energy storage and applications

MDPI

Article

Flexible Mobile Battery Energy Storage System Control Considering Traffic Congestion Risk

Zifan Liu [1,†], Jinglin Yu [2,†], Huan Zhao [3,*], Yuheng Cheng [2], Xuanang Gui [2] and Junhua Zhao [2]

[1] Department of Electronic and Computer Engineering, School of Engineering, Hong Kong University of Science and Technology, Hong Kong SAR 999077, China; zliuft@connect.ust.hk
[2] School of Science and Engineering, The Chinese University of Hong Kong–Shenzhen, Shenzhen 518172, China; 121090729@link.cuhk.edu.cn (J.Y.); yuhengcheng@link.cuhk.edu.cn (Y.C.); xuananggui@link.cuhk.edu.cn (X.G.); zhaojunhua@cuhk.edu.cn (J.Z.)
[3] Department of Building Environment and Energy Engineering, The Hong Kong Polytechnic University, Hong Kong SAR 999077, China
* Correspondence: huan-paul.zhao@polyu.edu.hk
† These authors contributed equally to this work.

Abstract

The volatility of renewable energy generation and nodal electricity prices provides an arbitrage opportunity for Mobile Battery Energy Storage Systems (MBESS) leveraging both temporal and spatial advantages. However, the inherent high complexity and strong randomness of both power and transportation systems lead to complex risks for MBESS control. Existing works mainly consider the market price risk and ignore the transportation system risk caused by traffic congestion. Specifically, they are constrained by two critical limitations: (1) decisions can only be made upon arrival at a destination, making the agent unresponsive on the road, and (2) traffic congestion risk is neither quantified nor controlled, leading to suboptimal routing strategies. To address these limitations, the MBESS needs more flexible "on the road" decision making and multiple risk management capabilities. Guided by this objective, a flexible deep reinforcement learning-based MBESS control framework is proposed, considering both market and traffic congestion risk. First, dynamic routing ability is integrated with the MBESS agent to provide more flexibility in making decisions, regardless of whether the agent has reached the designated location or not. Second, two risk metrics are proposed to quantitatively assess the traffic congestion risk based on moving time, and then the agent can make decisions considering both market and traffic congestion risk. Finally, considering the inefficiency of learning caused by introducing multiple risks, a risk curriculum learning method is proposed to improve the training efficiency and reduce learning costs. These components are unified in the Multiple Risk Estimation SDDPG (MRE-SDDPG) algorithm, which jointly maximizes profitability while controlling electricity price and traffic congestion risk. Simulations in the IEEE 30 bus environment show that the proposed framework can increase profit by 8.6% while reducing the traffic time by 15.8% on average, demonstrating the superiority of our design in considering traffic congestion risk.

Keywords: mobile battery energy storage system; deep reinforcement learning; dynamic routing ability; traffic congestion risk; risk curriculum learning

Check for updates

Academic Editor: Kyeon Hur

Received: 1 March 2026
Revised: 9 May 2026
Accepted: 5 June 2026
Published: 11 June 2026

Copyright: © 2026 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.

1. Introduction

The global energy landscape has recently undergone a paradigm shift due to the growing recognition of the environmental challenges associated with traditional energy

sources, such as air pollution and greenhouse gas emissions [1,2]. Consequently, the adoption of renewable energy sources, especially solar and wind power, has gained significant attention [3]. While renewable energy technologies present a sustainable alternative, they also introduce greater volatility into the power system, increasing uncertainty in electricity prices and power system stability concerns [4]. To address this issue, Battery Energy Storage Systems (BESS) have become a promising solution for enhancing the efficiency and reliability of smart grids [5]. BESS provides the ability to store excess renewable energy produced during peak generation times and release it during periods of low production [6]. It is essential for balancing fluctuations in supply and demand [7], improving grid stability [8], and promoting renewable energy source integration [9]. However, conventional BESS faces certain limitations, primarily related to its stationary nature [10]. For example, a stationary BESS has limited economic benefits [11] as it has to be on standby during the off-peak seasons. This limitation motivates researchers to incorporate mobility into BESS by integrating BESS into vehicles, such as trucks, and is named Mobile BESS (MBESS) [12]. The enhanced mobility allows MBESS to be easily transported and deployed in various locations, making it a suitable solution for addressing grid imbalances while also contributing to improving power quality [13,14].

Since the electricity market has witnessed higher price volatility caused by renewable energy, this volatility has prompted the emergence of profit opportunities from market arbitrage with MBESS [15,16]. Arbitrage in the electricity market requires participants to be sensitive to high-frequency changes in electricity prices at different locations [17], as it involves buying low and selling high to maximize profits, considering the moving costs. Traditional model-based control methods, such as Model Predictive Control (MPC) [18], first formulate this problem into a mixed-integer problem and address it using an optimization solver. Nonetheless, these approaches are impractical due to the expensive real-time computation costs and the necessity for precise system parameters [19,20]. As a result, Deep Reinforcement Learning (DRL), the data-driven control method, has gained attention as a more suitable approach for optimizing MBESS operations in arbitrage scenarios [21–23]. DRL leverages advanced deep learning and reinforcement learning techniques to learn optimal control policies, enabling it to be adapted to dynamic electricity price fluctuations in real time.

Although existing DRL-based MBESS control methods, such as [24], demonstrate a great real-time control ability, one important issue in these works is that decisions can only be made after the specified destination is reached, resulting in a time delay in making decisions. For example, the MBESS decides to move to node 4 from node 1, and traffic congestion occurs on the road after the move starts. The DRL agent needs to reach node 4 and then make the next decision due to the problem formulation. This delay limits the agent's responsiveness to dynamic environments, hindering its ability to optimize the overall profits. Thus, the ability to make decisions "on the road" can greatly enhance the system's adaptability to complex environments in practical scenarios.

Beyond maximizing profits in real time, managing multiple risks is also an important concern in MBESS arbitrage. Existing DRL methods [21,22,25] can maximize the average profit but fail to control the arbitrage risk caused by different factors, such as renewable energy, load, and traffic. To address the risk of electricity price, our previous work [24] introduced Conditional Value at Risk (CVaR) of charging/discharging decisions as charging/discharging risk, which has been shown to be superior in optimization applications due to the convexity property of CVaR. However, considering traffic congestion risk is not as intuitive as electricity price since it involves a comprehensive analysis of multiple factors, including traffic flow, vehicle capacity, peak hours, and environmental conditions [26].

Recognizing these limitations, an MBESS control framework with more flexible decision making and multiple risk management abilities is needed.

Learning from scratch in Mobile Battery Energy Storage Systems (MBESS) poses significant challenges, primarily due to the extensive environmental interactions, which can lead to inefficiencies and lengthy training times [27]. This is particularly problematic in MBESS, where data collection is often costly and time-consuming. Furthermore, the complexity of traffic risk compared to electricity price risk exacerbates these inefficiencies, making it difficult to learn effective policies simultaneously. In many domains, a pretraining procedure is needed in simulated environments where the cost of samples is virtually nonexistent. For example, in robotic learning, directly training in the real world from scratch is typically infeasible, but it is often possible to obtain a simulated environment and pretrain the robot on it to obtain an approximate model of the real-world environment [28].

To overcome the three above-identified challenges—inflexible destination-only decision making, the absence of explicit traffic congestion risk quantification, and the high cost of multi-risk learning from scratch—this paper proposes a flexible, risk-sensitive DRL-based MBESS control framework that addresses all three limitations in a unified architecture. The contributions of this paper are as follows:

1. To improve control flexibility, a dynamic routing module (DRM) is proposed that decomposes each journey into intermediate waypoints and enables the agent to revise its routing decision whenever timely arrival at the original destination is not expected to enable more flexible moving ability.
2. To quantify and control traffic congestion risk, two CVaR-based traffic risk metrics (the CVaR of total travel time and the CVaR of late time) are proposed, together with the traffic congestion risk estimation module (TREM) to estimate these defined metrics accurately. These elements are incorporated into the MRE-SDDPG algorithm to jointly control both electricity price risk and traffic congestion risk.
3. To improve training efficiency in the multi-risk setting, a risk curriculum learning (RCL) strategy is proposed that progressively introduces risk factors, starting from a risk-free baseline and sequentially adding charging/discharging and traffic congestion risks, substantially reducing the training costs to convergence.

2. Background

2.1. Mobile Battery Energy Storage System Model

To formulate the real-world MBESS accurately, the MBESS model should encompass the parameterization of the power system, BESS, transportation system, and vehicle mobility model. Since existing works [15,21,24] have already established sufficient model formulation, this paper will only mention the transportation system model and the vehicle mobility model, which are directly related to the traffic congestion risk. As for models of other systems, this paper follows the previous formulation in [24] without modification.

