates these processes based on the sensor measurements and user preference inputs, and upon convergence, the optimal DR policy of each room is obtained as the solution to (10).

IV. SINGLE TASK LEARNING BASED SOLUTION

There are a couple of issues associated with solving the optimization problem (10). First, the optimal policies need to be found jointly for all rooms due to constraint (10b), which couples all rooms. For a large building with many rooms, this may be prohibitively complex. Second, since the problem involves expectations, the knowledge of the probability distributions for $\{\boldsymbol{\xi}_i\}$ and $\{\mathbf{E}_i^*\}$ is required, which may not be readily available in practice. Even if the distributions can be estimated, calculating the integrals may significantly add to the computational complexity.

In the following, the Lagrange relaxation-based dual decomposition method is employed to decouple the overall problem into I subproblems for individual rooms. Subsequently, the relaxed problem is tackled taking a ML approach. Thus, one can work with samples of $(\{\boldsymbol{\xi}_i\}, \{\mathbf{E}_i^*\})$ to approximate the expectations. Once the optimal policies are learned, they can be evaluated very quickly without requiring complex optimization, facilitating the implementation of the real-time control in the EMS.

A. DUAL DECOMPOSITION

The constraint in (10b) can be relaxed via the Lagrange dual method, yielding a decoupled formulation, where each room's policy can be optimized separately, coordinated by the Lagrange multiplier [28]. Given a Lagrange multiplier $\nu \geq 0$, the Lagrangian for (10) can be expressed as

$$\mathcal{L}(\{\boldsymbol{\pi}_i\}; \nu)$$
$$= \sum_{i=1}^{I} \mathbb{E}[\hat{D}_i(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i), \mathbf{E}_i^*) - \nu E_i^{\text{red}}(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i))] + \nu E^{\text{DR}}. \tag{11}$$

The dual function can be obtained by minimizing the Lagrangian as

$$\mathcal{D}(\nu) = \min_{\{\boldsymbol{\pi}_i\}} \mathcal{L}(\{\boldsymbol{\pi}_i\}; \nu). \tag{12}$$

Note that (12) can be split into I subproblems, which can be solved independently for a given ν. That is, upon defining the cost function

$$C_i(\mathbf{a}_i, \mathbf{E}_i^*; \nu) := \hat{D}_i(\mathbf{a}_i, \mathbf{E}_i^*) - \nu E_i^{\text{red}}(\mathbf{a}_i), \ i = 1, 2, \ldots, I \tag{13}$$

the optimal control policies $\{\boldsymbol{\pi}_i^*\}$ can be obtained by solving the following subproblems:

$$\min_{\boldsymbol{\pi}_i} \mathbb{E}[C_i(\boldsymbol{\pi}_i(\boldsymbol{\xi}_i), \mathbf{E}_i^*; \nu)], \ i = 1, 2, \ldots, I. \tag{14}$$

Now, the optimal dual variable can be found by solving the dual problem defined as

$$\max_{\nu \geq 0} \mathcal{D}(\nu). \quad (15)$$

Thus, ν can be updated based on the subgradient method as

$$\nu_{t+1} = \left[\nu_t + \gamma_t \left(E^{\mathrm{DR}} - \sum_{i=1}^{I} \mathbb{E}\left[E_i^{\mathrm{red}}(\boldsymbol{\pi}_i^*(\boldsymbol{\xi}_i)) \right] \right) \right]^+ \quad (16)$$

where t denotes the iteration index, γ_t denotes a small positive step size, and $[\cdot]^+ := \max\{0, \cdot\}$.

The overall algorithm iterates the solution of (14) and dual update (16) until convergence. Upon convergence, the obtained optimal control policies $\{\boldsymbol{\pi}_i^*\}$ will satisfy the DR request given in (10b) and minimize the total average discomfort across the building at the same time.

B. LEARNING-BASED SOLUTION

The remaining issue is to solve the stochastic optimization problem in (14). A pragmatic approach to approximate the expectations would be to collect representative samples $(\boldsymbol{\xi}_{i,n}, \mathbf{E}_{i,n}^*)$, $n = 1, 2, \ldots, N_i$, from the sensors installed in each room i and approximate (14) as

$$\min_{\boldsymbol{\pi}_i} \frac{1}{N_i} \sum_{n=1}^{N_i} C_i(\boldsymbol{\pi}_i(\boldsymbol{\xi}_{i,n}), \mathbf{E}_{i,n}^*; \nu_t). \quad (17)$$

Our approach is to search for the optimal policy $\boldsymbol{\pi}^*$ in an appropriate function space using ML tools. To accommodate the nonlinear class of functions, kernel-based learning is employed. Given an RKHS $\mathcal{H}$ defined by a positive-definite kernel function $\kappa(\cdot, \cdot) : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$, the kernel function κ determines a nonlinear mapping $\phi : \mathbb{R}^2 \to \mathcal{H}$ that maps $\boldsymbol{\xi}_i$ to a high-dimensional feature space $\phi(\boldsymbol{\xi}_i) \in \mathcal{H}$. Let $\boldsymbol{\theta}_i^{\mathrm{light}}, \boldsymbol{\theta}_i^{\mathrm{chill}} \in \mathcal{H}$ and define $\boldsymbol{\theta}_i := \left[\boldsymbol{\theta}_i^{\mathrm{light}^\top}, \boldsymbol{\theta}_i^{\mathrm{chill}^\top} \right]^\top$. Then, it can be postulated that the control policy $\boldsymbol{\pi}_i(\boldsymbol{\xi}_i)$ for each room i can be well approximated by

$$\boldsymbol{\pi}(\boldsymbol{\xi}_i, \boldsymbol{\theta}_i) := \left[h(\langle \phi(\boldsymbol{\xi}_i), \boldsymbol{\theta}_i^{\mathrm{light}} \rangle), h(\langle \phi(\boldsymbol{\xi}_i), \boldsymbol{\theta}_i^{\mathrm{ac}} \rangle) \right]^\top \quad (18)$$

where $\langle \cdot, \cdot \rangle$ represents the inner product on $\mathcal{H}$, and $h(x) := 1/(1 + e^{-x})$ ensures that each action is always within $[0, 1]$. Therefore, the optimal policy for room i can be characterized by solving (17) with respect to $\boldsymbol{\theta}_i$ as

$$\boldsymbol{\theta}_i^* = \arg\min_{\boldsymbol{\theta}_i \in \mathcal{H}} \frac{1}{N_i} \sum_{n=1}^{N_i} C_i(\boldsymbol{\pi}(\boldsymbol{\xi}_{i,n}, \boldsymbol{\theta}_i), \mathbf{E}_{i,n}^*; \nu_t). \quad (19)$$

