Optimization of Dynamic Energy Management Strategy for New Energy Vehicles Based on Multi-Agent Reinforcement Learning

Xiaoyu Zhang
Automotive Academy, Henan Communications Vocational and Technical College, Zhengzhou Henan, 450000, China
E-mail: zxiaoyhappy@163.com

Keywords: battery degradation, energy management strategies, fuel economy, new energy vehicle (NEV), power distribution, scalable satin bowerbird optimizer-driven multi-agent deep Q-Network (SSB-MADQN)

Received: April 14, 2025

The development of New Energy Vehicles (NEVs), such as battery electric vehicles, is vital to addressing global issues like environmental pollution and fossil fuel depletion. However, optimizing their energy management strategies (EMSs) is complex due to conflicting goals, dynamic driving conditions, and system nonlinearity. This study proposes a dynamic EMS based on Multi-Agent Reinforcement Learning (MARL) using a Scalable Satin Bowerbird Optimizer-driven Multi-Agent Deep Q-Network (SSB-MADQN). The approach aims to enhance fuel economy, maintain battery State of Charge (SOC), and reduce battery degradation in real-time driving scenarios. Prior to training, data preprocessing—including min-max normalization and Principal Component Analysis (PCA)—improves learning efficiency. The MADQN framework consists of agents representing subsystems such as the engine, battery, and regenerative braking, each trained using a deep Q-network with three hidden layers (128-64-32 neurons). The dataset comprises 5,000 samples with 13 features, including vehicle speed, power demand, and battery performance. Evaluated on HWFET and WLTC driving cycles, the proposed strategy reduces fuel consumption by 0.912 L (WLTC) and 0.681 L (HWFET) compared to traditional methods. It effectively regulates SOC and reduces high-power discharge events, confirming the robustness of MARL for adaptive and efficient EMS in NEVs.

Povzetek: Raziskava predlaga dinamično strategijo upravljanja z energijo (EMS) za NEV na osnovi MARL (SSB-MADQN). Optimizira porabo goriva, stanje napolnjenosti baterije (SOC) in zmanjšuje degradacijo, s čimer izboljša učinkovitost v realnem času.

1 Introduction

The growing demand for NEVs, which includes hybrids and battery electric vehicles, occurs because they serve as an environmentally friendly replacement for traditional internal combustion engine vehicles that offer improved air quality decreased greenhouse gas emissions, and reliable energy systems [1]. Strong worldwide climate change understanding, along with decreasing fossil fuel reserves, has made NEV development essential for countries implementing sustainable transportation solutions [2]. Conventional EMS approaches, such as rule-based, fuzzy logic, or model predictive control methods, rely on pre-defined heuristics or offline optimization and often fail to adapt in real-time to complex, dynamic environments like varying road gradients, traffic conditions, and driving behaviours [3]. The growing complexity of NEVs and their need for adaptive, real-time decision-making have thus pushed the investigation toward leveraging artificial intelligence (AI) techniques such as machine learning (ML) and reinforcement learning (RL) [4]. Figure 1 shows the dynamic energy management strategy for NEVs.

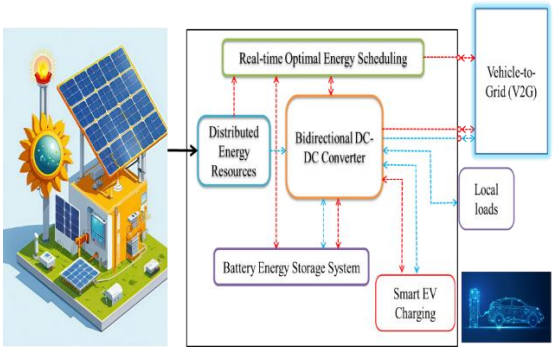

Figure 1: Dynamic energy management strategy for NEVs

Reinforcement learning has shown significant promise in EMS optimization by enabling systems to accumulate reward functions, such as fuel efficiency or battery health [5]. However, most existing RL-based EMS frameworks

operate under a single-agent paradigm, where the entire decision-making process is centralized, which limits scalability and does not fully represent the distributed nature of NEV components. In reality, energy management involves coordination between multiple subsystems [6].

The vehicle dynamics are modeled to include real-world constraints such as regenerative braking, load variations, and battery degradation metrics [7]. Despite various conventional EMS strategies yielding acceptable performance under ideal conditions, they often fail in unpredictable or highly dynamic driving environments. By leveraging the strengths of multi-agent systems and metaheuristic-optimized DL models, it offers a robust, adaptive, and intelligent EMS that is both scalable and energy-efficient. It highlights the transformative potential of AI-driven strategies in the automotive domain, particularly for real-time optimization and sustainable energy utilization in NEVs.

To address these limitations, MARL has emerged as an innovative solution for optimizing EMS in a decentralized and cooperative manner. In MARL-based EMS, different vehicle components are modeled as intelligent agents, such as a battery agent and an engine agent that learn to make decisions based on local observations and collaborate to achieve a global objective. It allows for distributed control, reduced computational complexity, and more effective adaptation to real-time driving dynamics. A novel MARL-based EMS framework is proposed using an SSB-MADQN. The SSB is a nature-inspired metaheuristic algorithm based on the mating behavior of satin bowerbirds, known for balancing exploration and exploitation efficiently. The aim is to enhance fuel economy, sustain battery SOC, and decrease battery degradation under dynamic driving conditions.

1.1 Key contribution

Data Collection: The dataset captures real driving conditions, fuel consumption, power distribution, and battery health metrics specific to NEV scenarios.

Data preprocessing: Applied data cleaning and min-max normalization to standardize input variables, ensuring consistent scale and reducing data noise for learning stability.

Feature extraction: Used PCA to extract 12 principal components, preserving 95% variance for improved training efficiency and dimensionality reduction.

Proposed method: SSB-MADQN, a MARL-based framework with decentralized agents and a Satin Bowerbird-optimized DQN for dynamic NEV energy management.

