


Article

Economic Dispatch for Smart Buildings with Load Demand of High Volatility Based on Quasi-Quadratic Online Adaptive Dynamic Programming

Kairui Chen ^{1,2} , Zhangmou Zhu ¹ and Jianhui Wang ^{1,*}¹ School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510006, China² School of Computer & Information, Qiannan Normal University for Nationalities, Guizhou 558000, China

* Correspondence: jhwang@gzhu.edu.cn

Abstract: In this paper, a quasi-quadratic online adaptive dynamic programming (QOADP) algorithm is proposed to realize optimal economic dispatch for smart buildings. Load demand of high volatility is considered, which is modeled by an uncontrollable state. To reduce residual errors of the approximation structure, a quasi-quadratic-form parametric structure was designed elaborately with a bias term to counteract effects of uncertainties. Based on action-dependent heuristic dynamic programming (ADHDP), an implementation of the QOADP algorithm is presented that involved obtaining optimal economic dispatch for smart buildings. Finally, hardware-in-loop (HIL) experiments were conducted, and the performance of the proposed QOADP algorithm is superior to that of two other typical algorithms.

Keywords: adaptive dynamic programming; smart buildings; economic dispatch; energy management systems

MSC: 37N35

Citation: Chen, K.; Zhu, Z.; Wang, J. Economic Dispatch for Smart Buildings with Load Demand of High Volatility Based on Quasi-Quadratic Online Adaptive Dynamic Programming. *Mathematics* **2022**, *10*, 4701. <https://doi.org/10.3390/math10244701>

Academic Editor: Javier Alcaraz

Received: 20 October 2022

Accepted: 8 December 2022

Published: 11 December 2022

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



Copyright: © 2022 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

The world is seeing tremendous growth in independent smart buildings due to the development of generators and battery techniques. When distributed generators and storage equipment are connected to smart buildings, there are new challenges for traditional energy management mechanisms due to uncertainties of user behaviors. These uncertainties are likely to result in a load demand sequence with high volatility. Therefore, an intelligent energy management system with the ability to confront with the uncertainties is necessary for smart buildings to realize optimal and economic dispatch.

1.1. Literature Review

Optimal economic dispatch is generally formulated as an optimization problem [1,2]. Therefore, traditional optimization techniques are widely adopted to obtain optimal economic dispatch solutions. The convex optimization algorithm is a fundamental and well-developed method for dispatch tasks. In [3], the authors developed a Lyapunov-based optimization method for a transformed problem of the original non-convex problem. In [4], a robust convex optimization algorithm was designed for single-phase or balanced three-phase microgrids with adverse conditions of random demand and renewable energy resources. In [5], a chance-constraint-based stochastic optimization technique was applied to design an optimal energy scheduling policy for multi-service battery energy storage. However, it is still very difficult for some nonlinear and strongly coupled problems to be transformed into a convex formulation. Motivated by the development mechanism in nature, heuristic algorithms were developed to search for optimal solutions in the feasible

domain. In [6], the authors developed a particle swarm optimization-based algorithm to form a reliable hybrid microgrid with optimal economic dispatch for non-sensitive loads and energy storage systems. In [7], a genetic-algorithm-based, online, real-time optimized energy management strategy was proposed for plug-in hybrid electric vehicles. Most of the literature above focused on static optimization problems whose solutions were in a space with finite dimensions. For optimization problems of dynamical systems, dynamic programming is a promising technique, since the optimality principle simplifies the numerical computation. In [8], the authors developed a two-scale dynamic-programming-based optimal energy management method for wind–battery–power systems. A multidimensional dynamic-programming-based energy control policy was proposed for global composite operating cost minimization in [9]. However, dynamic programming suffers from the so-called “curse of dimensionality” as the number of time steps increases. The solution space evolves with an exponential rate, which hinders the application of dynamic programming.

To avoid the “curse of dimensionality”, adaptive dynamic programming (ADP) was developed to solve the optimal economic dispatch problem with unknown system dynamics [10]. One can refer to [11,12] for the pioneering work and an overview of ADP. Due to its self-learning ability and adaptivity, it is popular in optimal economic dispatch problems of smart buildings. An optimal battery-control policy was obtained by designing a time-based ADP algorithm for residential buildings in [13]. In [14], the authors designed action-dependent, heuristic dynamic programming-based algorithms for optimal economic dispatch of multiple buildings. In [15], the authors proposed a temporal difference learning-based ADP algorithm for fast computation. In a recent work [16], a novel, model-free, real-time RT-ADP algorithm was verified in a hardware-in-the-loop experimental platform. However, models studied in the above literature are deterministic. In engineering practice, uncertainties are inevitable in most circumstances. Therefore, ADP was developed for optimal economic dispatch of smart buildings with uncertainties in the past few years [17–20]. To make the ADP algorithms more convincing, stability analysis was provided for ADP algorithms of energy systems in [21–25]. In [21], the authors proposed a dual, iterative Q-learning algorithm to solve the optimal battery-control law for a smart microgrid. Extensions of [21] were created in [23] for cases with renewable generation and in [22] for cases with actuator constraints. For application in appliance-level economic dispatch, a data-based ADP algorithm was studied for ice-storage air conditioning systems in [24]. The above works commonly assumed that the load demand of users is periodic. However, the load demand of users is random with high volatility in practice; thus, the periodic assumption cannot be satisfied in most real cases.

In the implementation of an ADP algorithm, the design of a function approximator shows significant effects on the performance of the overall algorithm. The feedforward neural network is the most commonly used as a function approximator. Leaving aside feedforward neural networks, the authors of [26] employed an echo-state networks as a parametric structure. The authors of [27] approximated the iterative value function by a fuzzy structure. However, these general-function approximators were employed without considering features of the considered systems, which can result in large residual errors.

1.2. Motivation and Contributions

According to the above discussion, the existing algorithms are mainly concerned with optimal economic dispatch for smart buildings with periodic load demands. In practice, load demands are not periodic due to uncertain user behaviors, and even with high volatility, which inspires the study of our work. In this paper, a QOADP algorithm is proposed to obtain the optimal economic dispatch for smart buildings with high volatility load demand. A quasi-quadratic form parametric structure is designed elaborately to approximate the value function precisely. Compared with the existing literature, the main contributions of this paper are listed as follows:

1. A kind of online ADP is proposed to iteratively obtain optimal economic dispatch for smart buildings with high volatility of load demand. The online algorithm allows parameters of controller to achieve optimal control with the changing of load demand.
2. A quasi-quadratic form parametric structure is designed for the implementation of QOAPD with a bias term to counteract the effects of uncertainties. To simplify the function approximation structure in the proposed algorithm, the feedforward information of the uncontrollable state is taken into account in the iterative value function and the iterative controller.

