

Article

Privacy-Preserving Energy Management of a Shared Energy Storage System for Smart Buildings: A Federated Deep Reinforcement Learning Approach

Sangyoon Lee ¹, Le Xie ² and Dae-Hyun Choi ^{1,*}

¹ School of Electrical and Electronics Engineering, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul 156-756, Korea; sangyoon1207@naver.com

² Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA; le.xie@tamu.edu

* Correspondence: dhchoi@cau.ac.kr; Tel.: +82-2-820-5101

Abstract: This paper proposes a privacy-preserving energy management of a shared energy storage system (SESS) for multiple smart buildings using federated reinforcement learning (FRL). To preserve the privacy of energy scheduling of buildings connected to the SESS, we present a distributed deep reinforcement learning (DRL) framework using the FRL method, which consists of a global server (GS) and local building energy management systems (LBEMSs). In the framework, the LBEMS DRL agents share only a randomly selected part of their trained neural network for energy consumption models with the GS without consumer's energy consumption data. Using the shared models, the GS executes two processes: (i) construction and broadcast of a global model of energy consumption to the LBEMS agents for retraining their local models and (ii) training of the SESS DRL agent's energy charging and discharging from and to the utility and buildings. Simulation studies are conducted using one SESS and three smart buildings with solar photovoltaic systems. The results demonstrate that the proposed approach can schedule the charging and discharging of the SESS and an optimal energy consumption of heating, ventilation, and air conditioning systems in smart buildings under heterogeneous building environments while preserving the privacy of buildings' energy consumption.

Keywords: building energy management system; shared energy storage system; federated reinforcement learning; deep reinforcement learning; smart buildings



Citation: Lee, S.; Xie, L.; Choi, D.-H. Privacy-Preserving Energy Management of a Shared Energy Storage System for Smart Buildings: A Federated Deep Reinforcement Learning Approach. *Sensors* **2021**, *21*, 4898. <https://doi.org/10.3390/s21144898>

Academic Editor: Michele Magno

Received: 15 June 2021

Accepted: 15 July 2021

Published: 19 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A shared energy storage system (SESS) is a promising technology to efficiently manage the energy consumption in residential and commercial sectors. Compared to individual ESSs, installment of the SESSs reduces the installation costs for individual consumers and minimizes the operation costs of ESS [1]. Recently, smart homes and buildings are equipped with advanced smart grid technologies, including distributed energy resources (DERs) such as a solar photovoltaic (PV) system and Internet of Things-based home/building appliances along with advanced energy sensors, namely smart meters, that monitor the real-time energy usage of consumer through advanced metering infrastructure network. The advancement of these smart grid technologies combined with the SESS suggests the need for more intelligent energy management systems that save electricity cost for consumers while maintaining their comfort levels and preferences.

Conventional algorithms for the energy management of buildings connected to the SESS have been developed using various model-based optimization methods. However, the model-based approach is formulated with approximated constraints for the operation characteristics of appliances so that it may yield an inaccurate and diverging solution. To address the limitation, this study presents a machine learning (ML) method for developing the energy management framework for a SESS with multiple smart commercial buildings.

Building energy management systems (BEMSs) are the core solutions for BEMS operators to monitor the real-time building energy usage through smart meters and accordingly, manage the optimal energy consumption of building appliances (e.g., heating, ventilation, and air conditioning (HVAC) systems and lights), thereby reducing the electricity cost while ensuring the comfort and satisfaction of the building's occupants. Approximately 45% of the total electricity consumption in the building comes from the HVAC system; therefore, the primary goal of the BEMS is to calculate the optimal energy consumption schedule of the HVAC system and consequently, reduce the electricity cost of the building.

There have been various optimization methods (e.g., robust optimization, stochastic programming, and model predictive control (MPC)) for the energy management of smart residential and commercial buildings while considering their uncertain and dynamic operation characteristics. For the robust optimization method, the range of the values of the uncertain parameters is determined by their continuous set. By contrast, for the stochastic programming method, the uncertain parameters are characterized by a number of discrete probabilistic scenarios. MPC is an optimization control approach that predicts the dynamic system operation characteristics over a predefined horizon time period for minimizing the cost function of the system via optimal control decisions. A robust optimization-based BEMS for a smart building with a wind turbine was developed in [2] where the total cost of heat/power co-generation and the total cost of emissions are minimized while considering the uncertainty of the wind speed and electrical and thermal loads. In [3], a robust optimization algorithm for the energy management of smart home with PV system, ESS, and electric vehicle (EV) was presented. This algorithm consisted of two stages where the exact energy consumption schedule for smart home is determined using a deterministic optimization method at the first stage and a heuristic algorithm is executed to reflect the uncertainty of the power production and energy demand for extended simulation time horizon. In [4], a stochastic programming was employed to construct home energy management system where the uncertainty of DERs and the EV availability for charging/discharging are incorporated in the scheduling of home energy consumption. In [5], a stochastic optimization algorithm was presented to minimize the energy cost and emissions of residential microgrids. In this algorithm, a scenario-based approach was employed to reflect the uncertainties of electrical market price, electrical and thermal load, and PV generation output. In [6], a BEMS method using the MPC method was presented where the energy consumption of commercial building is reduced by optimizing the operation of HVAC system in 24 h prediction horizon. In [7], a stochastic MPC strategy for the energy management of a smart home with DERs was formulated and executed in a real-time operation mode where the prediction horizon is set to 24 h.

Recently, considerable effort has been devoted to developing the optimization-based BEMS algorithms considering the operation of HVAC system. HVAC control methods for minimizing the electricity cost of commercial and university buildings were presented in [8,9]. In [10], an HVAC management algorithm that considers random occupancy in buildings was proposed. The BEMS method that involves multiple environmental variables of buildings, such as indoor temperature and lighting, was developed under a time-of-use (TOU) price-based demand response program in [11]. Currently, smart communities with various DERs are emerging; therefore, the SESS control algorithms have been developed as a core technology for efficient and economic energy management for multiple residential and commercial buildings. These algorithms range from the scheduling of the SESS for commercial building energy management under demand side management [12], the development of a novel cooperative algorithm between the SESS and residential buildings [13], credit-based operation of the SESS among households using Lyapunov optimization method [14], the scheduling of the SESS operation with a shared PV system based on welfare optimization and a game theoretical method according to consumers' preferences [15], and the impact analysis of the SESS operation subject to data privacy [16].

Note that the aforementioned optimization-based BEMS methods with the SESS highly rely on abstract optimization models and have a limited capability in larger system models with uncertain input data, thereby leading to a significantly high computational complexity and in some cases, even yielding a diverging solution. To handle these limitations from the model-based BEMS with the SESS, deep reinforcement learning (DRL) has recently attracted attention for being a model-free BEMS approach. A DRL algorithm using deep Q-network and deep policy gradient was adopted to classify the patterns of building energy usage and optimize the energy consumption schedule of the building in [17]. In [18], a joint DRL approach for datacenter and HVAC load control was proposed to reduce the total energy cost within the range of the preferred room temperature while satisfying datacenter workload deadline constraints. In [19], an imitation learning approach was employed to the DRL-based HVAC control algorithm to identify the characteristic of HVAC system controller in the building. A multi-agent DRL framework was developed in [20] where the energy cost of HVAC system in a multi-zone commercial building is minimized while considering random zone occupancy, consumer' thermal comfort, and air quality comfort. A concise review of DRL-based smart building energy management methods is summarized in [21]. More recently, various DRL-based BEMS methods have been developed, including the scheduling of the ESS and HVAC in residential buildings based on a deep deterministic policy gradient (DDPG) method [22], the management of utility-scale interruptible loads in a dueling deep Q network [23], actor-critic-based demand response management for industrial facilities [24], and the control of the state of charge of a group of multiple ESSs using the DDPG method [25].

However, in the aforementioned centralized DRL-based frameworks, the consumer's private data aggregated at a global server (GS) can be exposed to attackers who may exploit these data for finding the energy consumption behavior of consumers. There has been many studies to present energy management methods while considering the privacy of consumers in the residential and commercial buildings [26–29]. In [26], a Kullback–Leiber divergence was adopted to assess the privacy leakage in smart meters installed at the residential buildings and the energy management strategy was formulated with a weighted sum of the consumer privacy and energy cost. A service architecture of home energy management system for preserving the privacy of smart meter data was designed in [27] where the abuse of smart meter data by an adversary can be readily identified. In [28], a new privacy-preserving method for households with the ESSs was presented where an energy usage manipulation by an adversary can be prevented by using the ESS opportunistically. In [29], a privacy-enhanced architecture for the commercial buildings was proposed where the occupancy data are distorted to hide individual occupant location information while ensuring the performance of HVAC. Recently, federated learning (FL) [30] was proposed to preserve the consumer privacy. It is an ML method employed for training local agents and building their optimal models in a distributed manner. In FL, local and global neural network models between the local agents and the GS are exchanged without their private local data being shared to preserve consumer data privacy. Furthermore, for applying this FL concept to various engineering control problems, federated reinforcement learning (FRL) [31] was proposed wherein the optimal policy for individual agent was calculated as long as ensuring that the data were not shared among agents during the training process. The FRL method was employed for developing the navigation for cloud-based robotic systems [32], rapid personalization process of agents [33], and as a defense strategy for jamming attacks in the flying ad hoc network for unmanned aerial vehicles [34]. FRL has been adopted to resolve distributed control problems in a variety of engineering fields. Our recent study [35] developed a privacy-preserving FRL framework to manage the energy consumption of multiple smart homes with DERs. In this study, the DRL agents for home appliances such as air conditioners, washing machines and residential ESS iteratively interact with the GS to build their optimal energy consumption model in multiple homes.

However, to the best of the authors' knowledge, no study has been performed to construct a privacy-preserving DRL-based SESS model with an FRL-based BEMS model to manage the energy scheduling of the SESS and energy consumption of the HVACs in buildings in a distributed manner. In comparison with our previous study [35] where no SESS was considered in the FRL-based home energy management system, the proposed approach has applied the FRL method to multiple commercial buildings integrated with the SESS. In the previous study, the role of the GS was limited to only aggregating the local agents' models and broadcasting it back to the local agents without conducting the DRL training procedure. On the other hand, in the proposed approach, the GS is designed to train the charging and discharging schedule of the SESS through the DRL process as well as update the local models. Furthermore, different from the previous study, the proposed approach can further enhance the privacy-preserving of local agents in the FRL framework by adopting a selective parameter method [36] in which all local agents choose a randomly chosen part of their local model and deliver it to the GS. Lastly, a novel method for maintaining stable charging/discharging scheduling of the SESS was presented while preserving the private energy data of buildings.