1.2 Motivation

The motivation for this research is driven by the need for more effective and adaptive energy management strategies for new energy vehicles (NEVs). Current systems face challenges in optimizing fuel efficiency, battery health, and driving performance simultaneously, especially under dynamic driving conditions. By leveraging Multi-Agent Reinforcement Learning (MARL) and the novel SSB-MADQN approach, this research aims to reduce fuel consumption while maintaining optimal battery SOC and minimizing degradation, ultimately contributing to more sustainable and efficient NEV operation in real-world scenarios.

The research is comprised of the following sections: In Section 2, a list of relevant works was presented. In Section 3, the methodology is described. In Section 4, the findings are presented. The discussion portion is provided in Section 5, and Section 6 contains the conclusion.

2 Related work

A novel multiple-input and multiple-output (MIMO) control technique based on Multi-Agent Deep Reinforcement Learning (MDARL) was examined in [8] for the multi-mode photovoltaic EV. Two learning agents would collaborate under the MDARL, utilizing the deep deterministic policy gradient (DDPG) algorithm by implementing a handshaking technique that provided a relevance ratio. To improve fuel economy, [9] provided a unique EV EMS based on the MDARL architecture. Under power limits, the EMS effectively achieved optimal power transmission between the engine and battery.

The optimal functioning of a fleet of EVs that were directed to supply power to a group of clients at various places was covered in [10]. MARL was used in a Decentralised Markov Decision Procedure reformulation framework to be practicable for a fleet of EVs to function well and provide energy to numerous clients at various places. A unique optimum energy management approach based on the suggested MDARL technique was presented in [11]. It used a deep neural network to train a strategy based on multi-agent deep deterministic policy gradient (MADDPG) learning capacity and stacked denoising auto-encoders. By considering the different characteristics of both electrical and thermal energies.

A MADRL optimization approach was proposed in [12] for energy control with EV charging development. To determine the optimal choice, the aggregator and prosumers were designed to be intelligent agents that communicate with one another. Utilizing EV battery scheduling, prosumers might save on power costs. A new Multi-Agent ActorCritic (MA2C) system was examined in [13], which was specifically designed for mixed-traffic situations. The MA2C algorithm offers an extensive method of managing urban traffic that prioritizes effectiveness, safety, and passenger security.

To effectively recommend public charging stations, [14] anticipated a Multi-Agent Spatio-Temporal Reinforcement Learning (Master) that takes into consideration several

long-term spatiotemporal characteristics. The Demand Response potential in smart homes using a multi-agent reinforcement learning framework enhanced with BiLSTM and Attention Mechanism for improved data efficiency and handling stochastic household loads [15]. The BiLSTMA-MADDPG model improves data efficiency, convergence speed, and scalability in controlling household appliances under limited training samples. Table 1 presents recent advancements in multi-agent reinforcement learning (MARL) for energy management in smart systems. It highlights diverse applications ranging from EVs and smart grids to smart homes using algorithms like MADDPG, MA2C, and BiLSTMA-MADDPG. While most approaches show improved performance in energy savings and efficiency, common limitations include coordination complexity, high computational needs, and data inefficiency.

Table 1: Contrast examination of traditional works

Ref.	Year	Area Focused	Algorithms	Limitations	Performance
[8]	2023	Energy Management in Multi-mode plug-in hybrid EVs	MADRL, DDPG, Hand-shaking Strategy, Relevance Ratio	Requires careful tuning of DDPG parameters; learning performance is sensitive to learning rate	Energy savings can range from 4% to 23.54% when compared to a single-agent system and a rule-based system.
[9]	2025	Hybrid EVs, Energy Management Strategy	MADRL, MADDPG	Complexity in multi-agent coordination, simulation-based validation only	Fuel consumption was reduced by 26.91% (WLTC) and 8.41% (HWFET), improving EMS robustness.
[10]	2022	Smart Grids, Multi-Agent Systems, EVs.	MARL, Decentralized Markov Decision Process (Dec-MDP), Actor-Critic Networks	High initial training complexity assumes accurate agent-environment modeling.	Significant reduction in simulation time; superior scalability and efficiency
[11]	2023	Optimal Energy Management, Smart Grid, Multi-Energy MicroGrids.	MADRL, Stacked Denoising Auto-Encoders framework	Requires high computational resources, complexity in decentralized implementation, and training convergence	Achieved optimal dispatch of electric and thermal energies, and reduced emissions and costs.
[12]	2023	Smart Grid Energy Management, EV Scheduling, Solar Photovoltaic (PV) Integration	MADRL, Real-Time Pricing, Smart Agent Interaction	High computational requirements for real-time DRL.	Mean power consumption was reduced by 9.04% (vs. no EV usage) and reduced by 39.57% (vs. conventional pricing)
[13]	2024	Smart Cities, Autonomous Vehicles, Sustainable Mobility	MA2C, Reinforcement Learning, Actor-Critic Architecture	Complexity of multi-agent coordination; Requires realistic traffic data for deployment	Outperforms existing models in lane-changing efficiency, safety, comfort, and inter-vehicle cooperation.
[14]	2021	EVs Charging Recommendation, Smart Mobility, DRL	MA2C Framework, Centralized Attentive Critic, Delayed Access Strategy	Required coordination among distributed agents	Outperforms 9 baseline approaches in recommending charging stations.
[15]	2023	Demand Response in Smart Homes	BiLSTMA-MADDPG (Multi-Agent RL)	Non-stationary environment; data inefficiency	Improved data efficiency, faster convergence, and better scalability with small samples.

3 Methodology

The methodology involves modeling the NEV's energy system as a multi-agent environment with engine and battery agents. Real-time driving data undergoes data cleaning and min-max normalization, and PCA for feature extraction. AnSSB-MADQN is employed to optimize power distribution. Trained on WLTC and HWFET cycles, this strategy improves fuel efficiency, stabilizes SOC, and reduces battery degradation, enabling adaptive, real-time energy management under dynamic driving conditions. Figure 2 presents the proposed methodology's overview.