Upon defining $\boldsymbol{\Phi}(\boldsymbol{\Xi}_i) := [\phi(\boldsymbol{\xi}_{i,1}), \ldots, \phi(\boldsymbol{\xi}_{i,N_i})] \in \mathcal{H}^{N_i}$ and a block diagonal matrix $\boldsymbol{\Phi}_i^{\mathrm{d}} := \mathrm{diag}\{\boldsymbol{\Phi}(\boldsymbol{\xi}_i), \boldsymbol{\Phi}(\boldsymbol{\xi}_i)\}$, the representer theorem guarantees that the solution to the empirical problem (19) can be represented as a linear combination of features of the given input samples [29].

$$\boldsymbol{\theta}_i^* = \boldsymbol{\Phi}_i^{\mathrm{d}} \mathbf{w}_i^* \quad (20)$$

where $\mathbf{w}_i^* := [\mathbf{w}_i^{\mathrm{light}*^\top}, \mathbf{w}_i^{\mathrm{ac}*^\top}]^\top$ for coefficient vectors

$\mathbf{w}_i^{\mathrm{light}*} \in \mathbb{R}^{N_i}$ and $\mathbf{w}_i^{\mathrm{ac}*} \in \mathbb{R}^{N_i}$. The significance of this step is in converting the optimization problem in (19) formulated possibly in an infinite-dimensional space to that of finding vectors in $\mathbb{R}^{N_i}$. Note that (19) is a STL problem, viewing each room i as a task, since the policy for each room is learned in isolation based on its own input features. An extension based on the MTL approach is introduced in the next section.

V. KERNEL LIFELONG LEARNING FOR DR

In the STL formulation, the control policies for the individual rooms are learned independently. However, it is reasonable to expect the policies to share intrinsic structures, although there would be differences as well. For example, the control policies for campus classrooms would have a lot in common due to their similarities in sizes and usage patterns. Such shared structures can be exploited in an MTL framework.

Specifically, a recently developed kernel-based efficient lifelong learning algorithm, called KELLA, is adapted here for the building DR problem [23]. The KELLA performs kernel-based MTL in an online manner for a sequence of tasks, effectively transferring knowledge across different tasks, while significantly lowering the computational complexity. It also allows tracking of slow variations in the optimal policies.

A. KERNEL MTL-BASED DR FORMULATION

One approach to capture the shared structure for MTL is to adopt a union-of-subspaces model for the learned classifiers/policies [20]. Specifically, in our setup, given the libraries of K representative atoms for the lighting and AC system policies, denoted as $\mathbf{L}^{\mathrm{light}} := [\mathbf{l}_1^{\mathrm{light}}, \ldots, \mathbf{l}_K^{\mathrm{light}}] \in \mathcal{H}^K$ and $\mathbf{L}^{\mathrm{ac}} := [\mathbf{l}_1^{\mathrm{ac}}, \ldots, \mathbf{l}_K^{\mathrm{ac}}] \in \mathcal{H}^K$, respectively, and the combined library $\mathbf{L} := [\mathbf{L}^{\mathrm{light}^\top}, \mathbf{L}^{\mathrm{ac}^\top}]^\top$, it is postulated that for a sparse coefficient vector $\mathbf{s}_i$

$$\boldsymbol{\theta}_i = \mathbf{L}\mathbf{s}_i, \ i = 1, 2, \ldots, I. \quad (21)$$

The MTL problem for all rooms can then be stated as

$$\min_{\mathbf{L} \in \mathcal{H}^{2K}} \frac{1}{I} \sum_{i=1}^{I} \min_{\mathbf{s}_i} \left\{ \frac{1}{N_i} \sum_{n=1}^{N_i} C_i(\boldsymbol{\pi}(\boldsymbol{\xi}_{i,n}, \mathbf{L}\mathbf{s}_i), \mathbf{E}_{i,n}^*; \nu_t) \right.$$
$$\left. + \mu \|\mathbf{s}_i\|_1 \right\} + \lambda \|\mathbf{L}\|_{\mathcal{H}}^2 \quad (22)$$

where $\|\mathbf{s}_i\|_1$, the ℓ_1-norm of $\mathbf{s}_i$, is a regularizer that promotes sparsity in $\mathbf{s}_i$, and $\|\mathbf{L}\|_{\mathcal{H}}^2 := \sum_{k=1}^{K} [\|\mathbf{l}_k^{\mathrm{light}}\|_{\mathcal{H}}^2 + \|\mathbf{l}_k^{\mathrm{ac}}\|_{\mathcal{H}}^2]$ controls the complexity of the library. Parameters μ and λ are positive weights adjusting the strengths of the regularizers.

Solving (22) directly requires processing the samples for all rooms jointly, which may incur significant computational and memory requirements, and hinders the online implementation. To mitigate this issue, a proxy is derived for the cost function from the second-order Taylor approximation around

the STL optimal solutions $\{\boldsymbol{\theta}_i^*\}$ [23]. That is,

$$\frac{1}{N_i} \sum_{n=1}^{N_i} C_i(\boldsymbol{\pi}(\boldsymbol{\xi}_{i,n}, \mathbf{L}\mathbf{s}_i), \mathbf{E}_{i,n}^*; \nu_t) \approx \left\| \boldsymbol{\theta}_i - \boldsymbol{\theta}_i^* \right\|_{\mathbf{H}_i}^2 + \mathrm{const.} \tag{23}$$

where $\mathbf{H}_i := \frac{1}{2} \nabla_{\boldsymbol{\theta}_i, \boldsymbol{\theta}_i}^2 \frac{1}{N_i} \sum_{n=1}^{N_i} C_i(\boldsymbol{\pi}(\boldsymbol{\xi}_{i,n}, \boldsymbol{\theta}_i), \mathbf{E}_{i,n}^*; \nu_t)\big|_{\boldsymbol{\theta}_i = \boldsymbol{\theta}_i^*}$ is the Hessian, and $\|\mathbf{v}\|_{\mathbf{H}}^2 := \langle \mathbf{v}, \mathbf{H}\mathbf{v} \rangle$. Note that the first order term vanishes at $\boldsymbol{\theta}_i = \boldsymbol{\theta}_i^*$ due to the first-order optimality.

