Article | Received 13 November 2024; Accepted 17 December 2024; Published 31 December 2024
https://doi.org/10.55092/mt20240005

Adaptive learning-based energy management for HEVs using soft actor-critic DRL algorithm

Ozan Yazar[1], **Serdar Coskun**[1,*] **and Fengqi Zhang**[2]

[1] Department of Mechanical Engineering, Tarsus University, Tarsus, Mersin 33340, Turkey

[2] School of Automobile, Chang'an University, Xi'an 710054, China

* Corresponding author; E-mail: serdarcoskun@tarsus.edu.tr.

Abstract: In this work, we design an energy management strategy (EMS) for hybrid electric vehicles (HEVs) using a deep reinforcement learning (DRL) algorithm. Specifically, this paper introduces a soft actor-critic (SAC)-based EMS, tailored for devising optimal energy distribution for HEVs. The proposed SAC-based approach is useful for addressing inherent drawbacks that exist in many DRL methods such as slower convergence rate, discretization error, as well as suboptimal solutions. The designed SAC algorithm presents a self-adaptive efficiency in executing continuous decision-making policies through the balance of exploration and exploitation using an entropy-based action selection method and an entropy-added reward function. Extensive experiments are carried out to demonstrate the merits of the adaptive SAC algorithm over the widely adopted Q-learning (QL), deep-Q-network (DQN), and deep deterministic policy gradient (DDPG) approaches on fuel economy and battery charge sustainability. An unknown driving cycle is also employed to show the adaptability feature of the proposed scheme, revealing fuel savings of 6.26%, 3.01%, and 2.03% over the QL-based, DQN-based, and DDPG-based methods, respectively.

Keywords: adaptive energy management; hybrid electric vehicle; deep reinforcement learning algorithm; soft-actor-critic algorithm

1. Introduction

The increase in global energy demand and the environmental impacts of fossil fuels are bringing the need for more sustainable transportation solutions [1, 2]. Hybrid electric vehicles (HEVs) stand out as an innovative technology that responds to this need with their potential to increase energy efficiency and reduce carbon emissions [3]. HEVs operate by using an internal combustion engine and an electric motor together, thus benefiting from the advantages of both power sources [4]. The electric motor, especially at low speeds or in urban driving, reduces both fuel consumption and carbon emissions. In situations requiring high speeds, the internal combustion engine steps in and provides power support. With these features, HEVs reduce energy costs and provide a transportation alternative with a lower carbon footprint, making significant contributions to environmental sustainability [5, 6, 7].

Energy management strategy (EMS) in HEVs is of critical importance to minimize fuel consumption and to extend battery life. EMS aims to optimize power distribution between vehicle's internal combustion engine and electric motor, thus ensuring efficient use of both sources [8, 9]. However, energy management in HEVs presents several challenges due to their complex structures. It is necessary to effectively manage the power transfer between internal combustion engine and electric motor and to ensure the right balance between the two energy sources. In addition, constantly changing driving conditions such as speed, road slope, and traffic density require the EMS to make dynamic decisions in real-time. In order to optimize energy use and maintain battery health under varying conditions, more advanced and intelligent EMSs are required. Under an inefficiently-designed EMS, decreased vehicle performance and shortened battery life may occur [10, 11].

One of the innovative approaches that has emerged in energy management in recent years is the utilization of reinforcement learning (RL) techniques [12]. RL is a machine learning method that allows a system to develop the best action strategy by learning from its responses under different situations [13]. This technique is a valuable tool in complex systems, as it offers the ability to make decisions in real-time under dynamic and uncertain environments [14]. However, traditional RL methods can be limited with high-dimensional and continuous action fields [15, 16]. At this point, RL techniques combined with deep learning, forming deep reinforcement learning (DRL) overcomes the limitations of traditional RL and offers higher efficiency and flexibility [17, 18]. Wang *et al.* study the potential of electric vehicles to reduce carbon emissions in transportation networks and contribute to the energy grid with the vehicle-to-grid technique using a multi-agent RL method. The study shows that carbon reduction is achieved by eco-routing in transportation and energy networks [19]. Wang *et al.* develop a deep deterministic policy gradient (DDPG)-based multi-agent reinforcement learning algorithm to optimize active and reactive power control and voltage regulation services. Conducted simulations achieve a significant superiority in terms of speed and reward [20]. In this context, in the study by Chen *et al.*, an EMS is developed by combining model predictive control (MPC) with a double Q-learning algorithm to optimize power allocation for plug-in HEV (PHEV). Simulation results show that the strategy provides superior fuel economy by adapting to different battery charge levels [21]. In the study of Tresca *et al.*, a deep Q-learning (QL) algorithm is used to optimize the energy management of diesel PHEVs. The study tests the CO_2 emission reduction performance of the algorithm in the WLTC cycle and various driving conditions and reveals that it performs close to the dynamic programming (DP) optimization (7% difference) [22]. In the study of Zhu *et al.*, the design of the energy management strategy for a mild hybrid HEV greatly affects the potential fuel economy gains and the amount of calibration required under driving routes. An automatic EMS development process is presented using a DRL algorithm based on real-world routes to maintain the battery charge level when the driving cycle is not known in advance, and this strategy is compared with dynamic programming and adaptive energy consumption minimization strategy in [23]. Han *et al.* propose an eligibility trace-based EMS, which is an extension of the QL algorithm, to improve fuel economy and increase battery life for HEVs. The study increased the ability to adapt to various driving conditions by using an eligibility trace algorithm, which provides online learning and an adaptive environment model compared to traditional RL algorithms [24]. Ahmadian *et al.* develop a QL-based EMS for series-parallel HEVs. The study achieves 1.25% fuel saving and 65% battery life increase under HWFET driving cycle and the ability to adapt to different driving cycles is ensured [25]. In addition to these studies, the main methods in RL-based HEV energy management strategies are summarized in Table 1, providing a comprehensive comparison of recent studies.

Traditional algorithms have limitations such as optimal behavior execution under environments with continuous and high-dimensional action fields. In the application of EMSs, action fields are usually continuous and require continuous learning with a large dataset [39]. Moreover, QL and DQN may present instability in the learning process under uncertain and

Table 1. Summary of RL/DRL-Based energy management strategies for HEVs

Author	**Method**	**HEV Type**	**Description**
Lee *et al.*, 2020 [26]	Q-Learning	Parallel HEV	A RL strategy is compared with dynamic programming methods; shown to be more suitable for time-varying control and boundary conditions. Fuel efficiency under various driving cycles is evaluated and convergence properties are tested through transfer learning.
Xu *et al.*, 2022 [27]	Q-Learning	Parallel HEV	Adaptability of a Q-Learning-based supervisory control strategy is investigated. The effect of driving cycle, vehicle load condition, and road grade on fuel economy is analyzed, showing adaptability and fuel economy compared to other methods.
Tang *et al.*, 2022 [28]	DQN	Series HEV	A DRL-based EMS using DQN for throttle control and gear shifting is designed. A 0.55% reduction in fuel consumption and high computational efficiency is achieved. Learning- and rule-based control strategies are synchronized.
Zheng *et al.*, 2022[29]	Q-Learning, DQN, DDPG	Fuel Cell HEV	Q-Learning, DQN, and DDPG algorithms to FCHEV EMS are applied. Fuel economy is improved while considering fuel cell durability and algorithm performance. Convergence ability, fuel economy, durability, and adaptability are compared.
Lian *et al.*, 2020 [30]	DDPG	Parallel HEV	A DDPG-based EMS supported by expert knowledge is applied. Multi-objective energy management considering battery properties and BSFC curves is addressed. Accelerated learning and improved fuel economy demonstrate better performance and system stability as compared to other methods.
Yazar *et al.*, 2023 [31]	Q-Learning, DQN, DDPG, TD3	Power-Split HEV	TD3 for HEV EMS is proposed and compared with Q-Learning, DQN, and DDPG. Superior fuel economy, SOC sustainability, and training stability across various driving cycles are achieved.
Lin *et al.*, 2022 [32]	Q-Learning	Parallel HEV	A Q-Learning-based EMS using Markov Chains for transition probability is developed. EMS updates triggered by KL divergence rates, and convergence improved with an Exploration Factor (EF). Comparisons are carried out, showing significant improvements in fuel economy and energy efficiency.
Liu *et al.*, 2023 [33]	Reward-Directed Policy Optimization (RDPO)	Power-Split HEV	A DRL-based EMS using NN-based multi-constraint optimization and a rule-based restraint system is studied. RDPO is optimized for fuel economy while avoiding irrational control signals, achieving outstanding results under WLTC, NEDC, and CTUDC cycles.
Wang *et al.*, 2024 [34]	BO-SAC	Parallel HEV	A BO-SAC algorithm for EMS is proposed, enhancing stability and robustness via Bayesian optimization and SAR co-design, with over 3% energy consumption reduction across ten driving cycles.
Li *et al.*, 2022 [35]	SAC	Parallel HEV	A SAC-based EMS with automatic entropy tuning for energy efficiency optimization and adaptability to driving cycles is designed. Real vehicle data is utilized, achieving 4.37% energy savings and maintaining SOC at reference levels.
Liu *et al.*, 2021 [36]	TD3	Power-Split HEV	A DRL-based EMS with a novel reward function that penalizes irrational actions is proposed. TD3 achieves 10% faster computation and 7.28% lower fuel consumption compared to DDPG in complex tasks with physical constraints.
Liu *et al.*, 2024 [37]	SAC	Parallel HEV	A new auto-tune SAC algorithm is proposed to optimize the motor torque and gear shifting in hybrid action fields. The proposed algorithm has higher computational efficiency and lower energy efficiency compared with TD3.
Liu *et al.*, 2025 [38]	ATSAC	Light Duty HEV	A SAC-based EMS is proposed to improve generalization by automatically tuning the parameters and synthesizing a specific training cycle based on naturalistic driving big data. 52.32% higher computational efficiency is achieved compared to SAC and TD3. The synthetic cycle is found to reduce NTR by 18.37% compared to WLTC, better reflecting real-world scenarios.