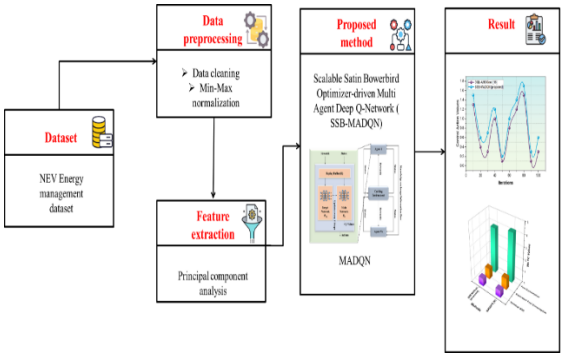

Figure 2: Proposed methodology overflow

3.1 Data collection

The NEV energy management dataset was collected from the Kaggle source. It is meant to assist in finding the most effective ways to save energy in NEVs, using the approach of MARL. It
includes data about real-world traffic, energy distribution, mileage, and battery health for multiple driving routines. 70% of the dataset was used for training and 30% for testing to evaluate performance under diverse scenarios. Source:https://www.kaggle.com/datasets/ziya07/nev-energy-management-dataset/data

3.1.1 Data Description

The NEV Energy Management Dataset features 5,000 records with 13 attributes for measuring vehicle speed along with acceleration, power demand, fuel usage, and battery performance across different driving conditions. The system combines essential variables such as engine power, battery power and SOC, battery degradation, and regenerative braking power to assess energy efficiency and sustainability levels.

3.1.2 Data Exploration

The pair plot demonstrates the relationship dynamics between speed, power demand, battery power, SOC, and fuel consumption variables for designing a dynamic energy management strategy in NEVs. The diagonal presentation displays distribution patterns to identify normal or skewed data shapes. The correlations and strong positive associations between power demand and battery power become visible through off-diagonal scatter plots. Figure 3 shows the data exploration.

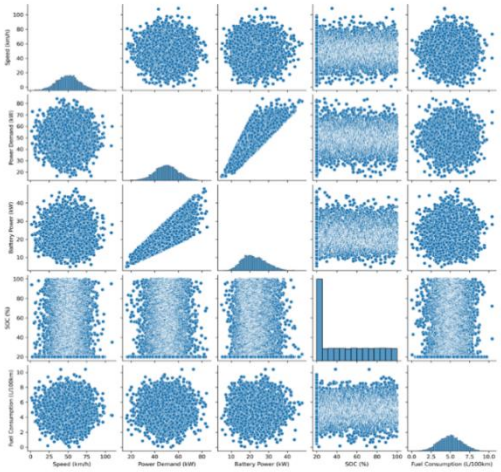

Figure 3: Data exploration outcomes

3.2 Data preprocessing using data cleaning

To clean the NEV energy management dataset, missing values should be handled through mean or median imputation techniques while maintaining sparse data rows. Convert data types to ensure consistency across numerical and categorical fields. The data types should be converted to achieve numerical and categorical field consistency. Reduction of redundant data will occur by eliminating duplicate records. The system needs to identify and handle unusual cases found in energy consumption alongside battery degradation trends. A final test must verify the data balance between driving cycles and efficiency classes.

3.2.1 Min-Max normalization

The process of min-max normalization transforms new energy vehicle energy management datasets into standardized ranges, which improves both model performance and speed of convergence, and accuracy during energy efficiency optimization. Using linear modifications of the original data, min-max normalization creates a balanced set of value comparisons between the data before and after the execution, as follows in Equation (1).

$$W_{new} = \frac{W - \min(W)}{\max(W) - \min(W)} \ldots. \quad (1)$$

W_{new}- The adjusted value derived from the normalized outcomes
W- Old Value
$\max(W)$-The dataset's maximum value
$\min(W)$- The dataset's minimum value

3.3 Feature extraction using PCA

The dynamic energy management technique becomes more efficient by eliminating unnecessary variables and focusing exclusively on critical factors. This results in faster convergence and more accurate decision-making via the MARL framework for energy distribution. PCA was used to minimize the dimensionality of the dataset while retaining the majority of its informational richness. In addition, 5 derived characteristics were designed to capture complicated energy dynamics such as power fluctuation, energy trends, and driving cycle behavior, which are crucial for intelligent EMS control.

By eliminating the class label, each observation in a data set of l observations is mathematically m-dimensional. Assuming that $w_1, w_2, \ldots\ldots, w_l \in \Re^m$. The subsequent procedures for calculating PCA.

Determine the mean vector μ in m-dimensions by Equation (2).

$$\mu = \frac{1}{l}\sum_{j=1}^{l} w_j \tag{2}$$

Determine the observed data's estimated matrix of covariance T by Equation (3).

$$T = \frac{1}{l}\sum_{j=1}^{l}(w_j - \mu)(w_j - \mu)^s \tag{3}$$

Determine the associated eigenvectors and eigenvalues of T, whereby $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_l \geq 0$. Determine the l primary components from the l original variables by Equation (4).

$$\begin{aligned} z_1 &= b_{11}w_1 + b_{12}w_2 + \cdots + b_{1l}w_l \\ z_2 &= b_{21}w_1 + b_{22}w_2 + \cdots + b_{2l}w_l \\ &\cdots \\ z_l &= b_{l1}w_1 + b_{l2}w_2 + \cdots + b_{ll}w_l \end{aligned} \tag{4}$$

It is orthogonal that z_l are uncorrelated. As much of the initial variation in the data set can be explained by z_1, as much of the residual variance can be explained by z_2, etc. In the most useful data sets, a small number of bigger eigenvalues often outnumber the others, as follows in Equation (4). Where the proportion maintained in the data format is denoted by z_l.

$$\gamma_l = \frac{\lambda_1+\lambda_2+\cdots+\lambda_n}{\lambda_1+\lambda_2+\cdots+\lambda_n+\cdots+\lambda_l} \geq 80\% \tag{5}$$

Principal Component Analysis (PCA) was applied to reduce the dimensionality of the input space. Although the original dataset consisted of 13 attributes, only 12 numeric features were used for PCA, excluding the non-numeric target column. PCA transformed this 12-dimensional feature space into 6 uncorrelated principal components, capturing over 95% of the total variance and improving model training efficiency by eliminating redundancy. After applying min-max normalization, PCA reduced the feature space to 6 principal components, maintaining more than 95% of the total variance while minimizing duplication, boosting the energy management model's learning efficiency. Figure 4 shows the PCA-based feature contribution to the first principal component, which explains the most variation. This information assists in determining the most significant elements for EMS optimization. Notably, this representation is based on the PCA loading matrix before dimensionality reduction. Figure 4 shows PCA-Based Feature Importance Output for Energy Management Optimization.