The Hessian can be computed as follows. For each system $j \in \{\mathrm{light}, \mathrm{ac}\}$, let us define $m_{i,n}^j := \langle \boldsymbol{\phi}(\boldsymbol{\xi}_{i,n}), \boldsymbol{\theta}_i^j \rangle$. Define also

$$\mathbf{Q}_i^j(m_{i,n}^j) := \frac{\partial^2}{\partial m_{i,n}^j \partial (m_{i,n}^j)^\top} C_i(\boldsymbol{\pi}(\boldsymbol{\xi}_i, \boldsymbol{\theta}_i), \mathbf{E}_{i,n}^*; \nu_t) \tag{24}$$

and $\mathbf{Q}_i^j := \mathrm{diag}\{Q_i^j(m_{i,1}^j), \ldots, Q_i^j(m_{i,N_i}^j)\}$. Then, upon defining $\mathbf{Q}_i^{\mathrm{d}} := \mathrm{diag}\{\mathbf{Q}_i^{\mathrm{light}}, \mathbf{Q}_i^{\mathrm{ac}}\}$, $\mathbf{H}_i$ is given by

$$\mathbf{H}_i = \frac{1}{N_i} \boldsymbol{\Phi}_i^{\mathrm{d}} \mathbf{Q}_i^{\mathrm{d}} \boldsymbol{\Phi}_i^{\mathrm{d}\top}. \tag{25}$$

With these, (22) can be approximated as

$$\min_{\mathbf{L} \in \mathcal{H}^{2K}} \frac{1}{I} \sum_{i=1}^{I} \min_{\mathbf{s}_i} \left\{ \left\| \boldsymbol{\theta}_i^* - \mathbf{L}\mathbf{s}_i \right\|_{\mathbf{H}_i}^2 + \mu \|\mathbf{s}_i\|_1 \right\} + \lambda \|\mathbf{L}\|_{\mathcal{H}}^2. \tag{26}$$

Note that (26) is reminiscent of the kernel dictionary learning formulation [30].

Invoking again the represener theorem, one can show that (26) can be reduced to a finite dimensional optimization problem. Define $\boldsymbol{\Phi}(\boldsymbol{\Xi}) := [\boldsymbol{\Phi}(\boldsymbol{\Xi}_1), \ldots, \boldsymbol{\Phi}(\boldsymbol{\Xi}_I)] \in \mathcal{H}^N$, where $N := \sum_{i=1}^{I} N_i$, and $\boldsymbol{\Phi}^{\mathrm{d}} := \mathrm{diag}\{\boldsymbol{\Phi}(\boldsymbol{\Xi}), \boldsymbol{\Phi}(\boldsymbol{\Xi})\}$. Then, the optimal $\mathbf{L}$ can be represented as a linear combination

$$\mathbf{L} = \boldsymbol{\Phi}^{\mathrm{d}} \mathbf{A} \tag{27}$$

where $\mathbf{A} := [(\mathbf{A}^{\mathrm{light}})^\top, (\mathbf{A}^{\mathrm{ac}})^\top]^\top$ is the coefficient matrix.

B. LIFELONG LEARNING ALGORITHM

As the number of rooms I increases, the law of large numbers (LLN) starts to kick in for the summation in (26). Thus, one can pursue the following stochastic optimization problem.

$$\min_{\mathbf{L}} \mathbb{E}\Big[\underbrace{\min_{\mathbf{s}} \|\boldsymbol{\theta}^* - \mathbf{L}\mathbf{s}\|_{\mathbf{H}}^2 + \mu \|\mathbf{s}\|_1}_{:= y(\mathbf{L}; \boldsymbol{\theta}^*, \mathbf{H})} \Big] + \lambda \|\mathbf{L}\|_{\mathcal{H}}^2 \tag{28}$$

where the expectation is taken with respect to $\boldsymbol{\theta}^*$ and $\mathbf{H}$. This problem can be solved in an online manner based on the stochastic gradient descent (SGD) method using $\{\boldsymbol{\theta}_i^*\}$ and $\{\mathbf{H}_i\}$ as samples.

Suppose that the input data samples $\{\boldsymbol{\xi}_{i,n}\}_{n=1}^{N_i}$ from room i are acquired in the i-th iteration. Then, the STL solution $\boldsymbol{\theta}_i$ is obtained from (19). Given the last iterate of the library $\mathbf{L}(i-1)$, the sparse coding is performed as

$$\mathbf{s}_i^* = \arg\min_{\mathbf{s}} \|\boldsymbol{\theta}_i^* - \mathbf{L}(i-1)\mathbf{s}\|_{\mathbf{H}_i}^2 + \mu \|\mathbf{s}\|_1. \tag{29}$$

Then, the instantaneous gradient of the objective of (28) is given by

$$\nabla \left\{ y(\mathbf{L}(i-1); \boldsymbol{\theta}_i^*, \mathbf{H}_i) + \lambda \|\mathbf{L}(i-1)\|_{\mathcal{H}}^2 \right\}$$
$$= \mathbf{H}_i(\mathbf{L}(i-1)\mathbf{s}_i^* - \boldsymbol{\theta}_i^*)\mathbf{s}_i^{*\top} + 2\lambda \mathbf{L}(i-1). \tag{30}$$

Using (30), the SGD update for $\mathbf{L}$ can be expressed as

$$\mathbf{L}(i) = \mathbf{L}(i-1) - \epsilon \nabla \{ y(\mathbf{L}(i-1); \boldsymbol{\theta}_i^*, \mathbf{H}_i) + \lambda \|\mathbf{L}(i-1)\|_{\mathcal{H}}^2 \}$$
$$= (1 - 2\lambda\epsilon)\mathbf{L}(i-1) - \epsilon \mathbf{H}_i(\mathbf{L}(i-1)\mathbf{s}_i^* - \boldsymbol{\theta}_i^*)\mathbf{s}_i^{*\top} \tag{31}$$

where $\epsilon > 0$ is a step size. Per (27), the update can be equivalently done in terms of the coefficient matrix $\mathbf{A} \in \mathbb{R}^{2N \times K}$. Plugging (20), (27), and (25) into (31), one can obtain

$$\boldsymbol{\Phi}^{\mathrm{d}} \mathbf{A}(i) = (1 - 2\lambda\epsilon)\boldsymbol{\Phi}^{\mathrm{d}} \mathbf{A}(i-1)$$
$$- \frac{\epsilon}{N_i} \boldsymbol{\Phi}_i^{\mathrm{d}} \mathbf{Q}_i^{\mathrm{d}} \boldsymbol{\Phi}_i^{\mathrm{d}\top} \left(\boldsymbol{\Phi}^{\mathrm{d}} \mathbf{A}(i-1)\mathbf{s}_i^* - \boldsymbol{\Phi}_i^{\mathrm{d}} \mathbf{w}_i^* \right) \mathbf{s}_i^{*\top}. \tag{32}$$