variable environmental conditions, and this situation is insufficient to provide the desired efficiency under objectives of fuel economy or battery performance [40]. In addition, DQN may face overestimation or underestimation problems when using deep neural networks to estimate the action-value function, which may cause fluctuations in the learning process [41]. The above limitations highlight the need for a more robust and generalizable algorithm for the HEV EMS problem. A method that improves learning stability in the continuous action domain, adapts to environmental changes and minimizes overestimation problems while achieving high efficiency is necessary.

Considering these shortcomings, the soft actor-critic (SAC) algorithm stands out as a promising solution for EMS design. SAC is useful for continuous action domains and optimizes the balance of exploration and exploitation during action selection using an entropy-based method with an entropy-added reward function. This allows the algorithm to encourage exploration while increasing the stability at each step, thus obtaining more reliable results. The entropy term allows uncertainty to be preserved during policy learning and helps the algorithm to learn more general solutions by balancing both exploration and exploitation [42]. SAC algorithm quickly adapts to environmental changes using its entropy-based reward function, thanks to its twin critic structure. It minimizes overestimation problems and learns the most appropriate strategy under constantly changing conditions; thus, it provides an effective response to varying energy demands and battery levels [42].

Although there are studies of the SAC algorithm in the literature, the lack of comprehensive research on the adaptability and efficiency of the SAC is the main motivation of this research. In particular, we test the overall adaptability and stability performance of SAC under different driving cycles, which is limited. Due to the features and limitations mentioned above, this paper investigates the adaptiveness and efficiency of the SAC algorithm for the HEV EMS problem. The main contributions of this work are twofold. First, an exploration and exploitation-based adaptive SAC is designed for HEV energy management for improved fuel economy and SOC charging sustainability. Second, we conduct comprehensive experiments to show fuel economy and battery charge sustainability advantages using a total of six driving cycles. An unknown driving cycle is employed to reveal adaptability and efficient training. In this study, the SAC algorithm for the HEV energy management strategy is optimized with an approach that considers the balance of exploration and exploitation. The designed model uses a dual-critic mechanism and a target network update strategy to improve the control performance of systems with continuous action space. In addition, the reward function used in the energy management control has been customized to improve fuel economy and ensure battery charging sustainability.

The rest of the paper is organized as follows. Section 2 introduces the study object of this research along with the detailed description of the proposed scheme in 3. The experiments and results discussion are demonstrated in 4. The conclusions and future research directions are drawn in 5.

2. Vehicle model

In this study, a power-split HEV model that combines the advantages of parallel and series structures is selected. Power-split hybrid systems are widely used in commercial vehicles due to their flexibility and energy efficiency.

A planetary gear system is used as the drive train of the hybrid system in this study. In the planetary gear system, the electric motor (EM) is connected to the carrier and the generator (GEN) is connected to the sun gear. The ring gear serves as the final drive and the power is shared between the EM and the ring gear. This structure allows the system to efficiently share torques from different power sources. The structure and power flow directions of the planetary gear system are shown in Figure 1. The planetary gear kinematic equation is expressed with the angular velocities of the sun gear, ring gear and carrier gear as follows:

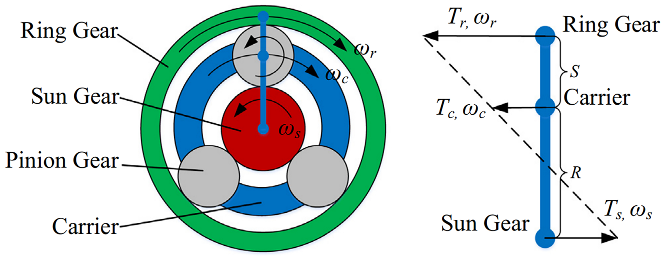


Figure 1. Structure and lever diagram of the planetary gear system equipped in a power-split HEV.

$$\omega_s(t)\cdot S+\omega_r(t)\cdot R=\omega_c(t)\cdot(S+R) \tag{1}$$

where, S and R represent the radius of the sun and ring gears, respectively. The angular velocities of the sun, ring, and carrier gears are represented by ω_s , ω_r , and ω_c , respectively. The generator (GEN) can charge the battery using the power produced by the internal combustion engine (ICE) or directly supply power to the electric motor (EM). The dynamics of the power transmission system can be modeled as follows, assuming the inertia of the pinion gears is negligible and the drive shafts are rigid:

$$J_{GEN}\dot{\omega}_{GEN}(t)=T_{GEN}(t)+F\cdot S \tag{2}$$

$$J_{ICE}\dot{\omega}_{ICE}(t)=T_{ICE}(t)-F\cdot(S+R) \tag{3}$$

$$J_{EM}\dot{\omega}_{EM}(t)=T_{EM}(t)-\frac{T_{shaft}(t)}{g_f}+F\cdot R \tag{4}$$

where, J_{GEN} , J_{ICE} , and J_{EM} represent the moments of inertia of GEN, ICE, and EM, respectively, while $T_{ICE}=T_r$, $T_{GEN}=T_g$, and $T_{EM}=T_c$ denote the torques. F represents the internal force in the pinion gears, g_f is the final drive gear ratio, and T_{shaft} is the torque on the drive shaft. To simplify the dynamic equations, the inertia terms are set to zero and ignored. Under this assumption, the rotational speed and torque requirements of EM are expressed as follows:

$$\omega_{EM}(t)=\frac{g_f}{R_{wheel}}V(t) \tag{5}$$

$$m\dot{V}(t)=\frac{T_{shaft}(t)+T_{gen}(t)}{R_{wheel}}-mg\sin(\theta(t))-\frac{1}{2}\rho AC_dV^2(t)-C_rmg\cos(\theta(t)) \tag{6}$$

where, R_{wheel} represents the wheel radius, V the vehicle speed, m the vehicle mass, T_{brake} the brake torque, θ the road slope, $\frac{1}{2}\rho AC_d$ the aerodynamic drag force, and C_r the rolling resistance coefficient.