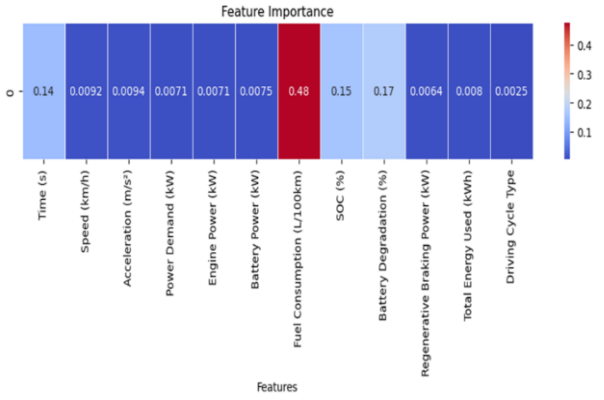

Figure 4: PCA-based feature importance output for energy management optimization

- **Data Cleaning (13 features):** Outliers, impossible values (e.g., negative fuel), and missing values were handled through imputation and filtering.
- **Normalization (13 features):** Each feature was scaled to a standard range (mean = 0, std = 1) for consistent learning performance.
- **PCA Application:** Principal component analysis reduced the final 18-dimensional space to 6 principal components, capturing >95% variance, enhancing model training speed and generalization.

While the original dataset contained 13 attributes, 5 additional derived features were introduced through feature engineering to enhance the model's ability to capture dynamic driving patterns and battery behavior. For instance, ΔSOC (change in State of Charge) reflects short-term battery discharge rates, offering temporal insights that static SOC cannot. Similarly, features like speed trend and regenerative efficiency were designed to capture vehicle acceleration patterns and energy recovery rates, respectively. These engineered features provide higher-level abstractions that improve the learning model's contextual awareness. PCA was then applied to this 18-dimensional space to reduce redundancy, improve

generalization, and retain the most informative patterns by selecting 6 principal components that preserved over 95% of the variance.

3.4 SSB-MADQN

The SSB-MADQN is a novel framework for dynamic energy management in NEVs. It integrates the SBO to enhance agent policy optimization and exploration within a MADQN environment. By enabling decentralized cooperation among energy management agents, SSB-MADQN effectively balances power delivery among both the engine and battery, optimizes fuel consumption, and mitigates battery degradation under diverse driving cycles. The scalable design ensures adaptability across vehicle platforms, while the optimizer enhances learning efficiency, making SSB-MADQN a robust solution for real-time, intelligent NEV energy management.

3.4.1 MADQN

The MADQN enables dynamic energy management in NEVs by allowing multiple agents (engine, battery, motor) to learn cooperative strategies. Through DRL, each agent optimizes energy distribution, improving efficiency, reducing fuel consumption, and adapting to varying driving conditions in real time. It uses a model-free reinforcement learning strategy, which eliminates the need to explicitly understand the environment's dynamics. Agent 1 observes state t_s and chooses the optimal action at time s to move to state t_{s+1} in traditional Q-learning, based on a value model-free approach. The agent then changes the Q-value after receiving an instant benefit $r(t_s, b, t_{s+1})$at time $s+1$, as shown in Equation (6).

$$Q_{s+1}(t_s, b_s) \leftarrow (1-\alpha)Q_s(t_s, b_s) + \alpha[r(t_s, b_s, t_{s+1}) + \gamma \max_b Q_s(t_{s+1}, b)] \quad (6)$$

In reinforcement learning, γ is a discount factor, $\gamma\, max_b' R_s(t', b')$ is the discounted reward, and $\alpha \in [0,1]$ is the learning rate. The Q-values for every potential state and action for agent 1 are stored in a two-dimensional look-up column with dimensions $\mathcal{T} \times \mathcal{B}$. Consequently, the number of actions and states in a complex system causes the Q-table's size to grow exponentially. Figure 5 presents the MADQN architecture. Every edge server is regarded as an agent in EV. Figure 5 depicts the MADQN framework utilized in the caching environment, with architectural details. The neural networks (Main and Target) are implemented as multilayer perceptrons, with an input layer matching the state dimension (e.g., 50 features), two hidden layers of 128 and 64 neurons, respectively, employing ReLU activation, and an output layer representing the number of potential actions (e.g., two for binary caching decisions). These features are critical to understanding the model's structure and ensuring repeatability.

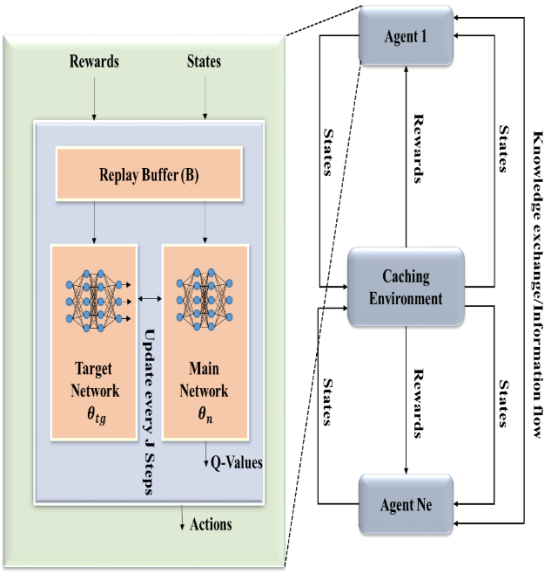

Figure 5: MADQN architecture

In multi-agent reinforcement learning, the replay buffer holds all agents' experiences, which frequently include shared observations, actions, and rewards to capture inter-agent relationships. Each agent's training is stabilized by the target network, which provides constant Q-value targets and is updated on a regular or soft basis. Q-value updates are changed by taking into account not just an agent's action and reward, but also the effect of other agents' activities, employing centralized training and decentralized execution. This allows agents to develop coordinated methods while functioning independently during deployment.