Figure 1 shows a system model of the proposed FRL-based BEMS with a SESS. In this model, we considered three entities: (i) an electric utility, (ii) a local BEMS (LBEMS) operator, and (iii) a non-profit building load aggregator (BLA) with the GS. The GS interacts with the LBEMSs to build the optimal energy consumption model of the HVACs and the optimal charging and discharging model of the SESS by performing the following two tasks: (i) the FRL process through an exchange of local neural network models and global neural network model for the energy consumption of the HVAC between the GS and LBEMSs, and (ii) SESS energy charging from the grid (i.e., energy purchase from the BLA) or energy discharging to the LBEMSs by the GS.

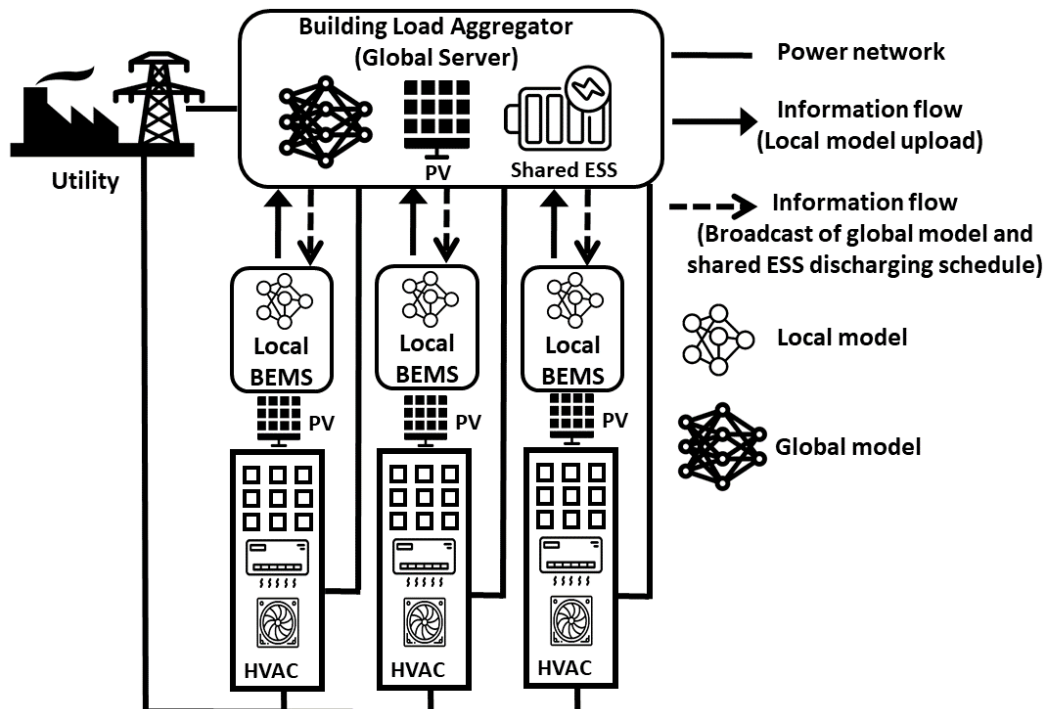


Figure 1. System architecture of a federated reinforcement learning (FRL)-based building energy management system (BEMS) with a shared energy storage system (SESS).

The main contributions of this paper are provided as follows:

- We present a distributed FRL architecture in which the energy consumption of smart buildings and the energy charging and discharging of the SESS are optimally scheduled within the heterogeneous environments of the buildings while preserving the privacy of energy usage information of individual buildings.
- We develop a robust privacy-preserving FRL-based BEMS algorithm against building privacy leakage in the hierarchically distributed architecture. During the FRL process, the HVAC DRL agent improves the energy consumption model of the LBEMS through an iterative interaction with the GS and preserves the privacy of energy usage data using a selective parameter sharing method. Subsequently, the SESS DRL agent trains the optimal energy charging and discharging model of the SESS by using the LBEMS agent's neural network model without sharing the relevant energy consumption data to preserve the privacy of the buildings' energy consumption.

The rest of the paper is organized as follows. Section 2 introduces the DRL and FRL methods in the Markov Decision Process (MDP). The mathematical formulation and methodology for the proposed approach are developed in Section 3. Section 4 provides the simulation results for the proposed algorithm with the SESS under heterogeneous building environments. Finally, discussions and concluding remarks are drawn in Sections 5 and 6, respectively.

2. Background of Reinforcement Learning

2.1. Markov Decision Process (MDP)

An MDP is defined as a 5-tuple $(\mathcal{S}, \mathcal{A}, P, R, T)$ in which \mathcal{S} is a finite set of all valid states and \mathcal{A} is a finite set of all valid actions. The function $P : \mathcal{S} \times \mathcal{A} \rightarrow P(\mathcal{S})$ represents a Markovian transition model that is the transition probability function, where $P(s_{t+1} | s_t, a_t)$ denotes the probability of transitioning from state s_t at time t into state s_{t+1} at time $t + 1$ after an agent selects an action a_t in state s_t at time t . $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ is the reward function such that a reward $R_{t+1} = R(s_t, a_t, s_{t+1})$ is obtained from transitioning from state s_t into state s_{t+1} after taking an action a_t . To solve an MDP problem having a finite time horizon T equals to determine a policy $\pi_\theta \in \Pi$ in which π is parameterized with θ that are the weights and biases of a neural network. The policy π_θ finds the action $a \in \mathcal{A}$ that should be conducted in any state s in order to maximize the discounted cumulative rewards that an agent receives from the state-action transition procedure during a finite time horizon T . In a Q-learning method, the policy π_θ relies on the value of $Q(s_t, a_t)$, namely the Q-value, which estimates the propriety of selected action a_t in given state s_t . This Q-value is estimated for the sum of the discounted cumulative future rewards and can be expressed as $Q(s_t, a_t) = E[\sum_{i=0}^t \gamma^i R_{t+1+i} | s = s_t, a = a_t]$. $\gamma \in [0, 1]$ is a discounting factor, and it represents the relative importance of present and upcoming rewards. The Q-learning method aims to find the optimal policy π_θ^* that maximizes the Q-value (i.e., $\pi_\theta^* = \arg \max_{\pi_\theta} Q(s_t, a_t)$) when the agent executes the action according to policy π_θ .

2.2. Deep Reinforcement Learning

An actor-critic algorithm is a contemporary DRL method that uses both policy-based DRL and value-based DRL [37]. In this DRL method, the agent applies two different types of networks to evaluate the goodness of the selected action in given state. The policy network which selects the action and returns its probability is actor network, and the network that estimates the value of the agent's given state is the critic network. The policy gradient method is suitable for handling the problem with a continuous action space; however, this method may cause a poor convergence performance. An additional critic network in the actor-critic algorithm can improve this convergence issue from the policy gradient method. The objective of the actor-critic algorithm is to proceed its learning to minimize the sum of two different loss functions, which correspond to $L_t^{\text{actor}}(\theta_a)$ and $L_t^{\text{critic}}(\theta_c)$ for the actor network and critic network, respectively:

$$\min L_t^{\text{actor}}(\theta_a) + L_t^{\text{critic}}(\theta_c) \quad (1)$$

s. t.

$$L_t^{\text{actor}}(\theta_a) = -\log p_{\pi_{\theta_a}}(a_t|s_t)Q_{\pi_{\theta_c}}(s_t, a_t) \quad (2)$$

$$L_t^{\text{critic}}(\theta_c) = (TD_t - V_{\pi_{\theta_c}}(s_t))^2. \quad (3)$$

In Equation (1), θ_a and θ_c represent the parameters of the actor network and the critic network, respectively. In Equation (2), $p_{\pi_{\theta_a}}(a_t|s_t)$ implies the probability of selecting action a_t given state s_t at time t under policy π_{θ_a} . Moreover, $Q_{\pi_{\theta_c}}(s_t, a_t)$ represents the Q-value of the agent's action a_t at state s_t under policy π_{θ_c} . In Equation (3), the target value of the critic network is denoted by $TD_t = R_{t+1} + \gamma V_{\pi_{\theta_c}}(s_{t+1})$. This target value can be calculated using a temporal difference (TD) method [38] for updating the critic network. $V_{\pi_{\theta_c}}(s_t)$ represents the value of state s_t under policy π_{θ_c} . $V_{\pi_{\theta_c}}(s_t)$ is written as the expectation of the discounted cumulative future reward that the agent will obtain in state s_t as follows: $V_{\pi_{\theta_c}}(s_t) = E[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | s = s_t]$.

2.3. Federated Reinforcement Learning

As mentioned earlier, FRL is one of the prospective ML methods to train the agents for different local devices in a distributed manner without allowing all agents to share their data for preserving their privacy. The learning process for FRL consists of the following two steps: (i) training for local models and (ii) assembling and updating local models with a newly generated global model. Assume that $\mathcal{N} := \{1, 2, \dots, N\}$ is a set of N local agents. Each agent n constructs its own optimal neural network model ω_n through its training process using its own dataset $\mathcal{D}_{n \in N}$. After finishing the training procedure of each local agent, each agent n transmits its model ω_n to a GS. Subsequently, the GS aggregates all local models to its batch $\varphi = [\omega_1, \omega_2, \dots, \omega_N]$ and builds the global model ω_G using the batch by $\omega_G = f(\varphi)$. Further, the global model ω_G is broadcasted to all connected local agents and their models are updated by ω_G : $\omega_G = \omega_1 = \omega_2 \dots = \omega_N$. Finally, each agent restarts its training using an identical ω_G along with dataset $\mathcal{D}_{n \in N}$ during the iterative communication round. This iterative interaction between all local agents and the GS continues until all local agents find their optimal model. In a centralized model for existing DRL methods, an agent in the GS must gather the data of all local devices at a single location to construct the neural network models for all local devices. This centralized approach may result in data privacy leakage from local devices. In addition, large volume of data collected at the GS may increase the training time of the agent significantly. However, in the FRL approach, no local data sharing is required in the GS. Consequently, FRL can successfully preserve the privacy of local data for agents. In addition, since FRL trains the agents for different local devices in a distributed manner, it can reduce the training time of the agents significantly. By contrast, a distributed multi-agent model without the GS may reduce the training time because each agent trains only its model by using its own data. However, in this model, some agent may not have enough data, and it results in an overfitting problem (i.e., the trained local model becomes biased, thereby leading to inaccurate policy). Note that during the FRL process the local models of devices are periodically updated by the global model that is constructed by the GS. The periodical local model update by the global model can improve the local models, thereby preventing the overfitting issue of local models.