The rest of this paper is organized as follows. In Section 2, the mathematical model for the EMS is established. In Section 3, the QOAPD is proposed for the EMS, and the optimality is analyzed. Then, parametric structures are designed for implementation. In Section 4, the data-driven approach is proposed. Section 5 provides hardware-in-loop (HIL) experimental results of the proposed method, verifying the effectiveness of QOAPD. In Section 6, conclusions are drawn.

2. EMS of Smart Buildings

A smart building is composed of an utility grid; an energy storage system, which is connected to the point of common coupling (PCC) with the power electronic converter; and the building load. The diagram of the EMS of a smart building is shown in Figure 1.

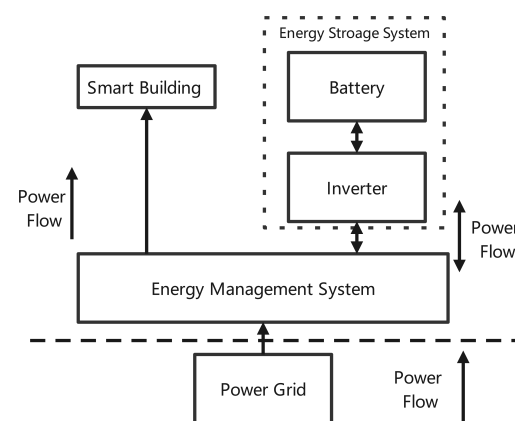


Figure 1. EMS of a smart building.

Power balance at the PCC is

$$P_G(t) = P_L(t) - P_B(t) \quad (1)$$

The battery model is

$$\frac{d[SOC(t)]}{dt} = \begin{cases} \frac{\eta}{E_{B,rate}} P_B(t), & P_B(t) \geq 0 \\ \frac{1}{\eta E_{B,rate}} P_B(t), & P_B(t) < 0 \end{cases} \quad (2)$$

where $P_B(t) \geq 0$ denotes battery charging and $P_B(t) < 0$ denotes battery discharging. In reality, dispatch of the storage system must satisfy the following constraints:

$$SOC_{min} \leq SOC(t) \leq SOC_{max} \quad (3)$$

$$|P_B(t)| \leq P_{B,rate} \quad (4)$$

Constraint (3) makes sure the battery energy is bounded within an allowable range. Constraint (4) limits the charging/discharging power of the battery.

The transition function of the load demand in continuous form can be defined as

$$\dot{P}_L(t) = g(P_L(t)) \quad (5)$$

The optimal energy management problem is formulated as a discrete-time control problem with $\tau = 1$ h, which is consistent with the real-time price (RTP). Using the Euler method with $\tau = 1$ h leads to

$$\begin{aligned} SOC(k\tau + \tau) = SOC(k\tau) &+ (sgn(P_B(k\tau)) \cdot \frac{\eta}{E_{B,rate}} \\ &+ sgn(P_B(k\tau)) \cdot \frac{1}{\eta E_{B,rate}}) P_B(k\tau) \end{aligned} \quad (6)$$

$$\begin{aligned} P_L(k\tau + \tau) &= P_L(k\tau) + \tau g(P_L(k\tau)) \\ &= \hat{g}(P_L(k\tau)) \end{aligned} \quad (7)$$

$$P_G(k\tau) = P_L(k\tau) - P_B(k\tau) \quad (8)$$

According to (6)–(8), the state of the system is defined as $\mathbf{x}_k = [P_G(k\tau), SOC(k\tau) - SOC_{ref}, P_L(k\tau)]^T$, i.e., $x_{1,k} = P_G(k\tau)$, $x_{2,k} = SOC(k\tau) - SOC_{ref}$, $x_{3,k} = P_L(k\tau)$. Let $\mathbf{u}_k = P_B(k\tau)$ be the control vector.

For convenience of analysis, delays are introduced in $P_B(t)$ and $P_L(t)$. Therefore, the discrete-time building management system is

$$\begin{aligned} \bar{\mathbf{x}}_{k+1} &= \mathbf{A}\bar{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k + \mathbf{D}\hat{\mathbf{x}}_k \\ \hat{\mathbf{x}}_{k+1} &= g'(\hat{\mathbf{x}}_k) \end{aligned} \quad (9)$$

where $\bar{\mathbf{x}}_k = [x_{1,k} \ x_{2,k}]^T$ is controllable and $\hat{\mathbf{x}}_k = x_{3,k}$ is uncontrollable. The drift, input and disturbance dynamics of controllable subsystem are $\mathbf{A} = \text{diag}(0, 1)$, $\mathbf{B} = [-1 \ \text{sign}(\mathbf{u}_k) \cdot \frac{\eta}{E_{B,rate}} + \text{sign}(\mathbf{u}_k) \cdot \frac{1}{\eta E_{B,rate}}]^T$ and $\mathbf{D} = [1 \ 0]^T$, respectively.

It should be noted that the building energy system (9) is partly controllable, since it is assumed that users in the building are not sensitive to the electricity price. Differently from completely controllable systems, the system state will deviate from the equilibrium point. There is not a continuous control law $\mu(\mathbf{x}_k)$ asymptotically stabilizing the system. Hence, no fixed optimal control law $\mu^*(\mathbf{x}_k)$ is able to maintain the optimality from step k to $+\infty$.

It is essential for smart building users to find appropriate discharging/charging power of battery to reduce the electricity cost under the RTP scheme. Hence, EMS is designed to maximize economic benefits of users and extend the battery life. Therefore, a utility function is defined as

$$\begin{aligned} U(\mathbf{x}_k, \mathbf{u}_k, k) &= m_1(\lambda_k x_{1,k})^2 + m_2 x_{2,k}^2 + m_3 x_{3,k}^2 + m_4 \mathbf{u}_k^T \mathbf{u}_k \\ &= \mathbf{z}_k^T \mathbf{M}_k \mathbf{z}_k \end{aligned} \quad (10)$$

where $\mathbf{M}_k = \text{diag}(m_1 \lambda_k^2, m_2, m_3, m_4)$ is a time-varying matrix due to dynamic price. The first term denotes the electricity cost, the second term represents the depth of discharge and the fourth term is to shorten the charging/discharging cycle. The utility function is continuous positive definite, which ensures that the performance index function is convex. Then, the performance index function can be defined as the sum of utility functions from time step k to ∞ —i.e.,

$$J(\mathbf{x}_k, \underline{\mathbf{u}}_k, k) = \sum_{i=k}^{\infty} \gamma^i U(\mathbf{x}_i, \mathbf{u}_i, i) \quad (11)$$

Now, our goal is to design an optimal control sequence $\underline{\mathbf{u}}_k^* = \{\mathbf{u}_k^*, \mathbf{u}_{k+1}^*, \dots\}$ which minimizes the performance index function (11). For $\forall k$, the optimal control is

$$u^*(x_k) = \arg \min_u \{U(x_k, u, k) + \gamma J^*(x_{k+1}, k+1)\} \quad (12)$$

where

$$J^*(x_k, k) = \min_u \{U(x_k, u, k) + \gamma J^*(x_{k+1}, k+1)\} \quad (13)$$

The discount factor γ should be less than one to ensure convergence of the series.