2.1.1. Transportation System Model

The transportation model consists of original and destination nodes (O-D pairs) in MBESS. Each O-D pair is assumed to have one road. The travel time along each road is determined by the volume of vehicles and baseline travel time, which is the travel time with no cars on the road [24,29]. As shown in Equation (1), the traffic volume between n and m at time t is denoted as V^{tra}_{nmt}. The uncertainty of the traffic volume is modeled with a normal distribution, with mean μ^{tra}_{nmt} and standard deviation $\sigma^{\text{tra}}_{nmt}$.

$$V^{\text{tra}}_{nmt} \sim N\left(\mu^{\text{tra}}_{nmt}, (\sigma^{\text{tra}}_{nmt})^2\right). \tag{1}$$

where μ^{tra}_{nmt} represents the average traffic volume from location n to m at t, and $\sigma^{\text{tra}}_{nmt}$ denotes the corresponding standard deviation. The traveling time between two nodes is estimated based on Equation (2), where the real-time travel time T^{mov}_{nmt} is influenced by the baseline travel time T^{0}_{nm} and current traffic conditions related to the road capacity C^{tra}_{nm} and real-time traffic volume V^{tra}_{nmt}, with χ and ψ serving as adjustment coefficients.

$$T^{\text{mov}}_{nmt} = T^{0}_{nm} * \left[1 + \chi\left(\frac{V^{\text{tra}}_{nmt}}{C^{\text{tra}}_{nm}}\right)^{\psi}\right]. \tag{2}$$

2.1.2. Vehicle Mobility Model

The vehicle mobility framework utilizes a series of constraints in Equations (3) to (9) to formulate the movement of the vehicle correctly. These include variables representing arrivals α_{nt}, departures β_{nt}, and movement γ_{nmt}, each constrained to binary values. Equations manage the logical conditions and physical limitations of traffic movement:

$$\sum_{n} \omega_{nt} \leq 1 - \sum_{n}\sum_{m} \gamma_{nmt}. \tag{3}$$

$$\alpha_{nt} - \beta_{nt} = \omega_{nt} - \omega_{n(t-1)}. \tag{4}$$

$$\sum_{n} (\alpha_{nt} + \beta_{nt}) \leq 1. \tag{5}$$

$$\sum_{m} \gamma_{nmt} \geq \beta_{nt}. \tag{6}$$

$$\alpha_{mt} - \theta_{mt} = \sum_{n} \left(\gamma_{nm(t-1)} - \gamma_{nmt}\right). \tag{7}$$

$$\sum_{n} (\alpha_{nt} + \theta_{nt}) \leq 1. \tag{8}$$

$$\gamma_{nmt} \geq \gamma_{nm(t-1)} - \gamma_{nm\left(t - T^{\text{mov}}_{nmt}/T_p\right)}. \tag{9}$$

where $\alpha_{nt} \in \{0,1\}, \beta_{nt} \in \{0,1\}, \gamma_{nmt} \in \{0,1\}, \omega_{nt} \in \{0,1\}, \theta_{nt} \in \{0,1\}$.

2.2. Risk Signal

In the realm of risk control, various measures have been developed to assess risks in diverse domains. Commonly utilized risk assessment measures, such as variance and the Sharpe ratio, have proven effective in quantifying risk but are limited in capturing one-sided risks. To address this issue, Value at Risk (VaR) and Conditional Value at Risk (CVaR) have gained prominence in risk assessment practices [30]. They facilitate the assessment and measurement of the level of single-tailed risk, offering a broader view of one-sided risk analysis. First, the VaR of a random variable Y is defined as

$$o^{\alpha} = \begin{cases} \arg_y F_Y(y) = 1 - \alpha, \text{ for left tail risk;} \\ \arg_y F_Y(y) = \alpha, \text{ for right tail risk.} \end{cases} \tag{10}$$

where $F_Y(y)$ denotes the cumulative distribution function of random variable y, and α represents the confidence level.

While VaR indicates the threshold beyond which a certain percentage of outcomes will fall, it does not provide any insight into the severity of outcomes beyond this threshold [31]. This limitation is addressed by CVaR, which gives a more comprehensive assessment by calculating the conditional expectation of the tail beyond the VaR threshold (o^{α}), effectively measuring the expected severity of extreme losses.

The formulation of CVaR is represented as

$$\phi^{\alpha} = \begin{cases} \frac{1}{1-\alpha}\int_0^{1-\alpha} o^{\alpha} d\alpha, & \text{for left tail risk;} \\ \frac{1}{1-\alpha}\int_{\alpha}^{1} o^{\alpha} d\alpha, & \text{for right tail risk.} \end{cases} \tag{11}$$

and the estimation version is

$$\phi^{\alpha} \approx \begin{cases} \frac{1}{(1-\alpha)N}\sum_{i=1}^{(1-\alpha)N} y_i, & \text{for left tail risk;} \\ \frac{1}{(1-\alpha)N}\sum_{i=\alpha N}^{N} y_i, & \text{for right tail risk.} \end{cases} \tag{12}$$

where y_i is the ith sample of a random variable y arranged in a certain order.

2.3. Risk-Sensitive Reinforcement Learning

Risk-sensitive reinforcement learning (RS-RL) emphasizes decision making in dynamic environments by taking into account the risks throughout the learning procedure. Effectively managing these risks is essential for various real-world applications, including autonomous driving, financial investments, and smart grids. Considering a Constrained Markov Decision Process (CMDP), denoted by (S, A, r, c, P), where S is the state space, A is the action space, r and c are reward and cost functions, and P is the transition function, the objective in RS-RL is formulated as

$$\max_{a \sim A} \sum_{t=1}^{T} \gamma^t r(s,a) + \beta \sum_{t=1}^{T} \gamma^t c(s,a), \tag{13}$$

where β is the risk parameter: $\beta > 0$ for the risk-seeking case and $\beta < 0$ for the risk-averse case.

In MBESS, reference [24], first applies RS-RL to MBESS, where the electricity price fluctuations are considered as risk. Although this design slightly improves the safety of arbitrage with MBESS, it ignores the risks brought by traffic uncertainty, leading to suboptimal strategies.

3. Control Framework

3.1. Overall Structure

3.1.1. Control Framework with Traffic Congestion Risk and Dynamic Routing

In MBESS, the dynamic routing and traffic congestion risk are significant considerations due to the potential profit reduction caused by the uncertainty of transportation, such as late arrival. Therefore, this paper proposes an MBESS control framework in the dynamic and uncertain transportation system to (1) dynamically change moving decisions and (2) measure and control the traffic congestion risk. The three major assumptions for the framework are as follows. First, due to the randomness of the transportation system, the transportation time within the system is very uncertain. Second, due to the first assumption, the decision may not be optimal before arriving and thus needs more real-time and flexible movement control. Third, MBESS does not have direct access to risk measurement, nor does it have accurate knowledge outside the system to measure risk, especially traffic risk.

Figure 1 demonstrates the overall control framework, which contains the real-time control module (RCM), dynamic routing module (DRM), and risk estimation module (REM).

The RCM is mainly responsible for making decisions and updating policies. According to real-time state information, it can obtain charging/discharging decisions a_c or moving decisions a_m from the policy and update itself based on the reward and risk.

After making the moving decision a_m, DRM can obtain the shortest path (with intermediate point) from the shortest path algorithm and estimate the expected minimum moving time according to the moving time distribution estimation model P_{nmt}^{mov} in REM. At

each time step, DRM will update this time and determine whether to maintain the original decision a_m or set the nearest intermediate point as the new destination to let RCM update the decision.

REM consists of a charging risk estimation module (CREM) and a traffic congestion risk estimation module (TREM). CREM is constructed based on [24]. At each time step, TREM collects the moving time from the starting point and destination into the moving time buffer and updates P^{mov}_{nmt} when used. During the training step, a mini-batch of MBESS transitions is randomly sampled from the predefined replay buffer and sent to REM to obtain the charging/discharging risk from CREM and traffic congestion risk from TREM. Then RCM is updated with these experiences.