3. Energy Management of a Shared ESS for Smart Buildings Using FRL

3.1. System Configuration

We assume a situation in which the LBEMS manages the economic operation of HVACs, a major appliance of smart buildings, with a PV system. Multiple smart buildings are connected to a single SESS to reduce the total electricity cost of purchasing conventional energy from the grid. The charging and discharging operations of the SESS are controlled by the GS. In this study, we present an FRL-based BEMS that consists of a single GS and

N LBEMSs, which is shown in Figure 2. Through the DRL process, under a TOU tariff, the HVAC agent in the LBEMS and the SESS agent in the GS conduct 24-h operation scheduling of their appliances with a 1-h scheduling resolution. As shown in Figure 2, the training process for the proposed approach includes the following two steps:

- Step (1) FRL for HVAC energy consumption scheduling: each HVAC agent in the LBEMS trains its own model to schedule the energy consumption of HVAC using the actor–critic method with its data. The randomly selected part of trained local models (i.e., the weights ω_n of its local neural network for LBEMS n) are periodically transmitted to the GS. Subsequently, the GS aggregates and updates the global model (i.e., the weights ω_G of the global neural network) using the federated stochastic gradient descent (FedSGD) algorithm [30] that averages the local models ($\omega_G = \frac{1}{N} \sum_{n=1}^N \omega_n$). The updated global model is distributed to all LBEMSs where all HVAC agents update their own models based on the global model. The updated local models and global model are exchanged iteratively until a predetermined stopping criterion is satisfied.
- Step (2) SESS charging/discharging: the optimal HVAC energy consumption models calculated from Step (1) along with the fixed loads in the building are fed back into the GS where the SESS agent trains the model for charging and discharging energy from and to the utility and the LBEMSs using the actor–critic method. The trained discharging schedules are transmitted to the LBEMSs, and these schedules are added to the HVAC energy consumption schedules that are calculated by the HVAC agents in Step (1).

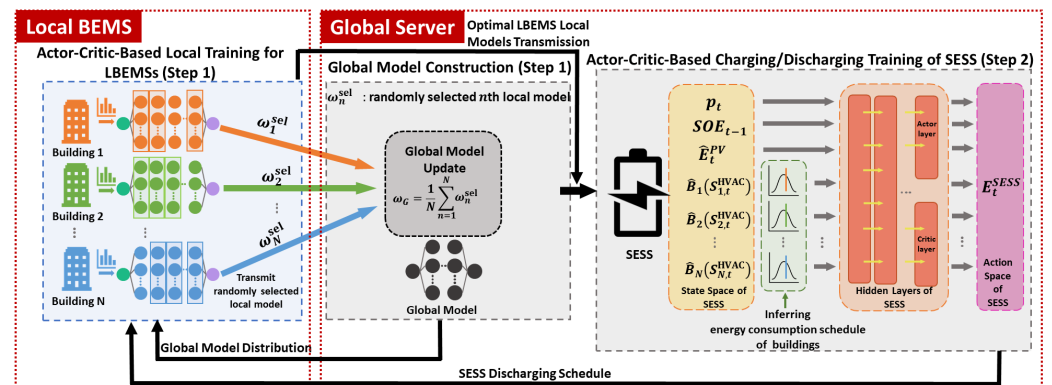


Figure 2. Training process for the proposed FRL-based energy management of smart buildings connected to a SESS.

Note that the GS could be hacked by potential adversaries, thereby leading to a more serious privacy threat to all LBEMSs. To resolve this, we adopted a selective parameter sharing model for the FRL process in Step (1). The primary feature of the selective parameter sharing method is that each LBEMS n chooses and delivers a part of its local neural network model ($\omega_n^{sel} \subset \omega_n$) to the GS. This method can prevent attackers from obtaining crucial information such as energy consumption of HVAC from the shared global neural network model. The mathematical formulation and the methodology for the proposed BEMS approach are described in the two subsections: the definition of state/action spaces and the reward functions for the DRL HVAC agent in the LBEMS and the DRL SESS agent in the GS in Section 3.2 and the proposed FRL-based energy management of the SESS with smart buildings along with the DRL method in Section 3.3.

3.2. System Description for HVAC and SESS Agents

3.2.1. State Space

$\forall t = 1, \dots, 24$, the state spaces for the HVAC agent in the LBEMS n and the SESS agent in the GS, respectively, are defined as follows:

$$\mathcal{S}_{n,t}^{\text{HVAC}} = \{p_t, \hat{T}_t^{\text{out}}, T_{n,t-1}^{\text{in}}, TC_{n,t}, \hat{E}_{n,t}^{\text{PV}}\} \quad (4)$$

$$\mathcal{S}_t^{\text{SESS}} = \{p_t, SOE_{t-1}, \hat{E}_t^{\text{PV}}, \hat{\mathbf{B}}(\mathcal{S}^{\text{HVAC}})\}. \quad (5)$$

In Equation (4), $\mathcal{S}_{n,t}^{\text{HVAC}}$ represents the state space of the HVAC agent for the LBEMS n at time t . The state p_t is the price under the TOU rates at time t . \hat{T}_t^{out} is the predicted outdoor temperature at time t . $T_{n,t-1}^{\text{in}}$ is the building n 's indoor temperature at time $t - 1$. $TC_{n,t} = \frac{OCC_{n,t}}{Cap_n}$ is the thermal capacity of the building n at time t and it provides the population density in the building n where $OCC_{n,t}$ and Cap_n represent the occupancy and thermal occupancy capacity of building n , respectively. $\hat{E}_{n,t}^{\text{PV}}$ is predicted PV generation output of building n at time t . $T_{n,t}^{\text{in}}$ is computed in terms of $T_{n,t-1}^{\text{in}}$, \hat{T}_t^{out} , $TC_{n,t}$, the energy consumption of HVAC in building n at time t ($E_{n,t}^{\text{HVAC}}$), generated PV output for energy usage directly at time t ($\hat{E}_{n,t}^{\text{PV}}$), and the environmental parameters ($\alpha_n, \beta_n, \gamma_n$) that characterize the indoor thermal and occupancy condition using the following equation: $T_{n,t}^{\text{in}} = T_{n,t-1}^{\text{in}} + \alpha_n(\hat{T}_{t-1}^{\text{out}} - T_{n,t-1}^{\text{in}}) + \beta_n(E_{n,t}^{\text{HVAC}} + \hat{E}_{n,t}^{\text{PV}}) + \gamma_n TC_{n,t}$. α_n , β_n , and γ_n are the parameters that identify the impact of temperature, energy consumption for the HVAC, and the building's thermal occupancy density, respectively, on the indoor temperature of the building. We consider a situation in which the energy generated from the PV system is preferentially used for HVAC energy consumption in the building. In Equation (5), $\mathcal{S}_t^{\text{SESS}}$ represents the state space of the SESS agent in the GS at time t . The state SOE_{t-1} is the state of energy (SOE) of the SESS at time $t - 1$. The state \hat{E}_t^{PV} is the predicted PV generation output of SESS at time t . $\hat{\mathbf{B}}(\mathcal{S}_t^{\text{HVAC}})$ is the $N \times 1$ vector for the inferred energy data from fully trained neural network model for LBEMS energy usage including the fixed loads in terms of the state $\mathcal{S}_t^{\text{HVAC}}$, where $\hat{\mathbf{B}}(\mathcal{S}_t^{\text{HVAC}}) = [\hat{\mathbf{B}}_1(\mathcal{S}_{1,t}^{\text{HVAC}}), \dots, \hat{\mathbf{B}}_N(\mathcal{S}_{N,t}^{\text{HVAC}})]$ and $\hat{\mathbf{B}}_n(\cdot)$ is the fully trained neural network of LBEMS n during the FRL process. For preserving the privacy of LBEMSs's energy consumption data, the SESS does not receive any exact energy consumption data from the LBEMSs. The SESS receives fully trained neural network from each LBEMS and infers the energy consumption schedules of all LBEMSs using $\hat{\mathbf{B}}(\cdot)$. The utilization of the fully trained LBEMS model can preserve the data privacy of buildings without explicitly sharing building' energy consumption pattern with the GS.

3.2.2. Action Space

In this study, the action corresponds to the energy consumption schedule of the HVAC and the energy charging and discharging schedule of the SESS. The action is a key component for the HVAC and SESS operators because the energy scheduling of the HVAC and SESS directly affects the energy cost and comfort level of the buildings. Therefore, the optimal action for the HVAC and SESS should be determined by their DRL agents according to the environment and the state described in Equations (4) and (5). The action spaces of the HVAC and SESS are written as

$$\mathcal{A}_{n,t}^{\text{HVAC}} = \{E_{n,t}^{\text{HVAC}}\}, \mathcal{A}_t^{\text{SESS}} = \{E_t^{\text{SESS}}\} \quad (6)$$

where $E_{n,t}^{\text{HVAC}}$ and E_t^{SESS} represent the energy consumptions of the HVAC for the LBEMS n and the charging/discharging energy of the SESS considering their own PV generation at time t , respectively. Given the predicted PV generation output, the operational dynamics of the SOE for the SESS at time t is expressed as follows: $SOE_t = SOE_{t-1} + E_t^{\text{SESS}} + \hat{E}_t^{\text{PV}}$. The HVAC agent and the SESS agent select continuous energy consumption and charging and discharging actions using normal distribution $N(\mu, \sigma^2)$ with the values of mean μ and

variance σ^2 received from the actor network. Furthermore, to prevent the adversary from inferring the energy consumption data of buildings, the mean μ and variance σ^2 obtained from the actor network are recalculated by two key functions, key_{mean} and key_{var} , which in turn yields new mean μ_{new} and variance σ_{new}^2 for action distribution as shown in Figure 3. Only the HVAC and SESS agents know these key functions; therefore, the adversaries can neither infer the exact energy consumption, nor the charging and discharging schedule from the neural network. In our proposed method, the SESS agent infers the energy consumption data of each building using neural network model and key functions of each LBEMS. Before the SESS agent selects the action, it observes the state that contains inferred LBEMSs' energy consumption data. Then, the SESS agent selects the action (e.g., energy charging or discharging) based on the given state. The SESS agent also chooses its action based on normal distribution with the outputs received from its actor network.

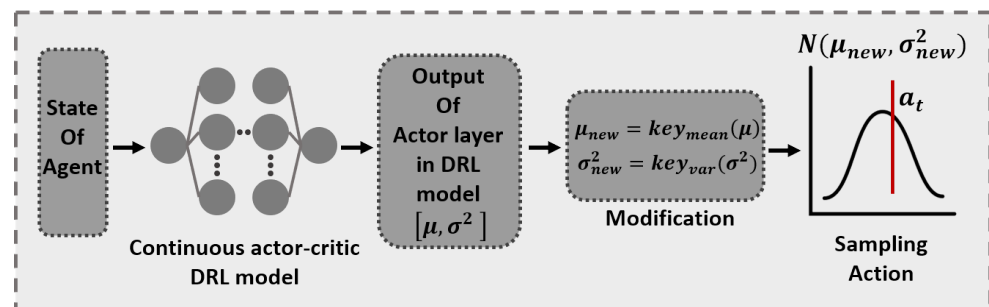


Figure 3. Sampling process of action for the proposed deep reinforcement learning (DRL) agent.