Thus, upon defining $N_{1:I'} := \sum_{i=1}^{I'} N_i$, $\boldsymbol{\Phi}(\boldsymbol{\Xi}_{1:i-1}) := [\boldsymbol{\Phi}(\boldsymbol{\xi}_1), \ldots, \boldsymbol{\Phi}(\boldsymbol{\xi}_{i-1})] \in \mathbb{R}^{N_{1:i-1}}$, $\mathbf{K}_{i,i} := \boldsymbol{\Phi}(\boldsymbol{\xi}_i)^\top \boldsymbol{\Phi}(\boldsymbol{\xi}_i) \in \mathbb{R}^{N_i \times N_i}$, and $\mathbf{K}_{i,1:i-1} := \boldsymbol{\Phi}(\boldsymbol{\xi}_i)^\top \boldsymbol{\Phi}(\boldsymbol{\Xi}_{1:i-1}) \in \mathbb{R}^{N_i \times N_{1:i-1}}$, the update rule for $\mathbf{A}^{\mathrm{light}}$ and $\mathbf{A}^{\mathrm{ac}}$ can be written as

$$\mathbf{A}^j(i) = \begin{bmatrix} (1 - 2\lambda\epsilon)\mathbf{A}^j(i-1) \\ -\frac{\epsilon}{N_i} \mathbf{Q}_i^j \left(\mathbf{K}_{i,1:i-1}\mathbf{A}^j(i-1)\mathbf{s}_i^* - \mathbf{K}_{i,i}\mathbf{w}_i^{j*} \right) \mathbf{s}_i^{*\top} \end{bmatrix}$$
$$\in \mathbb{R}^{N_{1:i} \times K}, \text{ for } j \in \{\mathrm{light}, \mathrm{ac}\}. \tag{33}$$

Note that $\mathbf{K}_{i,i}$ and $\mathbf{K}_{i,1:i-1}$ can be computed without actually specifying the nonlinear mapping ϕ, as long as the kernel function κ is given, which is often called the kernel trick [31]. In fact, the kernel trick is instrumental for the sparse coding step in (29), as it can be re-written as

$$\mathbf{s}_i^* = \arg\min_{\mathbf{s}} \left\| (\mathbf{Q}_i^{\mathrm{d}})^{\frac{1}{2}} \mathbf{K}_{i,i}^{\mathrm{d}} \mathbf{w}_i^* - (\mathbf{Q}_i^{\mathrm{d}})^{\frac{1}{2}} \mathbf{K}_{i,1:i-1}^{\mathrm{d}} \mathbf{A}(i-1)\mathbf{s} \right\|_2^2$$
$$+ \mu \|\mathbf{s}\|_1 \tag{34}$$

where $\mathbf{K}_{i,i}^{\mathrm{d}} := \mathrm{diag}\{\mathbf{K}_{i,i}, \mathbf{K}_{i,i}\}$ and $\mathbf{K}_{i,1:i-1}^{\mathrm{d}} := \mathrm{diag}\{\mathbf{K}_{i,1:i-1}, \mathbf{K}_{i,1:i-1}\}$. Problem (34) can be solved efficiently using various sparse coding solvers such as the SPAMS package [32].

Finally, $\boldsymbol{\theta}_i^*$ is computed as

$$\boldsymbol{\theta}_i^* = \mathbf{L}\mathbf{s}_i^* = \boldsymbol{\Phi}^{\mathrm{d}} \mathbf{A}\mathbf{s}_i^* \tag{35}$$

from which the policy function for each room i can be obtained as $\boldsymbol{\pi}(\boldsymbol{\xi}_i, \boldsymbol{\theta}_i^*)$ in (18), again using the kernel trick.

Overall, the lifelong learning is done concurrently with the dual variable update so that the DR request is satisfied. The algorithm is listed in Table 1. Matrix $\mathbf{A}$ can be initialized from solving (26) using a data set involving a small number I_0 of rooms. The inner loop in lines 3–8 performs the lifelong learning for the I rooms sequentially for the given value of Lagrange multiplier ν_t. Lines 10–12 represent the polishing step, in which the sparse codes for all the rooms are updated based on the most up-to-date library. Line 13 performs the dual variable update, where the expectation is taken again from the samples $\{(\boldsymbol{\xi}_{i,n}, \mathbf{E}_{i,n}^*)\}_{n=1}^{N_i}$ of each room i. The outer loop is repeated until the algorithm is converged.

Input: $(\boldsymbol{\xi}_{i,n}, \mathbf{E}^*_{i,n})$ for $n = 1, \ldots, N_i$, $i = 1, \ldots, I$, E^{DR}, $\mathbf{A}(I_0)$, κ, K, μ, ν_0, ϵ, and $\{\gamma_t\}$
Output: $\mathbf{A}$ and $\{\mathbf{s}^*_i\}^I_{i=1}$

1: Set $t = 0$
2: Repeat
3:   For $i = I_0, I_0 + 1, \ldots, I$
4:     Collect samples $\{(\boldsymbol{\xi}_{i,n}, \mathbf{E}^*_{i,n})\}^{N_i}_{n=1}$
5:     Obtain $\mathbf{w}^*_i$ via (19) and (20)
6:     Perform sparse coding via (34)
7:     Update $\mathbf{A}(i)$ via (33)
8:   End For
9:   Set $\mathbf{A} = \mathbf{A}(I)$
10:   For $i = 1, 2, \ldots, I$
11:     $\mathbf{s}^*_i = \arg\min_{\mathbf{s}} \|(\mathbf{Q}^{\mathrm{d}}_i)^{\frac{1}{2}} \mathbf{K}^{\mathrm{d}}_{i,i} \mathbf{w}^*_i$
          $- (\mathbf{Q}^{\mathrm{d}}_i)^{\frac{1}{2}} \mathbf{K}^{\mathrm{d}}_{i,1:I} \mathbf{A} \mathbf{s}\|^2_2 + \mu \|\mathbf{s}\|_1$
12:   End For
13:   $\nu_{t+1} = \big[\nu_t + \gamma_t \big(E^{\mathrm{DR}}$
          $- \sum_{i=1}^{I} \mathbb{E}\left[E^{\mathrm{red}}_i(\boldsymbol{\pi}(\boldsymbol{\xi}_i, \boldsymbol{\Phi}^{\mathrm{d}} \mathbf{A} \mathbf{s}^*_i))\right]\big)\big]^+$
14:   Set $t \leftarrow t + 1$
15: Until convergence

TABLE 1: Lifelong learning based building DR algorithm.