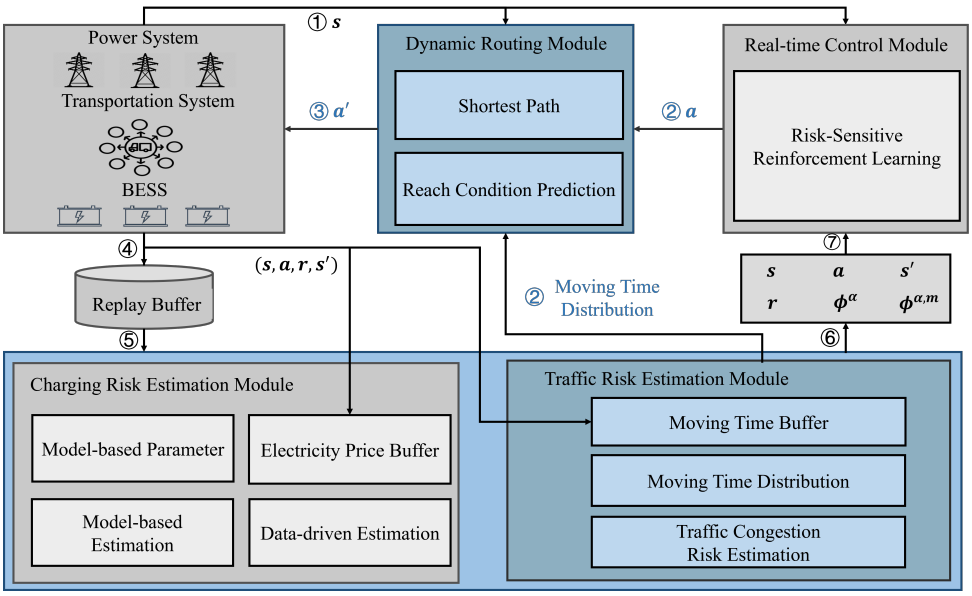


Figure 1. The MBESS control framework with traffic congestion risk and dynamic routing module. Modules and texts in blue color are the main contribution of this framework.

3.1.2. Problem Reformulation

Existing formulation in most DRL-based MBESS models only allows the agent to make the next decision after the current movement is completed, which is overly restrictive and rough for the model control, resulting in suboptimal performance. This formulation usually leads to a situation where the made movement decision is no longer optimal as the electricity price changes or when traffic congestion happens. In this situation, the agent cannot calibrate the policy in time, which greatly decreases the performance. To address the above problems, a new formulation of the vehicle mobility model for MBESS is extended to support dynamic routing ability for higher robustness.

The new formulation adds a logic Equation (14) to support dynamic routing. Equation (14) represents the vehicle arrives at the nearest intermediate place m' if it leaves place n at the former time step but does not arrive at place m at the current time step.

$$\alpha_{m't} = 1 \text{ if } \beta_{n(t-1)} = 1 \wedge \alpha_{mt} = 0. \tag{14}$$

Since the logic operation is not easy to solve, we formulate a new Equation (15) to represent the logic Equation (14).

$$\alpha_{m't} = (1 - \alpha_{mt}) * \beta_{n(t-1)}, \tag{15}$$

where $\alpha_{mt} \in \{0, 1\}, \beta_{nt} \in \{0, 1\}$.

To support the dynamic routing ability mentioned above, a modified MDP process is as follows:

State: Similar to the existing formulation, the state is formulated as the departure n and destination m, the current time t, and the quantity of electricity in the battery E_t. Due to the randomness of MBESS, at the next decision time, the vehicle may not get to the destination, and the decision may be changed in the intermediate waypoint, which is defined as m'. That means that if the vehicle can reach its destination before the next decision time, $m' = m$; otherwise, m' is the nearest intermediate waypoint that the vehicle can reach at the next time step. Thus, the state is modified to $s = (n, m', t, E_t)$, where m' is changeable based on traffic conditions.

Action: In the existing formulation, action a is divided into two categories. First, the discrete moving action consists of $a_m = \{0 \leq a_m \leq N, a_m \in \mathbb{N}\}$, where a_m indicates the target place index when $a_m \leq N-1$, and $a_m = N$ means the agent will charge/discharge. Second, the continuous charging action is $a_c = \left\{-P^{\text{mdis}} \leq a_c \leq P^{\text{mcha}}, a_c \in \mathbb{R}\right\}$, where $a_c < 0$ indicates the charging action and $a_c > 0$ indicates the discharging action.

When the vehicle cannot reach the original destination m due to complex road conditions, it will go to the nearest intermediate waypoint m' that the agent can reach at the next time step. Then, a new destination $a_m = m''$ will be selected as the moving action by RCM at m'. Compared with the previous formulation of the action, the formulation of the decision in this paper is no longer limited to the arrival of the destination, but can be made at every decision moment. This new formulation can improve the flexibility of the agent and thus has better profit maximization or traffic congestion risk control ability.

Reward: Similar to existing work, the reward consists of the revenue, degradation cost, and transportation cost.

$$\mathrm{r}(s_t, a_t) = \lambda_{nt}(P_{nt}^{\text{dis}} - P_{nt}^{\text{cha}})T_p - c^{\text{deg}}(q + (P_{nt}^{\text{dis}} + P_{nt}^{\text{cha}})T_p) - c^{\text{tra}}\gamma_{nmt}T_p. \tag{16}$$

Cost/Risk: Different from the existing formulation that only considers the electricity price risk, this paper also introduces the concept of traffic congestion risk $\phi_{nmt}^{\alpha,\text{mov}}$. The cost is considered as

$$\mathrm{c}(s_t, a_t) = \phi_{nt}^{\alpha,\text{cha}} + \phi_{nmt}^{\alpha,\text{mov}} + \phi_{nt}^{\alpha,\text{dis}}, \tag{17}$$

where $\phi_{nmt}^{\alpha,\text{mov}}$is the CVaR of the traffic congestion due to agent's movement action at confidence level α and $\phi_{nt}^{\alpha,\text{cha}}/\phi_{nt}^{\alpha,\text{dis}}$ is the similar risk of the agents charging/discharging action defined by Equation (18)/(19). The details of $\phi_{nmt}^{\alpha,\text{mov}}$ will be illustrated in Section 3.3.

$$\phi_{nt}^{\alpha,\text{cha}} = \frac{1}{(1-\alpha)N}\sum_{i=\alpha N}^{N} y_{i,c}, \tag{18}$$

where $y_{i,c}$ is the i^{th} charging sample arranged in ascending order.

$$\phi_{nt}^{\alpha,\text{dis}} = \frac{1}{(1-\alpha)N}\sum_{i=1}^{(1-\alpha)N} y_{i,d}, \tag{19}$$

where $y_{i,d}$ is the i^{th} discharging sample arranged in ascending order.

3.1.3. Multiple Risk Estimation SDDPG

To cope with the reformulated MDP, we improve the RE-SDDPG [24] by incorporating dynamic routing and traffic congestion risk into it. As the new algorithm can not only control the electricity price risk but also traffic congestion risk, it is named Multiple Risk

Estimation SDDPG (MRE-SDDPG), as shown in Algorithm 1. Different from RE-SDDPG, MRE-SDDPG makes the following upgrades:

Algorithm 1 MRE-SDDPG algorithm.

1: Initialize network $\mu(s|\theta^{\mu})$, $Q_r(s,a|\theta^{Q_r})$, $Q_c(s,a|\theta^{Q_c})$, $\mu'(s|\theta^{\mu'})$, $Q'_r(s,a|\theta^{Q'_r})$, $Q'_c(s,a|\theta^{Q'_c})$
2: Initialize replay buffer $\mathbb{B}^r$, price buffer $\mathbb{B}^p$ with capacity c^p, time buffer $\mathbb{B}^t$ with capacity c^t.
3: **for** episode $= 1, \cdots,$ M **do**
4: Reset the initial state s_0
5: **for** $t = 1, \ldots, T$ **do**
6: Get CHARGING ACTION:
7: $a_{c,t} = \mu(s_t|\theta^{\mu}) + N_t^n$.
8: $Q = (1-\beta)Q_r(s_t, a_{c,t}|\theta^{Q_r}) + \beta Q_c(s_t, a_t|\theta^{Q_c})$.
9: Get MOVING ACTION:
10: $$a_{m,t} = \begin{cases} \arg\max Q & \text{w.p. } 1-\epsilon \\ \text{rand}(N+1) & \text{w.p. } \epsilon \end{cases}.$$
11: **if** $a_{m,t} == N$ **then**
12: Execute $a_{c,t}$ and store price $\lambda_{n,t}$ into $\mathbb{B}^p$.
13: Obtain r_t and s_{t+1}.
14: Store transition $(s_t, a_{c,t}, r_t, s_{t+1})$ in $\mathbb{B}^r$.
15: **else**
16: Execute $a_{m,t}$.
17: Get P_{nmt}^{mov} and path $p_{e,t} = D(a_{m,t}, s_t)$.
18: **while** on the road **do**
19: **if** the agent cannot reach on time **then**
20: Stop at the nearest waypoint m'.
21: Set $a_{m,t} = m'$.
22: **end if**
23: **end while**
24: Store the moving time into the time buffer $\mathbb{B}^t$.
25: Obtain r_t and s_{t+1}.
26: Store transition $(s_t, a_{m,t}, r_t, s_{t+1})$ in $\mathbb{B}^r$.
27: **end if**
28: **end for**
29: Randomly sample a mini-batch of n experiences from $\mathbb{B}^r$.
30: Estimate the corresponding risk ϕ_i^{α}.
31: Merge the transitions to $(s_i, a_i, r_i, \phi_i^{\alpha}, s_{i+1})$.
32: Set $y_i = r_i + \gamma Q'_r(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'}))$.
33: $L_{Q_r} = \frac{1}{n}\sum(y_i - Q_r(s_i, a_i|\theta^{Q_r}))^2$.
34: Set $z_i = \phi_i^{\alpha} + \gamma Q'_c(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu}))$.
35: $L_{Q_c} = \frac{1}{n}\sum(z_i - Q_c(s_i, a_i|\theta^{c}))^2$.
36: Update Q_c and Q_r.
37: Update μ using the sampled policy gradient:
38: $\nabla_{\theta^{\mu}} J \approx \frac{1}{n}\sum \nabla_a((1-\beta)Q_r(s_i, a_i|\theta^{Q_r}) - \beta Q_c(s_i, a_i|\theta^{Q_c}))|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^{\mu}} \mu(s|\theta^{\mu})|_{s_i}$.
39: Update the target networks Q'_r, Q'_c and μ'.
40: **end for**