3.2.3. Reward Function

The reward functions for both HVAC and SESS agents are written as the sum of the negative energy cost and negative penalty cost, which are related to the building's thermal comfort preference and the SESS operation. To begin with, the reward function for the HVAC agent n is expressed as

$$R_{n,t}^{HVAC} = \begin{cases} -[p_t E_{n,t}^{HVAC} + \bar{\epsilon}_n (T_n^{\min} - T_{n,t}^{\text{in}})], & \text{if } T_{n,t}^{\text{in}} < T_n^{\min} \\ -[p_t E_{n,t}^{HVAC} + \underline{\epsilon}_n (T_{n,t}^{\text{in}} - T_n^{\max})], & \text{if } T_{n,t}^{\text{in}} > T_n^{\max} \\ -p_t E_{n,t}^{HVAC}, & \text{otherwise,} \end{cases} \quad (7)$$

where $\bar{\epsilon}_n$ and $\underline{\epsilon}_n$ are the penalties for thermal discomfort in the building n . The discomfort cost is formulated as the deviation in the building n 's preferred temperature $T_{n,t}^{\text{in}}$ from T_n^{\min} and T_n^{\max} . This cost is regarded as a reward with a negative sign when $T_{n,t}^{\text{in}}$ is out of the range of $[T_n^{\min}, T_n^{\max}]$. The reward function for the SESS agent is formulated as

$$R_t^{\text{SESS}} = \begin{cases} -[p_t E_t^{\text{SESS}} + \bar{\tau} (SOE_t - SOE^{\max})], & \text{if } SOE_t > SOE^{\max} \\ -[p_t E_t^{\text{SESS}} + \underline{\tau} (SOE^{\min} - SOE_t)], & \text{if } SOE_t < SOE^{\min} \\ -p_t E_t^{\text{SESS}}, & \text{otherwise.} \end{cases} \quad (8)$$

Here, $\bar{\tau}$ and $\underline{\tau}$ are denoted by the penalties for the overcharging and undercharging of SESS, respectively. When the SOE is smaller than SOE^{\min} (i.e., undercharging) or the SOE is greater than SOE^{\max} (i.e., overcharging), energy underutilization of the SESS happens.

3.3. Proposed Privacy-Preserving Energy Management of the SESS with Smart Buildings

We describe Algorithm 1 for privacy-preserving energy management of the SESS with FRL-based distributed BEMS. The HVAC agents in the LBEMSs and the SESS agent in the GS employ the DRL algorithm based on actor-critic method that is illustrated

in Section 2.2. In this algorithm, the LBEMS and GS first interact iteratively and train the energy consumption model for the HVAC agents in the LBEMSs during the FRL process until all HVAC agents find their own optimal consumption model (line 5~20 in Algorithm 1). All HVAC agents obtain their optimal model using DRL method, and these agents are assumed to start their learning procedure synchronously. The optimal models trained by the HVAC agents during the FRL process are then delivered to the GS where the charging and discharging schedule of the SESS agent is trained using DRL method (line 23~31 in Algorithm 1). Finally, the trained discharging schedule of the SESS agent is transmitted and added to the energy consumption schedule of the HVAC agents in the LBEMSs. The overall learning procedure followed by the proposed energy management of smart buildings in Algorithm 1 is illustrated as follows:

1. Prior to the learning procedure, the energy consumptions and discomfort parameters of both HVAC and SESS agent are initialized (line 1).
2. Probability of actions, weights of the actor network and the critic network, Q-value for the HVAC and the SESS agent are initialized (line 2).
3. The global neural network model ω_G in the GS along with the sharing batch φ for the FRL approach is initialized (line 3).
4. During every local training episode per communication round, each building's HVAC agent iterates the following procedures to compute its optimal energy consumption schedule from $t = 1$ to $t = 24$ (line 6~13).
 - (a) Sample action $a_{n,t}^{\text{HVAC}}$ based on distribution $N(\mu_{n,t}^{\text{HVAC}}, \{\sigma_{n,t}^{\text{HVAC}}\}^2)$ generated by actor network and key functions in state $s_{n,t}^{\text{HVAC}}$ (line 8).
 - (b) Execute action $a_{n,t}^{\text{HVAC}}$, receive reward $R_{n,t+1}^{\text{HVAC}}$ from the action and $V_{\pi_{\theta_c}^{\text{HVAC}}}(s_{n,t}^{\text{HVAC}})$ from the critic network, and finally, calculate the target value of critic network, TD_t (line 9).
 - (c) Compute the loss functions of actor network and critic network to minimize the losses and update the model of the HVAC agent using the ADAM optimizer [39] (lines 10, 11).
5. The HVAC agent n randomly selects a part of its local model ω_n^{sel} from the fully trained model ω_n and transmit it to the GS where it is stored in batch φ (lines 14, 15).
6. The GS yields a global neural network model ω_G^{new} by executing the FedSGD method with the selected weights in φ (line 17).
7. This newly generated global model ω_G^{new} is distributed to all HVAC agents in LBEMSs where those agents resume their own training based on ω_G^{new} (lines 18, 19).
8. All HVAC agents transmit their fully trained model $\widehat{\mathbf{B}}(\cdot)$ to the GS (line 21).
9. For training episodes, the SESS agent repeats the following procedures to search for an optimal charging and discharging schedule from $t = 1$ to $t = 24$ (line 23~31).
 - (a) The SESS agent infers the energy consumption of the LBEMS n using the model $\widehat{\mathbf{B}}_n(\cdot)$ and the state $S_{n,t}^{\text{HVAC}}$ (line 25).
 - (b) Sample an action a_t^{SESS} based on distribution $N(\mu_t^{\text{SESS}}, \{\sigma_t^{\text{SESS}}\}^2)$ generated by the actor network and the key functions given by state s_t^{SESS} , which includes the inferred energy consumption data for all LBEMSs (line 26).
 - (c) Compute action a_t^{SESS} , receive reward R_{t+1}^{SESS} and $V_{\pi_{\theta_c}^{\text{SESS}}}(s_t^{\text{SESS}})$ from the critic network, and calculate TD_t of the SESS agent (line 27).
 - (d) Estimate the loss functions of the actor network and the critic network by minimizing them, and update the model of the SESS agent using the ADAM optimizer (line 28, 29).

Algorithm 1: FRL-based energy management of a SESS with multiple smart buildings.

```

1 Initialize  $E_{n,t}^{\text{HVAC}}, E_t^{\text{SESS}}, \bar{\epsilon}_n, \epsilon_n, \bar{\tau}$  and  $\underline{\tau}$ 
2 Initialize  $p(a|s), Q(s,a)$ , two pairs of weights  $(\theta_a^{\text{HVAC}}, \theta_c^{\text{HVAC}})$  and  $(\theta_a^{\text{SESS}}, \theta_c^{\text{SESS}})$  of actor and critic
   networks for the HVAC agent and the SESS agent
3 Initialize the global neural network model  $\omega_G$  along with the sharing batch  $\phi$ 
4 %%Training an optimal energy consumption schedule for the HVACs
5 for communication episode = 1, Max communication episode do
6   for training episode = 1, iterative training episode do
7     for time=1,24 do
8       ▷ Sample an action  $a_{n,t}^{\text{HVAC}}$  using  $N(\mu_{n,t}^{\text{HVAC}}, \{\sigma_{n,t}^{\text{HVAC}}\}^2)$  from actor network for state  $s_{n,t}^{\text{HVAC}}$ 
9       ▷ Execute  $a_{n,t}^{\text{HVAC}}$ , receive  $R_{n,t+1}^{\text{HVAC}}, V_{\pi_{\theta_c^{\text{HVAC}}}}(s_{n,t}^{\text{HVAC}})$  and calculate  $TD_t$  of each building
10      ▷ Update the model by minimizing the loss of actor and critic network:
11       $L_t^{\text{actor}}(\theta_a^{\text{HVAC}}) = -\log p_{\pi_{\theta_a^{\text{HVAC}}}}(a_t|s_t)Q_{\pi_{\theta_c^{\text{HVAC}}}}(s_t, a_t),$ 
12       $L_t^{\text{critic}}(\theta_c^{\text{HVAC}}) = (TD_t - V_{\pi_{\theta_c^{\text{HVAC}}}}(s_t))^2$ 
13      ▷ Update the weight with weight optimizer:  $\nabla \omega_n = \Upsilon(L_t^{\text{actor}}(\theta_a^{\text{HVAC}}) + L_t^{\text{critic}}(\theta_c^{\text{HVAC}})),$ 
14       $\omega_n^{\text{new}} \leftarrow \omega_n^{\text{old}} + \nabla \omega_n$ 
15    end
16  end
17  ▷ Generate  $\omega_n^{\text{new,sel}}$  randomly and transmit it to the GS
18  ▷ Store  $\omega_n^{\text{new,sel}}$  in sharing batch  $\phi = [\omega_1^{\text{new,sel}}, \omega_2^{\text{new,sel}}, \dots, \omega_N^{\text{new,sel}}]$ 
19  %%Global model aggregation/update and HVAC agent update
20  ▷ Execute FEDSGD using the batch  $\phi$ :  $\omega_G^{\text{new}} = \frac{\sum_{n=1}^N \omega_n^{\text{new,sel}}}{N}$ 
21  ▷ Distribute  $\omega_G^{\text{new}}$  to all LBEMSs
22  ▷ Substitute the new global neural network model for the old local neural network model of the
   LBEMS:  $\omega_n \leftarrow \omega_G^{\text{new}}$ 
23 end
24 Transmit  $\hat{\mathbf{B}}(\cdot)$  to the GS
25 %%Training an optimal charging/discharging schedule for the SESS
26 for training episode = 1, iterative training episode do
27   for time=1,24 do
28     ▷ Infer the energy consumption of LBEMS  $n$  with  $\hat{\mathbf{B}}_n(\cdot)$ 
29     ▷ Sample an action  $a_t^{\text{SESS}}$  with  $N(\mu_t^{\text{SESS}}, \{\sigma_t^{\text{SESS}}\}^2)$  from actor network for state  $s_t^{\text{SESS}}$ 
30     ▷ Compute  $a_t^{\text{SESS}}$ , receive  $R_{t+1}^{\text{SESS}}, V_{\pi_{\theta_c^{\text{SESS}}}}(s_t^{\text{SESS}})$  and calculate  $TD_t$  of the SESS
31     ▷ Update the model by minimizing the loss of actor and critic network:
32      $L_t^{\text{actor}}(\theta_a^{\text{SESS}}) = -\log p_{\pi_{\theta_a^{\text{SESS}}}}(a_t^{\text{SESS}}|s_t^{\text{SESS}})Q_{\pi_{\theta_c^{\text{SESS}}}}(s_t^{\text{SESS}}, a_t^{\text{SESS}}),$ 
33      $L_t^{\text{critic}}(\theta_c^{\text{SESS}}) = (TD_t - V_{\pi_{\theta_c^{\text{SESS}}}}(s_t^{\text{SESS}}))^2$ 
34     ▷ Update the weight with weight optimizer:  $\nabla \omega^{\text{SESS}} = \Upsilon(L_t^{\text{actor}}(\theta_a^{\text{SESS}}) + L_t^{\text{critic}}(\theta_c^{\text{SESS}})),$ 
35      $\omega^{\text{SESS,new}} \leftarrow \omega^{\text{SESS,old}} + \nabla \omega^{\text{SESS}}$ 
36   end
37 end