3. Building Energy Management Strategy

In this section, a QOADP-based building energy management strategy is proposed to solve the optimal control problem formulated in Section 2. Moreover, the iterative QOADP algorithm and the corresponding parametric structure are designed.

3.1. Iterative QOADP Algorithm

ADP is an effective tool for solving the nonlinear optimal control problem based on the principle of optimality [11,12]. Generally, the iterative ADP algorithm can be classified into value iteration (VI) and policy iteration (PI). The main idea of ADP is the solution of the Hamilton–Jacobi–Bellman (HJB) equation can be iteratively approximated. We focus on the value iteration in this study.

The building energy system in Section 2 is time-varying, since the utility function varies with time. Hence, the performance index function J^* is time-varying; i.e., $J^* = J^*(x_k, k)$.

An initial value function $V_0(x, 0)$ is used to approximate $J^*(x, 0)$. Then, the initial iterative control policy $\mu_0(x, 0)$ can be solved by

$$\begin{aligned} \mu_0(x_k, 0) &= \arg \min_u \{U(x_k, u, 0) + \gamma V_0(x_{k+1}, 0)\} \\ &= \arg \min_u \{U(x_k, u, 0) + \gamma V_0(F(x_k, u, 0), 0)\} \end{aligned} \quad (14)$$

Next, the iterative value function at time index $k = 0$ can be improved by

$$V_1(x_k, 0) = U(x_k, \mu_0(x_k, 0), 0) + \gamma V_0(x_{k+1}, 0) \quad (15)$$

For $k = 0, i = 1, 2, \dots$, the iterative control policy and iterative value function can be updated by

$$\mu_i(x_k, 0) = \arg \min_u \{U(x_k, u, 0) + \gamma V_0(x_{k+1}, 0)\} \quad (16)$$

and

$$V_{i+1}(x_k, 0) = U(x_k, \mu_i(x_k, 0), 0) + \gamma V_i(x_{k+1}, 0) \quad (17)$$

For $k = 1, 2, \dots$, the real-time iterative control policy and the updating iterative value function will be obtained by

$$\mu_i(x_k, k) = \arg \min_u \{U(x_k, u, k) + \gamma V_i(x_{k+1}, k)\} \quad (18)$$

$$V_{i+1}(x_k, k) = U(x_k, \mu_i(x_k, k), k) + \gamma V_i(x_{k+1}, k) \quad (19)$$

The above description explains the offline iterative algorithm. However, it is nearly impossible to be applied in the real engineering tasks. The reasons are listed as follows:

(1) The number of iterations is infinite between time step k and time step $k+1$, which cannot be carried out.

(2) The iterative control policy obtained by finite iterations $\mu_i(x_k, k)$ may not be stable. Hence, it is likely to lead to abnormal operations of the system.

To overcome the difficulties, it is meaningful to develop an online algorithm, in which the time index k is equal to iteration index i . Hence, the online value iteration can be summarized as

$$\mu_k(x_k, k) = \arg \min_u \{U(x_k, u, k) + \gamma V_k(x_{k+1}, k)\} \quad (20)$$

and

$$V_{k+1}(x_k, k) = U(x_k, \mu_k(x_k, k), k) + \gamma V_k(x_{k+1}, k) \quad (21)$$

with a positive semidefinite initial function $V_0(x_k, 0)$. The whole structure diagram is shown in Figure 2.

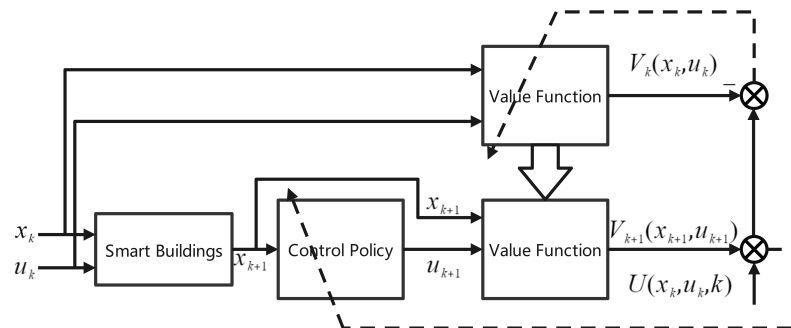


Figure 2. The structure diagram of online value iteration.

Note that the online value iteration cannot directly be applied to the EMS, as it is not completely controllable, which differs from most nonlinear systems in the literature [22,28,29]. For controllable systems, there is an admissible control law [30] to bring states of systems to the equilibrium point. Hence, the discount factor should not be omitted in the EMS control. Moreover, the dynamics of load demand is very complex and stochastic. However, it is not desirable to use a complex controller in the real application.

To solve this problem, load forecasting module is separated from the controller, and the controller $\mu_k(x_k, k)$ is modified by $\mu_k(x_k, \hat{x}_{k+1}, k)$, where \hat{x}_{k+1} is the predictive value of load demand at time step k . The predictive module can be implemented by an exponential smoothing model [31], neural network model, etc. Therefore, the complexity of the controller is reduced, which is beneficial for the real application.

3.2. Parametric Structure Design for Value Function

Function approximation structures should be specified to approximate V_k and μ_k . The most commonly used parametric structures are the feedforward neural network [32] and quadratic form neural network [33]. However, they are not the most fitting function approximators to be used in EMS. Here, the iterative value function and iterative control policy are defined as

$$V_k(x_k, k) = x_k^T P_{0,k} x_k + P_{1,k} x_k + P_{2,k} \quad (22)$$

and

$$\mu_k(x_k, k) = K_k x_k + b_k. \quad (23)$$

where the coefficients of (22) and (23) can be expressed as

$$K_k = -(2m_4 + 2\gamma B^T \tilde{P}_{011,k} B)^{-1} \times [2\gamma B^T \tilde{P}_{011,k}^T A \quad 2\gamma B^T \tilde{P}_{011,k}^T D] \quad (24)$$

$$b_k = -(2m_4 + 2\gamma B^T \tilde{P}_{011,k} B)^{-1} \times (2\gamma B^T \tilde{P}_{021,k}^T \hat{x}_{k+1} + \gamma \tilde{P}_{111,k} B) \quad (25)$$

$$P_{0,k+1} = M_s + K_k^T M_u K_k + \gamma([A \ 0]^T \tilde{P}_{011,k} [A \ 0] + [A \ 0]^T \tilde{P}_{011,k} B K_k + K_k^T B^T \tilde{P}_{011,k} [A \ 0] + K_k^T B^T \tilde{P}_{011,k} B K_k) \quad (26)$$