- MRE-SDDPG includes DRM from line 18 to line 23. When the agent cannot reach on time, it will stop at the nearest waypoint that can be reached at the next time step as the current transition. Then it can remake the decision at the next time step to achieve dynamic routing ability.
- MRE-SDDPG will collect the information needed by TREM and also estimate the traffic risk, as shown in lines 24 and 30.
- RE-SDDPG directly collects the risk into the replay buffer, which is not accurate as the risk in the history transition is not updated. As shown in line 30, MRE-SDDPG

re-estimates the risk based on the transition information at each update time to make sure the risk is estimated by the latest estimation module.

3.2. Dynamic Routing Module

In MBESS, the moving control is usually staged, meaning that the next decision can only be made at the next time after arriving at the destination. This staging significantly limits the decision-making capacity, complicating the ability to make optimal choices in complex and variable environments. Furthermore, in real-world cases, it is more inclined to move closer to real-time decision making to adapt to changing conditions. To address these limitations, a dynamic routing module (DRM) is proposed to relax the constraints on decision making by allowing the vehicle to adjust its moving decisions at intermediate waypoints.

DRM decomposes the shortest path to the destination into several intermediate waypoints and allows the moving decision to be changed at the intermediate waypoint. In particular, DRM first obtains the moving-time distribution P^{mov}_{nmt} from TREM and obtains the expected time to each point along the shortest path according to the current environment information and the shortest path algorithm. Then, DRM determines whether the agent can reach the destination at the expected time step based on P^{mov}_{nmt}. If not, DRM will let the agent stay at the nearest intermediate waypoint m' that can be reached at the next time step and make a new moving decision m''.

3.3. Traffic Congestion Risk Estimation Module

Existing MBESS control frameworks only include the charging/discharging risk but do not consider traffic congestion risk, which can lead to suboptimal decisions. To address this issue, the traffic risk estimation module (TREM) is proposed to accurately define and quantitatively estimate the traffic congestion risk.

3.3.1. Moving Time Distribution Model

To accurately estimate the moving time distribution P^{mov}_{nmt} between any two places at any time, TREM creates a moving time buffer $\mathbb{B}^t$ that stores the latest N samples for each starting point, destination, and time step and then estimates P^{mov}_{nmt} based on these samples. This distribution model is utilized by DRM to predict whether the agent can reach the destination on time.

3.3.2. Traffic Congestion Risk Definition

To represent the distribution of the moving time, we first create a moving time buffer to store the moving time between the two points at time t, denoted as $\mathbb{B}^t[nmt] = \{y^0_{nmt}, ..., y^i_{nmt}, ..., y^j_{nmt}, ... | \forall i < j, y^i_{nmt} < y^j_{nmt}\}$, where y are the moving time samples in ascending order. Based on the moving time buffer, two kinds of traffic congestion risk can be defined. First, it can be formulated as the CVaR of the total time:

$$\phi^{\alpha,\text{mov}}_{nmt} \approx \frac{1}{(1-\alpha) * N} \sum_{i=\alpha N}^{N} y^i_{nmt}, \tag{20}$$

where y^i_{nmt} is the i^{th} moving sample arranged in ascending order.

This definition provides a global perspective of traffic congestion, which is a straightforward measure of moving risk. However, this definition ignores the situation where the long moving time is not caused by congestion, but by other cases, like long intermediate

routes. To overcome these shortcomings, the risk can also be formulated as the CVaR of the late time:

$$\phi_{nmt}^{\alpha,\text{mov}} \approx \frac{1}{(1-\alpha)*N} \sum_{i=\alpha N}^{N} (y_{nmt}^{i} - \bar{y}_{nmt}), \tag{21}$$

where $\bar{y}_{nmt} = \frac{1}{N}\sum_{i=1}^{N} y_{nmt}^{i}$.

One thing that needs to be mentioned is that the cost/risk value function under late time is not equivalent to the subtraction of the cost/risk value function under total time and a constant. The key difference is that the expected time is different for each n, m, and t. As the cost/risk value function estimates the cumulative risk value, the meaning of estimating under these two definitions becomes totally different. Moreover, CVaR of total time (TT) penalizes long absolute journey durations, including those attributable to long routes. CVaR of late time (LT) isolates pure congestion-induced excess delay, normalized by the expected travel time for each O-D pair, making it invariant to route length. Thus, CVaR of TT is suitable for low-frequency moving and long moving time, while CVaR of LT is suitable for high-frequency moving, but each moving time is low.

3.4. Risk Curriculum Learning

Existing DRL-based MBESS implementations typically require learning from scratch, which can be inefficient and time-consuming in the real world. Moreover, this inefficient learning process will intensify with the additional consideration of different risks, especially since traffic congestion risk is much more complex in real-world conditions. To tackle this challenge, this paper proposes a risk curriculum learning (RCL) approach specifically for the risk control in MBESS, as shown in Figure 2.

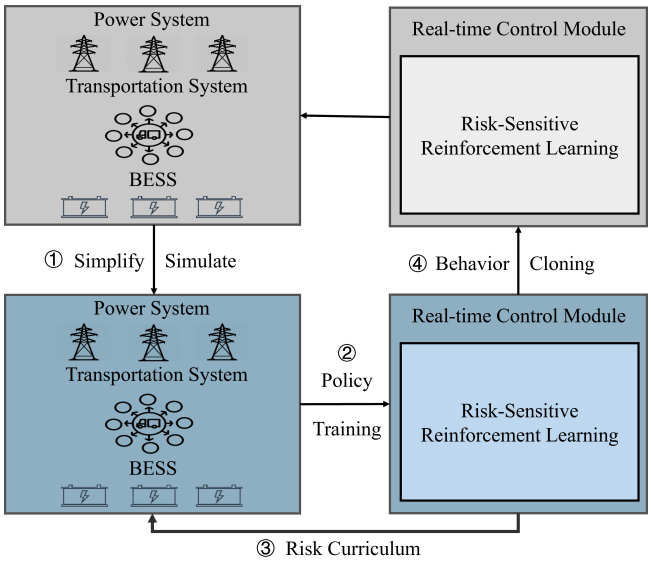


Figure 2. Risk curriculum learning in MBESS control.

First, we construct a simplified simulated environment that does not explicitly model any risk. Second, the agent is trained in the baseline simulated environment to develop an initial policy. Once the agent has learned effective strategies, one of the risks being considered, for example, the charging risk, will be introduced as a "risk curriculum" to facilitate further refinement. This step will be executed in a loop until all risks, charging/discharging and traffic congestion risks, have been taken into account. At last, once the agent has learned the risk-aware policy across all the risk sets, the policy will be deployed in the real-time control module through the behavior cloning method. The real-time control module will keep updating the policy with real-world operation data to reduce

the gap between the simulator and the real system. The key advantage of RCL is that gradually increasing risk starting from a simplified environment is both cost-effective and fast-converging, significantly enhancing overall learning efficiency.

4. Experiment

4.1. Settings

Environment: The test environment based on the IEEE 30-bus system, traffic volume data, and Australian Energy Market Operator (AEMO) load data [32] is built on the following settings.

The power system employs one-day load distribution data from AEMO to determine electricity prices through the resolution of an optimal power flow problem. To highlight the instability of new energy sources, a beta-distributed photovoltaic generator and a Weibull-distributed wind generator are located at nodes 26 and 28, respectively. And the node electricity prices for a single day are estimated using MATPOWER. To reflect the randomness, this paper repeats the simulation 100 times to obtain 100 lists of electricity prices independently, and the electricity price is selected randomly from these lists. Moreover, the influence of MBESS's capacity on the distribution of electricity prices at a given time can be ignored compared to the capacity of the entire electricity market. This allows the average electricity price to be considered a constant in the upper-level problem. Thus, we assume that the average electricity price is a stable price for each time.