```

4. Simulation Results

4.1. Simulation Setup

Under the TOU pricing, the predicted outdoor temperature and the PV generation output are plotted in Figure 4a–c. Accordingly, we considered three smart buildings with PV systems that are connected to a single SESS with the PV system. Each building was equipped with one controllable HVAC system and had different environment in terms of consumer comfort, thermal characteristics, HVAC capacity, occupancy capacity, and building occupancy which are detailed in Table 1 and plotted in Figure 4d. For the SESS, the battery capacity was 280 kWh; moreover, the maximum charging and discharging energy were both 120 kWh. The minimum, maximum and initial SOE of the SESS were set to 10%, 100% and 50% of its capacity, respectively. The penalties for SESS overcharging and undercharging were $\bar{\tau} = 600$ and $\underline{\tau} = 600$, respectively. The actor–critic DRL models of the HVAC and SESS agents consist of: (i) four hidden layers for a common body network that has 512 neurons in the first layer and 256 neurons each from the second layer to the fourth layer and (ii) two hidden layers for each of the actor network and the critic network having

128 neurons. We adopted a hyperbolic tangent function as a transfer function for the HVAC and SESS DRL models. The ADAM algorithm was selected to train the HVAC and SESS agents with a learning rate of 0.00005 and 0.00004, respectively. Two key functions, key_{mean} and key_{var} , for calculating new mean and variance of action distribution for the HVAC and SESS agents were selected as sigmoid function and exponential function, respectively. It is assumed that a single communication round between the LBEMS and the GS occurred every 150 iterations that are required for the LBEMS training. The test was executed using Python 3.7.0 and Pytorch 1.1.0.

Table 1. Simulation parameters in the three buildings.

Parameter	Building1	Building2	Building3
T^{\min}	23 °C	23 °C	22 °C
T^{\max}	25 °C	26 °C	26 °C
$\bar{\epsilon}, \epsilon$	13,000	16,000	21,000
α	0.85	0.92	0.88
β	−0.0004	−0.000325	−0.00022
γ	1.25	0.8	0.75
Cap	125	130	180
$E^{HVAC, \max}$	22 kWh	24 kWh	30 kWh

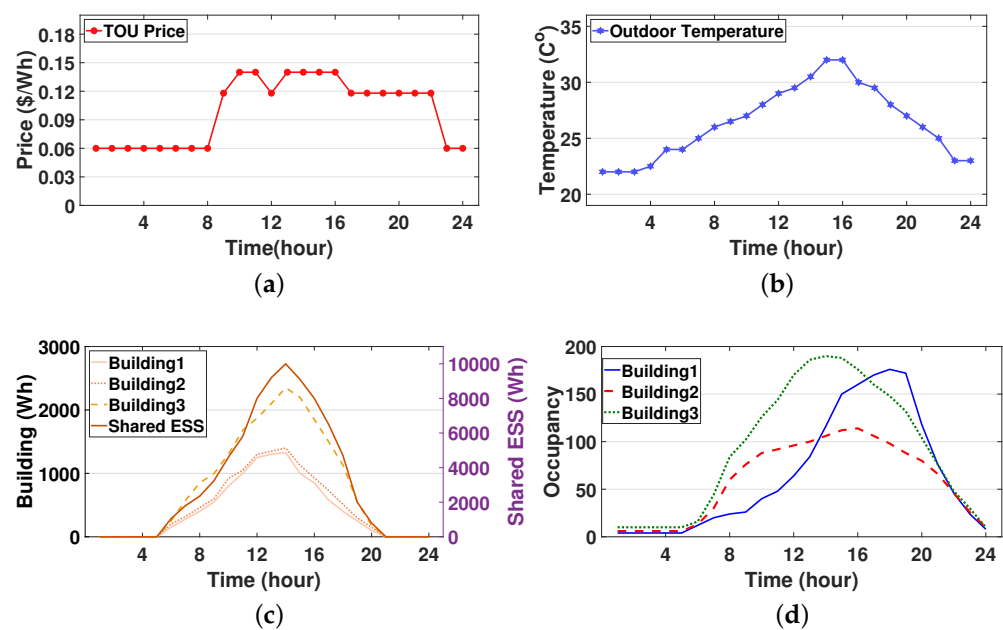


Figure 4. Parameters of electricity price, weather condition, and building occupancy: (a) time-of-use (TOU) price; (b) predicted outdoor temperature; (c) predicted photovoltaic (PV) generation output; and (d) daily occupancy.

4.2. Performance Assessment

4.2.1. Training Curve Convergence

Figure 5 depicts the training curves for average negative rewards for the HVAC agents in the three buildings under varying amounts of selected parameter sharing. We first observed that the training curves decreased and converged well during the learning process, while a sharp fluctuation occurred in every communication round (150 iterations). Another observation was that a smaller sharing of the selected parameters yielded a worse convergence performance. These results indicate a tradeoff relationship between the fast convergence of the training curve and the privacy preservation through sharing a lesser amount of selective parameters. In the subsequent simulation study, we considered that the HVAC agents share 80% of their DRL model with the GS to preserve energy usage information of the building occupants.

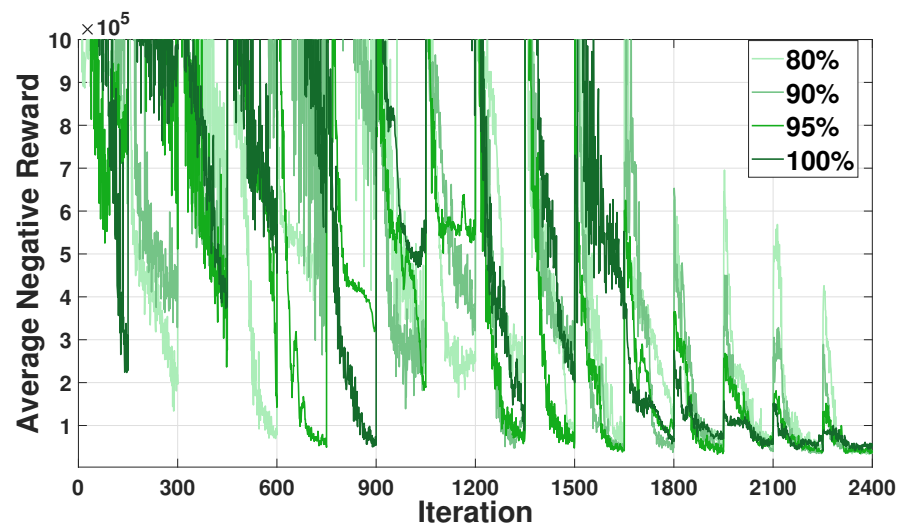


Figure 5. Average negative reward convergence for the heating, ventilation, and air conditioning (HVAC) agents with varying amounts of selective parameter sharing.

Figure 6 illustrates the training curves for negative reward (per one episode) and average negative reward (per 100 episodes) for the SESS agent. As mentioned in Section 3.2, the SESS agent utilizes fully trained neural networks of all LBEMSs' agents for preserving the privacy of buildings when learning the optimal policy of energy charging and discharging. We verified from Figure 6 that the training curves of the SESS agent converges well although the SESS agent uses neural network models rather than exact energy consumption data of LBEMSs.

Figure 7 compares the training curves for average negative rewards for the SESS agent between two schemes with perfect and without perfect energy consumption information of HVACs. In this figure, the scheme without the perfect energy consumption information of HVACs represents our proposed model where the neural network model of the HVAC agent is delivered to the GS and used as the element of the state space of the SESS agent. The SESS agent then infers the energy consumption schedules of HVACs using their neural network models. By contrast, the scheme with the perfect energy consumption information of HVACs considers the situation where the HVAC energy consumption schedule (i.e., the action of the HVAC agent in (6)) calculated by the HVAC agent is used as the element of the state space of the SESS agent instead of the neural network model of the HVAC agent. Note from Figure 7 that both the training curves without and with the perfect energy consumption information converge well, however the former converges rather slower than the latter. This is because inferring the energy consumption schedules of all HVACs using their neural network models requires additional training iterations of the SESS agent. We conclude from the result of Figure 7 that the proposed approach can preserve the privacy of HVAC energy consumption data at the expense of the convergence speed of the training curve.

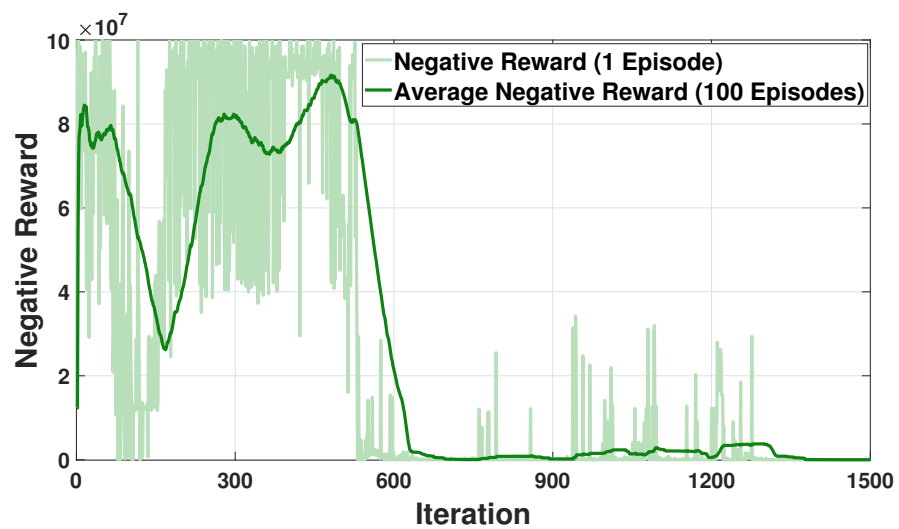


Figure 6. Negative reward convergence for the SESS agent.