$$\begin{aligned}
P_{1,k+1} = & 2b_k^T M_u K_k + \gamma \{ [\tilde{P}_{11,k} A \mathbf{0}] + \tilde{P}_{11,k} B K_k + \\
& [(A^T \tilde{P}_{012,k} \hat{x}_{k+1})^T + \hat{x}_{k+1}^T \tilde{P}_{021,k} A \mathbf{0}] + \\
& (K_k^T B^T \tilde{P}_{012,k} \hat{x}_{k+1})^T + \hat{x}_{k+1}^T \tilde{P}_{021,k} B K_k + \\
& (B b_k + D \hat{x}_k)^T \tilde{P}_{011,k} ([A \mathbf{0}] + B K_k) + \\
& (B b_k + D \hat{x}_k)^T \tilde{P}_{011,k}^T ([A \mathbf{0}] + B K_k) \}
\end{aligned} \quad (27)$$

$$\begin{aligned}
P_{2,k+1} = & b_k^T M_u b_k + \gamma [P_{2,k} + \tilde{P}_{11,k} (B b_k + D \hat{x}_k) \\
& + \tilde{P}_{112,k} \hat{x}_{k+1} + ((B b_k + D \hat{x}_k)^T \tilde{P}_{012,k} \hat{x}_{k+1} \\
& + \hat{x}_{k+1}^T \tilde{P}_{021,k} (B b_k + D \hat{x}_k) + \hat{x}_{k+1}^T \tilde{P}_{022,k} \hat{x}_{k+1}) \\
& + (B b_k + D \hat{x}_k)^T \tilde{P}_{011,k} (B b_k + D \hat{x}_k)]
\end{aligned} \quad (28)$$

The iterative control law is composed of a proportional control term and a bias vector. If the load demand $\hat{x}_k = 0$ for $\forall k = 0, 1, \dots$, it becomes a standard LQR problem. The challenge of this paper comes from the dynamic load demand \hat{x}_k . To solve this problem, the bias term b_k is designed to confront with effects of the uncontrollable state \hat{x}_k . It should be pointed out that the bias term at time k is a function of \hat{x}_{k+1} . Moreover, (25) shows that the bias term is a function of \tilde{x}_{k+1} , which means that the optimal controller contains feedback and feedforward signals.

4. Implementation of the Proposed Algorithm

The recursive formula for parameters of the iterative value function is given in Section 3, but one still can hardly identify parameters of the EMS. Therefore, an online data-driven implementation is necessary.

To carry out the above online iterative ADP algorithm, it is required to design a parametric structure to approximate $\mu_k(x_k)$ and $V_k(x_k)$. The output of the parametric structure is formulated as

$$\tilde{F}(X; \omega) = \omega^T \sigma(X) \quad (29)$$

where $\sigma(\cdot) \in \mathbb{R}^l$, and l is the dimension of vector $\sigma(\cdot)$.

The ADHDP technique is adopted to implement the QOADP algorithm, which combines the system dynamics and iterative value [34]. The Q function for ADHDP is defined as

$$\begin{aligned}
Q_k(x_k, u_k) = & \omega_c^T \sigma_c(z_k) \\
= & V_k(F(x_k, u_k))
\end{aligned} \quad (30)$$

Inputs of the parametric structure of iterative Q function are x_k and u_k , and the activation function $\sigma_c(z_k)$ is defined as

$$\begin{aligned}
\sigma_c(z_k) = & [x_{1,k}^2 \ x_{1,k} x_{2,k} \ x_{1,k} x_{3,k} \ x_{1,k} u_k \ x_{2,k}^2 \ x_{2,k} x_{3,k} \\
& x_{2,k} u_k \ x_{3,k}^2 \ x_{3,k} u_k \ u_k^2 \ x_{1,k} \ x_{2,k} \ x_{3,k} \ u_k \ 1]^T
\end{aligned} \quad (31)$$

The input of the iterative control policy is x_k , and the activation function $\sigma_a(x_k)$ is defined as

$$\sigma_a(x_k) = [x_{1,k} \ x_{2,k} \ x_{3,k} \ 1]^T \quad (32)$$

The target of function Q is

$$\begin{aligned}
Q_{k+1}(x_k, u_k) = & x_k^T Q_k x_k + u_k^T R u_k \\
& + \tilde{Q}_k(\tilde{x}_{k+1}, \mu_k(\tilde{x}_{k+1}))
\end{aligned} \quad (33)$$

The error of Q function can be defined as

$$E_{c,k}(\omega_{c,k}) = (\tilde{Q}_k(x_k, u_k; \omega_c) - Q_{k+1}(x_k, u_k))^2 \quad (34)$$

and the parameter-update rule for $\omega_{c,k}$ is a gradient-based iteration given by

$$\omega_{c,k}|_{m+1} = \omega_{c,k}|_m - \alpha_c \frac{\partial E_{c,k}|_m}{\partial \omega_{c,k}|_m} \quad (35)$$

where α_c is the learning rate of the iterative Q function and m is the iterative step for updating $\omega_{c,k}$. Define the maximum number of iteration steps as \mathcal{M} ; then,

$$\omega_{c,k+1} = \omega_{c,k}|_{\mathcal{M}} \quad (36)$$

The target of the iterative control policy is given by

$$\mu_k(x_k) = \arg \min_u \{U(x_k, u_k, k) + \tilde{Q}_k(x_{k+1}, u)\}. \quad (37)$$

Additionally, error of the iterative control policy is

$$E_{a,k}(\omega_{a,k}) = (\tilde{\mu}_k(x_k; \omega_a) - \mu_k(x_k))^2 \quad (38)$$

Similarly, the weight update is

$$\omega_{a,k}|_{m+1} = \omega_{a,k}|_m - \alpha_a \frac{\partial E_{a,k-1}|_m}{\partial \omega_{a,k-1}|_m} \quad (39)$$

where α_a is the learning rate of the iterative control policy

$$\omega_{a,k} = \omega_{c,k-1}|_{\mathcal{M}} \quad (40)$$

In summary, the data-driven QOADP algorithm implemented by parametric structures is shown in Algorithm 1.

Algorithm 1: Data-driven QOADP algorithm

Initialization:

- 1: Collect data of an EMS
- 2: Choose an initial array of $\omega_{c,0}$, which ensures initial $Q_0(\cdot)$ to be positive semi-definite.
- 3: Choose the maximum time step T_{final} .

Iteration (Online):

- 4: Let $k = 0$, and let $Q_0(x_k, u_k) = Q_0(x_k, u_k; \omega_{c,0})$.
 - 5: Calculate $\mu_0(x_k)$ and $Q_1(x_k, u_k)$ by (37) and (33) ($k = 0$).
 - 6: Let $k = k + 1$.
 - 7: Calculate $\mu_k(x_k)$ and $Q_{k+1}(x_k, u_k)$ by (37) and (33) ($k \neq 0$).
 - 8: If $k > T_{\text{final}}$, go to next step. Otherwise, go to Step 6.
 - 9: **return**.
-

5. Hardware-in-Loop Experimental Verification

In this section, two cases of numerical experiments were performed to examine the proposed QOADP algorithm. Comparisons will also be given to prove the superiority of QOADP. In case 1, QOADP was carried out under the assumption that the building load demand and electricity rate are periodic. In case 2, the dual iterative Q-learning (DIQL) algorithm [21], the fuzzy periodic value iteration (FPVI) algorithm [27] and the QOADP algorithm were, respectively, applied to the load demand of high volatility. The advantages of QOADP were verified by comparing the systems states and electricity cost of the three EMSs. For the purpose of comparison, the dual iterative Q-learning (DIQL) algorithm [21]

and the fuzzy periodic value iteration (FPVI) algorithm [27] were also applied to the same load demand. Some important parameters for experiment can be found in Table 1.