VI. NUMERICAL EXPERIMENTS

A. EXPERIMENT SETUP

The performance of the proposed DR algorithm is verified using numerical experiments. In order to conduct the experiments in a more realistic setup, real data sets are employed to the extent possible. First, an illuminance data set collected in a solar-powered house called SML System was used [33], [34]. We sampled I days' worth of data of direct sunlight illuminance levels and scaled them properly to simulate the ambient illuminance for I different rooms. The temperature levels were also taken by sampling I days' records from the data set in [35], which contains the data for 6 rooms during 31 days. Using the illuminance and temperature samples $\boldsymbol{\xi}_{i,n} = [l^{\mathrm{sun}}_{i,n}, t^{\mathrm{cur}}_{i,n}]^\top$ prepared in this way, the samples $\mathbf{E}^*_{i,n} = [E^{\mathrm{light}*}_{i,n}, E^{\mathrm{ac}*}_{i,n}]^\top$ were generated based on (1) and (2).

Here, the parameters were randomly generated as follows. For the lighting system, first the mean of the desired illuminance $\bar{l}^*_i$ for each room i was selected uniformly from the interval $[500, 700]$, i.e., $\bar{l}^*_i \sim \mathcal{U}[500, 700]$. Then, $l^*_{i,n}$ for $n = 1, 2, \ldots, N_i$ was sampled from a Gaussian distribution with mean $\bar{l}^*_i$ and variance 10^2, that is, $l^*_{i,n} \sim \mathcal{N}(\bar{l}^*_i, 10^2)$. The luminous efficacy was sampled from $\eta_i \sim \mathcal{N}(12.5, 0.2^2)$. The surface area was sampled as $S_i \sim \mathcal{U}[100, 150]$. For the AC system, the mean of the desired temperature $\bar{t}^*_i$ was set randomly from a uniform distribution $\mathcal{U}(22, 25)$, and $t^*_{i,n}$ was sampled as $t^*_{i,n} \sim \mathcal{N}(\bar{t}^*_i, 0.5^2)$. Similarly, with $\bar{A}_i \sim \mathcal{U}[1.5, 1.7]$ and $\bar{B}_i \sim \mathcal{U}[0.9, 1.1]$, the samples for $A_{i,n}$ and $B_{i,n}$ were taken as $A_{i,n} \sim \mathcal{N}(\bar{A}_i, 0.01^2)$ and $B_{i,n} \sim \mathcal{N}(\bar{B}_i, 0.01^2)$, respectively. The number of training
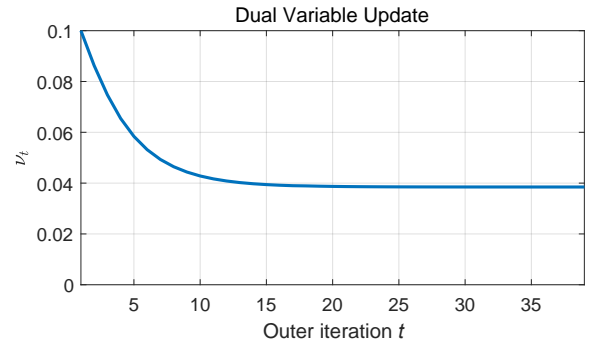
(a) Convergence of the dual variable ν_t
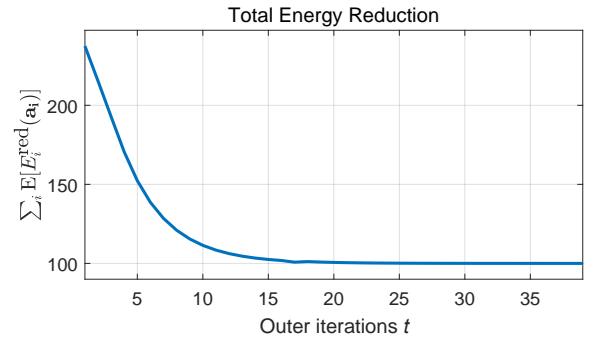
(b) Evolution of the average total energy reduction

FIGURE 3: Convergence of the algorithm.

samples was $N_i = 30$ for all i.

We experimented with $I = 20$ rooms. The DR request was set to $E^{\mathrm{DR}} = 100$. The model parameters $K = 6$, $\lambda = 10^{-4}$ and $\mu = 2 \cdot 10^{-6}$ were used. The resulting number of nonzero elements in $\mathbf{s}_i$ was around 3. The learning parameters used were $\gamma_t = 10^{-3}$ for all t, $\epsilon = 10^{-3}$, and $\nu_0 = 0.1$. A radial basis kernel function given by $\kappa(\boldsymbol{\xi}, \boldsymbol{\xi}') = \exp(-\|\boldsymbol{\xi} - \boldsymbol{\xi}'\|^2_2)$ was adopted.

B. TEST RESULTS

First, the convergence of the proposed algorithm is verified. Fig. 3(a) shows the convergence of the dual variable ν_t, and the corresponding average total reduction in the energy consumption is plotted in Fig. 3(b). The x-axis represents the outer iteration index t. Note that the requested DR amount E^{DR} is satisfied in about 20 iterations.

The convergence of a typical run of the inner SGD update loop is shown in Fig. 4, where the objective function values in (26) are plotted at different values of i. In order to ensure the convergence, here we actually used 5 epochs for the SGD updates, meaning that the samples of the I rooms were presented to the learning algorithm 5 times repetitively. Note that solving for $\{\mathbf{w}^*_i\}$ is needed only in the first epoch. It can be seen that the SGD update converges after processing around 100 rooms.

To assess the effectiveness of the proposed kernel lifelong learning-based DR algorithm, its performance is compared
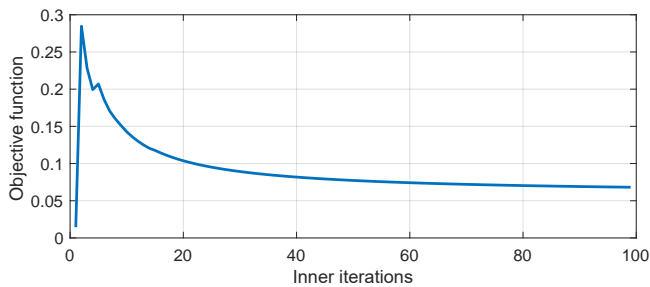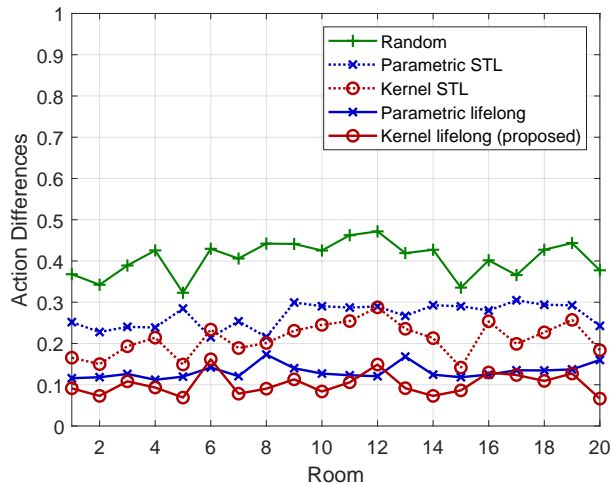
FIGURE 4: Convergence of SGD updates.