To achieve optimal power distribution, power is allocated between the internal combustion engine and motor/generator units (M/G1 and M/G2) to minimize energy consumption at each time step. Assuming the engine operates under optimal conditions and dynamic characteristics are neglected, the fuel consumption rate $\dot{m}_{fuel}$ and the efficiencies of $M/G1$ and $M/G2$ ($\eta_{M/G1}$ and $\eta_{M/G2}$) are derived from empirical data as functions of angular velocities and torques.

$$\dot{m}_{fuel}(t) = \Psi_{eng}(\omega_{eng}(t), T_{eng}(t)) \tag{7}$$

$$\eta_{M/G1}(t) = \Psi_{M/G1}(\omega_{M/G1}(t), T_{M/G1}(t)) \tag{8}$$

$$\eta_{M/G2}(t) = \Psi_{M/G2}(\omega_{M/G2}(t), T_{M/G2}(t)) \tag{9}$$

where Ψ_{eng}, $\Psi_{M/G1}$, and $\Psi_{M/G2}$ represent empirical maps for the ICE, generator, and motor, respectively. In a power-split HEV, the battery provides power or recovers energy through an inverter. A basic resistance model is used to describe the battery characteristics. The state of charge (SOC), representing battery charge sustainability, is calculated as follows:

$$S\dot{O}C(t) = -\frac{I_{batt}(t)}{Q_{max}} \tag{10}$$

$$P_{batt}(t) = V_{oc}I_{batt}(t) - I_{batt}(t)^2 R_{batt} \tag{11}$$

where $I_{batt}(t)$ is the battery current, Q_{max} denotes the maximum battery capacity, $P_{batt}(t)$ represents battery power, R_{batt} is the internal resistance, and V_{oc} is the open-circuit voltage. The terminal battery power requirement is expressed by the following equation:

$$P_{batt} = \frac{P_{M/G1}(t)}{(\eta_{M/G1}(t) \cdot \eta_{inv}(t))^{k_{M/G1}(t)}} + \frac{P_{M/G2}(t)}{(\eta_{M/G2}(t) \cdot \eta_{inv}(t))^{k_{M/G2}(t)}} \tag{12}$$

where $P_{M/G1}(t)$ and $P_{M/G2}(t)$ denote the shaft powers, and η_{inv} represents the inverter efficiency.

$$k_i(t) = \begin{cases} 1 & \text{if } P_i(t) > 0 \\ -1 & \text{if } P_i(t) < 0 \end{cases} \quad \text{for } i = \{M/G1, M/G2\} \tag{13}$$

where $P_i(t)$ denotes the instantaneous power output of the motor/generator units (M/G1 and M/G2). When $P_i(t) > 0$, the unit operates in motor mode, drawing power from the battery. In contrast, when $P_i(t) < 0$, the unit operates in generator mode, regenerating power back to the battery. Equations 1 to 13 describe the energy management-focused model used in this study. The primary parameters of the power-split HEV are listed in Table 2.

Table 2. Main parameters of power-split hybrid electric vehicle [43].

Component	Parameter	Value
Internal Combustion Engine	Type	Four-cylinder in-line gasoline engine
	Maximum power	57 kW @ 4500 RPM
	Maximum torque	110 Nm @ 4500 RPM
Electric motor	Type	AC motor
	Maximum power	35 kW @ 1040-5600 RPM
	Maximum torque	30 kW @ 3000-5500 RPM
Battery	Energy capacity	5 kWh/battery pack
	Charging capacity	2.3 Ah/battery unit
	Battery cell layout	110 serial x 6 parallel

3. Reinforcement learning-based energy management strategy

Reinforcement learning is a machine learning method that supports the learning process of an agent through its interactions within an environment. This learning process allows an agent to discover through trial and error for steps it should take to achieve a certain goal. The main goal of an RL agent is to obtain an optimal strategy that maximizes a certain goal through the interactions between the agent and the environment. This strategy means choosing the most appropriate action in each situation. The agent learns by experience which actions are more beneficial under different situations, and thanks to these experiences, it tends to choose the most useful action when faced with similar situations in the future. The biggest advantage of RL is that it allows the agent to discover on its own without providing pre-determined solutions [13, 44].

In the HEV energy management system, the RL mechanism aims to allocate the requested power in the most efficient way by making an intelligent distribution among different power sources to meet the power requirement of the vehicle. Here, the agent is the EMS itself and makes decisions to optimize the power management of the vehicle. The agent efficiently distributes the power requirement among energy sources such as engine, motor and battery, which increases the energy efficiency of the vehicle and preserves the battery health [45].

RL problems are usually addressed within the framework of Markov decision process (MDP), which models sequential decision processes under uncertainty. MDP forms the basic theoretical basis of complex learning problems such as RL and DRL, and its goal is to develop a strategy that will obtain the highest expected gain (reward) by learning from the interactions between the agent and the environment. In this process, the agent evaluates the current situation and discovers which action will maximize the reward. An MDP consists of four basic components: While the state (S) represents the current conditions of the agent; action Set (A) covers all possible actions that the agent can choose at that moment. Reward (R) indicates the feedback or gain that the agent receives as a result of an action taken in a situation; transition probability (P) expresses the probabilities that the agent will transition from one state to another as a result of an action.

These components of MDP are as follows: The MDP is represented by these fundamental components:

$$\mathrm{MDP} = \{S, A, R, P\} \tag{14}$$

In this study, to control the energy management system of a HEV, the agent observes particular states related to the vehicle's performance and energy efficiency, and selects the appropriate action based on these observations. Here, the "states" (S) are defined as the torque required to meet the vehicle's power demand (T_{dem}) and the battery's SOC. The T_{dem} varies according to the vehicle load and driving conditions, while the SOC indicates the battery's health and remaining energy capacity. By considering them as two state variables, the agent selects actions that optimize energy management, enhancing both the vehicle's performance and efficiency.

$$A = \{T_m\} \tag{15}$$

In the RL model, the reward function plays a critical role as it determines how favorable a particular action taken by the agent is in a given state. In this study, the reward function for the energy management controller consists of two main components: the instantaneous fuel consumption of the engine and the change in the battery's SOC. These two components aim to both maintain fuel efficiency and optimize the battery's health. Therefore, the objective function is then defined as the negative of the reward function as follows:

$$R = -\left[\alpha \cdot \dot{m}_{\text{fuel}}(t) + \beta \cdot (SOC_{\text{ref}} - SOC(t))^2\right] \tag{16}$$

where t represents the time step with the agent receives a new reward value at each time step. α is the coefficient term for fuel consumption, determining the impact of fuel consumption on the reward function. β is the weighting coefficient for the battery SOC variation, indicating the influence of the difference between the reference SOC value SOC_{ref} and the current SOC on the reward. SOC_{ref} is the reference state of charge of the battery, assumed to be kept as 60%, which represents the ideal SOC level to ensure optimal battery performance.

This function encourages the controller to choose the most appropriate control inputs to maintain energy efficiency and battery charge level simultaneously. When the instantaneous fuel consumption of the engine is reduced and the SOC value of the battery is kept close to the reference value, the reward value is increased, so that the controller prefers this situation in future decisions. This structure allows the controller to both optimize the vehicle's performance and manage energy in an efficient way. It encourages the controller to choose the most appropriate inputs to maintain energy efficiency and battery health.

3.1. Soft actor-critic (SAC) algorithm

The The SAC is an algorithm that is widely used in the field of DRL, which offers effective performance in continuous action spaces and stands out with its stable learning properties. Unlike traditional DRLs, SAC optimizes the exploration-exploitation balance with entropy regulation. Thus, it encourages agents to explore different actions and reduces the risk of getting stuck in local maxima by offering a wider exploration space during the learning process.

The algorithm adopts a "soft" policy approach, thanks to the entropy term added to the reward function, which maintains a certain degree of uncertainty and tries more diverse actions. In this way, policies not only target the maximum reward, but also become more flexible and durable during the exploration process.

SAC's actor-critic structure allows it to optimize both policy learning (actor) and the value function (critic) at the same time. This structure differs from algorithms such as QL and DQN, which focus only on the action-value function. QL and DQN are more suitable for discrete action domains and cannot show optimal performance in continuous action domains. On the other hand, SAC uses the actor-critic architecture to directly learn policies in the continuous action space of the actor and enables more precise decisions to be made in the action space [46]. In the SAC algorithm, the value functions are estimated by two separate critic networks using the twin critic structure. This prevents overestimation problems as in DQN and allows more stable results in the estimates. In addition, the target smoothing mechanism in SAC allows the actor to learn its actions more stably. SAC learns a more reliable and flexible policy by maintaining a certain level of uncertainty for each action, which provides a significant advantage under complex scenarios that require continuous adaptation [47, 48].

The SAC algorithm also uses a double Q-function to increase stability. The double Q-function structure prevents overestimation of target Q-values, increasing the reliability of learning. At the same time, while updating the policy networks, the values obtained from the Q-functions are taken as reference to ensure that the policy is directed correctly. The framework of the SAC algorithm is given in Figure 2.