Table 1. Parameters for experiment.

Parameters	Value
Capacity of battery in kWh	25
Upper bound of SOC	0.9
Lower bound of SOC	0.15
Rated power output of the battery in kW	3.5
Initial energy of battery in kWh	12.5
Discount factor	0.996

5.1. HIL Platform

The proposed QOADP algorithm was verified by an HIL experimental platform, as shown in Figure 3. The HIL platform was composed of an OPAL-RT OP5600 digital simulator as the controlled plant emulator and a dSPACE DS1103 as the controller. DS1103 is a controller board for digital control, which enables the connection between hardware and MATLAB/Simulink simulation environment. By Simulink, algorithms were implemented, compiled and converted to C-code, which was then loaded to the real-time dSPACE processor. The RT-LAB rapid control prototype (RCP) drive system loaded the smart building model, including the utility grid, battery and loads onto a real-time simulation platform configured with the I/O interface required for the study. MATLAB R2021a was the platform, and CPU was an 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz.

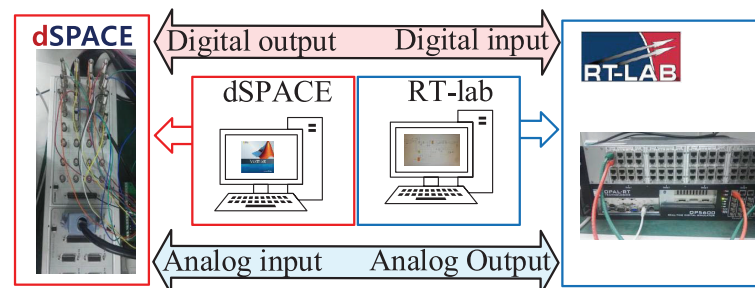


Figure 3. HIL experimental platform.

5.2. Dataset

The load demand data were taken from the IEEE Power-Energy Society (PES) Open Dataset. The load demand contained the active power of a single household in a family housing from 25 December 2011 to 15 March 2013. The sampling time was 1 h. Figure 4a shows the load demand of a smart building in a week, which has high volatility. In Figure 4b, the load demand in every single day is plotted. The bold line is the average daily load demand. The bold line shows the average load demand in a 24 h time horizon. The electricity price is consistent with the one in [13]. Figure 4c shows the electricity price. The electricity price is quasi-periodic, since it is determined by the aggregated load demand, which has a smaller fluctuation than a single building. Similarly, Figure 4d shows seven daily electricity prices ranging from 1 to 24, and the average electricity price is plotted by a bold line. It is assumed that users are not aware of the electricity price, which means that the load is not scheduled, and the target is to design a control strategy for the battery system.

5.3. Case 1: Periodic Load Demand

In this case, the load demand and electricity price are periodic. Without rescheduling of the energy storage system, the power cost over one week was 1999.2 cents.

The iterative weights of value function and controller are depicted in Figure 5. The learning process shows that weights converge to a periodic sequence. Since the iterative

algorithm has an exponential convergence rate, the initial weights have little influence on the performance.

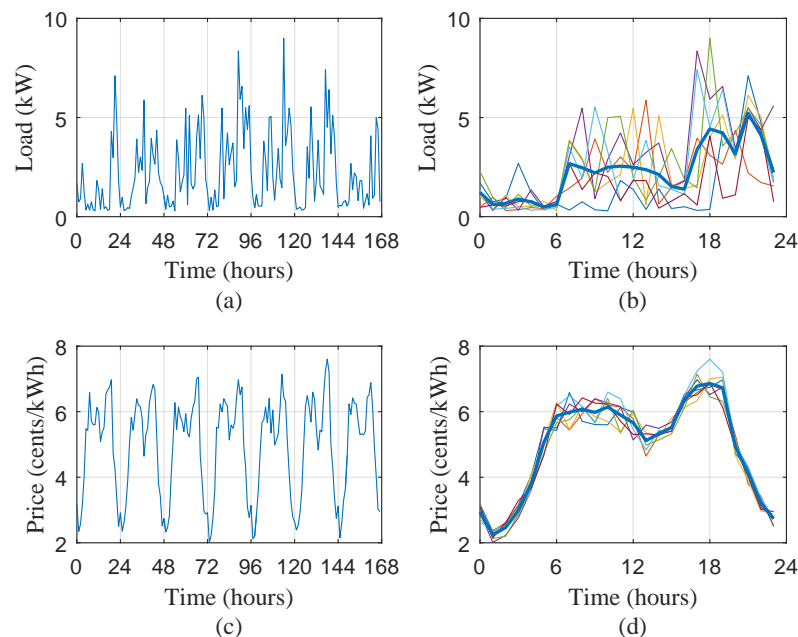


Figure 4. The weekly load demand and its average load curve. (a) Load demand of a smart building for one week. (b) Seven daily load demands and average load demand (bold line). (c) Electricity prices for one week. (d) Seven daily electricity prices and average electricity price (bold line).

The grid power is plotted in Figure 6a. Each of them works in a safe zone. The battery power is plotted in Figure 6b. Figure 6c shows the SOC of three EMSs. The battery charges when the electricity price is low at midnight and discharges in the daytime. Compared with QOADP, the maximum battery power levels of DIQL- and FPVI-based EMS are larger, which is not beneficial to battery life. Moreover, the total electricity cost of QOADP is less than those of DIQL and FPVI. Hence, QOADP has advantages in both economic costs and battery life degradation. QOADP-based EMS has higher grid power than DIQL and FPVI from 1 am to 6 am only when the electricity price is low.

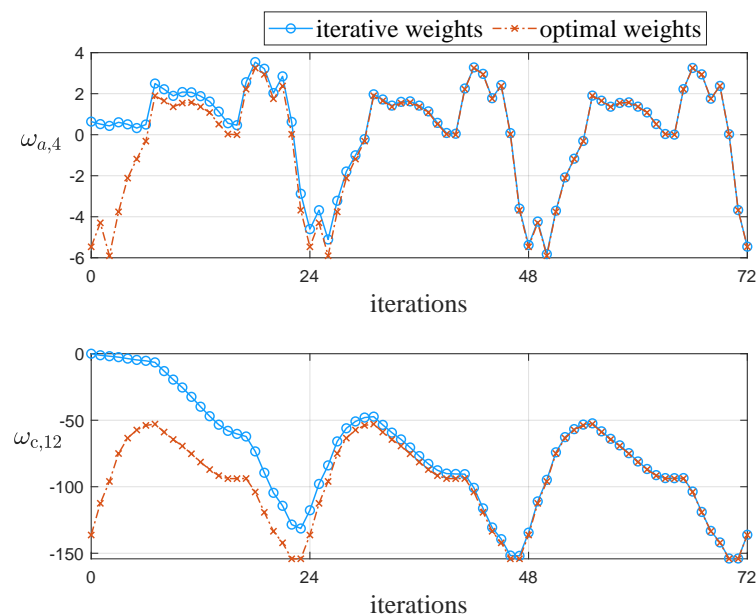


Figure 5. The iterative weights of the QOADP under periodic assumption.