FIGURE 5: Average differences from the ideal actions.

FIGURE 6: Average absolute deviations from the ideal costs.

FIGURE 7: User discomfort levels.

with that of four other methods, namely, the random, parametric STL, kernel STL, and parametric lifelong learning approaches. In the random policy, actions are randomly chosen among the actions satisfying the DR request. In the kernel STL approach, the control policy of each room is obtained solving (19) for individual rooms without leveraging any shared structure. The parametric STL policy is the STL policy without using the kernel-based learning. It can be thought of the kernel policy but using the linear kernel function. Likewise, the parametric lifelong learning policy can be thought of as the kernel lifelong policy with linear kernels [21].

Fig. 5 depicts the difference between the action from the learned policy and the ideal action, for each room. The difference is calculated as the average Euclidean distance $\mathbb{E}[\|\mathbf{a}_i^* - \mathbf{a}_i\|_2]$ between the action vector $\mathbf{a}_i = \boldsymbol{\pi}(\boldsymbol{\xi}_i, \boldsymbol{\theta}_i^*)$ based on $\boldsymbol{\theta}_i^*$ learned from the proposed algorithm, and the ideal action vector $\mathbf{a}_i^*$, which is obtained by solving (10) numerically assuming the full knowledge of $\{\mathbf{E}_{i,n}^*\}$, which is not available to the learning algorithm. Note that, for obtaining the performance metric, we used the test samples $(\boldsymbol{\xi}_{i,n}, \mathbf{E}_{i,n}^*)$ that are separate from the samples used for training. As can be seen from the figure, the proposed algorithm achieves the lowest action difference with an average of around 0.1, whereas the kernel STL approach yields roughly double the value. This highlights the benefit of lifelong learning, which
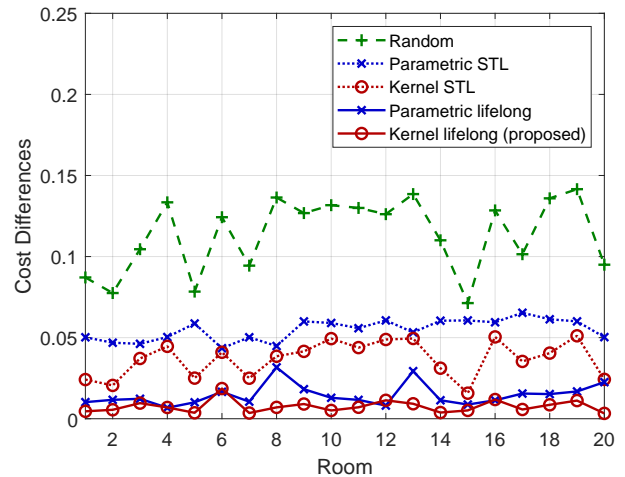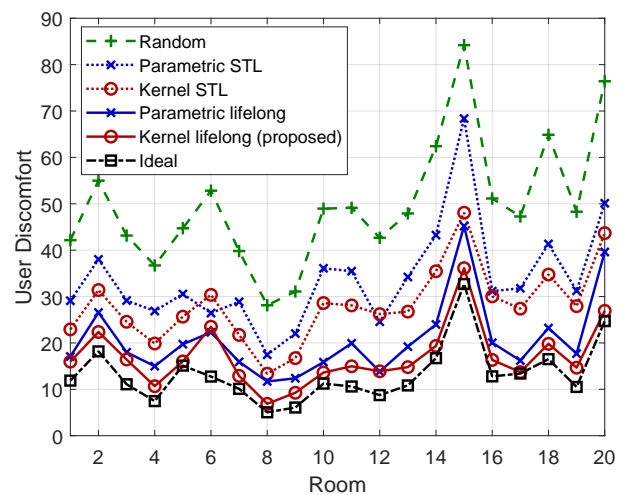
exploits the shared structure. Note also that the parametric approaches yield slightly worse performance compared to the kernel counterparts, illustrating the advantage of using nonlinear policies through kernel-based learning. Finally, the random policy is seen to yield far larger differences.

In Fig. 6, the average absolute difference of the cost function value due to the learned policy from that of the ideal actions is depicted. That is, $\mathbb{E}[|C_i(\mathbf{a}_i^*) - C_i(\mathbf{a}_i)|]$ is plotted for all i. A trend similar to what was observed in Fig. 5 emerges. That is, the proposed algorithm achieves the lowest cost differences compared to the other methods. In fact, the proposed policy is seen to achieve virtually the same cost as the ideal actions.

The user discomfort levels obtained from different approaches are compared in Fig. 7, which shows $\mathbb{E}[D_i]$ for all rooms. The minimization of the total discomfort is the prime objective of our DR formulation. It can be seen that the proposed algorithm achieves the lowest discomfort levels across all rooms among the compared five methods, and
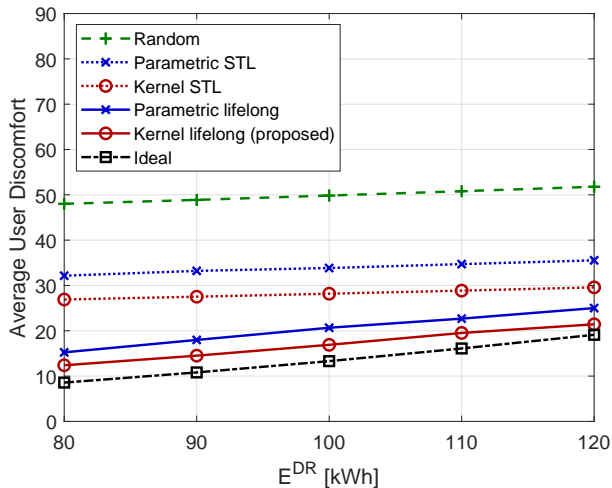
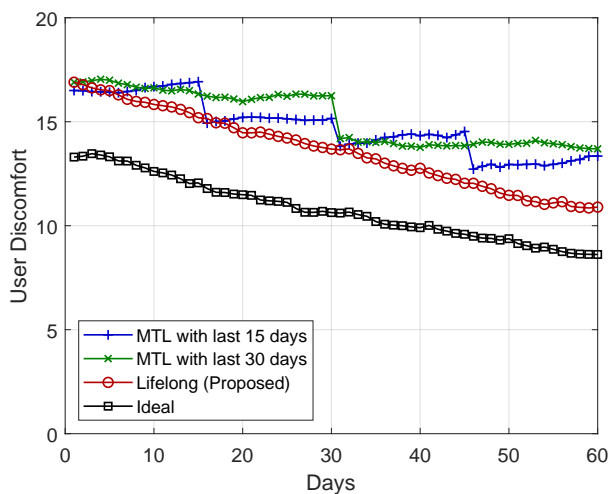FIGURE 8: Average user discomfort versus E^{DR}.