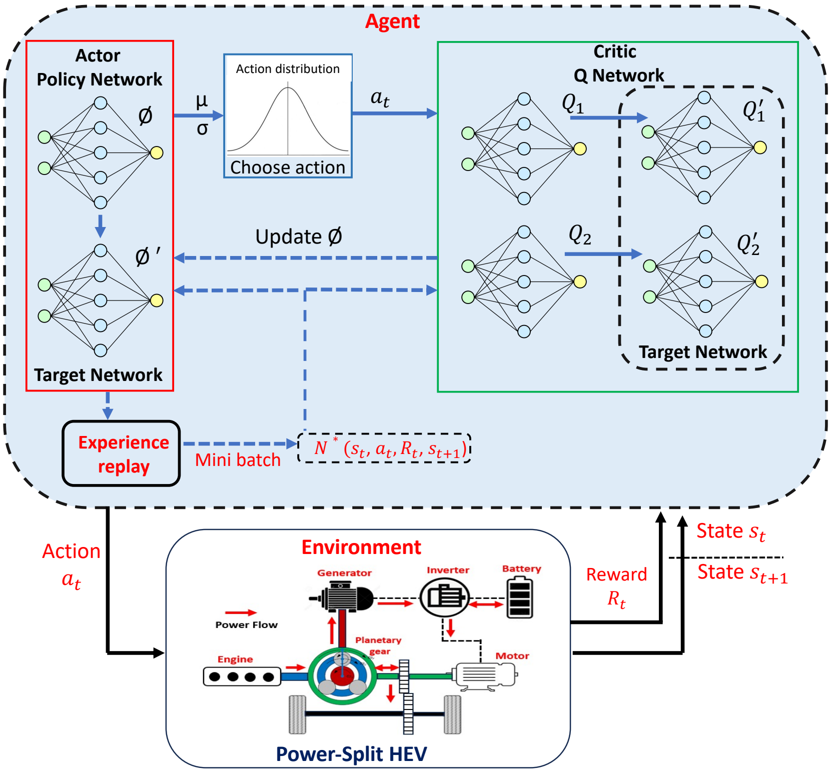


Figure 2. SAC algorithm structure.

In the SAC algorithm, the target value of the Q-functions, $y(r, s', d)$, is calculated by combining the reward obtained in the next state and the expected Q value of the policy. This target value is used as a reference during the update phase to enable the Q-functions to make more accurate predictions. Formulating the target value in this way in the SAC algorithm helps to stabilize the learning process. The formula for calculating this target value is as follows:

$$y(r, s', d) = r + \gamma(1-d)\left(\min_{i=1,2} Q_{\text{targ},i}(s', \tilde{a}) - \alpha \log \pi_\theta(\tilde{a}|s')\right) \tag{17}$$

where, r is the received reward, γ is the discount factor, d is the indicator of the terminal state, $Q_{\text{targ},i}$ represents the target Q-functions (indicating the double Q structure), $\tilde{a} \sim \pi_\theta(\cdot|s')$ is the action sampled by the policy, and α is the entropy regularization coefficient. The entropy term $\alpha \log \pi_\theta(\tilde{a}|s')$ is added to the reward function to enhance the exploration capacity of the policy.

Both Q-functions are updated by minimizing the squared difference between the predicted Q value and the target Q value in order to ensure reliable learning in the SAC algorithm. This update helps each Q-function better estimate the long-term reward expectation for each state-action pair. Updating the Q-functions in this way enables a more accurate evaluation of current and future rewards at each step. The update is performed using gradient descent, where the expression $\left(Q_{\phi_i}(s,a) - y(r,s',d)\right)^2$, known as the error term, is minimized. This error term represents the difference between the predicted Q value and the target Q value, and the smaller this difference, the higher the accuracy of the Q-functions.

This update process is carried out separately for each Q-function, and the parameters ϕ_i of each function are optimized using gradient descent. The update rule is as follows:

$$\phi_i \leftarrow \phi_i - \eta_Q \nabla_{\phi_i} \frac{1}{|B|} \sum_{(s,a,r,s',d)\in B} \left(Q_{\phi_i}(s,a) - y(r,s',d)\right)^2 \tag{18}$$

where, η_Q is the learning rate, B is the mini-batch of sampled experiences, and Q_{ϕ_i} represents the Q-function parameters. This update rule reduces the discrepancy between the predicted and target values for each Q-function, allowing the agent to make more accurate reward predictions and contribute to learning the optimal policy.

The policy is updated based on the values obtained from the Q-functions, taking into account the entropy regularization. The policy update rule is as follows:

$$\nabla_\theta J(\theta) \approx \nabla_\theta \frac{1}{|B|} \sum_{s \in B} \left(\min_{i=1,2} Q_{\phi_i}(s, \tilde{a}_\theta(s)) - \alpha \log \pi_\theta(\tilde{a}_\theta(s)|s) \right) \tag{19}$$

where, $\tilde{a}_\theta(s)$ is the action sampled by the policy in a given state, and the entropy term is used to maintain diversity in the policy's learning process. The target Q-functions are updated to slowly track the main Q-functions. The update rule is as follows:

$$\phi_{\text{targ},i} \leftarrow \rho \phi_{\text{targ},i} + (1-\rho)\phi_i \tag{20}$$

where, ρ is the target smoothing coefficient.

Algorithm 1 Soft Actor-Critic

```
**Input:** Initial policy parameters $\theta$, Q-function parameters $\phi_1$, $\phi_2$, empty replay buffer $\mathcal{D}$
Set target parameters equal to main parameters: $\phi_{\text{targ},1} \leftarrow \phi_1$, $\phi_{\text{targ},2} \leftarrow \phi_2$
**repeat**
    Observe state $s$ and select action $a \sim \pi_\theta(\cdot|s)$
    Execute $a$ in the environment
    Observe next state $s'$, reward $r$, and done signal $d$ to indicate whether $s'$ is terminal
    Store $(s, a, r, s', d)$ in replay buffer $\mathcal{D}$
    **if** $s'$ is terminal **then**
        Reset environment state
    **end if**
    **if** it's time to update **then**
        **for** $j$ in range (number of updates) **do**
            Randomly sample a batch of transitions, $B = \{(s, a, r, s', d)\}$ from $\mathcal{D}$
            Compute targets for the Q-functions:
                $y(r, s', d) = r + \gamma(1-d)\left(\min_{i=1,2} Q_{\text{targ},i}(s', \tilde{a}) - \alpha \log \pi_\theta(\tilde{a}|s')\right), \quad \tilde{a} \sim \pi_\theta(\cdot|s')$
            Update Q-functions by one step of gradient descent:
                $\nabla_{\phi_i} \frac{1}{|B|} \sum_{(s,a,r,s',d) \in B} \left(Q_{\phi_i}(s,a) - y(r,s',d)\right)^2 \quad \text{for } i = 1, 2$
            Update policy by one step of gradient ascent:
                $\nabla_\theta \frac{1}{|B|} \sum_{s \in B} \left(\min_{i=1,2} Q_{\phi_i}(s, \tilde{a}_\theta(s)) - \alpha \log \pi_\theta(\tilde{a}_\theta(s)|s)\right)$
    where $\tilde{a}_\theta(s)$ is a sample from $\pi_\theta(\cdot|s)$, differentiable with respect to $\theta$ via the reparameterization trick.
            Update target networks:
                $\phi_{\text{targ},i} \leftarrow \rho \phi_{\text{targ},i} + (1-\rho)\phi_i \quad \text{for } i = 1, 2$
        **end for**
    **end if**
**until** convergence
```

In summary, the SAC algorithm combines entropy-regularized policy updates with stable Q-function learning to achieve reliable performance in continuous action spaces. By balancing exploration and exploitation through entropy, SAC enables agents to learn robust policies that adapt well to complex environments. This combination of soft policy learning and double Q-function structure makes SAC a powerful tool in deep reinforcement learning, particularly in tasks requiring stability and effective decision-making in high-dimensional state and action spaces. The structure of SAC is presented in Algorithm 1.

4. Experiment results

This study performs a comprehensive analysis using traditional QL, DQN, and DDPG algorithms to evaluate and compare the proposed SAC-based EMS performance. SOC tracking, fuel consumption, and adaptiveness are evaluated under all four algorithms. The study aims to reveal the performance of the EMS against the changes in speed and torque demands and to analyze their performance under different settings. The hyperparameter settings of the SAC-based EMS used in this evaluation process are meticulously selected to stabilize the learning process of the model and to provide effective control. Critical parameters such as learning rates, entropy target, and network structure support the successful learning of the model under different driving cycles. Detailed hyperparameter settings of the SAC algorithm are presented in Table 3.

Table 3. Hyperparameters of the SAC Algorithm.