This paper utilizes the topology of the IEEE 30-bus system to construct a transportation system where the distances between different nodes are represented proportionally to line resistance, and traffic volume data from [33]. This assumption is a standard MBESS convention when detailed road information is unavailable. Although it has limitations in ignoring the differences in directionality and heterogeneous speed limits, it is reasonable because wires are commonly built along roads, especially in cities. Moreover, the proposed modules are topology-agnostic due to their data-driven, architecture-independent design, which remains accurate even if the transportation network is replaced. It should be noted that the current numerical results are obtained within this simplified simulation environment; further validation on real-world transportation networks with empirical traffic data can be a future work to fully establish the generality of the reported performance gains. To emphasize the traffic randomness, we construct three average-volume data matrices between different places and randomly select one as the volume data, written in Equation (22).

$$\bar{V}_{nm} = Random([\bar{V}^0_{nm}, \bar{V}^1_{nm}, \bar{V}^2_{nm}]). \tag{22}$$

As the original volume data $\bar{V}_{nm}$ does not involve the time concept, we utilize Equation (23), which is a bimodal (double peaks) function of time, to simulate the real-time traffic volume data with rush hours.

$$\bar{V}_{nmt} = (a^p * (|t - T/2| - T/4)^2 + b^p) * \bar{V}_{nm}, \tag{23}$$

where $a^p = ((-2 * \delta^{\max})/\frac{T}{4})^2$ and $b^p = 1 + \delta^{\max}$, T is the total number of time intervals and $\delta^{\max}$ is the fluctuation scale. After obtaining the real-time traffic volume data, the traffic condition is obtained based on Equations (1) and (2) with $\mu^{\text{tra}}_{nmt} = \bar{V}_{nmt}$, $\sigma^{\text{tra}}_{nmt} = 30\% * \bar{V}_{nmt}$, $\chi = 0.15$, and $\psi = 4$.

Moreover, we chose $T_p = 0.5\,\text{h}$ (30 min) to be one time step and $T = 48$ (24 h) to be one episode. We select $E^{\max} = 6.75\,\text{MWh}$, $P^{\text{mcha}} = P^{\text{mdis}} = 2.7\,\text{MW}$, and $\eta = 0.97$ as parameters for the battery system.

As for other experiment settings, this paper follows the same settings in [24], such as how to eliminate the impact of randomness and how to deal with the remaining energy.

The simulation is implemented in Python 3.7 with PyTorch 1.13. All training and evaluation are conducted on a server equipped with CPU: Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20G Hz, and GPU: NVIDIA TESLA K80. Each algorithm is trained for 4000 episodes. After training, the final policy is evaluated over 10 independent test episodes with different random seeds.

In validating the performance of our method, this study focuses on the following metrics: (1) total profit, defined as the cumulation of the reward in Equation (16); (2) moving times, which are defined by the moving time steps in one episode. For each metric, we evaluated the mean, standard deviation, and confidence interval to demonstrate the improvement comprehensively. The mean value is averaged over 10 results to show the average improvement. The standard deviation aims to show the stability of the improvement. The confidence interval is obtained from a Normal distribution with a 95% confidence level. Fewer moving times and smaller std indicate better traffic congestion risk control.

To validate the final performance gain of our method, we compare MRE-SDDPG with RL-based methods: (1) RE-SDDPG, which only considers rewards and the charging/discharging risk, (2) SAC and (3) TD3, which only consider the rewards, and the model-based method, (4) MPC, which constructs the mathematical forms of all systems in [24] and solves the average case of the model by mixed-integer programming. Details of the MRE-SDDPG are shown in Appendix A.

To ensure a fair and transparent comparison, we clarify the assumptions underlying each method. MPC constructs the mathematical forms of all systems and solves the deterministic average-case model via mixed-integer programming (MIP); it therefore assumes perfect knowledge of the mean system parameters but does not account for distributional uncertainty. SAC and TD3 are model-free RL algorithms that maximize expected cumulative reward without any explicit risk management. RE-SDDPG extends SDDPG with charging/discharging CVaR risk control but does not incorporate traffic congestion risk or dynamic routing. MRE-SDDPG further adds TREM, DRM, and RCL to enable traffic risk congestion control. All methods share identical environment parameters, initial conditions, battery specifications, and the same 10-episode evaluation protocol. More details are shown in Appendix B.

Table 1 shows the total profit, moving times, and corresponding standard deviation of MRE-SDDPG, RE-SDDPG, SAC, TD3, and MPC strategies at the same initial position. MRE-SDDPG successfully achieves the highest profits while controlling the lowest moving times and standard deviation. Compared with MPC, MRE-SDDPG can improve the average performance by 10.0%, and reduces the moving times by about 24.8%. Compared to RE-SDDPG, this method improves average performance by 8.6% and reduces the moving time by 15.8%. This indicates that MRE-SDDPG can improve total profit by saving more time from movement to charging/discharging. Without taking risk into consideration, even state-of-the-art algorithms, like SAC and TD3, cannot achieve the best performance due to suboptimal moving decisions.

Table 1. Results for different control methods. Bold values are the best results.

Method	Profit			Moving Times		
	Mean	**Std**	**95% CI**	**Mean**	**Std**	**95% CI**
MRE-SDDPG	**398.44**	**8.71**	[393.04, 403.84]	**10.53**	**0.50**	[10.22, 10.84]
RE-SDDPG	366.72	14.40	[357.79, 375.65]	12.50	0.84	[11.98, 13.02]
MPC	362.07	17.93	[350.96, 373.18]	14.00	3.00	[12.14, 15.86]
SAC	348.32	4.98	[345.23, 351.41]	20.05	4.92	[17.00, 23.10]
TD3	342.07	4.93	[339.01, 345.13]	18.46	5.23	[15.22, 21.70]

4.2. Ablation Study for Dynamic Routing Module

To validate the performance of DRM, we compared two sets of methods: (1) RE-SDDPG and RE-SDDPG w/DRM, and (2) MRE-SDDPG and MRE-SDDPG w/o DRM.

Table 2 demonstrates that DRM is beneficial to improve the final profit. After adding DRM, the profit of RE-SDDPG increases by 2.4%, and the moving times decrease by 13.4%. For MRE-SDDPG, removing DRM results in a 6.4% decrease in the profit and about one more moving time. This result indicates that DRM can help decrease traffic congestion risk by dynamically adjusting the path when the agent is going to be late.

Table 2. Results for dynamic routing module. Bold values are the best results.

Method	Profit			Moving Times		
	Mean	Std	95% CI	Mean	Std	95% CI
RE-SDDPG	366.72	14.40	[357.79, 375.65]	12.50	0.85	[11.97, 13.03]
RE-SDDPG w/DRM	375.63	9.85	[369.52, 381.74]	10.83	0.49	[10.53, 11.13]
MRE-SDDPG w/o DRM	372.77	12.11	[365.26, 380.28]	11.85	0.62	[11.47, 12.23]
MRE-SDDPG	**398.44**	**8.71**	[393.04, 403.84]	**10.53**	**0.50**	[10.22, 10.84]

The training procedure in profit and moving time of the four variants are shown in Figure 3. For MRE-SDDPG, the total profit without DRM is significantly lower than the profit of MRE-SDDPG, and this trend will become more obvious with the increase in the number of training rounds. However, as MRE-SDDPG contains TREM, removing DRM mainly influences the moving times before convergence. As for RE-SDDPG, adding DRM can result in higher profit and lower moving times. This result indicates the power of DRM in improving profit without sacrificing traffic congestion risk.

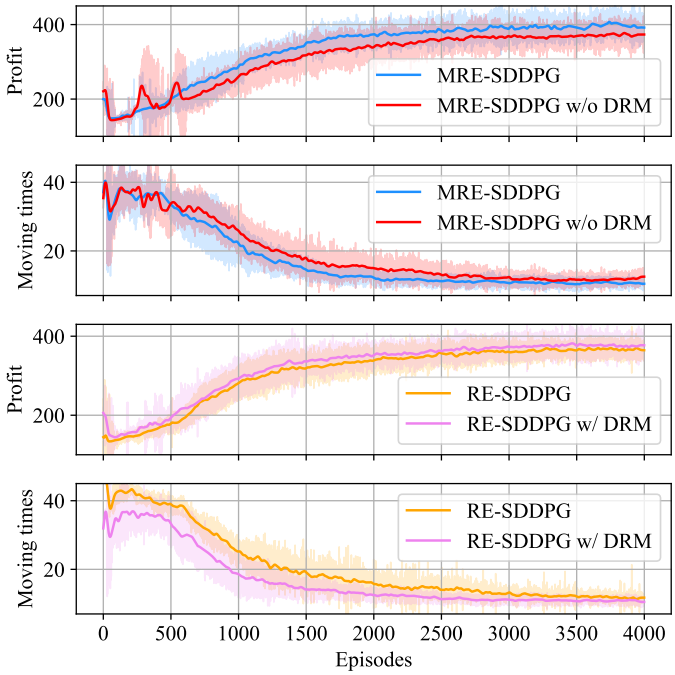


Figure 3. Training process for MRE-SDDPG and MRE-SDDPG w/o DRM, RE-DDPG and RE-DDPG w/DRM.