FIGURE 9: Tracking of optimal policy.

approaches quite close to the ideal discomfort levels. The discomfort levels obtained from the parametric and the kernel STL policies are much higher than those from the lifelong learning counterparts. This is because the STL approaches occasionally produce actions that reduce the energy consumptions excessively. In such cases, the actions are quite far from the actions of the proposed policy, which balances the DR constraint with the user discomfort effectively.

The average user discomfort across all rooms, i.e., $\frac{1}{I}\sum_{i}\mathbb{E}[D_{i}]$, is shown in Fig. 8 when the DR request E^{DR} is varied. It is seen that the average discomfort increases as E^{DR} increases, regardless of the policies employed. This is expected since increasing E^{DR} requires larger reduction in energy consumption, leading to higher discomfort for the occupants. Again, the proposed algorithm provides superior performance compared to the other considered methods, and produces the level closest to the discomfort level achieved by the ideal actions.

One of the advantages of the lifelong learning approach

is that the library and optimal policy can be continuously refined based on newly arrived samples. This is particularly appealing when the distributions of the random variables change over time, for instance, due to seasonal effects. On the other hand, the batch MTL solvers would need to re-compute the library and policies whenever enough new measurements are obtained, without taking advantage of the existing solutions. To verify the advantage, we performed an experiment, where the seasonal variations in the illuminance and temperature are simulated over a period of 60 days. The DR requests are assumed to arrive once per day. The proposed algorithm updates **L** whenever new samples for all rooms arrive, whereas the MTL counterpart computes the new library using the batch of samples from the last 15 or 30 days.

Fig. 9 shows the resulting user discomfort variations. It can be noticed that the proposed approach can track the variation of the ideal actions, whereas the MTL approach yields step-like curves. In fact, it can be seen that the performance gap between our method and the ideal one gradually decreases, as the library in the lifelong learning DR algorithm is refined further using the accumulated knowledge over many days. On the other hand, the MTL curves are sometimes seen to slightly worsen before the next update.

VII. CONCLUSION

A ML-based DR algorithm for controlling the adjustable loads such as the lighting and AC systems in individual rooms in buildings has been developed. The algorithm seeks a policy that can set the suitable energy levels for the loads as a function of the current ambient illuminance and temperature measurements, such that the discomfort levels of the occupants are minimized, while the DR requirement in overall energy reduction is met. The uncertainty in the environmental variables and user preferences has been tackled in a data-driven ML approach. Furthermore, an MTL framework has been adopted to exploit the structural similarities in the optimal policies across the rooms. In particular, a lifelong learning method has been derived, which can update the shared representation of the policies in an online fashion, for computational efficiency and the tracking capability of the optimal policies over time. The kernel-based learning was pursued to accommodate nonlinear policy structures as well. In order to cope with the DR constraint that couples all rooms, a dual decomposition technique was employed, which transformed the overall problem into a series of unconstrained stochastic optimization problems for individual rooms. The convergence and the performance advantage of the proposed method over benchmark policies were verified via numerical experiments designed based on semi-real data sets.

REFERENCES

[1] P. Siano, "Demand response and smart grids–a survey," Renew. Sust. Energ. Rev., vol. 30, pp. 461–478, 2014.

[2] U.S. Energy Information Administration (EIA), "International energy outlook 2016," Accessed on: Nov. 2, 2019. [Online]. Available: https://www.eia.gov/outlooks/ieo/pdf/0484(2016).pdf.

[3] G. T. Costanzo, G. Zhu, M. F. Anjos, and G. Savard, "A system architecture for autonomous demand side load management in smart buildings," IEEE Trans. Smart Grid, vol. 3, no. 4, pp. 2157–2165, 2012.

[4] S. Nan, M. Zhou, and G. Li, "Optimal residential community demand response scheduling in smart grid," Appl. Energy, vol. 210, pp. 1280–1289, 2018.

[5] I. Beil, I. Hiskens, and S. Backhaus, "Frequency regulation from commercial building HVAC demand response," Proc. IEEE, vol. 104, no. 4, pp. 745–757, 2016.

[6] H. Hao, C. D. Corbin, K. Kalsi, and R. G. Pratt, "Transactive control of commercial buildings for demand response," IEEE Trans. Power Syst., vol. 32, no. 1, pp. 774–783, 2016.

[7] S. A. Husen, A. Pandharipande, L. Tolhuizen, Y. Wang, and M. Zhao, "Lighting systems control for demand response," in Proc. IEEE PES Innov. Smart Grid Technol. (ISGT), Washington, DC, USA, 2012, pp. 1–6.

[8] D. T. Vedullapalli, R. Hadidi, and B. Schroeder, "Optimal demand response in a building by battery and HVAC scheduling using model predictive control," in Proc. IEEE/IAS Ind. and Commer. Power Syst. Tech. Conf. (I&CPS), Calgary, AB, Canada, 2019, pp. 1–6.

[9] D. Gao, Y. Sun, and Y. Lu, "A robust demand response control of commercial buildings for smart grid under load prediction uncertainty," Energy, vol. 93, pp. 275–283, 2015.

[10] L. Shen, Z. Li, and Y. Sun, "Performance evaluation of conventional demand response at building-group-level under different electricity pricings," Energ. Buildings, vol. 128, pp. 143–154, 2016.

[11] C. Ninagawa, S. Kondo, and J. Morikawa, "Prediction of aggregated power curtailment of smart grid demand response of a large number of building air-conditioners," in Proc. IEEE Int. Conf. Ind. Informat. and Comput. Syst. (CIICS), Sharjah, UAE, 2016, pp. 1–4.

[12] A. Haque, M. Pipattanasomporn, S. Rahman, R. Kothandaraman, A. Malekpour, E. A. Paaso, and S. Bahramirad, "An SVR-based building-level load forecasting method considering impact of HVAC set points," in Proc. IEEE PES Innov. Smart Grid Technol. (ISGT), Washington, DC, USA, 2019, pp. 1–5.

[13] D. Zhou, M. Balandat, and C. Tomlin, "Residential demand response targeting using machine learning with observational data," in Proc. IEEE Conf. Decis. Control (CDC), Las Vegas, NV, USA, 2016, pp. 6663–6668.