Parameter	Value
Simulation time T	7451 s
Sample time T_s	1 s
Episodes number M	500
Maximum steps per episode	7451
Actor learning rate	0.00001
Critic learning rate	0.0001
Discount factor γ	0.99
Entropy target	-0.5
Score averaging window length	10
Stop training criteria	Episode Reward
Stop training value	∞

A comprehensive driving cycle called “ALL-CYC” is employed by combining the widely used driving cycles. This driving cycle includes six standard driving cycles: New European Driving Cycle (NEDC), World Harmonized Light Vehicle Test Procedure (WLTP), Urban Dynamometer Driving Cycle (UDDS), Highway Fuel Economy Test (HWFET), New York City Cycle (NYCC) and LA92 cycle. NEDC is an old test procedure used especially in Europe and simulates low-speed, low-acceleration driving conditions, generally representing urban and light traffic driving scenarios [49]. WLTP, on the other hand, was developed based on more realistic driving data, covers various driving conditions with varying speed and acceleration profiles, and is currently accepted as the standard test procedure in many countries [50]. UDDS is a cycle used in the USA to model urban traffic conditions and has a low-speed profile with frequent stop-and-go movements [51]. HWFET is a cycle that represents highway driving and generally reflects high-speed steady driving conditions, so it is used to evaluate fuel economy [52]. NYCC tests low-speed driving performance by simulating heavy traffic conditions and frequent stop-and-go driving specific to New York City [53]. LA92 represents the dynamic structure of Los Angeles city traffic with complex speed changes and high acceleration [54].

The speed profile of this driving cycle is presented in detail in Figure 3, which shows the speed changes in the segments corresponding to each standard cycle. In addition, the basic features of the ALL-CYC cycle, such as distance, total time, average speed, and maximum speed, are summarized in Table 4. This comprehensive analysis allows for a more accurate evaluation of parameters such as vehicle performance and fuel consumption under different driving conditions, forming the ALL-CYC cycle.

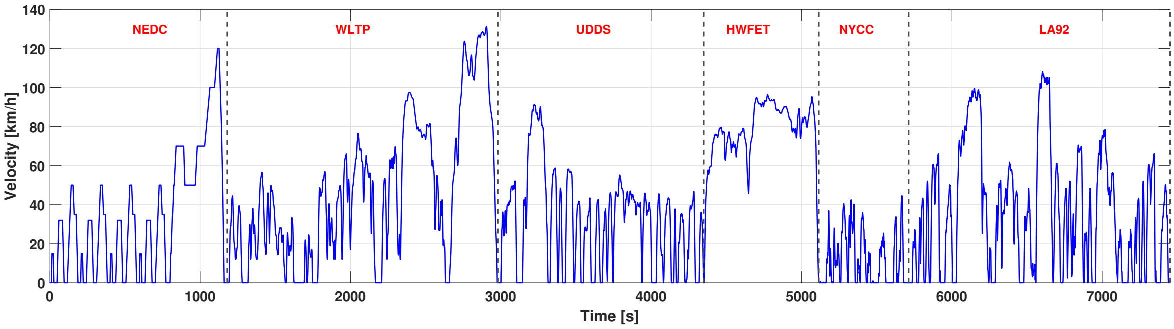


Figure 3. ALL-CYC Velocity Profile.

Table 4. Characteristics of driving cycles.

Driving Cycle	Distance (km)	Duration (s)	Average Speed (km/h)	Maximum Speed (km/h)
NEDC	11.0	1180	33.6	120
WLTP	23.3	1800	46.6	131
UDDS	12.0	1369	31.5	91
HWFET	16.5	765	77.8	97
NYCC	2.0	598	12.0	44
LA92	14.5	1435	36.4	106

The training results under the ALL-CYC driving cycle are examined in detail, presenting the final SOC values and fuel consumption in Table 5, while the SOC graph is shown in Figure 4. Note that we can capture all possible action spaces with ALL-CYC driving cycle since it presents a high possibility of driving scenarios. The engine operating points are shown in Figure 5. The color bar in the figures represents the brake specific fuel consumption (BSFC) that shows the efficiency of the engine. BSFC is measured in g/s and shows the amount of fuel consumed by the engine to produce a unit of power. Lower BSFC values (blue-green) indicate high efficiency, while higher values (yellow-red) indicate low efficiency. The contours define the BSFC regions, and the maximum torque curve defines the operating limits of the engine. All algorithms generally try to minimize fuel consumption by operating the engine in low BSFC regions. However, it is seen that the operating points of SAC are concentrated in low-consumption regions and closer to the optimal BSFC curves. This shows that SAC minimizes fuel consumption more effectively and adapts better to dynamic conditions. Moreover, there are significant differences in the battery SOC control and fuel consumption performance under different algorithms. The QL, DQN, and DDPG algorithms deviate significantly from the reference SOC value of 60%. The QL algorithm achieves an end of SOC of 67.93%, while the DQN algorithm remains at 65.63% and the DDPG algorithm remains at 63.82%. This shows that all three algorithms tend to overcharge the battery, potentially causing unnecessary energy consumption. Overcharging the battery can lead to a shortened battery life and increased energy losses in the long term. On the other hand, the SAC algorithm exhibits a good reference tracking performance, which is close to the reference SOC level with a value of 61.93%. This situation shows that SAC adapts better to driving conditions and can keep the battery level in the optimal range thanks to its adaptive structure. This provides a critical advantage in terms of both preserving battery health and adopting a balanced approach to energy management. When evaluated in terms of fuel consumption, the SAC algorithm achieves the lowest fuel consumption value of 2.807 L and demonstrates superior performance in terms of energy efficiency. In contrast, the QL algorithm shows inefficient energy usage due to battery overcharging with 3.17% higher fuel consumption. The DQN algorithm shows a moderate performance with 1.95% higher fuel consumption compared to SAC. In addition, the DDPG algorithm shows a closer performance to SAC with a fuel consumption difference of 1.13%. Overally, the SAC algorithm consumes 3.17% less fuel than QL, 1.95% less than DQN, and 1.13% less than DDPG. These differences clearly show that the SAC algorithm offers a more balanced and adaptive control strategy in terms of both energy management and fuel efficiency. This improved control capability of SAC enables it to dynamically adapt to different driving conditions and better optimize fuel consumption.

Table 5. Comparison of battery SOC and fuel consumption values of the proposed RL/DRL-based approaches.

Algorithm	End-of-cycle SOC (%)	Fuel Consumption (L)	Fuel Consumption per 100 km (L/100 km)	Fuel Consumption Difference (%)
QL	67.93	2.896	3.519	3.17
DQN	65.63	2.862	3.478	1.95
DDPG	63.82	2.839	3.449	1.13
SAC	61.93	2.807	3.411	-

Reward graphs obtained during the training process play a critical role in algorithm performance evaluation. These graphs visualize how the algorithm learns the environment, its learning speed and stability, and allow comparison of different methods. Figure 6 compares the training results of QL, DQN, DDPG and SAC algorithms in the ALL-CYC driving cycle and the obtained reward values. In the QL graph, it is seen that the reward values are quite fluctuating throughout the training process. Low reward values are obtained at the beginning, and although some improvements are recorded in the upcoming sections, the overall performance exhibited an unstable structure. In the DQN graph, the reward values became stable after approximately the 130th episode and exhibited a stable performance during the subsequent training process. However, slight fluctuations are observed in some sections. In the DDPG graph, it is observed that the reward values stabilize quickly and exhibit a stable structure throughout the training process, but exhibit offsets. In the SAC graph, the performance of the algorithm stands out clearly. In the training process, a stable and high reward value is reached after approximately the 50th episode. The average reward value is obtained in a fixed line and stability is preserved throughout the training process. This result shows that the SAC algorithm can learn the optimal policy more efficiently and provide higher performance compared to the other three algorithms.