4.3. Ablation Study for Traffic Congestion Risk

To validate the performance of TREM, we compare two sets of methods: (1) RE-SDDPG and RE-SDDPG w/TREM, (2) MRE-SDDPG and MRE-SDDPG w/o TREM.

Table 3 demonstrates that TREM is beneficial in reducing the traffic congestion risk. For MRE-SDDPG, removing TREM will lead to a decrease of 3.2% on the profit and a little

increase in the moving times. Because MRE-SDDPG contains DRM, the moving times will not increase significantly. After adding TREM, the profit increases slightly, and the move times decrease significantly with a smaller standard deviation. This result shows that TREM successfully controls the traffic congestion risk.

The training procedure in profit and moving times of the four variants are shown in Figure 4. RE-SDDPG w/TREM can achieve higher profits with smaller moving times during the whole training procedure. This result is consistent with that of MRE-SDDPG after removing TREM. This shows that TREM successfully decreases the moving times and thus the agent can charge/discharge more to achieve higher profits.

Table 3. Results for traffic congestion risk. Bold values are the best results.

Method	Profit			Moving Times		
	Mean	Std	95% CI	Mean	Std	95% CI
RE-SDDPG	366.72	14.40	[357.79, 375.65]	12.50	0.85	[11.97, 13.03]
RE-SDDPG w/TREM	369.25	11.35	[362.21, 376.29]	11.60	0.61	[11.22, 11.98]
MRE-SDDPG w/o TREM	385.45	9.62	[379.49, 391.41]	10.77	0.48	[10.47, 11.07]
MRE-SDDPG	**398.44**	**8.71**	[393.04, 403.84]	**10.53**	**0.50**	[10.22, 10.84]

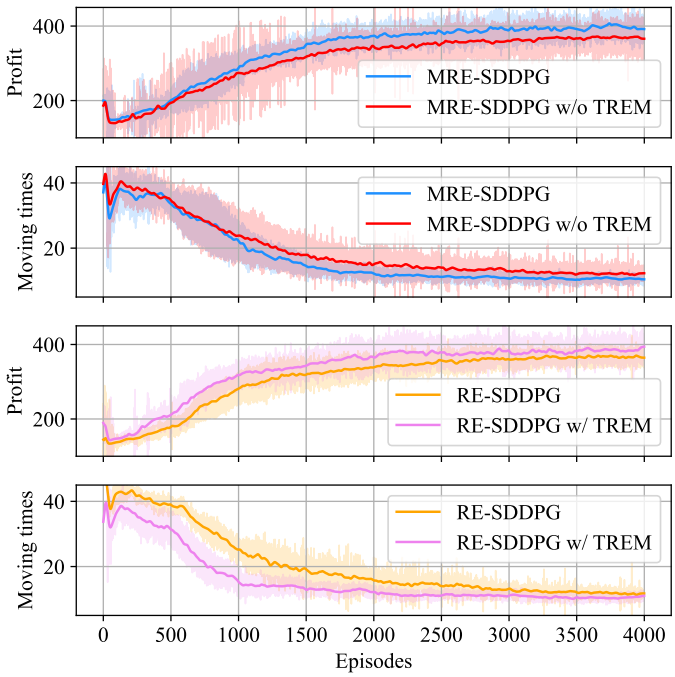


Figure 4. Training process for MRE-SDDPG and MRE-SDDPG w/o TREM, RE-DDPG and RE-DDPG w/TREM.

4.4. Comparison of Different Traffic Congestion Risk Definition

To validate the influence of two traffic congestion risk definitions, we tested MRE-SDDPG with CVaR of total time (MRE-SDDPG w/TT) and with CVaR of late time (MRE-SDDPG w/LT), respectively.

As shown in Figure 5, MRE-SDDPG w/LT achieves higher profit and smaller moving times compared to MRE-SDDPG w/TT. On average, the total profit of MRE-SDDPG w/LT exceeds MRE-SDDPG w/TT by 4.6%, with one fewer movement. This result shows that CVaR at a late time is a better criterion for traffic congestion risk, which demonstrates stronger advantages in both total profit and movement frequency.

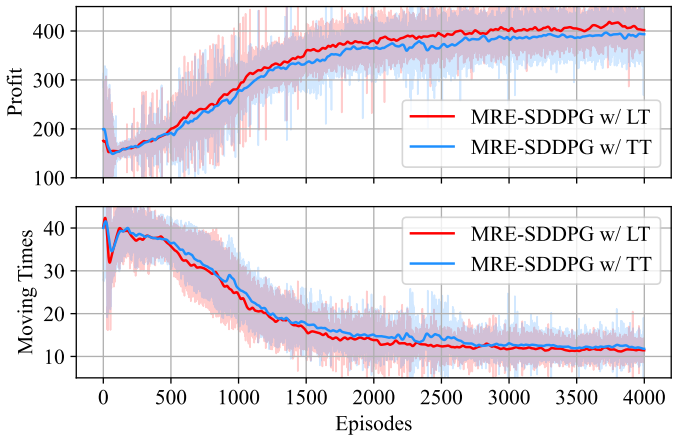


Figure 5. Ablation study on the two definitions of traffic congestion risk.

4.5. Ablation Study for Risk Curriculum Learning

To validate the improvement in training efficiency with RCL, we compared two methods: (1) MRE-SDDPG w/RCL and MRE-SDDPG w/o RCL.

As shown in Figure 6, MRE-SDDPG w/RCL achieves close conditions between the beginning stage and after convergence, while MRE-SDDPG w/o RCL needs to learn from scratch. MRE-SDDPG w/o RCL trains until 2500 episodes to catch up with MRE-SDDPG w/RCL. This result indicates that RCL takes advantage of the results in simulated environments and performs well in the real environment with traffic risk. Thus, RCL can significantly improve training efficiency and decrease training costs.

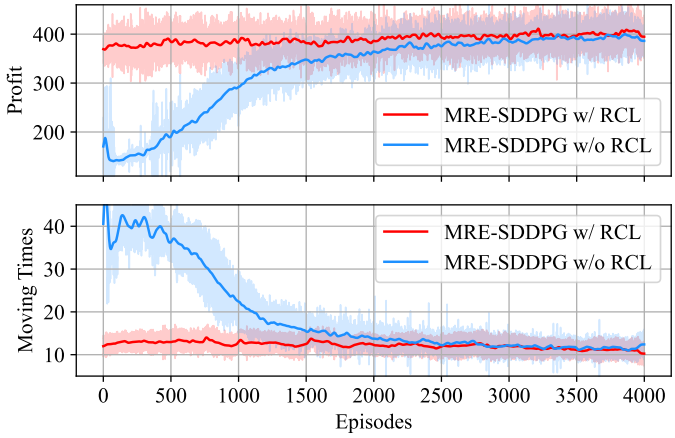


Figure 6. Ablation study on RCL.

4.6. Hyperparameter Analysis

To validate how the confidence level α of CVaR in traffic congestion risk influences the performance, we compared both LT and TT metrics under two commonly used confidence levels, 0.95 and 0.9.

As shown in Figure 7, both the mean and variance of the moving time decrease significantly as α decreases, no matter for MRE-SDDPG w/LT and MRE-SDDPG w/TT. Moreover, the profit also decreases slightly as α decreases. This result indicates that MRE-SDDPG tends to choose paths with low traffic congestion error as α decreases, even though the electricity prices for these paths are not very favorable.

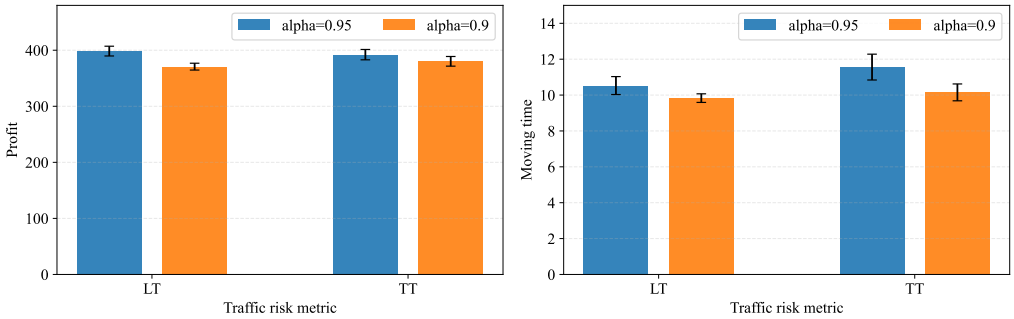


Figure 7. Hyperparameter analysis of CVaR under different confidence levels α for traffic congestion risk.