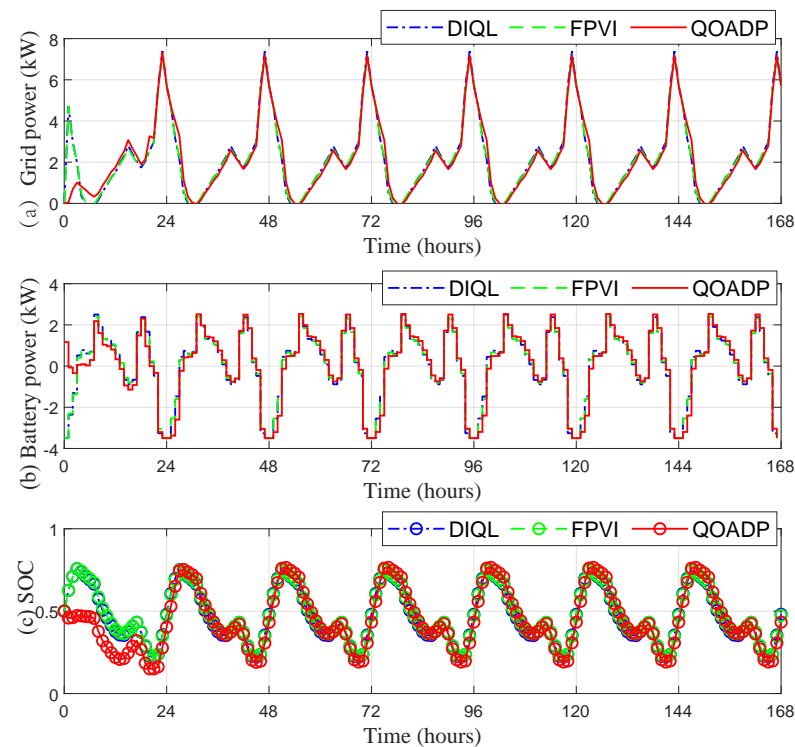


Figure 6. System state and control under periodic assumption: (a) Grid power; (b) Battery power; (c) SOC.

5.4. Case 2: Load Demand of High Volatility

Consider the load demand with high volatility shown in Figure 4a,c, which depicts the electricity price.

Figure 7 shows $\omega_{c,12}$ and $\omega_{a,4}$ of the iterative value function and iterative controller. Differently from the periodic case, weights will not converge to an array of parameters, but iterate towards the optimal. In Figure 7, optimal parameters are plotted under load demand of high volatility.

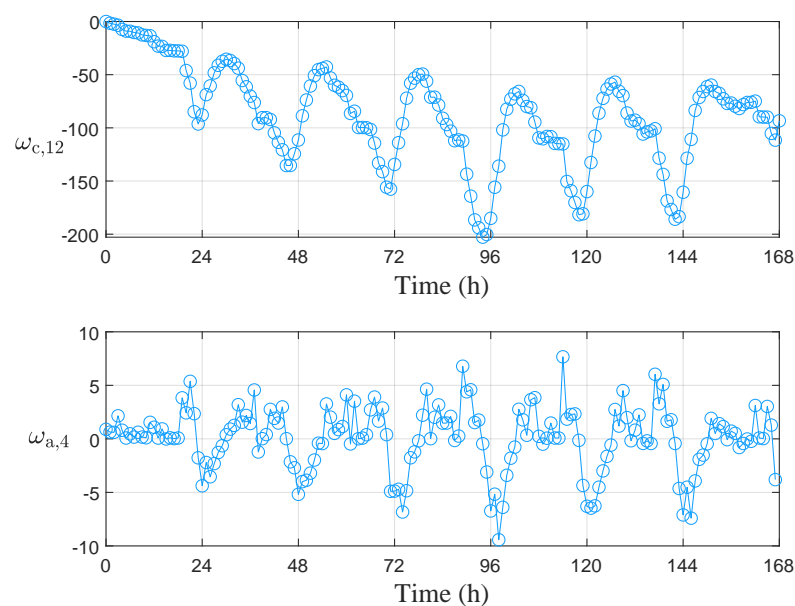


Figure 7. The iterative weights of parametric structures (load demand of high volatility).

The state and control variables of three EMSs are plotted in Figure 8. Similar to the first case, batteries charge when price is low and discharge when price is high. In QOAPD-based EMS, the control variable, i.e., P_B , can adapt to the real-time change in load demand. Hence, the battery power compensates the load demand in a good way. Furthermore, the grid power of QOAPD is more stable than those of DIQL and FPVI, which is a desirable feature in the utility grid.

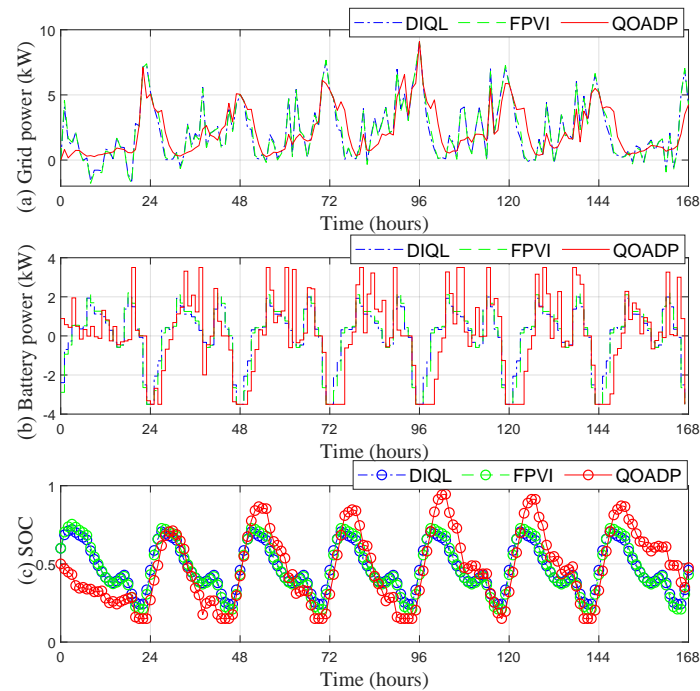


Figure 8. System state and control under load demand of high volatility: (a) Grid power; (b) Battery power; (c) SOC.