A replay buffer is used to retain the agent's experiences, a target network (θ_{tg}) replicates the main network to offer a steady target for learning, and a main network parameterized by (θ_n) is used to estimate Q-values in the multi-agent environment. First, agent 1 observes the energy demand signal and its states at the time s communicates with neighboring agents (states (t_s) and policies), and selects an action (b_s). For example, suppose that Agent 1 is unable to fulfill the energy storage request. Suppose that three collaborative NEV modules (engine, battery, motor) $(\{i, r\} \in \varepsilon_{nb})$ with a strategy for new energy $q_{F,ji}$and $q_{F,iq}$, where $q_{F,ji} < q_{F,iq}$, have the matching content. This situation results in the selection of the neighboring agent with energy cost, as shown in Equation (7).

$$b_s = \begin{cases} \arg\max_{b\in\mathcal{B}} Q(t_s, b) & o = 1-\epsilon_1-\epsilon_2 \\ random\ b \in \mathcal{B} & o = \epsilon_1 \\ Other\ replacement\ policy\ b \in \mathcal{B} & o = \epsilon_2 \end{cases} \quad (7)$$

Furthermore, it has ϵ_1 and ϵ_2 set to decrease with time. Consequently, the model will eventually choose the best

course of action. It is suggested to explore if the agent does not function well. A collection of recent rewards (R_G) is tracked, and $\in_y$ ($where \ \in_y \ \{1,2\}$)is updated, as shown in Equation (8). The step sizes for modifying the probability $\in_y$ are δ^+ and δ^-, and rth is a reward threshold.

$$\in_y = \begin{cases} \in_y + \delta^+, & \mathbb{E}(Q_G) < r_{th} \\ \in_y - \delta^-, & \mathbb{E}(Q_G) \geq r_{th} \end{cases} \quad (8)$$

The agent moves on to the next state $(t_s + 1)$ for the selected action (b_s), preserves moving in the replay buffer of size, and receives an instant benefit $(r_s + 1)$. During the training stage, agent 1 uses mini-batch descent to train the primary network after selecting a mini-batch of size A from the replay buffer. In every I step, the target network replicates the primary network to provide learning stability, as follows in Equation (9).

$$Q_{s+1} = (t_s, b_t) \leftarrow (1-\alpha) Q_s(t_s, b_t; \theta_n) + \alpha\left[r(t_s, b_t, T_{s+1}) + \gamma \max_b Q_s(t_{s+1}, b; \theta_{sh}) - Q_s(t_s, b_t; \theta_n)\right] + \sum_{i \epsilon M_f} w_{ji} Q_{s-1}(t_s, b_t; \theta_n) \quad (9)$$

Where ω_{ji} is modeled as inversely proportional to the EMS(rF, yx) among $i$and j, and is used to highlight the effect of neighbor I on agent 1.

3.4.2 SSB

The traditional Satin Bowerbird (SB) optimizer struggles to effectively manage the complex, dynamic, and multi-objective nature of energy management strategies in new energy vehicles (NEVs). It lacks the scalability and the ability to deal with several competing priorities, including fuel consumption, battery capacity, and reducing battery degradation. The basic SB algorithm lacks mechanisms for efficiently navigating high-dimensional search spaces or adapting to rapidly changing driving conditions. It also falls short in maintaining solution diversity and handling trade-offs among multiple objectives, often leading to premature convergence or local optima. Furthermore, its limited ability to handle real-time updates and high-dimensional decision spaces reduces its effectiveness in dynamic driving conditions, prompting the need for improved approaches like the Scalable SB (SSB) optimizer. SSB efficiently balances energy distribution between battery and engine systems, adjusts to various driving schedules, speeds up how policies are learned and helps achieve better fuel efficiency, fewer emissions, and longer life of the vehicle battery in complex driving situations.

- ***Logistic Chaos's initialization:***

Although the algorithm's initial population utilizes a random initialization mode according to natural law, a better initialization approach would greatly accelerate the intelligent optimization algorithm's convergence speed. The population is also initialized by the SB using random values. A logistic chaos map was created to improve the starting population's diversity, which in turn led to a better-starting population, which improved the algorithm's accuracy and speed of convergence. Equation (10) illustrates the logistic chaos map calculating method.

$$W_{j+1} = \mu W_j * (1 - W_j) \quad (10)$$

The control parameters μ have a value range of $0 \ to \ 4$. There will be more confusion when the number of μ is higher. The chaotic initialization effect will be amplifiedμ. Equation (11) is used as the population initialization.

$$pop(j).Position = Y(j,:).*(VarMax - VarMin) + VarMin \quad (11)$$

- ***The cauchy variation method:***

Instead of using the conventional SB mutation technique, which produces a shorter peak dispersed at the origin and a longer spread in the remainder, the Cauchy mutation strategy guarantees more disruption near the current population. Equation (12) shows the Cauchy variation approach.

$$W_{j,i}^{s+1} = W_{best} + Cauchy(0,1) \oplus W_{best}(s) \quad (12)$$

Where $W_{best}(s)$ is the location of an individual that requires variation, and $Cauchy$ $(0,1)$ is the typical Cauchy distribution. Equation (13) computes the relevant variation probability.

$$O_t = -\exp\left(1 - \frac{it}{MaxIt}\right)^{20} + o \quad (13)$$

Both the current is represented by $MaxIt$, where o is set at 0.05. The procedure of the Cauchy mutation will not be carried out if q and $< Ps$. Table 2 shows the hyperparameters of SSB.