5. Conclusions

A flexible MBESS control framework with DRM and TREM is proposed to effectively take traffic conditions into consideration. In particular, DRM significantly improves the "on the road" control ability by allowing the agent to make new decisions at intermediate waypoints. To better control the traffic congestion risk, this paper proposes two quantitative definitions, which are the CVaR of the total time and CVaR of the late time, to represent the traffic risk accurately. To integrate DRM and TREM, a new MRE-SDDPG algorithm is proposed. This algorithm can efficiently control both electricity price risk and traffic congestion risk. Additionally, this paper proposes simulated pre-training to enhance the training efficiency. Through extensive simulations, this paper effectively demonstrates that the proposed control framework outperforms existing methods in maximizing profits while managing both types of risk. Moreover, RCL significantly improves the training efficiency and provides a reasonable initial policy for real-world online training. While the proposed framework demonstrates consistent advantages in simulation, several limitations should be noted. First, the current formulation optimizes a single MBESS unit; in a fleet scenario, multi-agent coordination would be required to prevent competing agents from both electricity price risk and traffic congestion risk. Moreover, the traffic collision risk should also be considered, which is a further complex risk to be quantified in the real world. Second, battery degradation is modeled with a simplified linear capacity-fade function. Although this simplicity will not influence the short-term results of this paper, more accurate degradation should be considered for the long-term operation study. To handle these limitations, future research can consider (1) extending to multi-agent MBESS under cooperative multi-agent RL and (2) testing long-term operation cases with more accurate degradation.

Author Contributions: Conceptualization, H.Z.; methodology, Z.L., J.Y. and H.Z.; formal analysis, Z.L. and J.Y.; visualization, Z.L. and J.Y.; validation, Z.L. and J.Y.; investigation, Z.L. and J.Y.; resources, H.Z.; writing—original draft preparation, Z.L. and J.Y.; writing—review and editing, Z.L., J.Y., H.Z., Y.C., X.G. and J.Z.; supervision, H.Z. and J.Z.; project administration, J.Z.; funding acquisition, J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the PolyU Direct Grant (P0047700, P0043885, P0051105, P0042845); the National Natural Science Foundation of China (72331009, 72171206); the Shenzhen Key Lab of Crowd Intelligence Empowered Low-Carbon Energy Network (ZDSYS20220606100601002).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

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

Nomenclature

Symbol	Description
A. General Symbols	
m	Destination place
m'	The farthest point that the vehicle can reach if it cannot reach the destination on time
n	Departure place/node
N	Total number of samples
t	Current time interval/tth time interval
T	Total number of time intervals
T_p	Time value for each interval
B. Transportation Model	
a^p, b^p	Temporary variables
c^{tra}	The labor cost of driving vehicles per minute (the transportation cost)
C_{nm}^{tra}	Maximum capacity of the traffic volume from place n to m
$N(\mu_{nmt}^{\text{tra}}, (\sigma_{nmt}^{\text{tra}})^2)$	Normal distribution for random traffic formulation
T_{nmt}^{mov}	Real traveling time considering all randomness from departure n to destination m at time t
T_{nm}^0	The traveling time from departure n to destination m when no other vehicle is on the road (free traveling time)
$\bar{V}_{nm}$	Expected of $\bar{V}_{nmt}$ on time t
$\bar{V}_{nmt}$	Expected traffic volume from departure n to destination m at time t
V_{nmt}^{tra}	Real-time random traffic volume generated by $N(\mu_{nmt}^{\text{tra}}, (\sigma_{nmt}^{\text{tra}})^2)$ from place n to m at time t
$\delta^{\max}$	Maximum percentage that traffic volume can fluctuate
$\mu_{nmt}^{\text{tra}}, \sigma_{nmt}^{\text{tra}}$	Mean and standard deviation of the traffic volume distribution for each departure n to destination m at time t
χ, ψ	Retardation coefficients
P_{nmt}^{mov}	The moving time distribution model
C. Vehicle Mobility Model	
α_{nt}	Binary control variable of arriving n at time t
$\alpha_{m't}$	Binary control variable of the dynamic routing to intermediate waypoint m' at time t for lateness
β_{nt}	Binary control variable of leaving n at time t
γ_{nmt}	Binary control variable of moving from n to m at time t
θ_{nt}	Instrumental variable
ω_{nt}	Binary control variable of parking at node n at time t
D. Reinforcement Learning	
a	General action in reinforcement learning formulation
a_c	Continuous action for charging/discharging electricity
a_m	Discrete action for moving to another place
a'_m	New traveling action when the MBESS device cannot reach the original destination
a_t	Action at time t
$a_{c,t}, a_{m,t}$	Charging or moving action at time t

J	Expected cumulative reward returns
L_{Q_r}, L_{Q_c}	Loss of reward/cost Q function
n_t, m_t	Departure/destination place at time t
M	Total iteration for training
Q, Q'	Q function and target Q function
Q_r, Q'_r	Current/target reward Q function
Q_c, Q'_c	Current/target Cost Q function
r_t	Reward signal received at time t
$\mathbb{B}^r$	Replay buffer for reinforcement learning
s_t	State information received at time t
β	Hyper-parameter to adjust the weight cost value for policy optimization
γ	Discount factor
$\theta^Q, \theta^{Q'}$	Current/target Q neural network parameters
$\theta^\mu, \theta^{\mu'}$	Current/target policy neural network parameters
$\theta^{Q_r}, \theta^{Q'_r}$	Current/target reward Q neural network parameters
$\theta^{Q_c}, \theta^{Q'_c}$	Current/target cost Q neural network parameters
$\mu(\cdot)$	Policy function
τ	Hyper-parameter for soft update target network

E. Risk Estimation and CVaR

c^p	Capacity of history electricity price buffer
c^t	Capacity of history moving time buffer
$F_Y(y)$	Cumulative distribution function of Y
$\mathbb{B}^p / \mathbb{B}^p_{n,t}$	The buffer to store history electricity price to formulate the electricity price risk
$\mathbb{B}^t$	History moving time buffer
Y, y	Random variable and its value
α	Confidence level
o^α	VaR of Y
ϕ^α	CVaR of Y
ϕ^α_t	General risk represented by CVaR at time t in confidence level α
$\phi^{\alpha,\text{cha}}_{n,t}, \phi^{\alpha,\text{dis}}_{n,t}$	Charging risk represented by CVaR at place n and time t
$\phi^{\alpha,\text{mov}}$	Moving/transportation risk represented by CVaR at place n and time t

F. Other Symbols

$F(x_t)$	Total profit of MBESS
$C^{\text{rev}}(x_t)$	Revenue of MBESS
$C^{\text{tra}}(x_t)$	Traveling cost of MBESS
$C^{\text{deg}}(x_t)$	Degradation cost of MBESS
λ_{nt}	Electricity price at place n at time t

https://doi.org/10.3390/esa3020009

Appendix A. Network Structure and Hyperparameter of MRE-SDDPG

Table A1. Hyperparameter settings for the MRE-SDDPG algorithm.

Parameter	Value
Network Architecture	
Actor Network	Input: state dimension (after embedding: 31) Hidden layers: [64, 64] (Fully Connected) Activation: ReLU Output: 1-dim continuous action (Tanh)
Critic Networks (Q_r, Q_c)	Input: state–action pair (after embedding) Hidden layers: [128, 256, 128, 64] Activation: ReLU Output: Return & risk values
Embedding Dimensions	Start node: $30 \rightarrow 10$ Target node: $30 \rightarrow 10$ Time index: $48 \rightarrow 10$
Training Hyperparameters	
Optimizer	Adam
Actor Learning Rate	1×10^{-4}
Critic Learning Rate	1×10^{-4}
Discount Factor (γ)	1.0
Risk Parameter (β)	0.25
Soft Update Rate (τ)	0.01
Replay Buffer Size ($\mathbb{B}^r$)	6000 transitions
Batch Size	128
Exploration Noise	Gaussian noise $\mathcal{N}(0, 0.01)$
Exploration Probability (ϵ)	0.2
Target Network Update	Hard update every 20 learning steps
Risk and Environment Parameters	
Confidence Level (α) for CVaR	0.95
Price Buffer Size	$48 * 30 * n$
Moving Time Buffer Size	$48 * 30 * 30 * n$
Charging Efficiency (η)	0.97
Battery Capacity	1.0 (normalized)
Max Charge Power	± 0.2 per time step
Transportation Cost c^{tra}	0.35
Degradation Coefficient c^{deg}	0.98

Appendix B. Systemic Comparison Between MRE-SDDPG and Baselines

Table A2. Differences among compared approaches. DD is short for Dynamic Decision ability, and DR is short for dynamic routing ability.