The electricity costs of three methods are listed in Figure 9. The proposed algorithm is not only able to adapt to the change of load demand, but also saves more electricity cost for residential users.

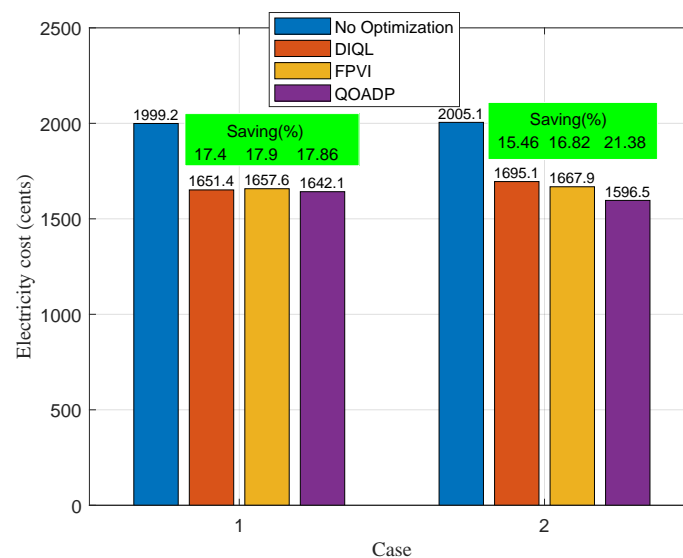


Figure 9. Electricity costs of three methods.

6. Conclusions

This paper proposes a QOAP algorithm to realize optimal economic dispatch in smart buildings. The load demand is allowed to be uncertain with high volatility, which is modeled by an uncontrollable state in the EMS. Firstly, the framework of iterative QOAP algorithm is designed to obtain optimal economic dispatch control law. Then, a quasi-quadratic-based function approximation structure with a bias term is utilized to approximate the value function precisely. Moreover, the implementation of the proposed algorithm is given out with detailed steps. The design of QOAP has paved the way for applying ADP technique to EMS of smart buildings and some other systems with uncertainties. Finally, experimental results reveal that QOAP not only can be applied to periodic load demand but also can be employed to load demand of high volatility with superior performance. Our future work will be concerned with expanding the proposed algorithm to more practical engineering systems and formulating a general way to apply ADP.

Author Contributions: Methodology, J.W.; Validation, Z.Z.; Investigation, K.C.; Data curation, Z.Z.; Writing-original draft, K.C.; Writing-review & editing, Z.Z. and J.W.; Project administration, J.W.; Funding acquisition, K.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded in part by the National Natural Science Foundation of China under grant 62103115, in part by the Natural Science Foundation of Guangdong Province under grants 2021A1515011636, in part by the Science and Technology Research Program of Guangzhou under grant 202102020975 and in part by the basic and applied basic research projects jointly funded by Guangzhou and schools (colleges) under grant 202201020233.

Data Availability Statement: Load demand data for the experiment can be found in IEEE Power-Energy Society (PES) Open Dataset at <https://open-power-system-data.org/> (accessed on 20 October 2022).

Conflicts of Interest: No conflict of Interest.

Abbreviations

i	iteration steps.
t, k	time steps (hours).
$P_G(t)$	Power from the utility grid at time t (kW).
$P_B(t)$	Battery discharging/charging power at time t (kW).
$P_L(t)$	Building load at time t (kW).
η	Charging/discharging efficiency of Power electronic converter in the storage system.
$SOC(t)$	State of charge of the battery at time t .
$E_{B,rate}$	Rated energy of battery (kWh).
SOC_{min}	Minimum value of SOC.
SOC_{max}	Maximum value of SOC.
$P_{B,rate}$	Battery rated discharging/charging power at time t (kW).
$g(\cdot)$	Transition function of building load (continuous time).
$\hat{g}(\cdot)$	Transition function of building load (discrete time).
τ	Sampling time.
\bar{x}_k	Controllable system state vector.
\hat{x}_k	Uncontrollable system state vector.
x_k	System state vector, $x_k \in \mathbb{R}^n$, $x_k = [\bar{x}_k^T \hat{x}_k^T]^T$.
u_k	System control vector, $u_k \in \mathbb{R}^m$.
z_k	$z_k = [x_k^T u_k^T]^T$
\underline{u}_k	control sequence from time k to ∞ .
$F(\cdot)$	System transition function.
A, B, D	Drift dynamics, input dynamics, disturbance dynamics.
$U(\cdot)$	Utility function.
λ_k	Real-time electricity price at time k (cents/kWh)
$J(\cdot)$	Performance index function.
γ	Discount factor, $0 < \gamma < 1$.