[14] D. O'Neill, M. Levorato, A. Goldsmith, and U. Mitra, "Residential demand response using reinforcement learning," in Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm), Gaithersburg, MD, USA, 2010, pp. 409–414.

[15] 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, 2018.

[16] L. Jia, Q. Zhao, and L. Tong, "Retail pricing for stochastic demand with unknown parameters: an online machine learning approach," in Proc. Annual Allerton Conf. Commun., Cont., and Comput., Monticello, IL, USA, 2013, pp. 1353–1358.

[17] C. K. Lee, H. Liu, D. Fuhs, A. Kores, and E. Waffenschmidt, "Smart lighting systems as a demand response solution for future smart grids," IEEE Trans. Emerg. Sel. Topics Power Electron, 2019.

[18] N. Mahdavi, J. H. Braslavsky, M. M. Seron, and S. R. West, "Model predictive control of distributed air-conditioning loads to compensate fluctuations in solar power," IEEE Trans. Smart Grid, vol. 8, no. 6, pp. 3055–3065, 2017.

[19] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in Proc. Adv. Neural Inf. Process. Syst., Montreal, QC, Canada, 2018, pp. 527–538.

[20] A. Kumar and H. Daumé III, "Learning task grouping and overlap in multi-task learning," in Proc. Int. Conf. Mach. Learn., Edinburgh, Scotland, 2012, pp. 1723–1730.

[21] P. Ruvolo and E. Eaton, "ELLA: An efficient lifelong learning algorithm," in Proc. Int. Conf. Mach. Learn., Atlanta, GA, USA, 2013, pp. 507–515.

[22] M.-F. Balcan, A. Blum, and S. Vempala, "Efficient representations for lifelong learning and autoencoding," in Proc. Conf. Learn. Theory, Paris, France, 2015, pp. 191–210.

[23] S.-J. Kim and R. Mowakeaa, "Kernel-based efficient lifelong learning algorithm," in Proc. IEEE Data Sci. Wksp. (DSW), Minneapolis, MN, USA, 2019, pp. 175–179.

[24] RapidTables, "How to convert lux to watts," [Online]. Available: https://www.rapidtables.com/calc/light/how-lux-to-watt.html. Accessed on: Dec. 23, 2019.

[25] J. Son, R. Hara, H. Kita, and E. Tanaka, "Operation scheduling considering demand response in a commercial building with chiller system and energy storage system," in Proc. IEEE PES Asia-Pacific Power and Energy Eng. Conf. (APPEEC), Brisbane, QLD, Australia, 2015, pp. 1–5.

[26] J. H. Yoon, R. Bladick, and A. Novoselac, "Demand response for residential buildings based on dynamic price of electricity," Energ. Buildings, vol. 80, pp. 531–541, 2014.

[27] A. Albert and R. Rajagopal, "Finding the right consumers for thermal demand-response: An experimental evaluation," IEEE Trans. Smart Grid, vol. 9, no. 2, pp. 564–572, 2016.

[28] D. P. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," IEEE J. Sel. Areas Commun., vol. 24, no. 8, pp. 1439–1451, 2006.

[29] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in Proc. Int. Conf. Computat. Learn. Theory, Amsterdam, Netherlands, 2001, pp. 416–426.

[30] S.-J. Kim, "Online kernel dictionary learning," in Proc. IEEE Global Conf. Signal and Inf. Process. (GlobalSIP), Orlando, FL, USA, 2015, pp. 103–107.

[31] C. Wang, L. Zhang, L. Xie, and J. Yuan, "Kernel cross-correlator," in Proc. AAAI Conf. Artif. Intell., New Orleans, LA, USA, 2018.

[32] J. Mairal, "SPArse Modeling Software (SPAMS)," Accessed on: Jan. 22, 2020. [Online]. Available: http://spams-devel.gforge.inria.fr/.

[33] F. Zamora-Martinez, P. Romeu, P. Botella, and J. Pardo, "On-line learning of indoor temperature forecasting models towards energy efficiency," Energ. Buildings, vol. 83, pp. 162–172, 2014.

[34] F. Zamora-Martinez, P. Romeu, and J. Pardo, "UCI machine learning repository," Accessed on: Oct. 14, 2019. [Online]. Available: http://archive.ics.uci.edu/ml/index.php, 2014.

[35] J. M. White, "Room temperature datasets (under MIT license)," Accessed on: Aug. 12, 2019. [Online]. Available: https://github.com/johnmyleswhite.

SUNYONG KIM received the B.S. degree from the School of Computer Science and Electrical Engineering, Handong Global University, Pohang, South Korea, in 2014. He is currently pursuing the Ph.D. degree at the School of Electrical Engineering and Computer Science (EECS), Gwangju Institute of Science and Technology (GIST), Gwangju, South Korea and was a Visiting Scholar in the Department of Computer Science and Electrical Engineering at University of Maryland, Baltimore County, MD, USA, in 2019. His research interests include machine learning for smart grids, and design, optimization, and analysis for wireless networks.

RAMI MOWAKEAA received the B.S. and M.S. degrees in computer engineering from the University of Maryland, Baltimore County in 2004 and 2010 respectively. After nearly a decade of designing electronics for 3 Phoenix, Inc. (now part of Ultra Electronics), he is currently pursuing the Ph.D. degree in electrical engineering at University of Maryland, Baltimore County, MD, USA. His research interests include machine learning, statistical signal processing, optimization and related fields.

SEUNG-JUN KIM (Senior Member, IEEE) received the B.S. and M.S. degrees from Seoul National University, Seoul, South Korea, in 1996 and 1998, respectively, and the Ph.D. degree from the University of California at Santa Barbara in 2005, all in electrical engineering. From 2005 to 2008, he was a Research Staff Member with NEC Laboratories America, Princeton, NJ, USA. He was with the Digital Technology Center and the Department of Electrical and Computer Engineering, University of Minnesota, from 2008 to 2014, where his final title was Research Associate Professor. Since 2014, he has been an Assistant Professor with the Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, MD, USA. His research interests include statistical signal processing, optimization, and machine learning, with applications to wireless communication and networking, future power systems, and big data analytics. He is serving as a Senior Area Editor for the IEEE Signal Processing Letters.

HYUK LIM (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from the School of Electrical Engineering and Computer Science, Seoul National University, Seoul, South Korea, in 1996, 1998, and 2003, respectively. From 2003 to 2006, he was a postdoctoral research associate in the Department of Computer Science, University of Illinois at Urbana–Champaign, Champaign, IL, USA. He is currently a full professor in the AI Graduate School and the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology (GIST), Gwangju, Republic of Korea. His research interests include network system design, optimization, and performance evaluation.

• • •