SSB's chaotic initialization improves exploration by ensuring diverse initial solutions, avoiding local optima, and speeding up convergence. The Cauchy variation, with its heavy-tailed distribution, enables larger step sizes, improving the algorithm's capacity to escape local minima and strike a better balance between exploration and exploitation. These traits exceed typical heuristics, allowing for faster and more efficient optimization.

Table 2: Hyperparameters of SSB

No.	Hyperparameter	Symbol / Name	Typical Value / Range	Description
1	Population size	P	5 – 50	Number of candidate solutions (bowerbirds)
2	Maximum iterations	MaxIter	10 – 100	Maximum SBO optimization cycles
3	Attraction coefficient	α	0.05 – 0.3	Strength of movement toward better solutions
4	Random scaling factor	rand ()	[0, 1]	Random noise for solution diversification
5	Learning rate search range	LR_range	[0.0001, 0.01]	Search space for learning rate
6	Epsilon search range	ε_range	[0.1, 1.0]	Exploration rate range
7	Discount factor search range	γ_range	[0.8, 0.99]	Reward discount factor range
8	Fitness function	F(x)	Avg episodic reward	Evaluate solution quality
9	Movement formula	x_new = x + α * rand () * (x_best - x)	—	Bowerbird movement update
10	Dimensionality of solution	D	3	Parameters optimized (LR, ε, γ)

4 Results and discussion

The result comparison parameters, such as EMS optimization results for different strategies under WLTC, EMS optimization results for different strategies under HWFET, and control action, are used to demonstrate the comparison of the proposed model, SSB-MADQN, for energy management strategy for new energy with the existing techniques, such as MADDPG [9] and Deep Q-learning Adaptive Moment Estimation (DQL-AMSGrad) [16]. The experimental setup is presented in Table 3.

Table 3: Experimental setup

Projects	Environment
Operating System	Windows 10(x64)
CPU	i5-9500HF CPU@2.40GHz
Memory Size	32GB
GPU	NVIDIA GeForce GTX 2080 Ti
CUDA Version	10.2
Python Version	3.8
Episode count Batch size Convergence criteria	1000 64 Training stops when reward, loss, episodes, or epsilon criteria are met.

4.1 Confusion matrix

The results of the confusion matrix are shown in Figure 6. The model accurately predicted all classes: 152 samples as class 0, 777 as class 1, and 71 as class 2, with zero misclassifications. This indicates that the energy management model is highly effective in correctly categorizing vehicle energy efficiency levels or strategies with no false positives or negatives across all classes. The predicted classes represent EMS efficiency levels: 0 (High), 1 (Medium), and 2 (low).

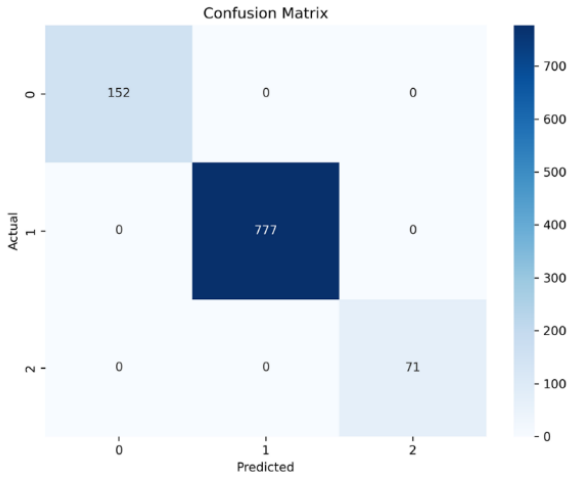

Figure 6: Confusion matrix outcomes

4.2 Battery degradation distribution

The distribution of battery degradation in NEV highlights a concentration of around 10%. It suggests significant wear under certain conditions, necessitating a dynamic energy management strategy. By integrating real-time degradation data, NEVs can optimize engine-battery energy distribution, extend battery life, and improve energy efficiency, especially under high-degradation scenarios. It supports adaptive, data-driven decision-making for sustainable vehicle performance. Figure 7 presents the distribution of battery degradation outcomes.

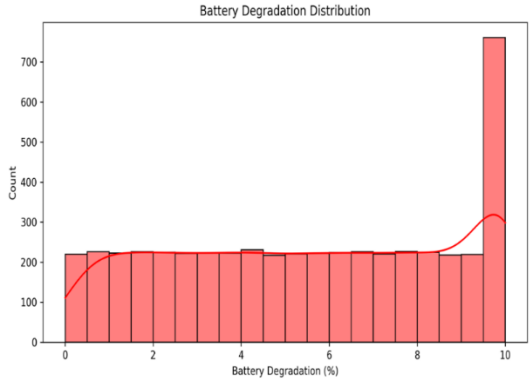

Figure 7: Distribution of battery degradation outcomes

4.3 WLTC

The EMS optimization results under the WLTC driving cycle show that the proposed SSB-MADQN method outperforms the existing method, MADDPG. SSB-MADQN achieves a higher terminal SOC (0.643 vs. 0.598), lower equivalent fuel consumption (0.912 L vs. 0.977 L), and improved fuel efficiency (3.864 L/100km vs. 4.199 L/100km), demonstrating its effectiveness in dynamic energy management for NEVs by enhancing energy utilization and reducing fuel use. Figure 8 presents the EMS optimization under WLTC.