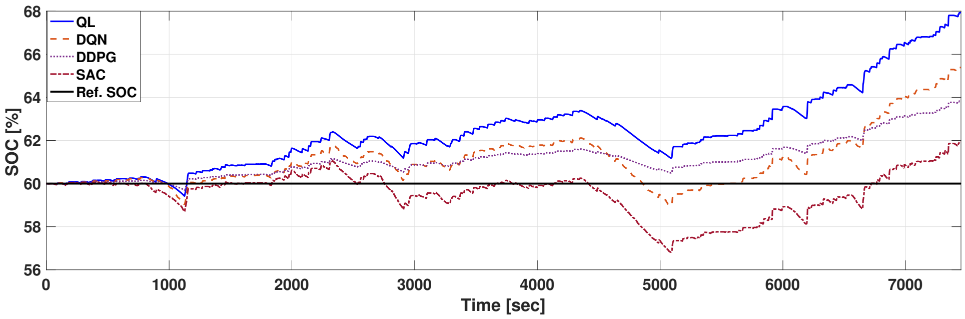
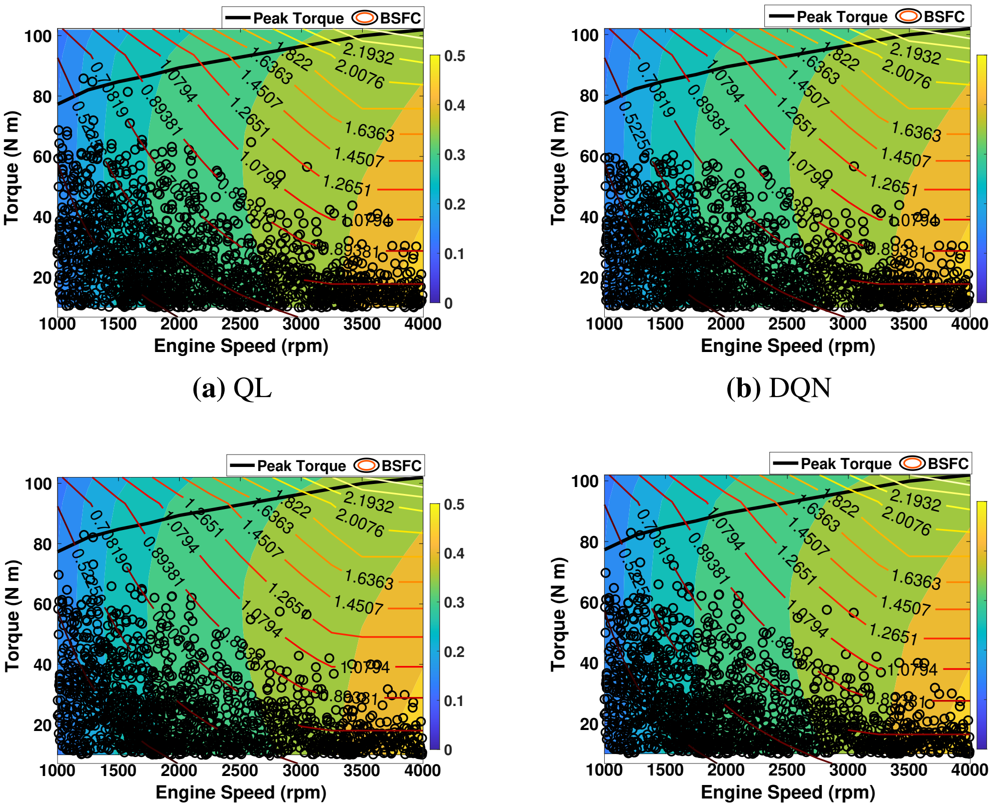


Figure 4. SOC graph in ALL-CYC driving cycle of RL/DRL-based EMS models.



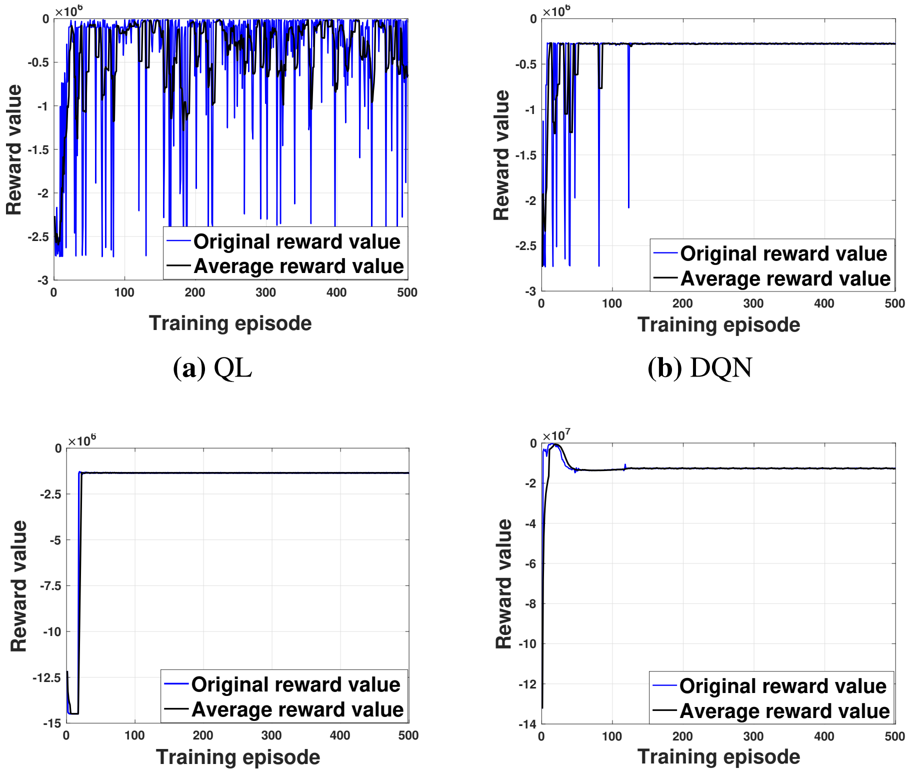
(**a**) QL

(**b**) DQN

(**c**) DDPG

(**d**) SAC

Figure 5. Engine operating points for the ALL-CYC driving cycle.



(**a**) QL

(**b**) DQN

(**c**) DDPG

(**d**) SAC

Figure 6. Training results for RL/DRL methods under ALL-CYC driving cycle.

In this section, the SAC algorithm is trained using the ALL-CYC driving cycle. ALL-CYC includes a widely used and standard driving cycle. To evaluate the adaptiveness feature of the proposed SAC-based EMS model, we use a completely different driving cycle, which is not included in the ALL-CYC dataset. Adaptiveness is a vital feature of the SAC algorithm, presenting good performance due to its ability to dynamically adapt to different environments and conditions.

The FTP-75 (Federal Test Procedure 75) Driving Cycle is a test driving cycle used by the U.S. Environmental Protection Agency to evaluate automobile emissions, fuel consumption, and energy management strategies. It was developed specifically to simulate urban driving conditions. This driving cycle covers a distance of approximately 17.7 km in a total time of 1875 seconds and has an average speed of 34.1 km/h. The speed profile of the FTP-75 driving cycle is given in Figure 7.

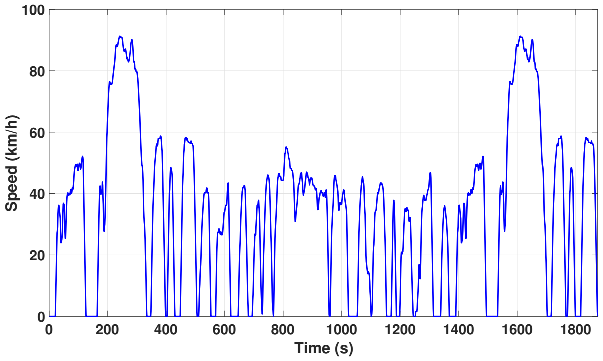


Figure 7. FTP-75 driving cycle velocity profile.

The training results under the FTP-75 driving cycle are examined in detail, demonstrating the final SOC values and fuel consumption in Table 6, and the SOC graph is presented in Figure 8. Additionally, the engine operating points are illustrated in Figure 9. Since the reward definition includes fuel consumption minimization, we can see that the controllers aim to run the engine in a fuel-efficient way, the SAC methods still outperform the other methods. According to the table and SOC graph results, the SAC algorithm exhibits a more adaptive and balanced performance in terms of both fuel consumption and SOC control. QL, DQN, and DDPG algorithms result in a higher SOC level, which leads to overcharging of the battery; this may cause energy losses and shorten the battery life. In contrast, the SAC algorithm obtains a SOC level closer to the reference SOC value, ensuring that the battery is kept at the optimum level. In terms of fuel consumption, SAC offers the lowest value (3,258 L/100 km), showing 6.26% less fuel consumption than QL, 3.01% less than DQN, and 2.03% less than DDPG. This shows that the SAC algorithm dynamically adapts to different driving conditions and optimizes energy efficiency.