References

1. Zhou, B.; Zou, J.; Chung, C.Y.; Wang, H.; Liu, N.; Voropai, N.; Xu, D. Multi-microgrid Energy Management Systems: Architecture, Communication, and Scheduling Strategies. *J. Mod. Power Syst. Clean Energy* **2021**, *9*, 463–476. [\[CrossRef\]](#)
2. 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**, *8*, 12046–12063. [\[CrossRef\]](#)
3. Zhong, W.; Xie, K.; Liu, Y.; Yang, C.; Xie, S.; Zhang, Y. Online Control and Near-Optimal Algorithm for Distributed Energy Storage Sharing in Smart Grid. *IEEE Trans. Smart Grid* **2020**, *11*, 2552–2562. [\[CrossRef\]](#)
4. Giraldo, J.S.; Castrillon, J.A.; López, J.C.; Rider, M.J.; Castro, C.A. Microgrids Energy Management Using Robust Convex Programming. *IEEE Trans. Smart Grid* **2019**, *10*, 4520–4530. [\[CrossRef\]](#)
5. Zhong, W.; Xie, K.; Liu, Y.; Xie, S.; Xie, L. Chance Constrained Scheduling and Pricing for Multi-Service Battery Energy Storage. *IEEE Trans. Smart Grid* **2021**, *12*, 5030–5042. [\[CrossRef\]](#)
6. Zhang, G.; Yuan, J.; Li, Z.; Yu, S.S.; Chen, S.Z.; Trinh, H.; Zhang, Y. Forming a Reliable Hybrid Microgrid Using Electric Spring Coupled With Non-Sensitive Loads and ESS. *IEEE Trans. Smart Grid* **2020**, *11*, 2867–2879. [\[CrossRef\]](#)
7. Fan, L.; Wang, Y.; Wei, H.; Zhang, Y.; Zheng, P.; Huang, T.; Li, W. A GA-based online real-time optimized energy management strategy for plug-in hybrid electric vehicles. *Energy* **2022**, *241*, 122811. [\[CrossRef\]](#)
8. Zhang, L.; Li, Y. Optimal Energy Management of Wind-Battery Hybrid Power System With Two-Scale Dynamic Programming. *IEEE Trans. Sustain. Energy* **2013**, *4*, 765–773. [\[CrossRef\]](#)
9. Meng, X.; Li, Q.; Zhang, G.; Chen, W. Efficient Multidimensional Dynamic Programming-Based Energy Management Strategy for Global Composite Operating Cost Minimization for Fuel Cell Trams. *IEEE Trans. Transp. Electrification* **2022**, *8*, 1807–1818. [\[CrossRef\]](#)
10. Liu, D.; Xue, S.; Zhao, B.; Luo, B.; Wei, Q. Adaptive Dynamic Programming for Control: A Survey and Recent Advances. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 142–160. [\[CrossRef\]](#)
11. Lewis, F.L.; Vrabie, D. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits Syst. Mag.* **2009**, *9*, 32–50. [\[CrossRef\]](#)
12. Lewis, F.L.; Vrabie, D.; Vamvoudakis, K.G. Reinforcement Learning and Feedback Control: Using Natural Decision Methods to Design Optimal Adaptive Controllers. *IEEE Control. Syst. Mag.* **2012**, *32*, 76–105. [\[CrossRef\]](#)
13. Fuselli, D.; De Angelis, F.; Boaro, M.; Squartini, S.; Wei, Q.; Liu, D.; Piazza, F. Action dependent heuristic dynamic programming for home energy resource scheduling. *Int. J. Electr. Power Energy Syst.* **2013**, *48*, 148–160. [\[CrossRef\]](#)
14. Liu, D.; Xu, Y.; Wei, Q.; Liu, X. Residential energy scheduling for variable weather solar energy based on adaptive dynamic programming. *IEEE/CAA J. Autom. Sin.* **2018**, *5*, 36–46. [\[CrossRef\]](#)
15. Zhao, Z.; Keerthisinghe, C. A Fast and Optimal Smart Home Energy Management System: State-Space Approximate Dynamic Programming. *IEEE Access* **2020**, *8*, 184151–184159. [\[CrossRef\]](#)
16. Yuan, J.; Yu, S.S.; Zhang, G.; Lim, C.P.; Trinh, H.; Zhang, Y. Design and HIL Realization of an Online Adaptive Dynamic Programming Approach for Real-Time Economic Operations of Household Energy Systems. *IEEE Trans. Smart Grid* **2022**, *13*, 330–341. [\[CrossRef\]](#)
17. Shuai, H.; Fang, J.; Ai, X.; Tang, Y.; Wen, J.; He, H. Stochastic Optimization of Economic Dispatch for Microgrid Based on Approximate Dynamic Programming. *IEEE Trans. Smart Grid* **2019**, *10*, 2440–2452. [\[CrossRef\]](#)
18. Li, H.; Wan, Z.; He, H. Real-Time Residential Demand Response. *IEEE Trans. Smart Grid* **2020**, *11*, 4144–4154. [\[CrossRef\]](#)
19. Shang, Y.; Wu, W.; Guo, J.; Ma, Z.; Sheng, W.; Lv, Z.; Fu, C. Stochastic dispatch of energy storage in microgrids: An augmented reinforcement learning approach. *Appl. Energy* **2020**, *261*, 114423. [\[CrossRef\]](#)
20. Shuai, H.; He, H. Online Scheduling of a Residential Microgrid via Monte-Carlo Tree Search and a Learned Model. *IEEE Trans. Smart Grid* **2021**, *12*, 1073–1087. [\[CrossRef\]](#)
21. Wei, Q.; Liu, D.; Shi, G. A novel dual iterative Q-learning method for optimal battery management in smart residential environments. *IEEE Trans. Ind. Electron.* **2015**, *62*, 2509–2518. [\[CrossRef\]](#)
22. Wei, Q.; Liu, D.; Liu, Y.; Song, R. Optimal constrained self-learning battery sequential management in microgrid via adaptive dynamic programming. *IEEE/CAA J. Autom. Sin.* **2017**, *4*, 168–176. [\[CrossRef\]](#)
23. Wei, Q.; Shi, G.; Song, R.; Liu, Y. Adaptive Dynamic Programming-Based Optimal Control Scheme for Energy Storage Systems With Solar Renewable Energy. *IEEE Trans. Ind. Electron.* **2017**, *64*, 5468–5478. [\[CrossRef\]](#)
24. Wei, Q.; Liao, Z.; Song, R.; Zhang, P.; Wang, Z.; Xiao, J. Self-Learning Optimal Control for Ice-Storage Air Conditioning Systems via Data-Based Adaptive Dynamic Programming. *IEEE Trans. Ind. Electron.* **2021**, *68*, 3599–3608. [\[CrossRef\]](#)
25. Wei, Q.; Liao, Z.; Shi, G. Generalized Actor-Critic Learning Optimal Control in Smart Home Energy Management. *IEEE Trans. Ind. Inform.* **2021**, *17*, 6614–6623. [\[CrossRef\]](#)
26. Shi, G.; Liu, D.; Wei, Q. Echo state network-based Q-learning method for optimal battery control of offices combined with renewable energy. *IET Control. Theory Appl.* **2017**, *11*, 915–922. [\[CrossRef\]](#)
27. Zhu, Y.; Zhao, D.; Li, X.; Wang, D. Control-Limited Adaptive Dynamic Programming for Multi-Battery Energy Storage Systems. *IEEE Trans. Smart Grid* **2019**, *10*, 4235–4244. [\[CrossRef\]](#)
28. Al-Tamimi, A.; Lewis, F.L.; Abu-Khalaf, M. Discrete-Time Nonlinear HJB Solution Using Approximate Dynamic Programming: Convergence Proof. *IEEE Trans. Syst. Man Cybern. Part (Cybernetics)* **2008**, *38*, 943–949. [\[CrossRef\]](#)
29. Wang, D.; Liu, D.; Wei, Q.; Zhao, D.; Jin, N. Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming. *Automatica* **2012**, *48*, 1825–1832. [\[CrossRef\]](#)

30. Liu, D.; Wei, Q. Policy Iteration Adaptive Dynamic Programming Algorithm for Discrete-Time Nonlinear Systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2014**, *25*, 621–634. [[CrossRef](#)]
31. Yang, X.; Zhang, Y.; He, H.; Ren, S.; Weng, G. Real-Time Demand Side Management for a Microgrid Considering Uncertainties. *IEEE Trans. Smart Grid* **2019**, *10*, 3401–3414. [[CrossRef](#)]
32. Wei, Q.; Lewis, F.L.; Liu, D.; Song, R.; Lin, H. Discrete-Time Local Value Iteration Adaptive Dynamic Programming: Convergence Analysis. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *48*, 875–891. [[CrossRef](#)]
33. Xue, S.; Luo, B.; Liu, D. Event-Triggered Adaptive Dynamic Programming for Zero-Sum Game of Partially Unknown Continuous-Time Nonlinear Systems. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *50*, 3189–3199. [[CrossRef](#)]
34. Liu, D.; Wei, Q.; Wang, D.; Yang, X.; Li, H. *Adaptive Dynamic Programming with Applications in Optimal Control*; Springer: Berlin/Heidelberg, Germany, 2017.