Method	Model	Risk Awareness	DD	DR	Information Oracle
MPC	MIP (model-based)	None	No	No	Average setting
SAC	DRL (model-free)	None	Yes	No	Current observation
TD3	DRL (model-free)	None	Yes	No	Current observation
RE-SDDPG	DRL (model-free)	Price	Yes	No	Current observation
MRE-SDDPG	DRL (model-free)	Traffic + price risk	Yes	Yes	Current observation

References

1. Yang, H.; Wang, C.A.; Dong, Y. Energy ceramic design for robust battery cathodes and solid electrolytes. *Adv. Powder Mater.* **2024**, *3*, 100185. [CrossRef]
2. Zhang, T.; Lin, Z.; Li, C.; Cai, X.; Shen, S.; Xiang, J.; Xia, X.; Qiu, Z.; Tang, C.; Song, M.; et al. Plasma-Assisted Multiscale Interface Engineering for Advanced Solid/Liquid Lithium Batteries. *cMat* **2025**, *2*, e70025. [CrossRef]
3. As' ad, S. Why renewable energy gained attention and demand globally? *Nat. Environ. Pollut. Technol.* **2024**, *23*, 467–473. [CrossRef]
4. Shahzad, S.; Jasińska, E. Renewable revolution: A review of strategic flexibility in future power systems. *Sustainability* **2024**, *16*, 5454. [CrossRef]
5. Yang, Y.; Bremner, S.; Menictas, C.; Kay, M. Battery energy storage system size determination in renewable energy systems: A review. *Renew. Sustain. Energy Rev.* **2018**, *91*, 109–125. [CrossRef]
6. Li, R.; Tu, Q.; Feng, H.; Zou, Z. Demand response based battery energy storage systems design and operation optimization. *Energy Build.* **2025**, *338*, 115738. [CrossRef]
7. Javaid, S.; Kaneko, M.; Tan, Y. Energy balancing of power system considering periodic behavioral pattern of renewable energy sources and demands. *IEEE Access* **2024**, *12*, 70245–70262. [CrossRef]
8. Kumar, A.; Meena, N.K.; Singh, A.R.; Deng, Y.; He, X.; Bansal, R.; Kumar, P. Strategic integration of battery energy storage systems with the provision of distributed ancillary services in active distribution systems. *Appl. Energy* **2019**, *253*, 113503. [CrossRef]
9. Wang, W.; Yuan, B.; Sun, Q.; Wennersten, R. Application of energy storage in integrated energy systems—A solution to fluctuation and uncertainty of renewable energy. *J. Energy Storage* **2022**, *52*, 104812. [CrossRef]
10. Arifujjaman, M.; Andaya, G.; Xioco, Z.; Araiza, J. Development, Demonstration, and Validation of Mobile Battery Energy Storage System (MBESS) by Southern California Edison (SCE). In *Proceedings of the 2024 IEEE Electrical Energy Storage Application and Technologies Conference (EESAT)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–5.
11. Zheng, Y.; Meng, K.; Luo, F.; Qiu, J.; Zhao, J. Optimal integration of MBESSs/SBESSs in distribution systems with renewables. *IET Renew. Power Gener.* **2018**, *12*, 1172–1179. [CrossRef]
12. Zheng, Y.; Dong, Z.; Huang, S.; Meng, K.; Luo, F.; Huang, J.; Hill, D. Optimal integration of mobile battery energy storage in distribution system with renewables. *J. Mod. Power Syst. Clean Energy* **2015**, *3*, 589–596. [CrossRef]
13. Armenta-Déu, C.; Demas, L. Optimization of grid energy balance using vehicle-to-grid network system. *Energies* **2024**, *17*, 1008. [CrossRef]
14. Yan, Z.; Zhang, Y.; Yu, J. An allocative method of stationary and vehicle-mounted mobile energy storage for emergency power supply in urban areas. *Energy Storage* **2024**, *6*, e681. [CrossRef]
15. He, G.; Michalek, J.; Kar, S.; Chen, Q.; Zhang, D.; Whitacre, J.F. Utility-scale portable energy storage systems. *Joule* **2021**, *5*, 379–392. [CrossRef]
16. Tian, K.; Zang, Y.; Wang, J.; Zhang, X. Electricity arbitrage for mobile energy storage in marginal pricing mechanism via bi-level programming. *Int. J. Electr. Power Energy Syst.* **2024**, *162*, 110330. [CrossRef]
17. He, G.; Zhang, D.; Pi, X.; Chen, Q.; Kar, S.; Whitacre, J. Spatiotemporal arbitrage of large-scale portable energy storage for grid congestion relief. In *Proceedings of the 2019 IEEE Power & Energy Society General Meeting (PESGM)*; IEEE: Piscataway, NJ, USA, 2019; pp. 1–5.
18. Tamrakar, U.; Bhujel, N.; Nguyen, T.A.; Byrne, R.H.; Chalamala, B. A Model Predictive Control Framework for Combining Energy Arbitrage and Power Quality Applications From Energy Storage Systems. *IEEE Open Access J. Power Energy* **2024**, *11*, 469–480. [CrossRef]
19. Jeon, S.; Nguyen, H.T.; Choi, D.H. Safety-Integrated Online Deep Reinforcement Learning for Mobile Energy Storage System Scheduling and Volt/VAR Control in Power Distribution Networks. *IEEE Access* **2023**, *11*, 34440–34455. [CrossRef]
20. Michailidis, P.; Michailidis, I.; Minelli, F.; Coban, H.H.; Kosmatopoulos, E. Model Predictive Control for Smart Buildings: Applications and Innovations in Energy Management. *Buildings* **2025**, *15*, 3298. [CrossRef]
21. Zhao, H.; Liu, Z.; Mai, X.; Zhao, J.; Qiu, J.; Liu, G.; Dong, Z.Y.; Ghias, A.M. Mobile battery energy storage system control with knowledge-assisted deep reinforcement learning. *Energy Convers. Econ.* **2022**, *3*, 381–391. [CrossRef]
22. Ding, Y.; Qu, G.; Chen, X.; Wang, J.; Song, J.; He, G. Deep Reinforcement Learning-Based Spatiotemporal Decision of Utility-Scale Highway Portable Energy Storage Systems. *IEEE Trans. Ind. Appl.* **2023**, *60*, 966–975. [CrossRef]
23. He, G.; Ding, Y.; Wu, Z.; Chen, X.; Zhang, D.; Song, J. Environment-adaptive online learning for portable energy storage based on porous electrode model. *IEEE Trans. Autom. Sci. Eng.* **2024**, *22*, 8386–8399. [CrossRef]
24. Liu, Z.; Zhao, H.; Liu, G.; Liang, G.; Zhao, J.; Qiu, J. Risk-Sensitive Mobile Battery Energy Storage System Control with Deep Reinforcement Learning and Hybrid Risk Estimation Method. *IEEE Trans. Smart Grid* **2024**, *15*, 4143–4158. [CrossRef]
25. Ding, Y.; Chen, X.; Wang, J. Deep reinforcement learning-based method for joint optimization of mobile energy storage systems and power grid with high renewable energy sources. *Batteries* **2023**, *9*, 219. [CrossRef]

26. Malin, F.; Norros, I.; Innamaa, S. Accident risk of road and weather conditions on different road types. *Accid. Anal. Prev.* **2019**, *122*, 181–188. [CrossRef]
27. Sheng, Z.; Huang, Z.; Chen, S. Traffic expertise meets residual RL: Knowledge-informed model-based residual reinforcement learning for CAV trajectory control. *Commun. Transp. Res.* **2024**, *4*, 100142. [CrossRef]
28. Wagenmaker, A.; Huang, K.; Ke, L.; Jamieson, K.; Gupta, A. Overcoming the Sim-to-Real Gap: Leveraging Simulation to Learn to Explore for Real-World RL. *Adv. Neural Inf. Process. Syst.* **2024**, *37*, 78715–78765.
29. Wang, Y.; Qiu, D.; Strbac, G. Multi-agent deep reinforcement learning for resilience-driven routing and scheduling of mobile energy storage systems. *Appl. Energy* **2022**, *310*, 118575. [CrossRef]
30. Sarykalin, S.; Serraino, G.; Uryasev, S. Value-at-risk vs. conditional value-at-risk in risk management and optimization. In *State-of-the-Art Decision-Making Tools in the Information-Intensive Age*; Informs: Catonsville, MD, USA, 2008; pp. 270–294.
31. Supervision, O.B. *BC: Fundamental Review of the Trading Book: A Revised Market Risk Framework*; Bank for International Settlments: Basel, Switzerland, 2014.
32. Zhao, H.; Zhao, J.; Qiu, J.; Liang, G.; Wen, F.; Xue, Y.; Dong, Z.Y. Data-driven risk preference analysis in day-ahead electricity market. *IEEE Trans. Smart Grid* **2020**, *12*, 2508–2517. [CrossRef]
33. Stabler, B.; Bar-Gera, H.; Sall, E. Transportation Networks for Research Core Team. Transportation Networks for Research. Available online: https://github.com/bstabler/TransportationNetworks (accessed on 4 June 2026).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.