Table 6. Comparison of battery SOC and fuel consumption values of the proposed RL/DRL-based approaches under the FTP-75 driving cycle.

Algorithm	End-of-cycle SOC (%)	Fuel Consumption (L)	Fuel Consumption per 100 km (L/100 km)	Fuel Consumption Difference (%)
QL	64,60	0.6152	3.462	6.26
DQN	63.33	0.5964	3.356	3.01
DDPG	62.85	0.5901	3.325	2.03
SAC	62.34	0.578	3.258	-

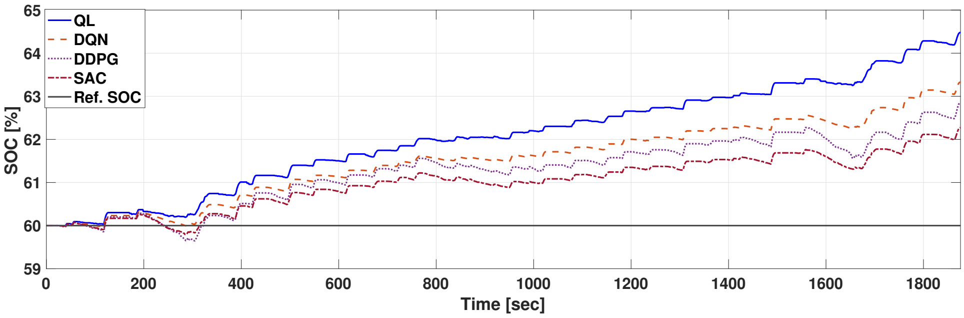
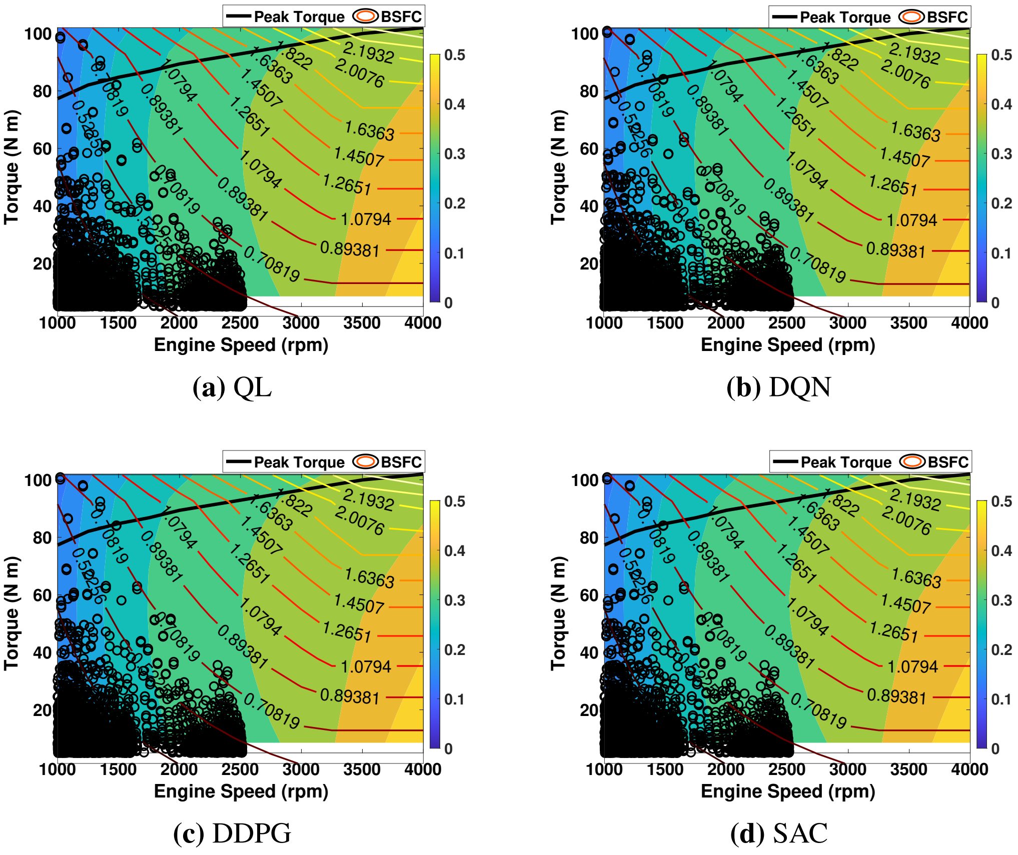


Figure 8. SOC graph in FTP-75 driving cycle of RL/DRL-based EMS models.



(a) QL

(b) DQN

(c) DDPG

(d) SAC

Figure 9. Engine operating points for the FTP-75 driving cycle.

5. Conclusion

In this study, a SAC-based EMS is presented for HEV energy management problem. Due to its soft policy approach via an added entropy term during action selection in policy computation, it maintains an improved performance under more diverse drive conditions. We train the SAC actor-critic structure under a set of six driving cycles, demonstrating a good exploration and exploitation feature in performance metrics assessment. A completely different driving cycle is also utilized to light out the adaptiveness feature of the SAC-based EMS design. Specifically, the fuel consumption and battery charge sustainability are evaluated against the QL-based EMS, the DQN-based EMS, and the DDPG-based EMS benchmarks in the generalization of the proposed SAC-based EMS. It is found that the SAC-based EMS outperforms in adaption to dynamic driving conditions and improves the energy efficiency of HEV, thanks to its pursuit of maximum entropy feature for encouraging more exploration and obtaining more stable training performance. These findings demonstrate that the SAC-based approach offers significant advantages in EMS design. In the future, testing such approaches on large scales and real-world applications will make significant contributions to the field of connected driving.

Acknowledgments

This work is supported by the Scientific and Technological Research Council of Türkiye with project number 121E260 under the grant name CAREER.

Conflicts of Interests

There is no conflict of interest to declare.

Authors contribution

Validation, Ozan Yazar; methodology, Serdar Coskun; formal analysis, Ozan Yazar; investigation, Fengqi Zhang; writing, Fengqi Zhang; writing - original draft, Ozan Yazar; writing - review and editing, Serdar Coskun; supervision, Ozan Yazar, Fengqi Zhang. All authors have read and agreed to the published version of the manuscript.

References

[1] Holechek JL, Geli HM, Sawalhah MN, Valdez R. A global assessment: can renewable energy replace fossil fuels by 2050? *Sustainability* 2022 14(8):4792.

[2] Umar M, Ji X, Kirikkaleli D, Alola AA. The imperativeness of environmental quality in the United States transportation sector amidst biomass-fossil energy consumption and growth. *Journal of Cleaner Production* 2021 285:124863.

[3] Li Z, Khajepour A, Song J. A comprehensive review of the key technologies for pure electric vehicles. *Energy* 2019 182:824–839.

[4] Hannan MA, Azidin F, Mohamed A. Hybrid electric vehicles and their challenges: A review. *Renewable and Sustainable Energy Reviews* 2014 29:135–150.

[5] Hawkins TR, Gausen OM, Strømman AH. Environmental impacts of hybrid and electric vehicles—a review. *The International Journal of Life Cycle Assessment* 2012 17:997–1014.

[6] Balali Y, Stegen S. Review of energy storage systems for vehicles based on technology, environmental impacts, and costs. *Renewable and Sustainable Energy Reviews* 2021 135:110185.

[7] Li L, Coskun S, Langari R, Xi J. Incorporated vehicle lateral control strategy for stability and enhanced energy saving in distributed drive hybrid bus. *Applied Soft Computing* 2021 111:107617.

[8] Zhang F, Wang L, Coskun S, Pang H, Cui Y, *et al.* Energy management strategies for hybrid electric vehicles: Review, classification, comparison, and outlook. *Energies* 2020 13(13):3352.

[9] Onori S, Serrao L, Rizzoni G. *Hybrid electric vehicles: Energy management strategies*, vol. 13, Springer2016.

[10] Sulaiman N, Hannan M, Mohamed A, Majlan E, Daud WW. A review on energy management system for fuel cell hybrid electric vehicle: Issues and challenges. *Renewable and Sustainable Energy Reviews* 2015 52:802–814.

[11] Saiteja P, Ashok B. Critical review on structural architecture, energy control strategies and development process towards optimal energy management in hybrid vehicles. *Renewable*

and Sustainable Energy Reviews 2022 157:112038.

[12] Ganesh AH, Xu B. A review of reinforcement learning based energy management systems for electrified powertrains: Progress, challenge, and potential solution. *Renewable and Sustainable Energy Reviews* 2022 154:111833.