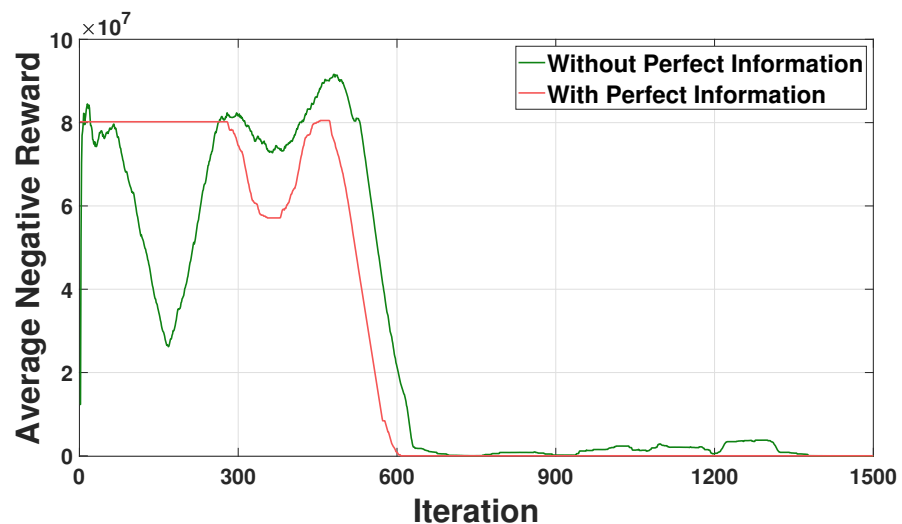


Figure 7. Comparison of average negative reward convergence for the SESS agent between with and without perfect energy consumption information of HVACs.

4.2.2. HVAC Energy Management

The graphs in Figure 8 show the aggregated scheduled optimal energy consumption (action $E_{n,t}^{HVAC}$) of the HVAC system considering predicted PV generation and the fixed loads for the three buildings. Specifically, in individual building, the optimal energy consumption schedule for the HVAC was computed after the training curves converge as shown in Figure 5. It can be observed from these graphs that the optimal energy consumption schedule of the HVAC in each building varies according to its heterogeneous building environment, such as the preferred indoor temperature, thermal characteristics, and the building occupancy. In Figure 8a, building1 consumes a large amount of the HVAC energy in the time period (5:00 p.m. to 8:00 p.m.) even though the outdoor temperature decreases. This is because building1 has a high occupancy with a large value of γ during this time period as shown in Figure 4d and Table 1. By contrast, it is observed from Figure 8b that building2 consumes more (or less) HVAC energy as the outdoor temperature increases (or decreases). This is because larger α and β in building2 makes the HVAC energy consumption rely increasingly on the outdoor temperature. Furthermore, because the outdoor temperature during the time period (1:00 a.m. to 4:00 a.m.) is lower than the preferred minimum indoor temperature of building2, the indoor temperature decreases gradually even though the HVAC turns off during this time period. Moreover, it should be

noted that among the three buildings considered, building3 consumes the largest HVAC energy for most of the time periods. This derives from the fact that building3 has the largest building occupancy and β that represents the sensitivity of current indoor temperature to the HVAC energy consumption.

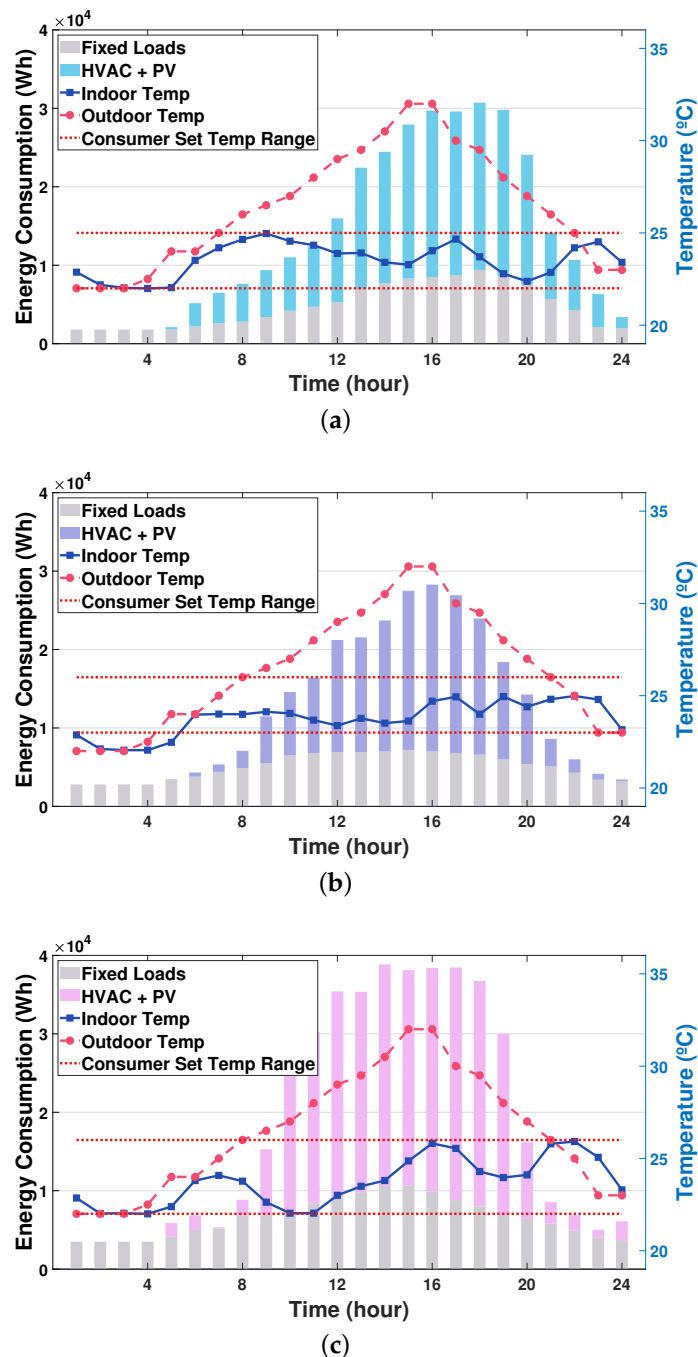


Figure 8. Sum of scheduled energy consumption for HVAC ($E_{n,t}^{HVAC} + \hat{E}_{n,t}^{PV}$) and the fixed loads: (a) building1; (b) building2; and (c) building3.

4.2.3. SESS Charging and Discharging Management

Figure 9a,b depict the SOE schedules of the SESS and the discharging energy schedules from the SESS to the three buildings, respectively. From Figure 9a, it can be observed that a sudden increase of the SOE occurs at five different times (9:00 a.m., 1:00 p.m., 3:00 p.m., 6:00 p.m., and 9:00 p.m.). At these times, the SESS charges energy from the

grid (i.e., the SESS purchases energy from the grid) to supply building loads. At other times, it discharges energy, thus resulting in decreased SOE to satisfy the energy demand for the three buildings. The discharging energy schedules for these three buildings are plotted in Figure 9b where no bar graphs are shown for the charging times mentioned above. Note from Figure 9b that building3 receives more discharging energy from the SESS than buildings 1 and 2 during daytime. This is because the aggregated energy consumption schedule for building3 is greater than that for the other buildings, as illustrated in Figure 8. Another observation is that the SESS discharges more energy to building1 than building2 during the time period (4:00 p.m. to 10:00 p.m.). This is due to the fact that building1 has larger occupancy and aggregated energy demand than building2 during this time period. From Figure 9a,b, it can be concluded that the proposed DRL-based SESS control method calculates a moderate charging and discharging schedule of the SESS for multiple buildings with heterogeneous environments. In addition, to preserve the privacy for buildings' energy consumption data, the SESS agent infers the corresponding schedules of all buildings based on the LBEMS energy consumption model $\hat{\mathbf{B}}(\cdot)$ in Section 3.2 without using the optimal building energy consumption schedule calculated by the HVAC agent. Figure 10 shows the sum of energy consumption schedules for three buildings during a day, which is calculated by DRL HVAC agents and DRL SESS agent, respectively. Note that the sum of DRL SESS agent-based energy consumption schedules is inferred using the LBEMS energy consumption model $\hat{\mathbf{B}}(\cdot)$. We verify from this figure that the energy consumption schedule inferred by the SESS agent using the LBEMS model deviates only 0.2~1% from the optimal energy consumption schedule computed by the HVAC agent. In sum, the aforementioned results demonstrate that the SESS agent successfully calculates optimal charging or discharging action while preserving the privacy of buildings' energy consumption.

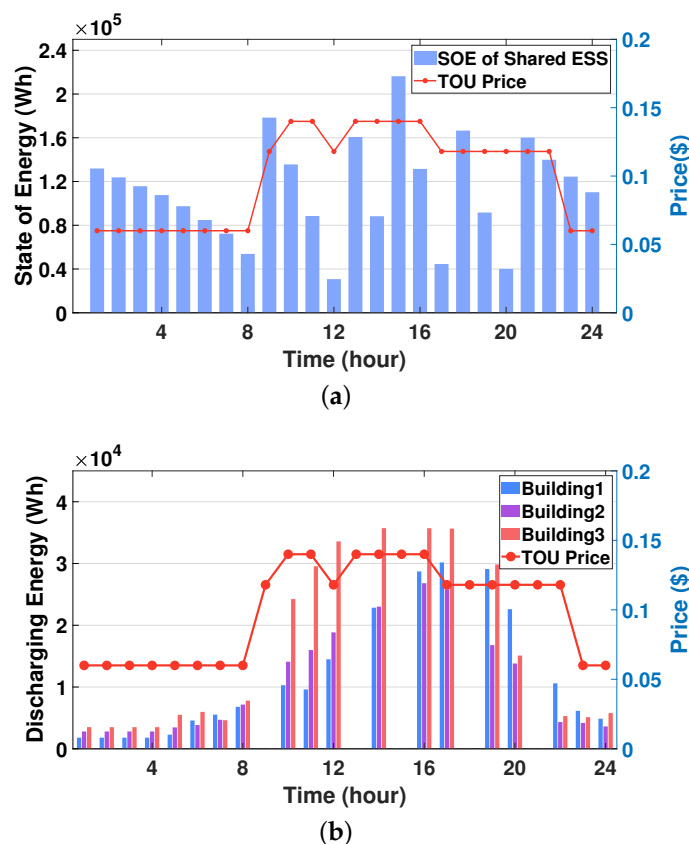


Figure 9. Simulation results for the SESS: (a) state of energy (SOE) (state SOE_t) of the SESS; (b) discharging energy schedules (action E_t^{SESS}) for the three buildings.