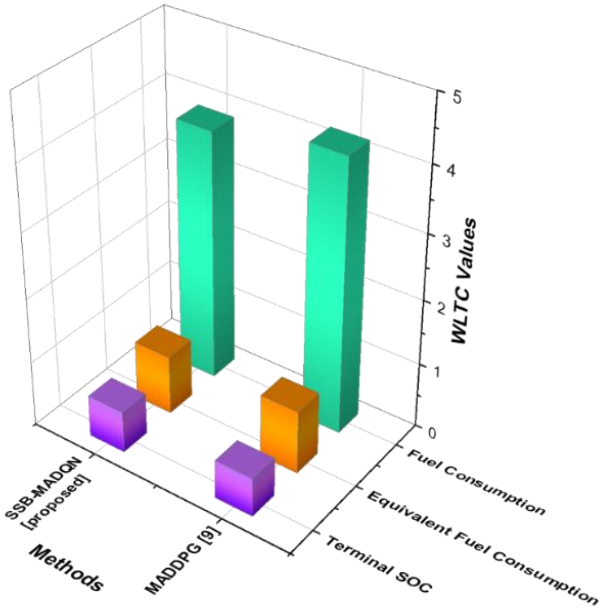

Figure 8: Graphical representation of WLTC

4.4 HWFET

According to the HWFET driving cycle, SSB-MADQN performs better than MADDPG when optimizing the EMS system. It achieves a higher terminal SOC (0.603 vs. 0.556), reduced equivalent fuel consumption (0.681 L vs. 0.734 L), and better fuel efficiency (4.121 L/100km vs. 4.446 L/100km), indicating improved energy recovery and reduced fuel usage in dynamic energy management for NEVs. Figure 9 presents the EMS optimization under HWFET.

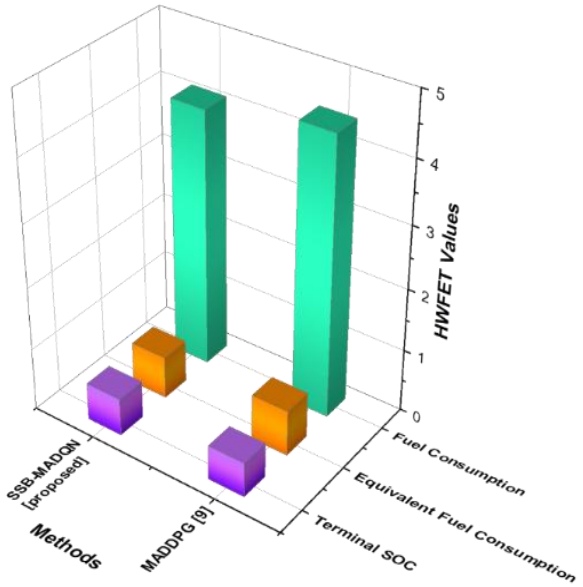

Figure 9: Graphical Representation of HWFET

4.5 Control action

A comparison of control action variations over time in dynamic energy management for NEVs. DQL-AMSGrad shows fluctuating control values, peaking at 1.5, indicating moderate adaptability. The proposed SSB-MADQN model consistently yields slightly higher control actions, with smoother transitions and a peak of 1.7, reflecting improved responsiveness and stability. It suggests SSB-MADQN's superior performance in managing energy distribution dynamically and efficiently in NEV systems. Table 4 and Figure 10 show control action outcomes.

Table 4: Control action outcomes

Model	10	20	30	40	50	60	70	80	90	100
DQL-AMSGrad [16]	1.3	0.4	0.3	1.0	0.1	0.8	1.2	1.5	0.1	0.3
SSB-MADQN [proposed]	1.5	0.6	0.7	1.2	0.2	1.0	1.4	1.7	0.3	0.6

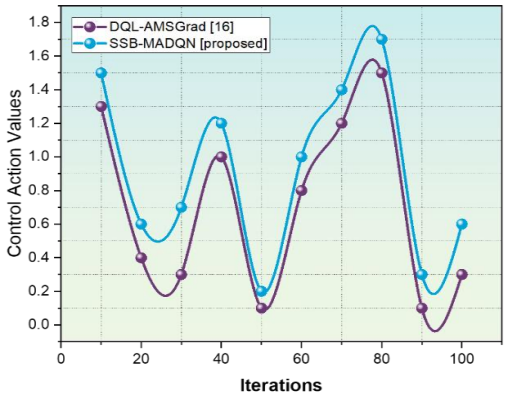

Figure 10: Graphical representation of control action

4.6 Performance metrics summary of SSB-MADQN for NEV energy management

The primary performance metrics of the proposed multi-agent deep reinforcement learning framework applied to dynamic energy management in new energy vehicles (NEVs). Metrics include fuel consumption, battery SOC limits, battery degradation rate, and computational efficiency during both training and real-time inference. These results demonstrate the framework's effectiveness in balancing energy usage and system longevity. Table 5 displays the SSB-MADQN performance.

Table 5: Key results of SSB-MADQN performance

Performance metric	SSB-MADQN (Proposed)
Fuel Usage	3.4 L/100km
SOC Bounds	20% – 80%
Degradation Rate (%)	0.72%
Training Time	4.1 hours
Inference Time	14 ms

5 Comparative analysis with existing systems

A dynamic EMS for NEVs optimizes power distribution between the battery and engine in real-time, enhancing energy efficiency, reducing emissions, and adapting to varying driving conditions. MADDPG faces limitations in scalability and convergence stability when managing complex multi-agent interactions in dynamic NEV energy environments. Such technology mandates a large amount of training
material alongside powerful computing capabilities. The integration of DQL-AMSGrad with adaptive learning rates facilitates better convergence, but it performs poorly with the continuous action spaces regularly found in NEV energy systems. The decision-making processes of these

methods show poor adaptation to sudden driving condition changes, along with restricted performance across different driving cycles, which affects real-time decisions in NEVs. The proposed SSB-MADQN enhances scalability and convergence stability by integrating the SSB with MADQN, enabling efficient exploration and exploitation in complex NEV environments. The system successfully deals with complex action spaces together with dynamic driving conditions because it learns quickly and provides reliable real-time energy management functionality that outperforms MADDPG and DQL-AMSGrad by showing better adaptability and generalization over several driving cycles. The proposed strategy relies heavily on high-quality simulations, which may not fully capture real-world complexities. Additionally, there is a lack of real-world validation, and the interpretability of multi-agent reinforcement learning models remains a challenge, hindering broader practical adoption.