[13] Kaelbling LP, Littman ML, Moore AW. Reinforcement learning: A survey. *Journal of artificial intelligence research* 1996 4:237–285.

[14] Vázquez-Canteli JR, Nagy Z. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied energy* 2019 235:1072–1089.

[15] Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, *et al.* The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164* 2018 .

[16] Wang L, Cui Y, Zhang F, Coskun S, Liu K, *et al.* Stochastic speed prediction for connected vehicles using improved bayesian networks with back propagation. *Science China Technological Sciences* 2022 65(7):1524–1536.

[17] Wang X, Wang S, Liang X, Zhao D, Huang J, *et al.* Deep reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems* 2022 35(4):5064–5078.

[18] Li SE. Deep reinforcement learning. In *Reinforcement learning for sequential decision and optimal control*, Springer2023, pp. 365–402.

[19] Wang Y, Qiu D, He Y, Zhou Q, Strbac G. Multi-agent reinforcement learning for electric vehicle decarbonized routing and scheduling. *Energy* 2023 284:129335.

[20] Wang Y, Qiu D, Strbac G, Gao Z. Coordinated electric vehicle active and reactive power control for active distribution networks. *IEEE Transactions on Industrial Informatics* 2022 19(2):1611–1622.

[21] Chen Z, Gu H, Shen S, Shen J. Energy management strategy for power-split plug-in hybrid electric vehicle based on MPC and double Q-learning. *Energy* 2022 245:123182.

[22] Tresca L, Pulvirenti L, Rolando L, Millo F. Development of a deep Q-learning energy management system for a hybrid electric vehicle. *Transportation Engineering* 2024 16:100241.

[23] Zhu Z, Liu Y, Canova M. Energy management of hybrid electric vehicles via deep Q-networks. In *2020 American Control Conference (ACC)*, IEEE2020 pp. 3077–3082.

[24] Han L, Yang K, Ma T, Yang N, Liu H, *et al.* Battery life constrained real-time energy management strategy for hybrid electric vehicles based on reinforcement learning. *Energy* 2022 259:124986.

[25] Ahmadian S, Tahmasbi M, Abedi R. Q-learning based control for energy management of series-parallel hybrid vehicles with balanced fuel consumption and battery life. *Energy and AI* 2023 11:100217.

[26] Lee H, Song C, Kim N, Cha SW. Comparative analysis of energy management strategies for HEV: Dynamic programming and reinforcement learning. *IEEE Access* 2020 8:67112–67123.

[27] Xu B, Tang X, Hu X, Lin X, Li H, *et al.* Q-learning-based supervisory control adaptability investigation for hybrid electric vehicles. *IEEE Transactions on Intelligent Transportation Systems* 2021 23(7):6797–6806.

[28] Tang X, Chen J, Pu H, Liu T, Khajepour A. Double deep reinforcement learning-based energy management for a parallel hybrid electric vehicle with engine start–stop strategy. *IEEE Transactions on Transportation Electrification* 2021 8(1):1376–1388.

[29] Zheng C, Zhang D, Xiao Y, Li W. Reinforcement learning-based energy management strategies of fuel cell hybrid vehicles with multi-objective control. *Journal of Power Sources* 2022 543:231841.

[30] Lian R, Peng J, Wu Y, Tan H, Zhang H. Rule-interposing deep reinforcement learning based energy management strategy for power-split hybrid electric vehicle. *Energy* 2020 197:117297.

[31] Yazar O, Coskun S, Li L, Zhang F, Huang C. Actor-critic TD3-based deep reinforcement learning for energy management strategy of HEV. In *2023 5th International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, IEEE2023 pp. 1–6.

[32] Lin X, Zhou K, Mo L, Li H. Intelligent energy management strategy based on an improved reinforcement learning algorithm with exploration factor for a plug-in PHEV. *IEEE Transactions on Intelligent Transportation Systems* 2021 23(7):8725–8735.

[33] Liu ZE, Zhou Q, Li Y, Shuai S, Xu H. Safe deep reinforcement learning-based constrained optimal control scheme for HEV energy management. *IEEE Transactions on Transportation Electrification* 2023 9(3):4278–4293.

[34] Wang J, Du C, Yan F, Duan X, Hua M, *et al.* Energy Management of a Plug-in Hybrid Electric Vehicle Using Bayesian Optimization and Soft Actor-Critic Algorithm. *IEEE Transactions on Transportation Electrification* 2024 .

[35] Li T, Cui W, Cui N. Soft actor-critic algorithm-based energy management strategy for plug-in hybrid electric vehicle. *World Electric Vehicle Journal* 2022 13(10):193.

[36] Liu ZE, Zhou Q, Li Y, Shuai S. An intelligent energy management strategy for hybrid vehicle with irrational actions using twin delayed deep deterministic policy gradient. *IFAC-PapersOnLine* 2021 54(10):546–551.

[37] Liu ZE, Li Y, Zhou Q, Li Y, Shuai B, *et al.* Deep Reinforcement Learning based Energy Management for Heavy Duty HEV considering Discrete-Continuous Hybrid Action Space. *IEEE Transactions on Transportation Electrification* 2024 .

[38] Liu ZE, Li Y, Zhou Q, Shuai B, Hua M, *et al.* Real-time energy management for HEV combining naturalistic driving data and deep reinforcement learning with high generalization. *Applied Energy* 2025 377:124350.

[39] Xu D, Zheng C, Cui Y, Fu S, Kim N, *et al.* Recent progress in learning algorithms applied in energy management of hybrid vehicles: A comprehensive review. *International Journal of Precision Engineering and Manufacturing-Green Technology* 2023 10(1):245–267.

[40] Shen S, Gao S, Liu Y, Zhang Y, Shen J, *et al.* Real-time energy management for plug-in hybrid electric vehicles via incorporating double-delay Q-learning and model prediction control. *IEEE Access* 2022 10:131076–131089.

[41] Dong H, Dong H, Ding Z, Zhang S, Chang T. *Deep Reinforcement Learning*, Springer2020.

[42] Haarnoja T, Zhou A, Abbeel P, Levine S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, PMLR2018 pp. 1861–1870.

[43] Rousseau A, Kwon J, Sharer P, Pagerit S, Duoba M. Integrating data, performing quality assurance, and validating the vehicle model for the 2004 Prius using PSAT. Tech. rep., SAE Technical Paper, 2006.

[44] Ernst D, Louette A. Introduction to reinforcement learning. *Feuerriegel, S., Hartmann, J., Janiesch, C., and Zschech, P* 2024 pp. 111–126.

[45] Coskun S, Yazar O, Zhang F, Li L, Huang C, *et al.* A multi-objective hierarchical deep reinforcement learning algorithm for connected and automated HEVs energy management. *Control Engineering Practice* 2024 153:106104.

[46] Christodoulou P. Soft actor-critic for discrete action settings. *arXiv preprint arXiv:1910.07207* 2019 .

[47] Xu D, Cui Y, Ye J, Cha SW, Li A, *et al.* A soft actor-critic-based energy management strategy for electric vehicles with hybrid energy storage systems. *Journal of Power Sources* 2022 524:231099.

[48] Wang Y, Wu Y, Tang Y, Li Q, He H. Cooperative energy management and eco-driving of plug-in hybrid electric vehicle via multi-agent reinforcement learning. *Applied Energy* 2023 332:120563.

[49] Shim BJ, Park KS, Koo JM, Jin SH. Work and speed based engine operation condition analysis for new European driving cycle (NEDC). *Journal of mechanical science and technology* 2014 28:755–761.

[50] Demuynck J, Bosteels D, De Paepe M, Favre C, May J, *et al.* Recommendations for the new WLTP cycle based on an analysis of vehicle emission measurements on NEDC and CADC. *Energy Policy* 2012 49:234–242.

[51] Kruse RE, Huls TA. Development of the federal urban driving schedule 1973 .

[52] Mersky AC, Samaras C. Fuel economy testing of autonomous vehicles. *Transportation Research Part C: Emerging Technologies* 2016 65:31–48.

[53] Seers P, Nachin G, Glaus M. Development of two driving cycles for utility vehicles. *Transportation Research Part D: Transport and Environment* 2015 41:377–385.

[54] Lin J, Niemeier DA. An exploratory analysis comparing a stochastic driving cycle to California's regulatory cycle. *Atmospheric Environment* 2002 36(38):5759–5770.