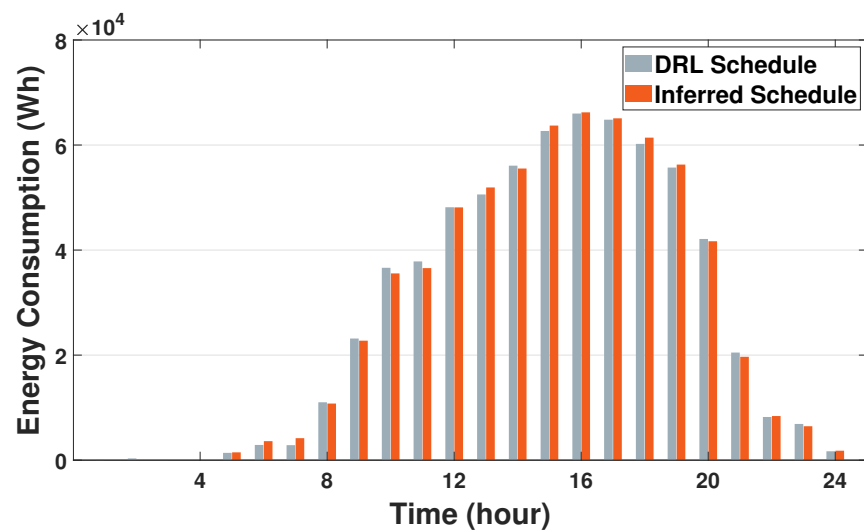


Figure 10. Comparison of the sum of energy consumption schedules for three buildings using the energy consumption schedule ($E_{n,t}^{HVAC}$) of DRL HVAC agents and using the inferred energy consumption schedule ($\hat{\mathbf{B}}_n(S_{n,t}^{HVAC})$) of DRL SESS agent.

4.2.4. Flexibility with Varying Number of the HVAC Agents

Figure 11 compares the training curves for average negative rewards for the HVAC agents in terms of the number of the HVAC agents (N^{HVAC}). We first observe from this figure that all training curves converge well with increasing number of the HVAC agents; however, the rate of convergence is rather slowing down as the number of the HVAC agents increases. Next, in the plot for $N^{HVAC} = 6$, we considered the situation in which an additional building (sixth HVAC agent) with the HVAC joined the FRL network that has the five HVAC agents at the 10th communication round (1350 iterations). In this situation, we observe from Figure 11 that after the 10th communication round, the training curve with $N^{HVAC} = 6$ starts increasing more significantly than the training curves with $N^{HVAC} = 1 \sim 5$. This observation is because the new HVAC agent joins the FRL network. After the 15th communication round (2100 iterations), we verify that the training curve with $N^{HVAC} = 6$ converges to its optimal policy. We conclude in Figure 11 that our FRL framework is flexible to a varying number of agents including the agent joining scenario.

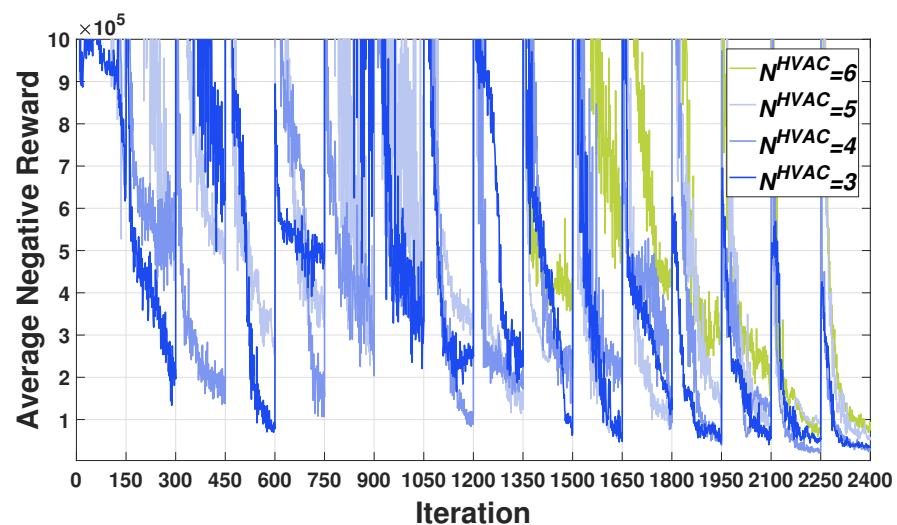


Figure 11. Average negative reward convergence with different number of HVAC agents (N^{HVAC}).

4.2.5. Performance Comparison between the Proposed Approach and Existing Methods

We compared the performance of our proposed FRL method to that of two existing methods that employ the distributed multi-agent model without FRL and the mixed-integer linear programming (MILP) optimization model, respectively. The simulation result showed that the electricity cost of HVAC systems using the proposed approach deviates only 1.1~1.4% from the electricity cost of HVAC systems using the distributed multi-agent model without FRL. In addition, we verified from the simulation study that compared to the MILP method, our proposed method enables all three HVAC agents to reduce 24~32% of energy consumption and 18.6~20.6% of electricity cost. Figure 12 compares the charging or discharging energy of the SESS between the MILP method and the proposed FRL method. In this figure, a positive (or negative) value represents the amount of charging (or discharging) energy. It can be verified from Figure 12 that the proposed FRL method discharges more energy than the MILP method during a day; the SESS agent using the proposed method can support 66% of the total energy consumption of the buildings whereas the MILP method can support 35% of the total energy consumption of the buildings. Therefore, these results indicated that building energy management integrated with the DRL-based HVAC and SESS agents leads to a more economic energy consumption scheduling compared to the conventional optimization approach.

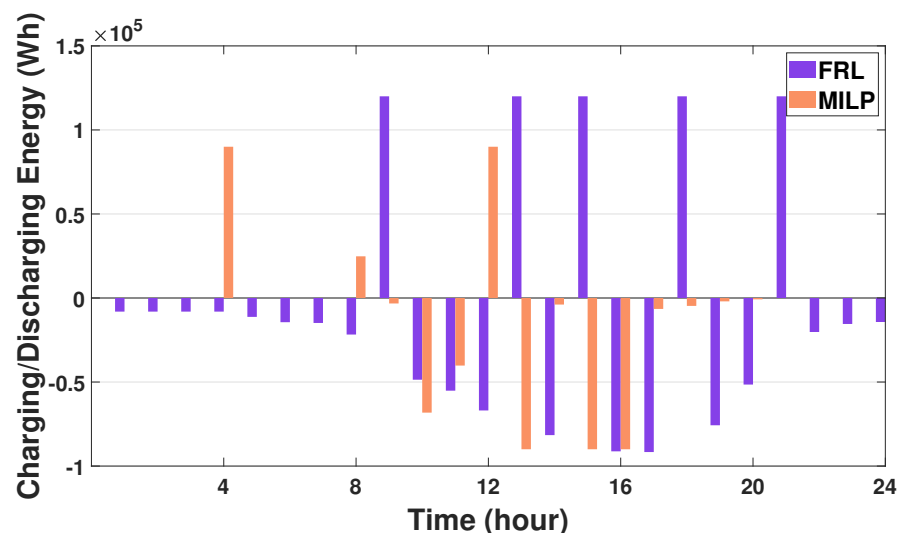


Figure 12. Comparison of hourly charging and discharging energy (E_t^{SESS}) of SESS between the proposed FRL method and the mixed-integer linear programming (MILP) method.

4.2.6. Computational Efficiency

We evaluated the total computation time for the proposed algorithm, which is defined as the sum of the maximum training time among the three HVAC agents and the training time for the SESS agent. The proposed method requires a total of 512 s (368 s and 144 s for the HVAC and SESS agent trainings, respectively) to find their optimal policy. The proposed algorithm is carried out for one day with a 1-h scheduling resolution; it is hence computationally efficient. In addition, we quantified the total computation time for the distributed multi-agent method without FRL. This method requires a total of 430 s (292 s and 138 s for the HVAC and SESS agent trainings, respectively). Compared to the distributed multi-agent method without FRL, the proposed approach requires only additional 82 s.

5. Discussions

5.1. Various Types of Controllable Appliances in the Smart Building

In this study, we considered that each LBEMS calculates the energy consumption schedule of only HVAC appliance that consumes the largest energy consumption in the

smart building. However, it is possible that various types of building appliances such as refrigerators, elevators and electric vehicles can be controlled by the LBEMS to reduce the electricity cost of the building [40,41]. In the previous studies, the controllable building appliances can be managed by the LBEMSs to find their economic energy consumption schedule according to their operation characteristics. To include these controllable appliances in our proposed framework, a key part would be to design state and action spaces and reward functions for the DRL agents of these appliances based on their unique operation characteristics. Thus, the DRL-based LBEMS model with additional appliance agents can be merged into the FRL-based framework proposed in Section 3 to schedule the optimal energy consumption of multiple smart buildings while preserving the privacy of the energy consumption of each smart building.

5.2. Practical Model of Building Thermal Dynamics

For simplicity, we approximated a real-world HVAC system model by focusing on analyzing the impact of the outdoor temperature, the building occupancy, and the HVAC energy consumption on the indoor temperature of the building. In addition, the DRL-based LBEMS was implemented in a single-zone building model rather than a multizone building that has multiple temperature zones. However, we emphasize that our study is the first step toward constructing the privacy-preserving FRL framework to schedule the energy consumption of smart buildings connected to the SESS. An important extension of our study is to construct the FRL-based energy management framework for multizone buildings with a complete model of the HVAC including the operation and air temperature of supply fan and external thermal disturbance [8,42], and it is referred to as a future work.

6. Conclusions

This paper presented a distributed and privacy-preserving FRL algorithm that conducts the energy management of a smart community that comprised of one SESS and multiple smart buildings equipped with the HVACs. The presented framework has the following two key components: (i) an FRL module that enables the HVAC agent and the GS to collaboratively learn the optimal consumption model of the HVAC while preserving the privacy of buildings' energy consumption data using a selective parameter sharing method, and (ii) an SESS management module that enables the SESS agent to train the optimal charging and discharging model of the SESS without having the knowledge of the optimal building energy consumption schedule. Simulation results are presented for a smart community comprised of one SESS and three smart buildings with PV systems. The results demonstrated the effectiveness of the proposed approach in terms of training convergence, energy consumption schedule of the HVAC, charging/discharging schedule of the SESS, flexibility under different number of the HVAC agents, decreased energy consumption as compared to the conventional optimization approach, and computation time.