6 Conclusion

Energy efficiency and operational performance in NEVs have significantly improved through the application of AI-driven optimization strategies. The suggested SSB-MADQN architecture used MARL to allow cooperative agents to control the engine and battery's power allocation in real time under various driving circumstances. Data preprocessing methods, such as data cleaning and min-max normalization, and PCA employed for feature extraction, ensured consistency, reduced dimensionality, and enhanced model learning. Experimental results revealed notable improvements, with fuel consumption reduced under WLTC compared to MADDPG, achieving a final consumption of 3.864 L/100km, and similarly under HWFET with a reduction to 4.121 L/100km. These outcomes confirm the effectiveness of intelligent EMS in achieving adaptive and globally optimized energy strategies for NEVs. The limitations of relying solely on simulation-based testing and plans to incorporate real-world ECU-in-the-loop evaluation to enhance validation. Another key challenge is the interpretability of the MARL model, for which we plan to adopt explainability techniques such as SHAP or LIME to analyze Q-values and better understand agent decisions. Additionally, potential deployment on edge computing platforms like NVIDIA Jetson is being considered to assess real-time feasibility. The proposed approach shows strong potential for real-time EMS in NEVs by leveraging decentralized agents and a powerful optimizer for high-dimensional spaces. However, to strengthen its scientific contribution, future work should focus on improving algorithm transparency, ensuring rigorous experimentation, and incorporating advanced statistical techniques for deeper validation and performance comparison.

References

[1] Wang, Y., Wu, Y., Tang, Y., Li, Q., & He, H. (2023). Cooperative energy management and eco-driving of plug-in hybrid electric vehicle via multi-agent reinforcement learning. *Applied Energy*, *332*, 120563. https://doi.org/10.1016/j.apenergy.2022.120563

[2] Yang, N., Han, L., Liu, R., Wei, Z., Liu, H., & Xiang, C. (2023). Multiobjective intelligent energy management for hybrid electric vehicles based on multiagent reinforcement learning. *IEEE Transactions on Transportation Electrification*, *9*(3), 4294-4305. https://doi.org/10.1109/TTE.2023.3236324

[3] Gautam, A. K., Tariq, M., Pandey, J. P., Verma, K. S., & Urooj, S. (2022). Hybrid sources powered electric vehicle configuration and integrated optimal power management strategy. *IEEE Access*, *10*, 121684-121711.https://doi.org/10.1109/ACCESS.2022.3217771

[4] Jiang, Q., & Wang, H. (2025). Risk Assessment Method for New Energy Vehicle Supply Chain Based on Hierarchical Holographic Model and Matter Element Extension Model. *Informatica*, *49*(7). https://doi.org/10.31449/inf.v49i7.6953.

[5] Hu, H., Yuan, W. W., Su, M., & Ou, K. (2023). Optimizing fuel economy and durability of hybrid fuel cell electric vehicles using deep reinforcement learning-based energy management systems. *Energy Conversion and Management*, *291*, 117288. https://doi.org/10.1016/j.enconman.2023.117288

[6] Bakare, M. S., Abdulkarim, A., Shuaibu, A. N., & Muhamad, M. M. (2024). Energy management controllers: strategies, coordination, and applications. *Energy Informatics*, *7*(1),57.https://doi.org/10.1186/s42162-024-00357-9

[7] Rawat, R., Borana, K., Gupta, S., Ingle, M., Dibouliya, A., Bhardwaj, P., & Rawat, A. (2025). Enhancing OSN Security: Detecting Email Hijacking and DNS Spoofing Using Energy Consumption and Opcode Sequence Analysis. *Informatica*, *49*(2). https://doi.org/10.31449/inf.v49i2.6956.

[8] Hua, M., Zhang, C., Zhang, F., Li, Z., Yu, X., Xu, H., & Zhou, Q. (2023). Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning. *Applied Energy*, *348*, 121526.https://doi.org/10.1016/j.apenergy.2023.121526

[9] Li, X., Zhou, Z., Wei, C., Gao, X., & Zhang, Y. (2025). Multi-objective optimization of hybrid electric vehicles energy management using multi-agent deep reinforcement learning framework. *Energy and AI*, *20*, 100491. https://doi.org/10.1016/j.egyai.2025.100491

[10] Alqahtani, M., Scott, M. J., & Hu, M. (2022). Dynamic energy scheduling and routing of a large fleet of electric vehicles using multi-agent reinforcement learning. *Computers & Industrial Engineering*, *169*, 108180. https://doi.org/10.1016/j.cie.2022.108180

[11] Monfaredi, F., Shayeghi, H., & Siano, P. (2023). Multi-agent deep reinforcement learning-based optimal energy management for grid-connected multiple energy carrier microgrids. *International Journal of Electrical Power & Energy Systems*, *153*, 109292. https://doi.org/10.1016/j.ijepes.2023.109292

[12] Kaewdornhan, N., Srithapon, C., Liemthong, R., & Chatthaworn, R. (2023). Real-time multi-home energy management with EV charging scheduling using multi-agent deep reinforcement learning optimization. *Energies*, *16*(5), 2357. https://doi.org/10.3390/en16052357

[13] Louati, A., Louati, H., Kariri, E., Neifar, W., Hassan, M. K., Khairi, M. H., ... & El-Hoseny, H. M. (2024). Sustainable smart cities through multi-agent reinforcement learning-based cooperative autonomous vehicles. *Sustainability*, *16*(5), 1779.https://doi.org/10.3390/su16051779

[14] Zhang, W., Liu, H., Wang, F., Xu, T., Xin, H., Dou, D., & Xiong, H. (2021, April). Intelligent electric vehicle charging recommendation based on multi-agent reinforcement learning. In *Proceedings of the Web Conference 2021* (pp. 1856-1867). https://doi.org/10.1145/3442381.3449934

[15] Al-Saffar, M., & Gül, M. (2023). Data-efficient MADDPG based on self-attention for IoT energy management systems. *IEEE Access*, *11*, 109379-109389. https://doi.org/10.1109/ACCESS.2023.3322193.

[16] Montaleza, C., Arévalo, P., Gallegos, J., & Jurado, F. (2024). Enhancing energy management strategies for extended-range electric vehicles through deep Q-learning and continuous state representation. *Energies*, *17*(2), 514.https://doi.org/10.3390/en17020514 s