Author Contributions: S.L. developed the proposed FRL-based energy management algorithm and conducted the simulation analysis for the research. L.X. reviewed the original and revised manuscript. D.-H.C. instructed and supervised the research as the corresponding author. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2020R1F1A1049314, and in part by the Chung-Ang University Graduate Research Scholarship in 2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bhusal, N.; Abdelmalak, M.; Benidris, M. Optimum locations of utility-scale shared energy storage systems. In Proceedings of the 2019 8th International Conference on Power Systems (ICPS), Jaipur, India, 20–22 December 2019; pp. 1–6. [\[CrossRef\]](#)
2. Golpira, H.; Khan, S.A.R. A multi-objective risk-based robust optimization approach to energy management in smart residential buildings under combined demand and supply uncertainty. *Energy* **2019**, *170*, 1113–1129. [\[CrossRef\]](#)
3. Melhem, F.Y.; Grunder, O.; Hammoudan, Z.; Moubayed, N. Energy management in electrical smart grid environment using robust optimization algorithm. *IEEE Trans. Ind. Appl.* **2018**, *54*, 2714–2726. [\[CrossRef\]](#)
4. Shafie-Khah, M.; Siano, P. A stochastic home energy management system considering satisfaction cost and response fatigue. *IEEE Trans. Ind. Inform.* **2018**, *14*, 629–638. [\[CrossRef\]](#)
5. Sedighzadeh, M.; Esmaili, M.; Mohammadkhani, N. Stochastic multi-objective energy management in residential microgrids with combined cooling, heating, and power units considering battery energy storage systems and plug-in hybrid electric vehicles. *J. Clean. Prod.* **2018**, *195*, 301–317. [\[CrossRef\]](#)
6. Gomez-Romero, J.; Fernandez-Basso, C.J.; Cambronero, M.V.; Molina-Solana, M.; Campana, J.R.; Ruiz, M.D.; Martin-Bautista, M.J. Probabilistic algorithm for predictive control with full-complexity models in non-residential buildings. *IEEE Access.* **2019**, *7*, 38748–38765. [\[CrossRef\]](#)
7. Yousefi, M.; Hajizadeh, A.; Soltani, M.N.; Hredzak, B. Predictive home energy management system with photovoltaic array, heat pump, and plug-in electric vehicle. *IEEE Trans. Ind. Inform.* **2021**, *17*, 430–440. [\[CrossRef\]](#)
8. Yu, L.; Xie, D.; Zou, Y.; Wang, K. Distributed real-time HVAC control for cost-efficient commercial buildings under smart grid environment. *IEEE Internet Things J.* **2018**, *5*, 44–55. [\[CrossRef\]](#)
9. Jindal, A.; Kumar, N.; Rodrigues, J.J.P.C. A heuristic-based smart HVAC energy management scheme for university buildings. *IEEE Trans. Ind. Inform.* **2018**, *14*, 5074–5086. [\[CrossRef\]](#)
10. Yu, L.; Jiang, T.; Zou, Y. Online Energy Management for a Sustainable Smart Home With an HVAC Load and Random Occupancy. *IEEE Trans. Smart Grid.* **2017**, *10*, 1646–1659. [\[CrossRef\]](#)
11. Wang, F. Multi-objective optimization model of source-load-storage synergetic dispatch for a building energy management system based on TOU price demand response. *IEEE Trans. Ind. Appl.* **2017**, *54*, 1017–1028. [\[CrossRef\]](#)
12. Jo, J.; Park, J. Demand-side management with shared energy storage system in smart grid. *IEEE Trans. Smart Grid.* **2020**, *11*, 4466–4476. [\[CrossRef\]](#)
13. Ye, G.; Li, G.; Wu, D.; Chen, X.; Zhou, Y. Towards cost minimization with renewable energy sharing in cooperative residential communities. *IEEE Access.* **2017**, *5*, 11688–11699. [\[CrossRef\]](#)
14. Zhu, H.; Ouahada, K. Credit-based distributed real-time energy storage sharing management. *IEEE Access.* **2019**, *7*, 185821–185838. [\[CrossRef\]](#)
15. Fleischhacker, A.; Auer, H.; Lettner, G.; Botterud, A. Sharing solar PV and energy storage in apartment buildings: Resource allocation and pricing. *IEEE Trans. Smart Grid.* **2019**, *10*, 3963–3973. [\[CrossRef\]](#)
16. Yao, J.; Venkitasubramaniam, P. Privacy aware stochastic games for distributed end-user energy storage sharing. *IEEE Trans. Signal Inf. Process. Netw.* **2018**, *4*, 82–95. [\[CrossRef\]](#)
17. Mocanu, E.; Mocanu, D.C.; Nguyen, P.H.; Liotta, A.; Webber, M.E.; Gibescu, M.; Slootweg, J.G. On-Line Building energy optimization using deep reinforcement learning. *IEEE Trans. Smart Grid.* **2019**, *10*, 3698–3708. [\[CrossRef\]](#)
18. Wei, T.; Ren, S.; Zhu, Q. Deep reinforcement learning for joint datacenter and HVAC load control in distributed mixed-use buildings. *IEEE Trans. Sustain. Comput.* **2019**. [\[CrossRef\]](#)
19. Chen, B.; Cai, Z.; Berges, M. Gnu-RL: A precocious reinforcement learning solution for building HVAC control using a differentiable MPC policy. In Proceedings of the 6th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, New York, NY, USA, 13–14 November 2019; pp. 316–325. [\[CrossRef\]](#)
20. Yu, L.; Sun, Y.; Xu, Z.; Shen, C.; Yue, D.; Jiang, T.; Guan, X. Multi-agent deep reinforcement learning for HVAC control in commercial buildings. *IEEE Trans. Smart Grid.* **2021**, *12*, 407–419. [\[CrossRef\]](#)
21. Yu, L.; Qin, S.; Zhang, M.; Shen, C.; Jiang, T.; Guan, X. A review of deep reinforcement learning for smart building energy management. *IEEE Internet Things J.* **2021**. [\[CrossRef\]](#)
22. Yu, L. Deep reinforcement learning for smart home energy management. *IEEE Internet Things J.* **2019**, *7*, 2751–2762. [\[CrossRef\]](#)
23. Wang, B.; Li, Y.; Ming, W.; Wang, S. Deep reinforcement learning method for demand response management of interruptible load. *IEEE Trans. Smart Grid.* **2020**, *11*, 3146–3155. [\[CrossRef\]](#)
24. Huang, X.; Hong, S.H.; Yu, M.; Ding, Y.; Jiang, J. Demand response management for industrial facilities: A deep reinforcement learning approach. *IEEE Access.* **2019**, *7*, 82194–82205. [\[CrossRef\]](#)
25. Gorostiza, F.S.; Gonzalez-Longatt, F. Deep reinforcement learning-based controller for SOC management of multi-electrical energy storage system. *IEEE Trans. Smart Grid.* **2020**, *11*, 5039–5050. [\[CrossRef\]](#)
26. You, Y.; Li, Z.; Oechtering, T.J. Energy management strategy for smart meter privacy and cost saving. *IEEE Trans. Inf. Forensics Secur.* **2021**, *16*, 1522–1537. [\[CrossRef\]](#)
27. Rahman, M.S.; Basu, A.; Nakamura, T.; Takasaki, H.; Kiyomoto, S. PPM: Privacy policy manager for home energy management system. *J. Wirel. Mob. Netw. Ubiquitous Comput. Dependable Appl.* **2018**, *9*, 42–56.
28. Sun, Y.; Lampe, L.; Wong, V.W.S. Smart meter privacy: Exploiting the potential of household energy storage units. *IEEE Internet Things J.* **2018**, *5*, 69–78. [\[CrossRef\]](#)

29. Jia, R.; Dong, R.; Sastry, S.S.; Sapnos, C.J. Privacy-enhanced architecture for occupancy-based HVAC control. In Proceedings of the 2017 ACM/IEEE 8th international conference on cyber-physical systems (ICCPS), Pittsburgh, PA, USA, 18–20 April 2017; pp. 177–186. [CrossRef]
30. McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. 2019; pp. 1–11. Available online: <https://arxiv.org/pdf/1602.05629> (accessed on 3 May 2021).
31. Zhou, H.H.; Feng, W.; Lin, Y.; Xu, Q.; Yang, Q. Federated Deep Reinforcement Learning. 2020; pp. 1–9. Available online: <https://arxiv.org/abs/1901.08277> (accessed on 3 May 2021).
32. Liu, B.; Wang, L.; Liu, M. Lifelong federated reinforcement learning: A learning architecture for navigation in cloud robotic systems. *IEEE Trans. Robot. Autom.* **2019**, *4*, 4555–4562. [CrossRef]
33. Nadiger, C.; Kumar, A.; Abdelhak, S. Federated reinforcement learning for fast personalization. In Proceedings of the 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering(AIKE), Sardinia, Italy, 3–5 June 2019; pp. 123–127. [CrossRef]
34. Mowla, N.I.; Tran, N.H.; Doh, I.; Chae, K. AFRL: Adaptive federated reinforcement learning for intelligent jamming defense in FANET. *J. Commun. Netw.* **2020**, *22*, 244–258. [CrossRef]
35. Lee, S.; Choi, D.-H. Federated reinforcement learning for energy management of multiple smart homes with distributed energy resources. *IEEE Trans. Ind. Inform.* **2020**. [CrossRef]
36. Shokri, R.; Shmatikov, V. Privacy-preserving deep learning. In Proceedings of the 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 29 September–2 October 2015; pp. 1310–1321. [CrossRef]
37. Silver, D.; Lever, G.; Heess, N.; Degris, T.; Wierstra, D.; Riedmiller, M. Deterministic policy gradient algorithms. In Proceedings of the 31st International Conference on Machine Learning (ICML), Beijing, China, 21–26 June 2014; pp. 387–395.
38. Tesau, C.; Tesau, G. Temporal difference learning and TD-gammon. *Commun. ACM* **1995**, *38*, 58–68. [CrossRef]
39. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. 2017; pp. 1–15. Available online: <https://arxiv.org/abs/1412.6980> (accessed on 3 May 2021).
40. Liu, Z.; Wu, Q.; Shahidehpour, M.; Li, C.; Huang, S.; Wei, W. Transactive real-time electric vehicle charging management for commercial buildings with PV on-site generation. *IEEE Trans. Smart Grid.* **2019**, *10*, 4939–4950. [CrossRef]
41. Martirano, L.; Parise, G.; Greco, G.; Manganelli, M.; Massarella, F.; Cianfrini, M.; Parise, L.; di Laura Frattura, P.; Habib, E. Aggregation of users in a residential/commercial building managed by a building energy management system (BEMS). *IEEE Trans. Ind. Appl.* **2019**, *55*, 26–34. [CrossRef]
42. Ostadijafari, M.; Dubey, A. Tube-based model predictive controller for building’s heating ventilation and air conditioning (HVAC) system. *IEEE Syst. J.* **2020**. [CrossRef]